A Performance-Consistent and Computation-Efficient CNN System for High-Quality Brain Tumor Segmentation

Juncheng Tong

A Thesis
In the Department
of
Electrical and Computer Engineering

Presented in Partial Fulfillment of the
Requirements for the Degree of
Master of Applied Science (Electrical and Computer Engineering)
at Concordia University
Montreal, Quebec, Canada

April 2022

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: Juncheng Tong

Entitled: A Performance-Consistent and Computation-Efficient CNN System for High-Quality Brain Tumor Segmentation

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Electrical and Computer Engineering)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final Examining Committee:

______________________________________Chair
Dr.Wei-Ping Zhu

______________________________________Examiner
Dr.Wei-Ping Zhu

______________________________________Examiner
Dr. Wenfang Xie (MIE)

______________________________________Supervisor
Dr. Chunyan Wang

Approved by ______________________________________
Dr. Yousef Shayan, Chair
Department of Electrical and Computer Engineering

____

Dr. Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

A Performance-Consistent and Computation-Efficient CNN System for High-Quality Brain Tumor Segmentation

Juncheng Tong

Brain tumors cause serious health problems and brain tumor detection is important for the diagnosis. The detection is a very challenging task due to the complexity in brain structures and in brain tumor patterns. Manual segmentation requires an expertise of highly trained medical specialists and is very time-consuming. Therefore, it's imperative to develop fully automated brain tumor segmentation systems, i.e., CNN based systems, to accelerate the diagnosis process. The research on developing such systems has been progressed rapidly in recent years. For the systems to be applicable in practice, a good processing quality and reliability are required. Moreover, for a wide range of applications of such systems, a minimization of computation complexity is desirable, which can also result in a minimization of randomness in computation and, consequently, a better performance consistency.

In this thesis, a new CNN system for brain tumor segmentation is proposed. The CNN in the proposed system is custom-designed with 2 distinguished characters dedicated to optimizing the feature extraction and classification processes. Firstly, there are three paths in its feature extraction block, designed to extract, from the multi-modality input, comprehensive feature information of mono-modality, paired-modality and cross-modality, respectively. Also, it has a particular three-branch classification block to identify the pixels of 4 classes, namely, whole tumor, enhancing tumor, non-enhancing core/necrotic tumor and those in the background. By means of the three branches, a complex multi-class classification problem is decomposed into several simple binary classification problems. Each branch is trained separately so that the parameters are adjusted specifically to suit the detection of one specific kind of tumor areas. The parameters of the convolution layers in the proposed system are determined to suit the specific purposes so that the computation volume for each filtering operations in each layer are just-sufficient, which results in a very simple config of 61,843 parameters in total, while most existing CNN systems require multi-millions.

The performance of the proposed system has been tested extensively with BraTS 2018 and BraTS 2019 data samples. A good mean Dice scores in each experiment has been obtained. The average of the mean Dice scores obtained from ten experiments are very close to each other with very small deviations. In the case of the 10 experiments on BraTS 2018 validation samples, the average Dice scores and their standard deviations are 0.787±0.003, 0.886±0.002, 0.801±0.007, respectively, for enhancing tumor, whole tumor and tumor core. For the validation samples of BraTS 2019 in 10 experiments, the average Dice scores and standard deviations of enhancing tumor, whole tumor and tumor core are 0.751±0.007, 0.885±0.002, 0.776±0.004, respectively. The test results demonstrate that the proposed system is able to perform high-quality segmentation in a consistent manner. Furthermore, it only requires 146G FLOPs to complete a segmentation of the four 3D images (155x240x240x4 voxels) of a single patient case. The extremely low computation complexity of the proposed system will facilitate its implementation/application in various environments.

The high processing quality and low computation complexity of the proposed system make it implementable in various environments. It can be expected that such system will have wide applications in medical image processing.

Acknowledgement

I would firstly like to thank my supervisor Dr. Chunyan Wang for her support and help during my study in Concordia University. Her insightful feedback and guidance pushed me to sharpen my thinking and brought my work to a higher level. It’s my great honor to work under her supervision.

I would also like to thank my colleges, Mingze Ni, Yanming Sun and Yijian Zhao for their help during my study.

Last but least, thank Compute Canada for providing the powerful GPUs.

Contents

List of Tables

List of Figures

List of Acronyms and Abbreviations

Acronym/ Abbreviation	Full name	First appearance
Adam	Adaptive moment estimation	Page 12
BraTS	The Multimodal Brain Tumor Segmentation Bench Mark	Page 21
CBICA	Center for Biomedical Image Computing and Analysis	Page 33
CNN	Convolutional Neural Network	Page 3
ED	Edema	Page 2
ET	Enhancing Tumor	Page 2
FE	Feature Extraction	Page 22
Flair	Fluid-attenuated inversion recovery	Page 2
FLOPs	Floating-Point Operations	Page 4
MRI	Magnetic Resonance Imaging	Page 2
ReLU	Rectified Linear Units	Page 9
SGD	Stochastic Gradient Descent	Page 12
T1	T1-weighted imaging	Page 2
T1ce	T1-weighted contrast-enhanced imaging	Page 2
T2	T2-weighted imaging	Page 2
TC	Tumor Core	Page 2
VGG	Visual Geometry Group	Page 8
WT	Whole Tumor	Page 2

List of Symbols

Symbols	Meaning	First appearance
m	Number of samples in a mini-batch	Page 8
P_0	Tumor-free regions in a predicted image	Page 35
P_1	Tumor regions in a predicted image	Page 35
STDDEV	Standard deviation	Page 42
T_0	Tumor-free regions in the ground truth	Page 35
T_1	Tumor-free regions in the ground truth	Page 35
μ_B	The mean of a mini-batch	Page 8
σ_{B^2}	The standard deviations of a mini-batch	Page 8
x_i	A sample in a mini-batch	Page 8
y_i	The output of batch normalization	Page 8

1. Introduction

1.1. Brain Tumor Segmentation and Challenges

Brain tumors cause serious health problems that affects life span of human being. In 2021, there are about 3,100 patients in Canada diagnosed with brain tumors [1], and much more are waiting anxiously for their diagnosis results. Brain tumor detection is important for the diagnosis and treatment. Manual segmentation is time-consuming and demands the expertise of neuroradiologist, which may lead to lengthy waits in many patient cases. Therefore, it's imperative to develop fully automated brain tumor segmentation systems to reduce the work loads of medical specialists and to accelerate the diagnosis process in order to improve the patients' chance of survival.

Human brain is probably the most complex organ in human body that contains a staggering one hundred billion nerve cells, or neurons, each of which has quite different structures [3]. A brain tumor is a growth of abnormal cells in the brain. Brain tumors can develop in any part of a brain, including its protective lining, the underside of the brain (skull base), the brainstem, the sinuses and the nasal cavity, and many other areas. There are more than 120 different types of tumors that can develop in the brain, depending on what tissue they arise from [4].

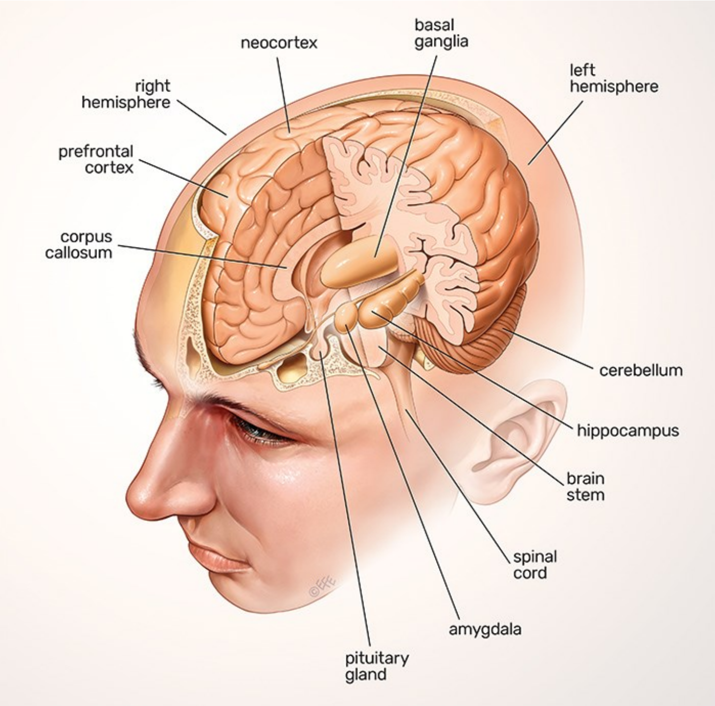

Figure 1.1 Complexity of brain structures and potential location of brain tumors [2].

The structure of brain tumor can be segmented into 3 intra-tumoral parts, namely edema (ED), non-enhancing core/necrotic tumor (NET) and enhancing tumor (ET). The segmentation of 3D brain images is, ultimately, to classify the voxels into 4 classes, i.e., 3 classes of pathological tissues, namely enhancing tumor (ET), tumor core (TC) comprising ET and NET, whole tumor (WT), and the background, i.e., the healthy area. In the case of brain scanning by Magnetic Resonance Imaging (MRI), four 3D brain images of 4 modalities, i.e., Flair, T2, T1ce and T1, are acquired for each patient case. Each 3D image is usually sliced into a large number of 2D slices. Samples of brain image slices of 4 patients and the segmentation result approved by board-certified neuroradiologists [5] are illustrated in Figure 1.2. One can observe that the tumor sizes, the tumor locations, the textural patterns and the intensity ranges in tumor areas vary from patient to patient. These variations make the detection of brain tumors a very challenging task.

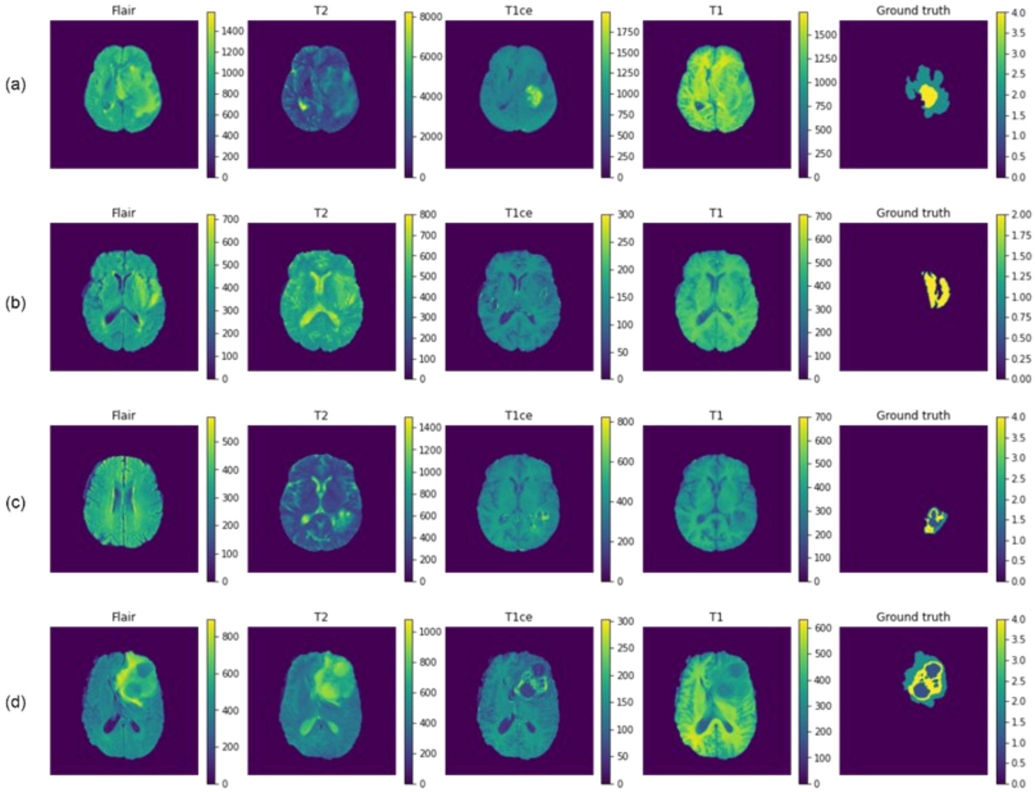

Figure 1.2 Samples of the four-modality MRI image and its corresponding ground truth of tumor area of four patient cases. Each row represents a single patient case.

In general, brain tumor segmentation is actually a multi-class classification problem from the multi-modality input. The variations of brain tumors and the complexity of brain structures make the segmentation a very difficult and challenging task.

1.2. Motivation and Objective

Computer vision algorithm is a potential solution to develop fully automated brain tumor segmentation system. Brain tumor segmentation by computer vision comprises two processes, extraction of the image feature data representing tumor patterns and regrouping the feature data for classification. The feature extraction is done mainly by filtering operations performed sequentially and/or concurrently. Various filters, such as Sobel filters [6] and Gabor filters [7], have been used to extract feature data of different patterns. Nevertheless, it's not feasible to apply a large number of filters of handcrafted coefficients in a short time to deal with the various tumor patterns, which may cause a loss of critical pathological feature information and thus affects the quality of the classification in the succeeding process. The feature data regrouping for classification can also be done by means of filtering operations, but it also has the difficulty in manually selecting a large number of suitable filters to correctly classify various tumor areas.

Convolutional neural networks (CNN) have been used effectively for varieties of image processing purposes [8]-[11]. As the filtering coefficients in a CNN can be determined by means of progressive update in a training process, without the need to choose them manually, one can apply a sufficient number of filters to handle various kinds of feature data. Moreover, as the network is trained with the ground truth data verified by medical specialists, the filters of the network can be set-up and fine-tuned with their knowledge in the topic area.

It should, however, be mentioned that, operating a large number of filters in a CNN requires a huge number of calculations, let alone training the filters, which may be a problem of implementation/application in a computation resource restricted environment. Even though cloud computing might be a potential solution to handle the limited resources problem, for highly sensitive medical data, the privacy issues should be concerned. Also, while inadequacy of filtering capacity affects the processing quality, excessively use of filters can result in problems, such as performance inconsistency due to the randomness in the training process. For a reliable system of medical application, performance consistency is an issue of critical importance and the randomness is not tolerable.

The objective of the work presented in the thesis is to design a CNN system specifically for brain tumor segmentation. The design focus is on the efficiency in computation, i.e., while achieving a high segmentation quality, minimizing the computation complexity, measured by the number of trainable filtering coefficients and the number of FLOPs required in detection. A performance consistency can then be obtained with a high computation efficiency.

1.3. Scope and Organization

To achieve the objective, the proposed CNN system is designed to have three blocks, i.e., a pre-processing block for data reduction, a CNN block for main segmentation, and a post-processing block for refinement. The CNN will be custom-designed to receive the 4-modality data and to generate the specific classification results of 4 classes. To this end, signal processing techniques should be applied to optimize the feature extraction and classification processes.

- Feature extraction is focused on the exploitation of the data acquired with different modalities that the feature data generated from the feature extraction block can capture the critical pathological patterns from brain images
- Classification process is designed to suit the feature selections of each of the 3 sub-tumoral areas and to optimize the training process. Different criteria should be applied to precisely detect different kinds of tumor areas.

The rest of the thesis is organized as follows. The background of CNN and related work on brain tumor segmentation by CNN system are presented in Chapter 2. In Chapter 3, a new CNN system is proposed and designing details are elaborated. The training details and performance evaluation of the proposed system are described in Chapter 4. The comparison between the results of the proposed system and those of other CNN systems recently reported in reputed journals is also presented in Chapter 4. The conclusion of the work of this thesis is given in Chapter 5.

2. Background and Related Work

2.1. Introduction

Brain tumor segmentation is a very challenging task. From computer vision point of view, this task consists of two phases, feature extraction and classification. They can be done by means of filtering operations [6], [7]. The coefficients of the filtering kernels can be manually determined based on the knowledge obtained by data analysis, which requires low computation complexity and is independent of training samples. Understanding the basics of filtering operation is essential for optimization of filter designs. The knowledge-based filter design has, however, its limitations, and filtering operations implemented in CNN are considered as a potential solution to it.

In this chapter, basic filtering operations is firstly presented in Sub-chapter 2.2. Sub-chapter 2.3 is dedicated for a brief presentation of the basic elements in CNN. The CNN based brain tumor segmentation systems are described in Sub-chapter 2.4, followed by a brief summary of the chapter.

2.2. Basic Filtering Operations

In computer vision systems, spatial convolutions are the basic filtering operations. An example of 2D convolution on an image is shown as below. The filter is sliding over the image by performing point-wise multiplication to generate feature maps. Supposing an $s \times s$ image is convolved with an $n \times n$ filter and the stride is 1, then the size of output feature map is $(s - n + 1) \times (s - n + 1)$. An example is shown in Figure 2.1.

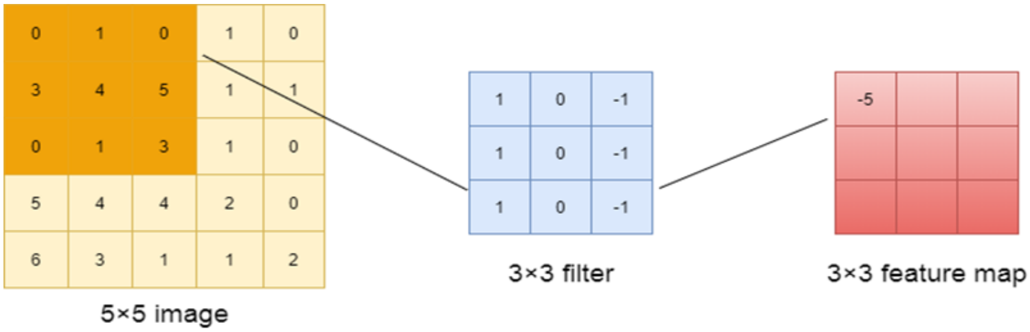

Figure 2.1 An example of how convolution works.

Different choices of parameters in the filter will generate various feature maps. There are three kinds of 2D filters, i.e., high-pass filter, low-pass filter and band-pass filter. An example of low-pass and high-pass filters is shown in Figure 2.2

As shown in Figure 2.3, the high-pass filter detects the edges of the image, while the low-pass filter can smoothen the image at the cost of fine-grained details.

$$1/16\begin{bmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{bmatrix} \qquad \begin{bmatrix}-1 & -2 & -1\\ 0 & 0 & 0\\ 1 & 2 & 1\end{bmatrix}$$

(a) (b)

Figure 2.2 An example of low-pass filter and high-pass filter. (a) Gaussian filter is a low-pass filter. (b) Sobel filter is a high-pass filter.

Figure 2.3 A comparison between high-pass and low-pass filter. High-pass filter detects the edges of the image, while low-pass smoothens the image.

2.3. CNN

Brain tumors have a lot of variations and it is too time-consuming to manually determine a large number of suitable coefficients for the filtering kernels to tackle the complex brain tumor segmentation problem. Therefore, CNN, whose coefficients can be updated, based on the error generated in the comparison of the predicted data and the ground truth, is chosen. The update is done progressively by means of backpropagation. The most commonly used backpropagation method is gradient descent. Gradient descent is an optimization algorithm that minimizes the loss function by iteratively moving in the direction of steepest direction, i.e., its derivatives. Like the traditional filtering method, the most important operation in CNN is still the convolution. There are three most commonly used categories of convolutions in CNN. Supposing the kernel size is 3×3, the number of input and output channels is 4 and 16 respectively:

1. Standard convolution. Each output channel is generated by the convolution between filters and all input channels. The number of parameters in this mode is 592 (3×3×4×16+16).
2. Group convolution. In this type of convolution, the input channels are divided into different groups and each of them is performing a standard convolution. Assume group equals to 2, the number of parameters used is 304 (3×3×4×16/2+16).
3. Depthwise convolution. This kind of convolution can be viewed as a special case of group convolution where the number of groups equals to the number of input channels. Therefore, the number of parameters of depthwise convolution is 160 (3×3×4×16/4+16).

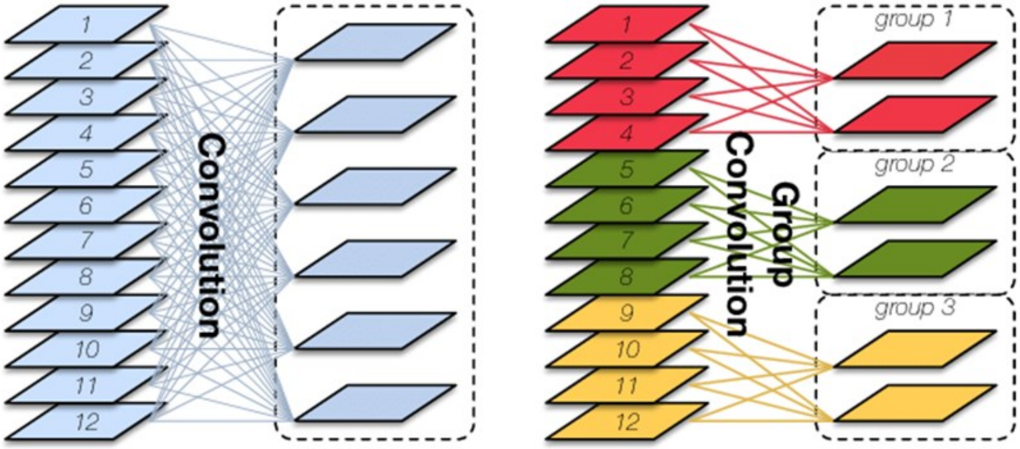

Figure 2.4 Examples of standard convolution and group convolution.

Modern CNN layer, as shown in Figure 2.5, not only consists of learnable convolution kernels but also involves in several other key components, i.e., batch normalization, activation functions, pooling operation and upsampling. Some hyperparameters choices, i.e., batch size, loss function, training epochs, will also have a huge effect on the training process and the overall model performance.

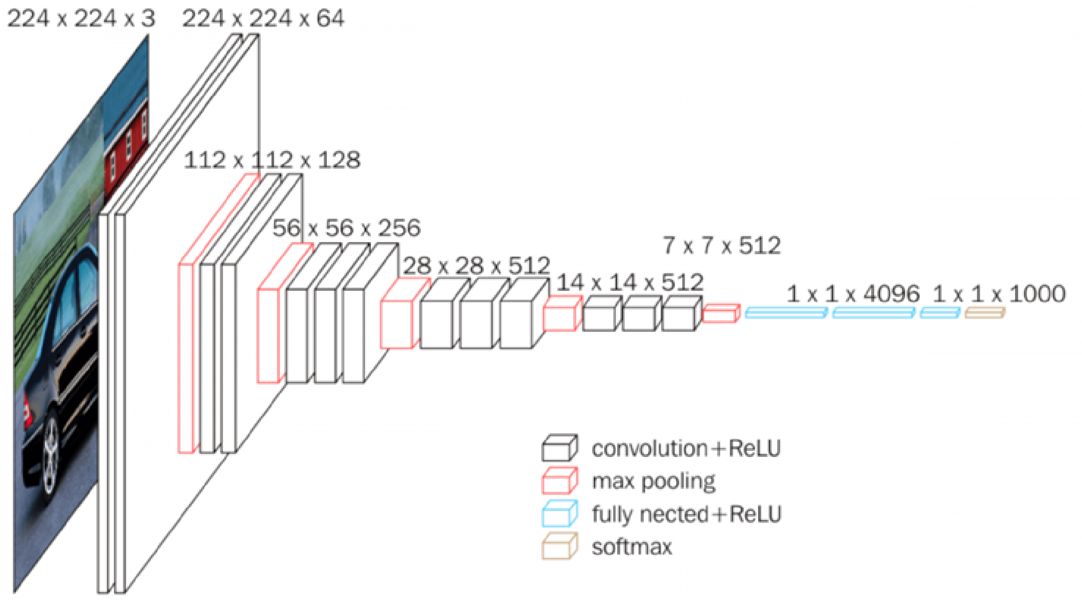

Figure 2.5 Architecture of VGG [10].

2.3.1. Batch Normalization

Batch normalization is often applied after convolution operation, it normalizes the feature data across the batch dimension and maintains the contribution of each feature so that the network is uniformed and unbiased. This technique can accelerate the training process and make the training more stable. Supposing x_i is a sample in a mini batch with m samples, μ_B and σ_B^2 are the mean and variance of the mini batch, ϵ, γ, β are the hyperparameters that need to be selected. Then the formula of batch normalization is defined as following:

$$\mu_\beta = \frac{1}{m}\sum_{i=1}^{m} x_1 \tag{2.1}$$

$$\sigma_B^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_B)^2 \tag{2.2}$$

$$\widehat{x_i} = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 - \epsilon}} \tag{2.3}$$

$$y_i = \gamma\widehat{x_i} + \beta \tag{2.4}$$

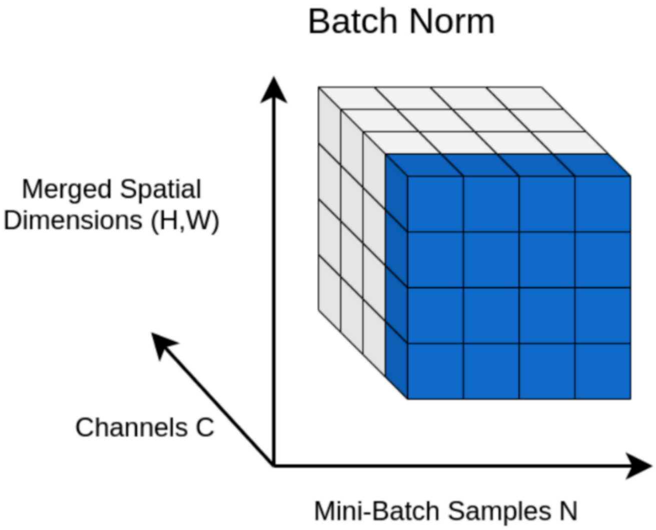

Figure 2.6 Batch Normalization [12].

In the testing time, the mean and variance of each batch normalization layer are determined by a record of empirical mean and variance taken at the training time, instead of calculating the mean and variance for the test batch.

2.3.2. Non-linear Activation function

Many complex features can't be simply represented by linear combination. Therefore, non-linear activation function is applied which can increase the expressiveness of the neural network. The most commonly used activation function in a CNN layer, except for the last layer, is Rectified Linear Unit (ReLU). The function of ReLU is defined as:

$$y = \max(0, x) \tag{2.5}$$

Only positive values can pass through it. Unlike some activation functions. i.e., Sigmoid and Tanh, who have the vanishing gradient problem, ReLU activation function normally doesn't suffer from this problem and therefore, is widely used in the hidden layer.

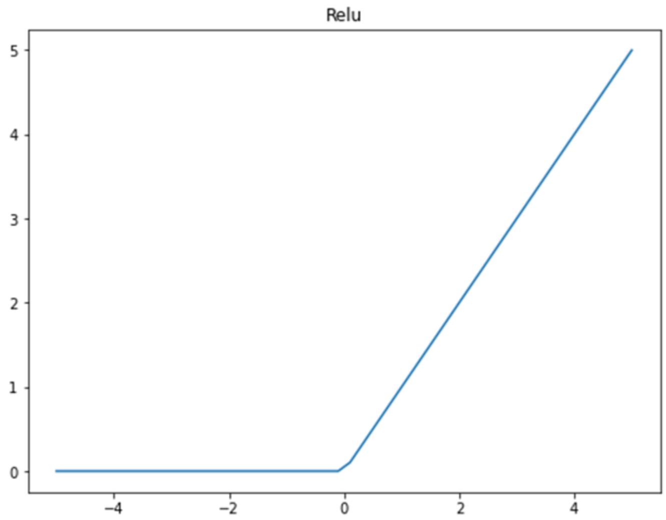

Figure 2.7 Curve of ReLU.

2.3.3. Pooling operation

Pooling layers are used to reduce the dimension of output feature maps so that the information density can be increased and the amount of computation can be decreased. In computer vision, this operation can also increase the receptive filed of each pixel. Two popular pooling techniques are max-pooling and mean pooling.

1. Max-pooling. Supposing the stride is 2 and the filter size is 2×2, max-pooling selects the maximum number from the region covered by the filter.
2. Mean pooling. Unlike max-pooling which selects the maximum number, mean pooling is to select the average from the region covered by the filter.

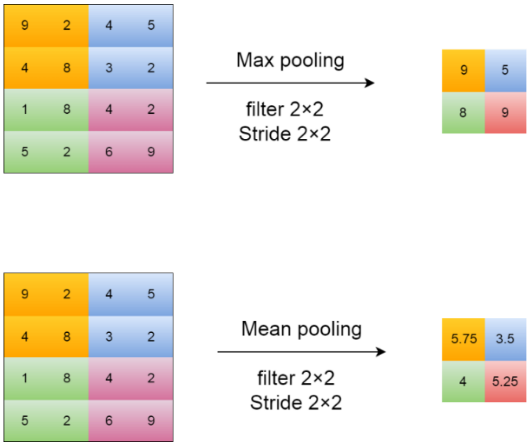

Figure 2.8 An example of max-pooling and mean pooling operation.

2.3.4. Upsampling

Unlike image classification problem, image segmentation requires the output image has the same dimension as the input image. Since max pooling reduces the dimension of the input image, upsampling is needed in order to restore the original image dimension. There are two commonly used upsampling method. One is linear interpolation and the other is transposed convolution.

1. Linear interpolation. Bilinear interpolation is one of the most widely used and efficient upsampling techniques. The value of an unknown function f at point (x, y) is estimated by four known points (x_1, y_1), (x_1, y_2), (x_2, y_1), (x_2, y_2). This process is illustrated in Figure 2.9.

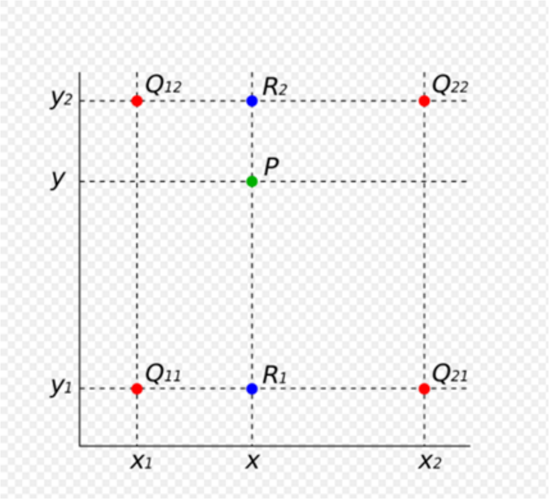

Figure 2.9 Illustration of Bilinear Interpolation [13].

2. Transposed convolution. In spite of efficiency of bilinear interpolation, there is no learnable parameters in the process and the result might not be accurate. Consequently, transposed convolution with learnable parameters is used. Unlike in conventional convolution where the dimension of output channels will be reduced, in transposed convolution, the dimension of output channels will be increased. Supposing the kernel size is k, the stride equals to s, the padding is p, the size of input image is n, then the output image size m equals to:

$$m = (n - 1) * s - 2p + k \tag{2.6}$$

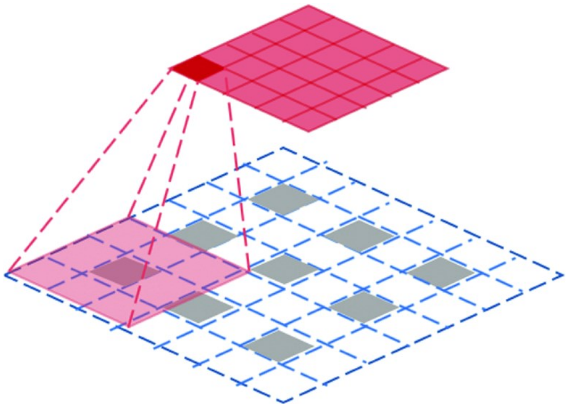

Figure 2.10 Illustration of transposed convolution [14].

2.3.5. Training Details

Learnable parameters in CNN model are trained by trying to minimize the loss between the predicted result from the network and the ground truth data. The choice of several hyperparameters is essential for the training of the model.

1. Image augmentation. In order to make the model more robust and generalized, a simple way is to perform image augmentation which can create variations of images. Affine transformations such as rotation, flip and translation are broadly used.
2. Batch size. The choice of batch size shouldn't be too small, otherwise batch normalization will potentially have negative effect on the training process. However, due to the limit of GPU memory, large batch size is also an infeasible choice. It should be chosen based on the volume of dataset and GPU memory in order to optimize training process.
3. Loss function. Cross-entropy is widely used in image classification task. While in image segmentation, class imbalance is a huge problem where other loss function is introduced such as Dice loss.
4. Learning rate. In the initial training epochs, large learning rate should be chosen to fasten the convergence. But, learning rate should decrease as the number of epochs increases to make sure that the loss will reach to its minimum value.
5. Number of epochs. The training shouldn't be stopped until the loss reaches its minimum value.
6. Optimizer. Stochastic gradient descent (SGD) and adaptive moment estimation Adam [46] are two broadly used techniques to fasten the convergence of loss function.

2.4. Brain Tumor Segmentation Models Using CNN

Many CNN based systems have been developed to tackle the brain tumor segmentation problems in the past few years. U-net [15] is widely used in developing CNNs for image segmentation. It consists of a contracting path to extract features and an expanding path with skip connection to perform precise localization and classification. The success of U-net on medical image segmentation is attributed to its skip connections. Since precise medical image segmentation requires fine-grained details which is lost due to the max-pooling operations in the contracting path, those feature maps generated before the max-pooling operations are brought in by means of skip connections to provide details for the expanding path. Nevertheless, U-net is designed for a general purpose of image segmentation, to precisely segment 3D brain images, the U-net structure has been adapted with different variations.

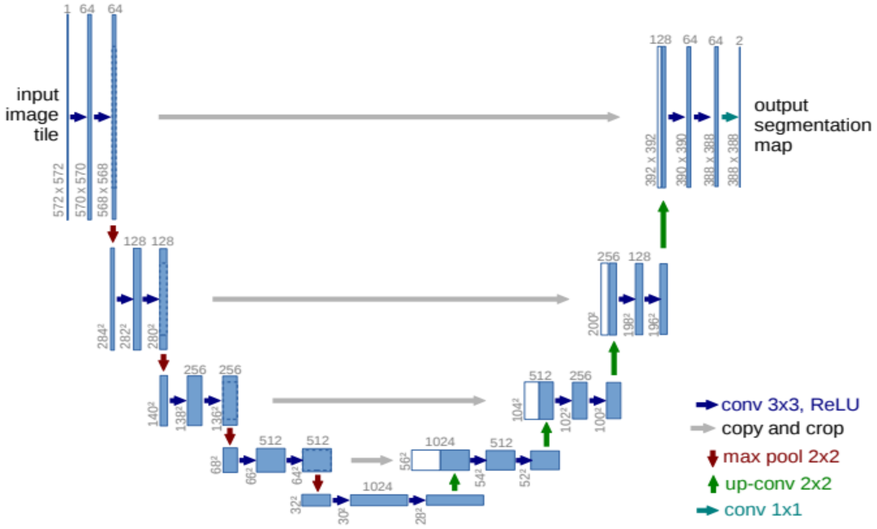

Figure 2.11 Architecture of U-net [15].

As the 3D brain images are often seen as sequences of 2D slices, many segmentation systems involve 2D U-net based CNNs, of which the input and ground truth data are 2D samples. There are also 3D CNNs, operating with 3D input samples, in order to use the inter-slice feature information. In the implementation of such 3D CNNs, one of the challenging issues is to get training samples of sufficient quality and quantity. In some cases, the limitation of computation resources, including memory usage, can also be an issue. To bypass such issues, varieties of pseudo-3D CNNs have appeared. Some used the combination of 2D and 3D convolutions where 2D ones were for extracting intra-slice features, while 3D ones were for extracting inter-slice features [16]. Others sliced each 3D input image into three orthogonal axes to obtain 3 sequences, i.e., coronal, sagittal and axial, then performed 2D convolutions on each axis in order to capture spatial information with low dimensional convolutions [17]-[20]

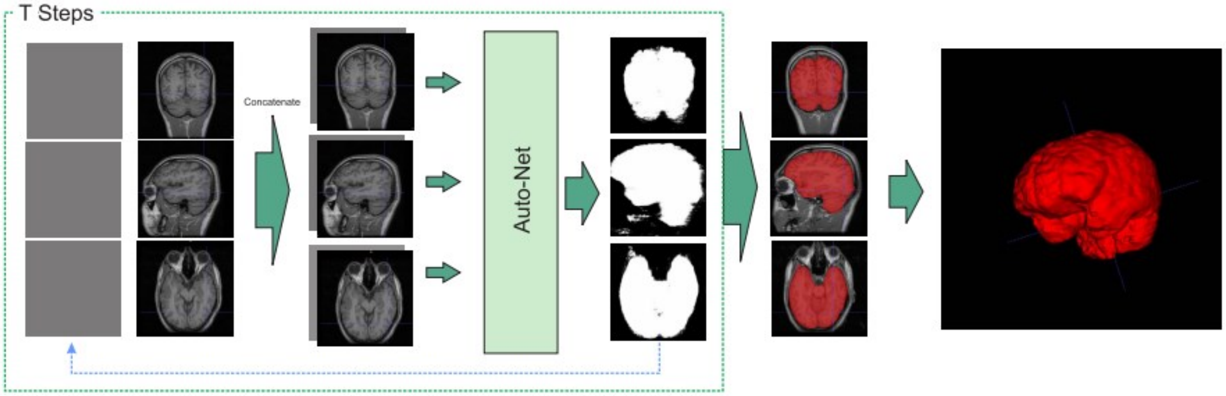

Figure 2.12 Architecture of the system in [17]. Three 2D CNN networks are applied separately to three orthogonal axes. The final output is based on the segmentation results from three axes.

To enhance the computation in convolutional layers, in many U-net based CNNs, multi-convolution blocks were used as the basic computational units. For example, a standard convolution layer was replaced by a residual block [20], [21], [27], inception block [22], [23], or dense block [16], [24]-[26]. In some U-net designs, the skip connections were enhanced by employing attention gates [20], [27], or squeeze-and-excitation blocks [26], [28].

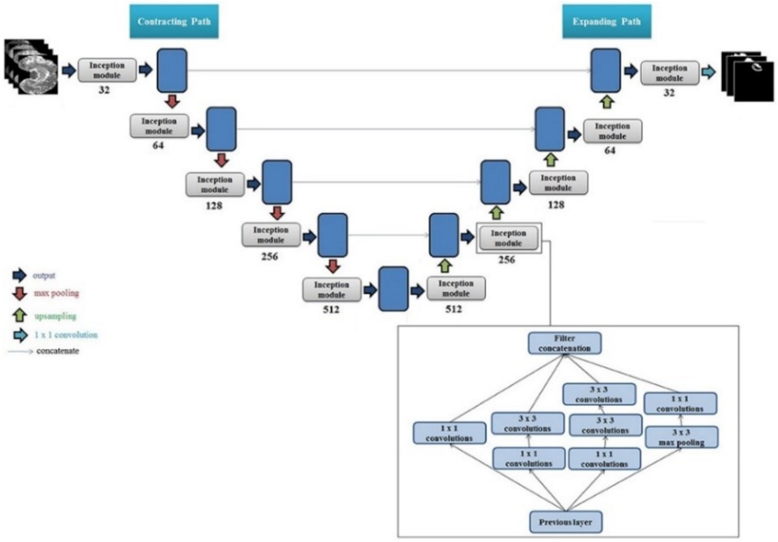

Figure 2.13 U-net with Inception Module [23].

In the case of Unet++ [29], skip connections were redesigned to aggregate features of varying semantic scales at the decoder sub-networks, leading to a highly flexible feature fusion scheme.

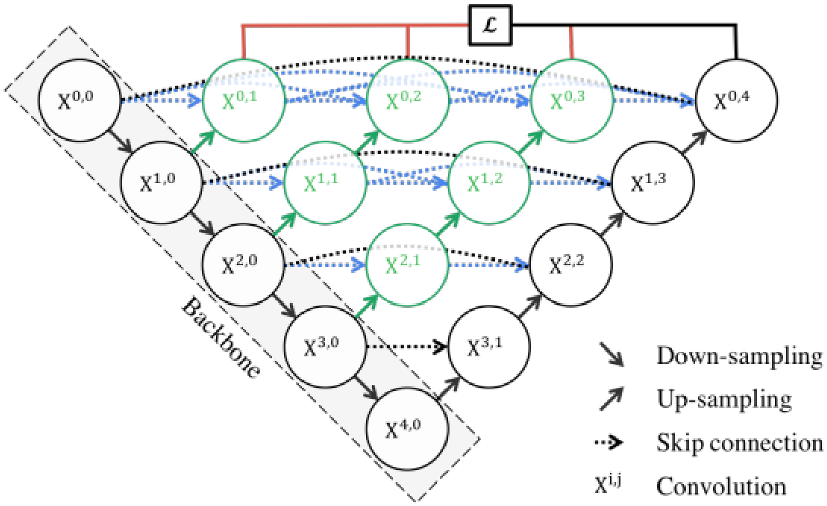

Figure 2.14 Architecture of U-net++ [29].

The variations of U-net can also be in network structure. In order to perform precise classification, the feature data generated from the feature extraction block must carry critical information of pathological tissues. As a result, many tried to modify the contracting path to optimize the feature extraction process. For example, one could use multiple contracting paths in order to process the input data of different modalities. To handle the input data of each modality independently, instead of single one in the original U-net, four contracting paths were used to extract feature information from the individual modality in order to avoid the false-adaptation between the modalities [30]. There's also a U-net with 2 contracting paths to extract features from flair-t2 pair and t1ce-t1 pair respectively in order to capture the discriminative patterns between brain tumor areas and normal brain areas [31]. Nevertheless, as each modality has its own enhanced features for certain tumor patterns and is correlated one another, the four contracting paths will inevitably miss the important cross-modality features and the two contracting paths will miss the feature data of mono-modality.

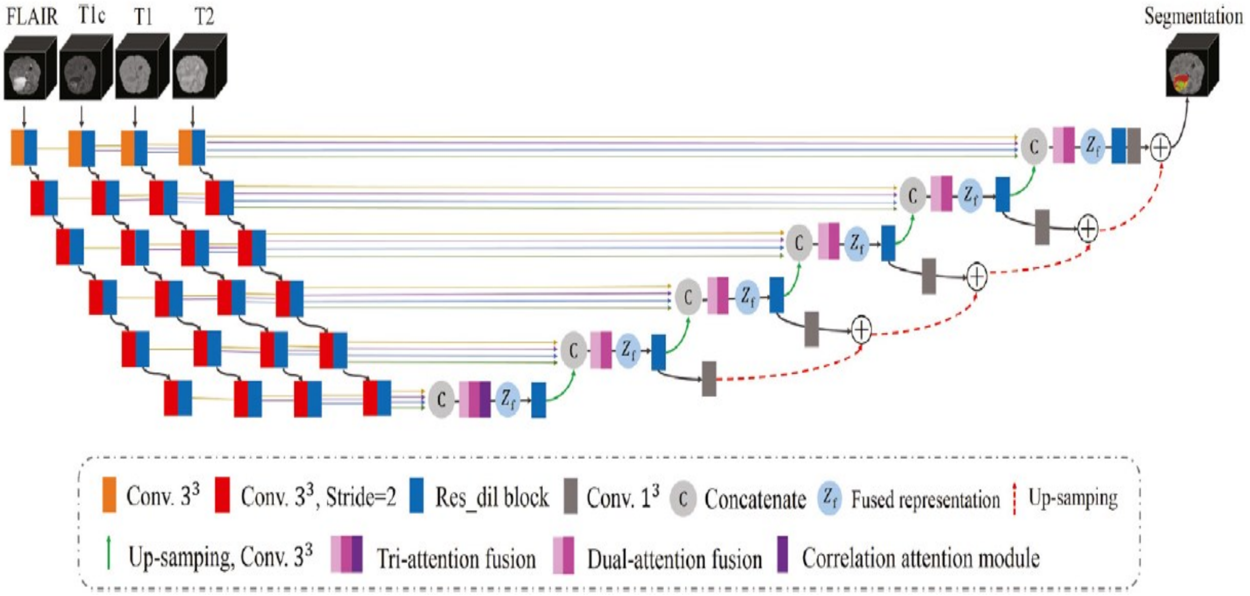

Figure 2.15 Architecture of a four contracting paths U-net [30].

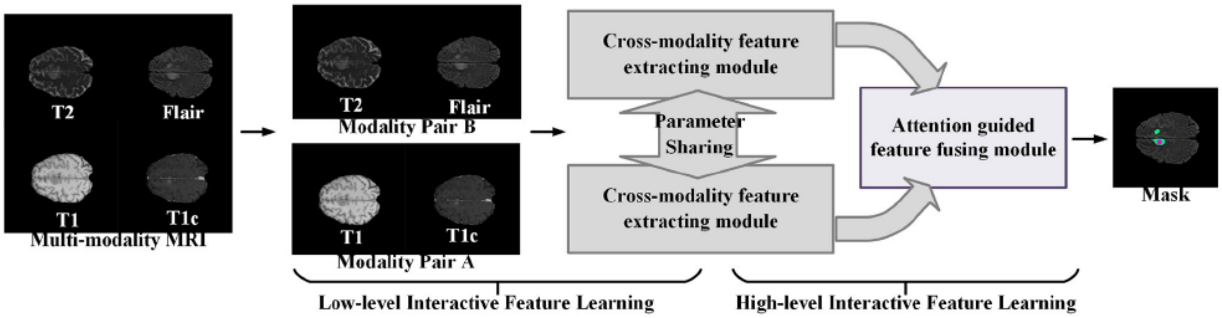

Figure 2.16 Architecture of a two contracting paths U-net [31].

Since the brain tumor detection is a fine classification of pixels in a hierarchical structure of tumor sub-regions, multiple U-nets architecture was used where the result of the former U-net was utilized as the prior information for the next U-net to perform finer segmentation [32]-[35]. For example, in [32], three independent U-nets were used to detect WT, TC and ET, successively. The input image will be cropped based on the segmentation results from previous U-net. This can remove those regions of non-interest and greatly reduce the computation burdens in the following U-net.

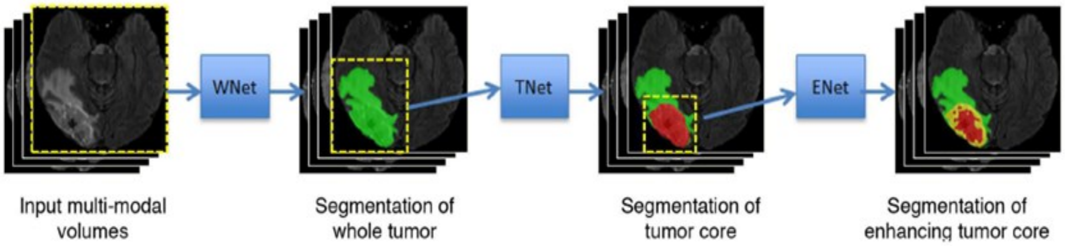

Figure 2.17 Flowchart of three independent cascaded U-net [32].

Nonetheless, this cascade strategy results in more training time and computation cost since there are three independent U-nets needed to be trained. To reduce the training time and computation cost, one could also cascade multiple U-nets into a single network to reduce the training time and computation cost [33]-[35]. In [33], there's a two-stage cascaded U-net structure where the first U-net was used to segment the WT area, the segmentation results of WT were used as an additional input information for the second U-net to predict the TC and ET area.

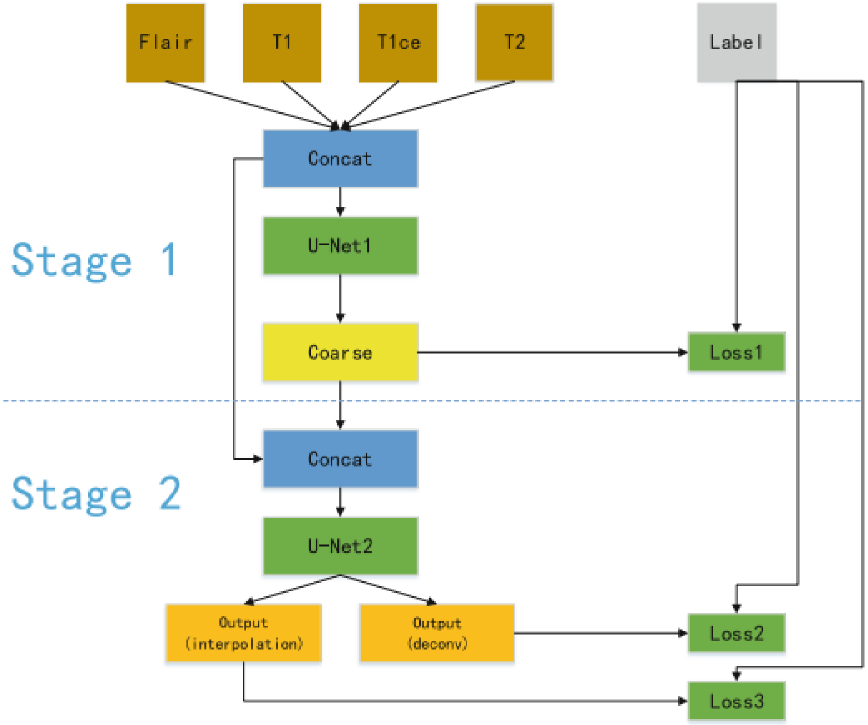

Figure 2.18 Architecture of 2-stage cascaded U-net [33].

The CNN systems presented in [34] and [35] also cascaded multiple U-nets into a single network. Unlike network reported in [33] where there're actually two U-nets in total, networks designed in [34] and [35] both cascaded three-U-nets into a single network.

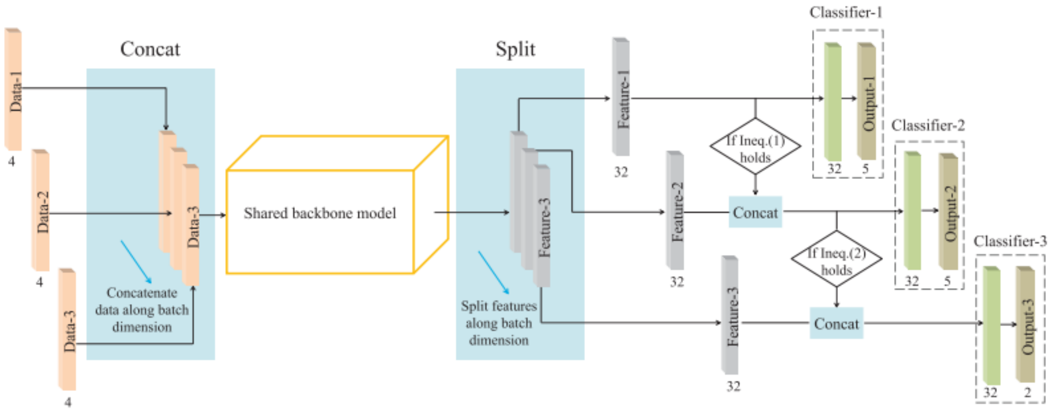

Figure 2.19 Architecture of OM-Net [34].

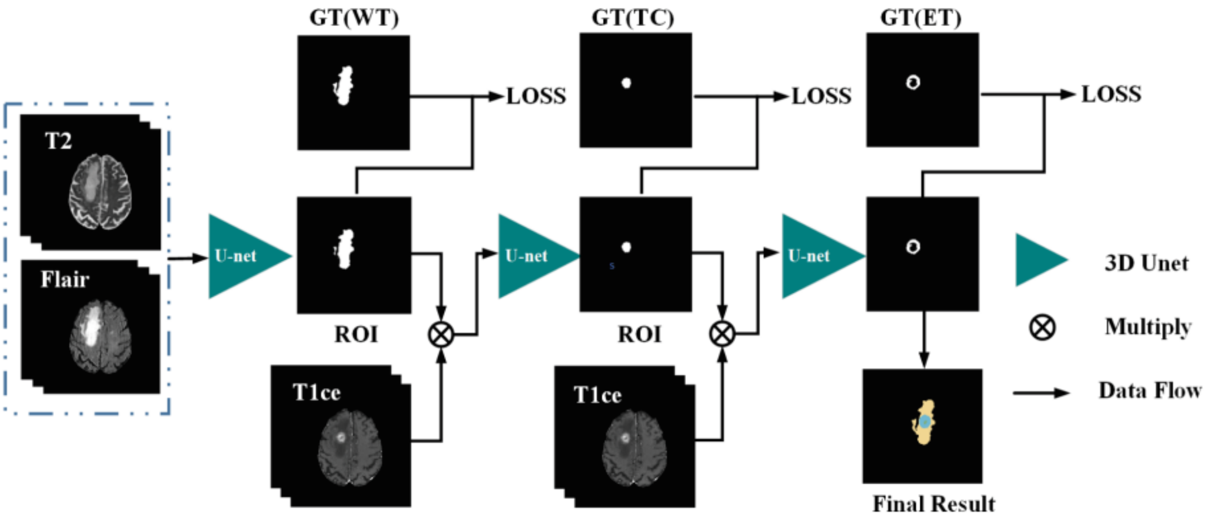

Figure 2.20 Architecture of multi-step cascaded U-net [35].

2.5. Summary

In this chapter, the basics of knowledge-based filtering operations and convolutional neural networks are presented. Some CNN systems for brain tumor segmentation published in reputed journals in recent years are also described to illustrate the latest development in this topic area. Most of existing CNNs for brain tumor segmentation are constructed based on U-net structure and have resulted in a much higher segmentation quality compared to that given by knowledge-based filtering systems. However, the improvement in the quality is often at the expense of large computation resources. In particular, among different CNNs for brain tumor segmentation, one attempts to gain more in quality by putting more complex convolution layers. The heavy computation loads limit the application of such systems in a computation resource restricted environment and results in a risk of performance inconsistency. Therefore, the work presented in this thesis aims at custom-designing a light-weight CNN system that is able to deliver high-quality brain tumor segmentation results in a consistent manner at very low computation cost.

3. Proposed system

3.1. Introduction

A computer vision system for medical applications must be made to operate in a reliable manner. The reliability, involving the performance consistency and result reproducibility, is not considered as a strong point of current neural networks. Also, CNN systems often require a lot of computation, which may limit the applications of CNN systems for medical signal processing. The objective of the work presented in this thesis is to develop a CNN system for brain tumor segmentation that is able to perform high-quality processing in a consistent manner, while requiring a very low volume of computation. The approach to the objective is custom-designing the CNN structure to maximize the computation efficiency and performance consistency.

The input of the proposed system consists of four 3D MRI images, corresponding to the 4 modalities, namely Flair, T2, T1ce and T1. Each 3D image is sliced into 2D slices. Thus, the voxels in 3D images become pixels in 2D space. The output is a sequence of slices, in which each pixel is labeled to be in one of the 4 classes, i.e., ED, NET, ET and the background. The proposed system consists of the following three parts.

1. Pre-processing. It is to exclude the excessive margins and evident tumor-free slices, which reduces the regions of non-interest and the data volume to be processed.
2. CNN. It's the main part of the proposed system to perform brain tumor segmentation. It involves a multi-path feature extraction block to exploit the comprehensive feature information from the multi-modality input, a bottleneck to perform tumor localization and coarse segmentation, and, a multi-branch classification block to optimize the segmentation.
3. Refinement block. It aims at correcting falsely labeled enhancing tumor pixels to improve segmentation quality.

The pre-processing block is presented in Sub-chapter 3.2. Sub-chapter 3.3 is dedicated to elaborating the CNN in detail, The refinement block is given in Sub-chapter 3.4. Network configuration is described in Sub-chapter 3.5. A summary is given in Sub-chapter 3.6.

3.2. Pre-processing Block

The pre-processing block is placed to facilitate the operations in the following CNN block. It is to reduce the volume of data to be processed in the CNN, by removing some of the regions that are apparently out of the regions of interest, from the 4 series of slices in each patient case. As a result, the information density will be increased. Two kinds of regions of non-interest are targeted.

- Excessive margins in each slice. In terms of samples from BraTS datasets where the size of each image slice is 240×240, there are many background areas where no regions of interest are found. To reduce some evident regions of non-interest, one need to calculate the boundary of brain area in each slice. The leftmost point, the rightmost point, the uppermost point and the lowermost point of the brain area are 25, 225, 34 and 202 respectively. Therefore, one can crop each slice to reduce its dimensions from 240×240 to 200×168 pixels, cutting off more than 40% of the input data for the CNN, without losing any pixels in the brain areas.

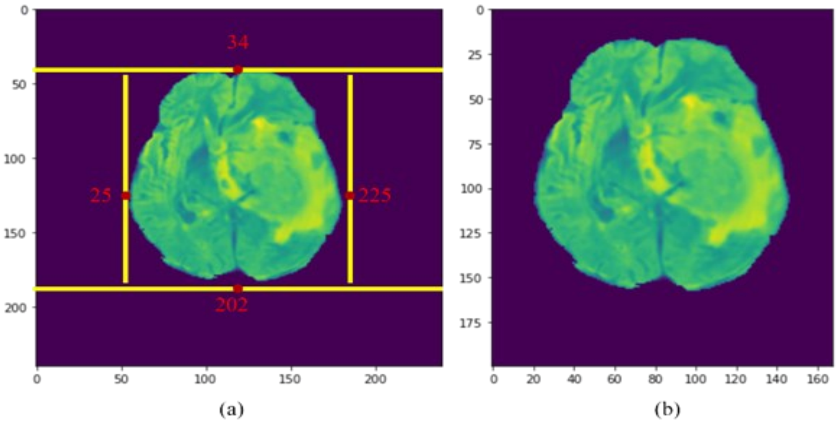

Figure 3.1 (a) Example of the original input slice. (b) Cropped slice.

- Tumor-free slices located in the two ends of each sequence. It is known that the location of brain tumors is variable, but they are hardly found in the two ends of each sequence. In the case of Brats samples, the first fifteen slices and the last twelve slices do not contain any tumor areas and are thus removed. As a result, the total slices in each sequence are reduced from 155 to 128.

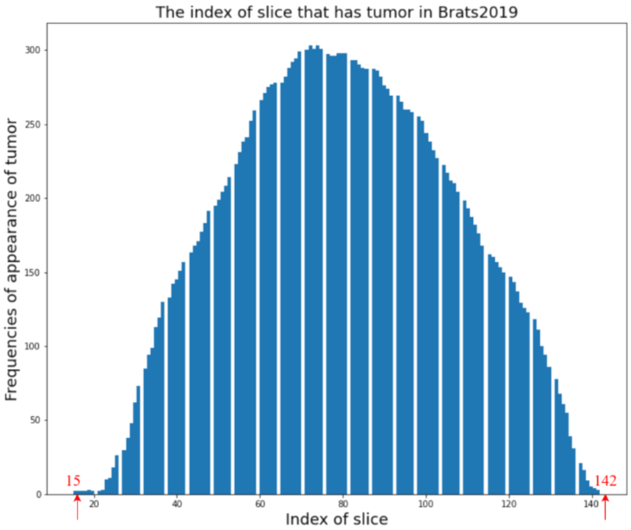

Figure 3.2 Histogram of the index of slices that has tumors in BraTS 2019.

The removal of the 2 kinds of regions of non-interest by the pre-processing block results in a significant data reduction. In terms of BraTS samples, for each patient case, the volume of the input data is reduced from 240×240×155×4 to 200×168×128×4， resulting in the total data reduction of 52% in terms of pixels, which helps not only to reduce the computation burden in the succeeding CNN, but also to increase the information density of its input data.

3.3. CNN

The CNN is the major part of the system for the brain tumor segmentation proposed in this thesis. The emphasis in the design of the network is on the computation efficiency and processing quality, i.e., maximizing the processing quality while minimizing the computation cost and randomness, which results in a performance consistency.

The detailed architecture of the proposed CNN is shown in Figure 3.3. It consists of the following three blocks.

1. Feature extraction (FE) block. It has three-path convolutions, which are dedicated to extracting different feature data from the four-modality input.
2. Bottleneck. It is to generate feature maps representing tumor locations of low resolution.
3. Three-branch classification block. Each of the branches is made to identify the pixels of one specific kind of tumor areas.

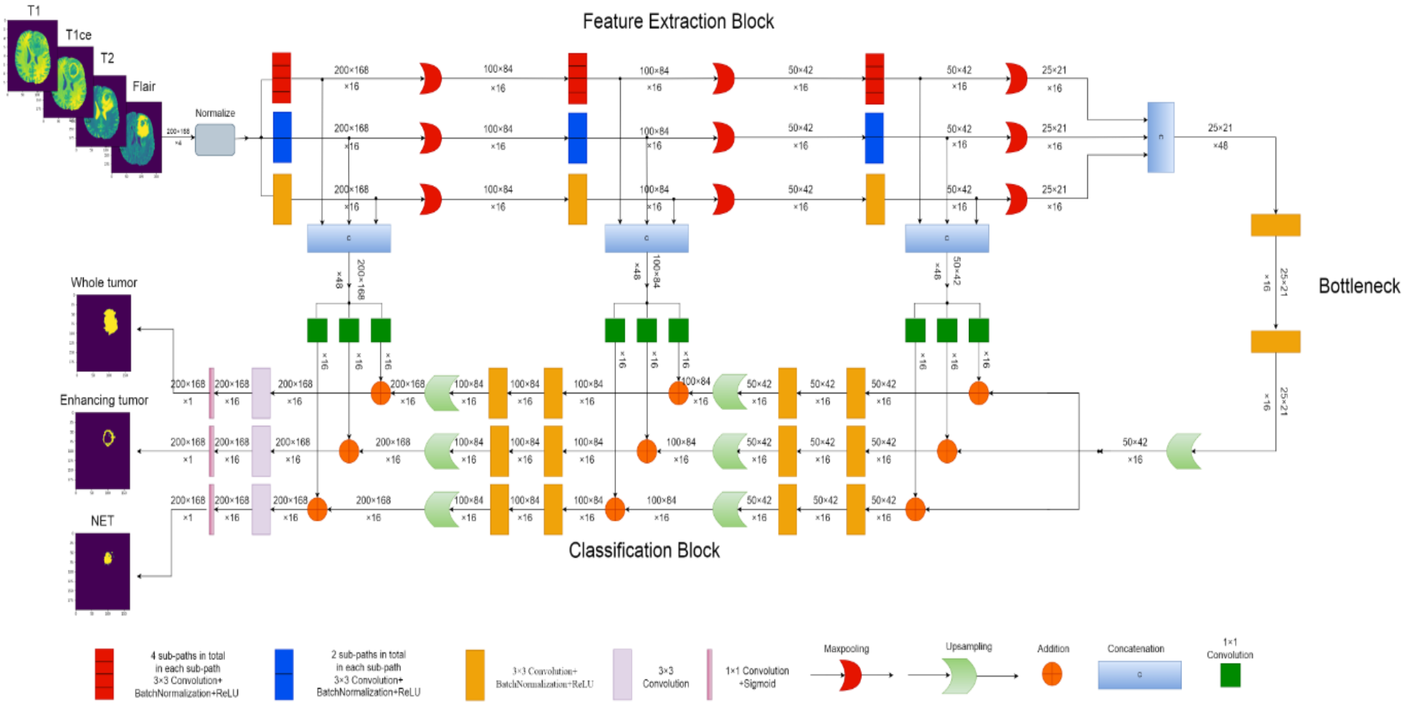

Figure 3.3 Diagram of the proposed CNN.

3.3.1. Feature Extraction Block

The function of the FE block is to extract comprehensive image features from 2D MRI image slices of the 4 modalities, namely Flair, T2, T1ce and T1. A set of 4 slices from the 4 modalities is illustrated in the first row of Figure 3.4. The brain images are acquired by the 4 modalities to enhance various kinds of pathological tissues to facilitate diagnosis, and the intensity range of different modalities may not be uniformed. Hence, the input data need an appropriate normalization in order to facilitate the following filtering operations.

The aforementioned normalization will be applied to the 3D data of each modality. The mean and standard deviation are calculated from the pixels in the brain areas of all input slices, and those in the background with no region of interest are excluded. As a result, the mean and standard deviation obtained can better represent the statistic distribution of data within the brain area, and consequently, certain features in the brain areas can be enhanced and the contrast of image can be improved as shown in Figure 3.4.

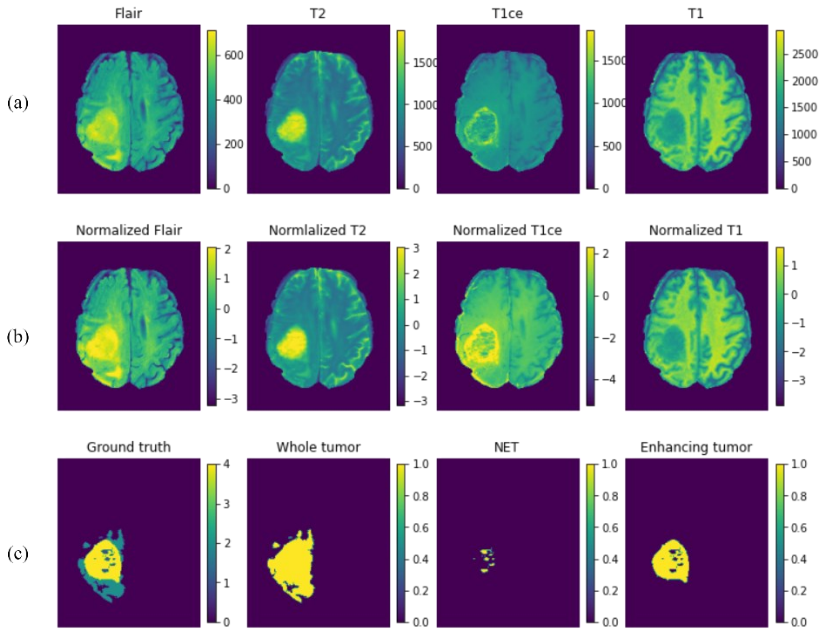

Figure 3.4 (a) Input slices of the 4 modalities, (b) normalization results, and (c) ground truth images.

The purpose of the comprehensive feature extraction is to capture the data representing critical pathological patterns from brain images. Considering the different emphasis on the varieties of tissues, the input data of each of the modalities need to be filtered individually in order to make good use of their feature characters. For example, T1ce has enhanced features for the enhancing tumor. However, as the data of all the 4 modalities are acquired from the same source and therefore correlated, they also need to be processed collectively.

It is known, on one hand, that the patterns of the whole tumor region are distinguishable in Flair and T2, and one can get more whole tumor features by means of convolutions applied to these 2 modalities combined. On the other hand, the pair of T1 and T1ce display more differences between the patterns in tumor core and those in the rest of brain areas, and filtering them together can produce high-quality feature data representing tumor cores and enhancing tumors.

Taking the above-mentioned points into consideration, the FE block is designed with 3 paths, as shown in Figure 3.5. The input data of all the 4 modalities are applied to all of the 3 paths, and get convolved 3 times in each path. The input data are organized, however, into 3 different groupings so that the 3 paths can produce the following 3 kinds of features.

- Mono-modality features. The upper path of the FE block has 4 sub-paths, in each of which the input data of mono-modality are processed independently in the 3 successive layers of convolution and max-pooling operations. Each of the convolutions has 4 kernels, producing adequately 4 data maps per modality. The 4 sub-paths generate, in total, 16 mono-modality feature maps from the input data of Flair, T2, T1 and T1ce.

- Paired-modality features. In the mid path, there are 2 sub-paths, where the input data of Flair-T2 are applied to one sub-path for whole tumor features, and those of T1-T1ce are applied to the other for tumor core and enhancing tumor features. Each convolution has 8 kernels, considering that it involves the data of 2 modalities. There are 16 feature maps generated by the 2 sub-paths.
- Cross-modality features. There is no separated sub-path in the lower path of the FE block. The 3 successive convolution and max-pooling operations are applied to the data of all the 4 modalities combined. This path produces 16 cross-modality feature maps.

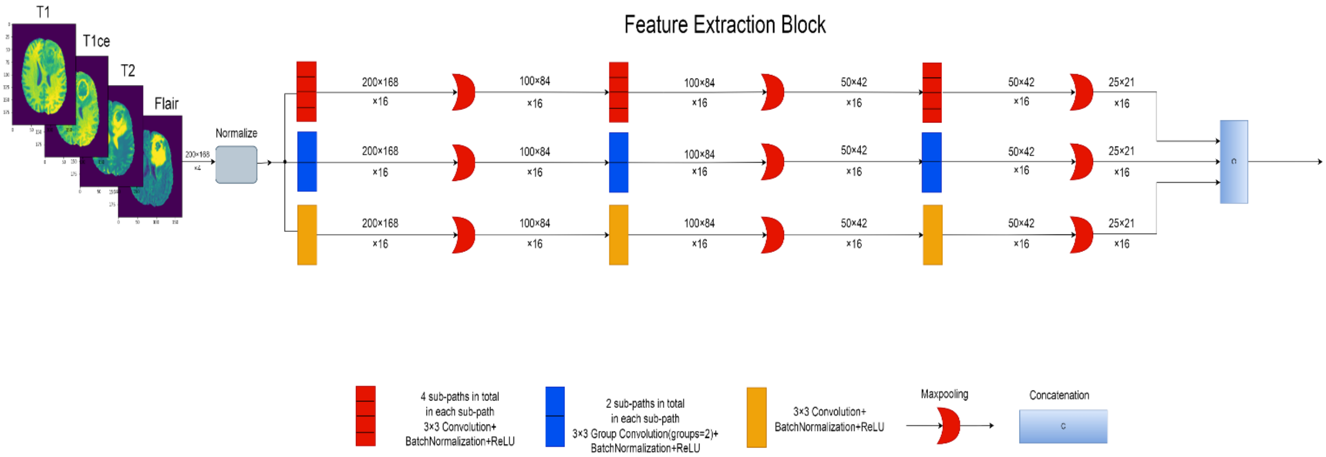

Figure 3.5 Diagram of FE Block.

Accurate brain tumor segmentation requires each pixel to have relatively large receptive field. Increasing the depth of CNN can meet this end at the expanse of large computation volume and at the risk of model overfitting. Therefore, a max-pooling operation is applied after each convolution in order to enlarge the receptive field and increase the information density without introducing additional parameters. Nevertheless, max-pooling reduces the resolution of the feature maps and may lose the location information of tumor area. The 3-path feature extraction block generates, in total, 48 maps of low-resolution (less than 1000 pixels per map), representing, comprehensively, features of cross-modality, individual modalities, and Flair-T2 pair & T1-T1ce pair.

3.3.2. Bottleneck

As shown in Figure 3.3, the bottleneck is located between the 3-path FE block and the 3-branch classification block. Its function is to transform the input data, i.e., 48 low-resolution feature maps, into data maps indicating the potential tumor locations in order to perform coarse segmentation.

The function of this block is implemented by means of two simple convolution layers. After three convolution and max-pooling operations, each feature map in the bottleneck contains a lot of semantic information. Therefore, the number of kernels per layer is modestly 16 to minimize the risk of introducing randomness, with a view to achieving a good consistency of the network's performance. Since the input data maps are of low resolution and convolutions do not increase the resolution. The output maps can carry the information of potential tumor locations, without details of tumor areas. The information will be used in the classification block, presented in the following sub-subchapter.

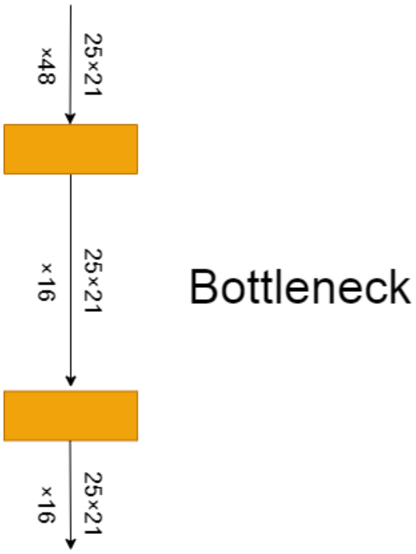

Figure 3.6 Diagram of Bottleneck

3.3.3. Classification Block

The classification block is to classify all the pixels into 4 classes, i.e., WT, ET, NET, and the background. The prime input of this block is a set of low-resolution data maps produced by the bottleneck.

The basic processing elements in this block, shown in Figure 3.7, are convolution, upsampling and skip connection. The purpose of the upsampling, by means of bilinear interpolation, is to restore, gradually, the initial image dimension. Because the fine classification needs detailed image information which is lost due to the three max-pooling operations in the feature extraction block, the feature maps produced before each maxpooling operation are brought in by means of the skip connections. They are scaled with learnable coefficients, and then used to modulate the upsampled data maps. Two layers of convolutions are employed in the following to fine-tune the modulated signal in order to get precise segmentation result.

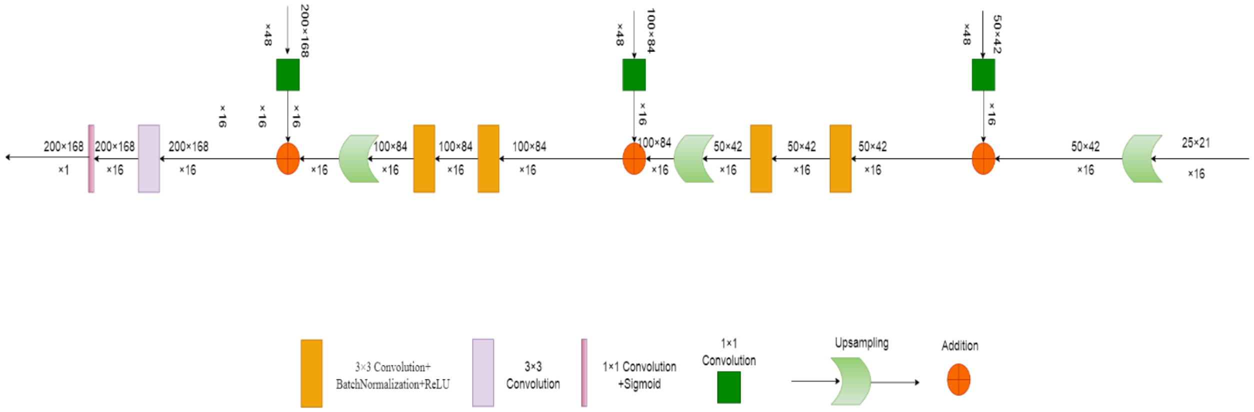

Figure 3.7 Basic computation block in Classification Block. The input is from the right.

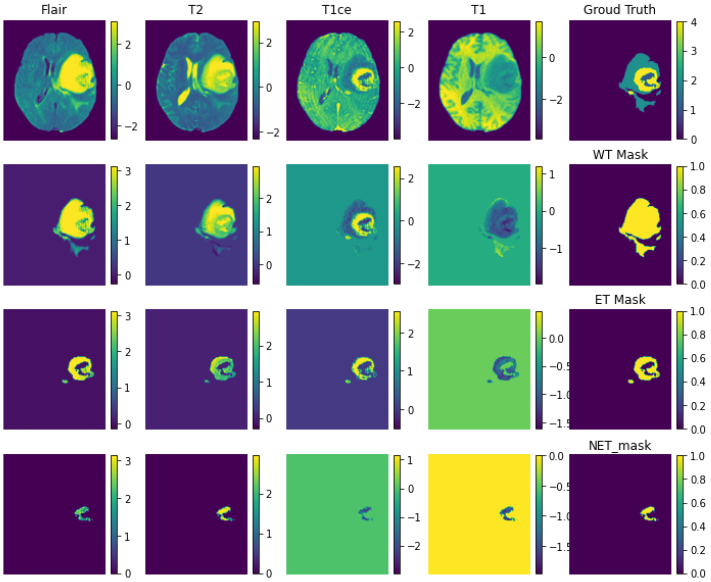

Figure 3.8 Different tumor areas have different patterns.

As mentioned earlier, there are 3 kinds of tumor areas needed to be identified, i.e., WT, ET and

NET. They are different in sizes, locations, textural patterns, and intensity ranges. The areas of whole tumor, enhancing tumor and NET in a slice at a certain position of a 3D image of a patient case are illustrated in Figure 3.8 as examples. By examining images of different patient cases, one can have the following observations.

1. Whole tumor areas include various pathological tissues. They have relatively large dimensions. Although the texture patterns have a lot of variations inside the areas, they are different from those outside the areas. Identifying whole tumor areas from a brain image is essentially to distinguish the patterns of pathological tissues from those of the normal brain structures.
2. The other kinds of tumor areas, namely ET, NET and edema, have patterns different from one another. If a set of filters is trained to detect one kind of tumor areas, it may not be suitable for the others.

In light of above observation, to precisely classify the pixels in the 3 different kinds of tumor areas, different criteria to differentiate image patterns are needed, which may not be easily done by a single filtering path. Hence, the classification block of the proposed system is designed to have 3 branches in parallel, as shown in Figure 3.3, dedicated to identifying the pixels in the whole tumor, enhancing tumor, and NET areas, respectively. Each branch is specifically made to differentiate the particular patterns of the certain tumor areas and those in the rest of the brain image. As a result, a complex multi-class classification task is decomposed into several simple binary classification tasks.

The three branches are configured identically, and each has three layers. In each layer, the upsampled data maps are modulated by the feature maps of the same dimensions from the feature extraction block. To optimize the modulation result, the feature maps are appropriately scaled by means of 1x1 convolutions. The modulated data maps are then convolved twice in each layer to perform a precise detection of targeted tumor area.

In order to make each of the 3 branches capable of identifying one specific kind of tumor areas, they are trained independently with the binary masks of the whole tumor, enhancing tumor and NET, respectively. As a result, the separate supervision allows the filtering coefficients to be adjusted properly to suit its own pattern detection.

In the final layer of each branch, there will be a 1x1 convolution with a Sigmoid activation function, generating one feature map which indicates the areas of whole tumor, enhancing tumor and NET, respectively. The Sigmoid function is defined as:

$$y = \frac{1}{1 + e^{-x}} \tag{3.1}$$

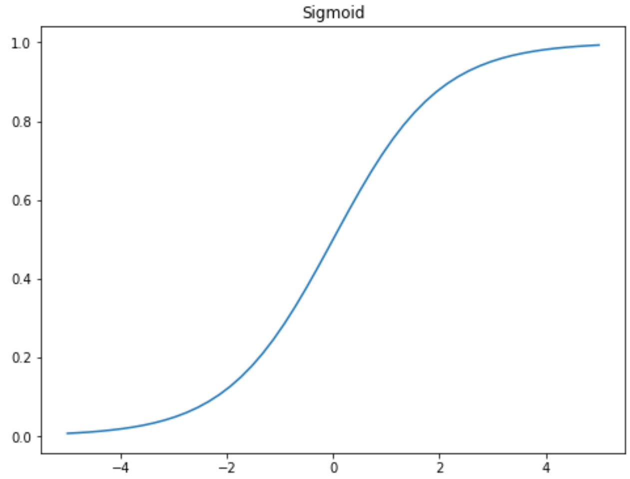

Figure 3.9 Curve of Sigmoid function.

The Sigmoid function will squash the values between 0 and 1. Since each branch is performing a binary pixel-wise classification problem, those pixels labeled with 1 indicate the target tumor area, and the rest indicate the non-target area.

3.4. Refinement Block

The final output mask is actually a 3D image, where the 2D maps are concatenated together. Thus, the pixels in the 2D image become voxels in the 3D image. Since the final output requires each voxel to be labeled to one of the following classes, i.e., ED, ET and NET instead of WT, ET and NET, the refinement block is firstly to assign the new labels to the voxels based on the output from CNN, and secondly to verify if any of the voxels placed, by the CNN, in the ET areas are falsely classified.

1. Reassign the labels. The CNN outputs three maps, indicating WT (consisting of ED, ET, NET), ET and NET, but it’s required that each voxel to be classified as one of the 4 categories, i.e., ED, ET, NET and background. The label reassignment is based on the following principles. (i) If there is no overlap between the WT area and the ET or NET area, then those voxels classified as ET or NET will be relabeled as background. (ii) If a voxel is labeled as 1 by WT classifier, but also labeled as 1 by NET and ET classifier, then the label of this voxel is dependent on the comparison between the output probability of NET and ET classifier. If the output probability of NET classifier is larger than that of ET

classifier, then this voxel will be labeled as NET, and vice versa.

2. Correct the falsely classified ET areas. As a tumor area is a 3D object, the voxels of the area are expected to appear in a certain number of slices and the total number of these voxels should be above a certain threshold. As shown in Figure 3.10, if a patient case has ET, then the ET area must appear in at least 6 slices. Therefore, if the ET area appear in less than 6 slices or the total number of the voxels in the area is less than 1000, then the voxels are considered as misclassified and the area will be relabeled as NET.

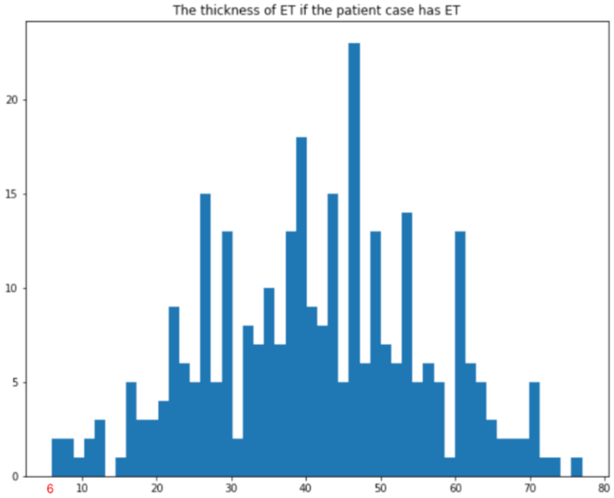

Figure 3.10 Histogram of the thickness of enhancing tumor. If a patient has enhancing tumor, then the minimum number of slices of ET should be six.

The simple yet efficient refinement process can increase the mean Dice scores of ET area about 2.5% on both BraTS 2018 and BraTS 2019 validation datasets.

3.5. Network Configuration

The network configuration is shown is Table 3.1. The total number of parameters is only 61,843, which is merely a small percentage of that of many existing CNNs in the reported literatures. Based on the analysis of the complexity of brain tumor segmentation, the network configuration, i.e., the number of kernels per layer and the number of layers, is carefully designed and just-sufficient for delivering high processing quality.

With such simple configuration, it can be expected that the proposed system only demands for a very low computation volume both in training and testing. Even in a computation resource limited environment, the proposed system is expected to be implementable and applicable.

Table 3.1 Network Configuration.

Layer	**Kernel Size**	**Input Size**	**Output Size**	**Kernel Number**	**Parameters**
1	3×3	200×168	100×84	48	1,248
2	3×3	100×84	50×42	48	4,272
3	3×3	50×42	25×21	48	4,272
4	3×3	25×21	25×21	16	6,992
5	3×3	25×21	25×21	16	2,384
6	3×3	25×21	50×42	48	7,152
7	3×3	50×42	50×42	48	7,152
8	3×3	50×42	100×84	48	7,152
9	3×3	100×84	100×84	48	7,152
10	3×3	100×84	200×186	48	6,960
11	1×1	200×186	200×186	3	51
SC1	1×1	25×21	25×21	48	2,352
SC2	1×1	50×42	50×42	48	2,352
SC3	1×1	100×84	100×84	48	2,352
Total					**61,843**

*SC stands for Skip Connections.

3.6. Summary

In this chapter, the detailed description of the proposed system has been presented. It consists of three parts:

1. Pre-processing block is to remove the excessive margins and evident tumor-free slices to reduce the input data volume and increase the information density.
2. CNN block is the main part of the proposed system, dedicated to performing accurate brain tumor segmentation. It is designed with two unique characters: (i) the feature extraction block has 3 paths and each of them is to extract comprehensive features of mono-modality, paired-modality and cross-modality, respectively, from the multi-modality input data, and, (ii) there are three branches in the classification block, which decomposes a complex multi-class classification problem into three simple binary classification tasks. This structure enables an appropriate training of these branches so that each of them can be adjusted effectively to suit the classification of the pixels of the specific kind of tumor areas.
3. Refinement block is to reassign the labels based on the output and remove the falsely classified ET pixels to improve the segmentation accuracy.

The network structure, comprising the above-described blocks, has very simple configuration, and it requires only 61,843 trainable filtering coefficients. It is expected to have an easy training process and requires an amount of computation much smaller than most of the existing designs. In the next chapter, the performance of the proposed system will be evaluated.

4. Performance Evaluation

4.1. Introduction

The performance evaluation of the proposed system has been conducted on the datasets of BraTS 2018 and BraTS 2019. A good number of experiments have been carried out to assess the quality of the proposed system in different aspects.

In this chapter, the datasets and the training details are described in Sub-chapters 4.2 and 4.3 respectively. In Sub-chapter 4.4, the evaluation metrics are presented. The consistency study and the ablation study of the proposed system are presented in Sub-chapters 4.5 and 4.6, respectively. The comparison between the proposed system and those recently reported in reputed journals is shown in the Sub-chapter 4.7, followed by a summary in Sub-chapter 4.8.

4.2. Dataset

BraTS 2019 and BraTS 2018 are used to train the system and also for the performance evaluation. In BraTS 2019, there are 335 patient cases in the training pool and additional 125 cases for validation. In BraTS 2018, the number of cases is much smaller and all of them are found in BraTS 2019. The assessment of the validation results is done by means of CBICA Image Processing Portal [36], [37], as the ground truth data are not available in the validation datasets.

A simple data augmentation, by flipping and rotating, is applied to the training samples to increase their number for a better training. Moreover, the training samples are shuffled in each epoch so that they are rearranged differently in batches.

Table 4.1 Number of training and validation samples of BraTS 2019 and BraTS 2018 dataset.

	Training samples	Validation samples
BraTS 2019	335	125
BraTS 2018	285	66

4.3. Implementation Details

The hyperparameters determine, to some extent, how the filtering coefficients of the proposed CNN are derived. For the training process, the most important hyperparameters are as follows.

1. Batch size. There are 335 patient cases for training, i.e., 335 * 155 slices per modality.

Taking this number of slices and the limit of GPU memory usage into consideration, each batch is made to have 128 slices.

2. Number of epochs. The training requires only 75 epochs for the loss value to reach its minimum level.
3. Learning rate. It is variable with staircase decay. The initial learning rate is 0.005 and will be reduced by half every 15 epochs.
4. Loss function. Dice loss is applied to calculate the loss in the WT branch. For NET and ET branches, Tversky loss [38] is chosen to reduce the risk of divergence caused by small-size targets. In this paper, α is chosen to be 0.7 and β is chosen to be 0.3.

$$Tversky\ Loss = \frac{TP}{TP + \alpha FN + \beta FP} \tag{4.1}$$

5. Optimizer. Adam [46] is chosen as the optimizer for the training process.
6. Kernel initialization. The kernel coefficients are initialized with a uniform distribution by he_uniform [47].

Tensorflow is used to implement the proposed system. The total training time is 7 hours for BraTS 2018 dataset and 9 hours for BraTS 2019 dataset with NVIDIA Tesla P100 GPU.

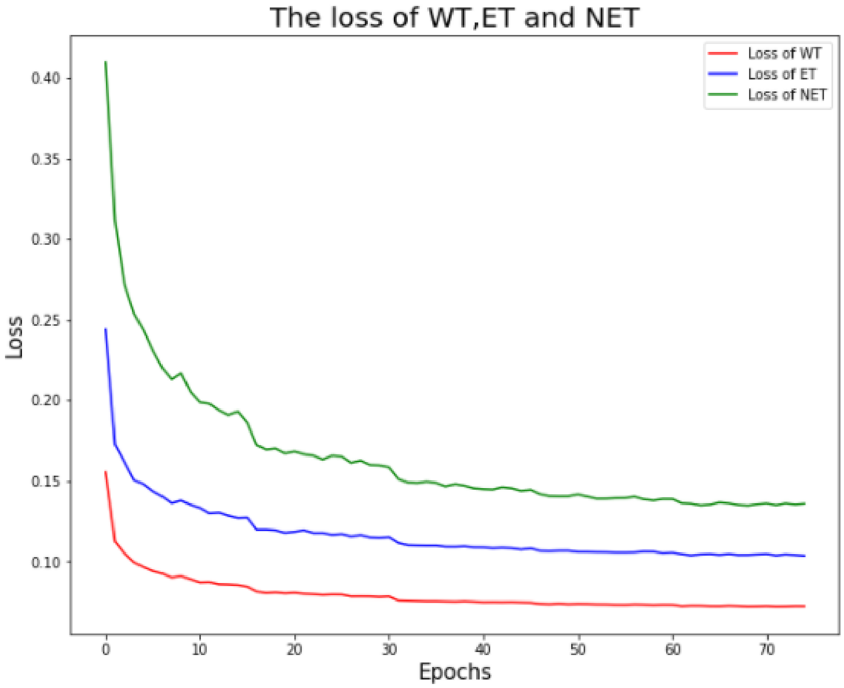

Figure 4.1 Curve of the loss of WT, ET and NET.

4.4. Evaluation metrics

The segmentation result is mainly evaluated by 4 metrics: Dice score, Sensitivity, Specificity and Hausdorff95. Supposing P_0 and P_1 are respectively the predicted result of tumor-free and tumor regions, T_0 and T_1 are the ground truth of tumor-free and tumor regions. Then, the first three metrics can be defined as:

$$Dice(P_1, T_1) = \frac{P_1 \cap T_1}{\frac{(P_1 + T_1)}{2}} \tag{4.2}$$

$$Sensitivity(P_1, T_1) = \frac{P_1 \cap T_1}{T_1} \tag{4.3}$$

$$Specificity(P_0, T_0) = \frac{P_0 \cap T_0}{T_0} \tag{4.4}$$

Hausdorff95 is 95th percentile of the maximum distance of a point on the predicted result to the nearest point on the ground truth. Dice score is the most important and comprehensive metric for evaluating brain tumor segmentation result. It measures volumetric overlap between segmentation results and ground truth.

4.5. Consistency Study and Test Results

Due to certain randomness in training a CNN system, one cannot expect such a system to reproduce the exactly same results after each training. Nevertheless, since the proposed system is developed for a medical application, a good reproducibility is required for it to be usable in practice. Hence, it is critical to measure the consistency of its performance to test the reliability of the result.

Ten experiments are conducted, using each of the two datasets, i.e., BraTS 2018 and 2019, to measure the consistency, and the results, i.e., the mean and median Dice scores produced in each experiment, are presented in Tables 4.2 and 4.3. From the mean Dice scores, presented in the 2nd section in each of the two tables, one can have the following observations concerning the performance consistency.

- The system yields high Dice scores in each experiment. For example, the mean Dice scores of ET in ten experiments are ranged from 0.782 to 0.792 on BraTS 2018 validation dataset, and 0.745 to 0.767 on BraTS 2019 validation dataset. In terms of WT, the mean Dice scores are ranged from 0.885 to 0.887 on BraTS 2018 validation dataset, and from 0.883 to 0.886 on BraTS 2019 validation dataset. In terms of TC, the mean Dice scores are ranged from 0.792 to 0.813 on BraTS 2018 validation dataset, and from 0.774 to 0.784 on BraTS 2019 validation dataset.

- The average mean Dice scores of all three tumor areas. Obtained from ten experiments, have a high degree of consistency. The standard deviations of three tumor areas are all below 1%.

The median Dice scores of the ten experiments, presented in the 3rd section in each of the two tables, also demonstrate a similar degree of performance excellence and consistency of the proposed system.

Table 4.2 Ten Experiments Results on BRATS 2018 Validation Dataset.

Exp	Dice-Mean			Dice-Median		
	ET	WT	TC	ET	WT	TC
1	0.792	0.887	0.813	0.861	0.908	0.871
2	0.786	0.881	0.797	0.860	0.913	0.872
3	0.788	0.887	0.803	0.845	0.917	0.865
4	0.789	0.886	0.799	0.861	0.911	0.869
5	0.782	0.885	0.792	0.857	0.908	0.871
6	0.789	0.887	0.803	0.861	0.912	0.874
7	0.790	0.888	0.792	0.859	0.908	0.859
8	0.785	0.888	0.797	0.847	0.905	0.858
9	0.784	0.889	0.806	0.866	0.912	0.868
10	0.789	0.886	0.808	0.865	0.912	0.870
Best case	**0.792**	**0.887**	**0.813**	**0.861**	**0.908**	**0.870**
Worst case	**0.782**	**0.885**	**0.792**	**0.857**	**0.908**	**0.871**
Average	**0.787**	**0.886**	**0.801**	**0.858**	**0.911**	**0.868**
STDEV	**0.003**	**0.002**	**0.007**	**0.007**	**0.003**	**0.005**

Table 4.3 Ten Experiments Results on BRATS 2019 Validation Dataset.

Exp	Dice-Mean			Dice-Median		
	ET	WT	TC	ET	WT	TC
1	0.748	0.884	0.776	0.843	0.912	0.860
2	0.767	0.886	0.774	0.861	0.913	0.859
3	0.747	0.883	0.771	0.848	0.912	0.861
4	0.747	0.882	0.776	0.842	0.911	0.850
5	0.754	0.889	0.779	0.844	0.916	0.866
6	0.753	0.885	0.775	0.849	0.910	0.846
7	0.751	0.884	0.768	0.856	0.916	0.857
8	0.752	0.882	0.776	0.844	0.915	0.869
9	0.745	0.883	0.784	0.854	0.917	0.862
10	0.746	0.887	0.779	0.841	0.915	0.862
Best case	**0.767**	**0.886**	**0.774**	**0.861**	**0.913**	**0.859**
Worst case	**0.745**	**0.883**	**0.784**	**0.854**	**0.917**	**0.862**
Average	**0.751**	**0.885**	**0.776**	**0.848**	**0.914**	**0.859**
STDEV	**0.007**	**0.002**	**0.004**	**0.006**	**0.002**	**0.007**

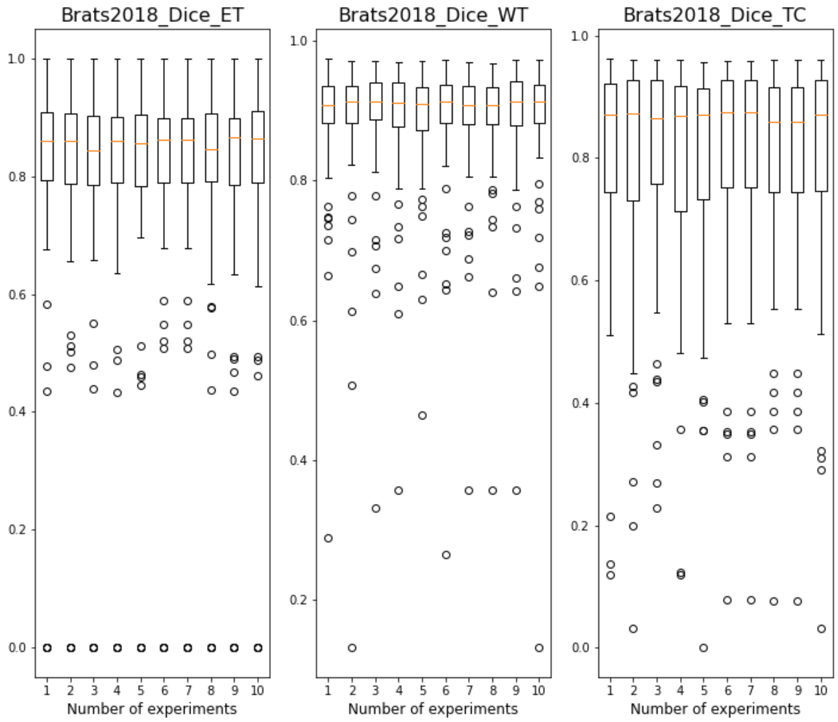

Figure 4.2 Boxplots of the test results obtained in ten experiments on BraTS 2018 validation dataset.

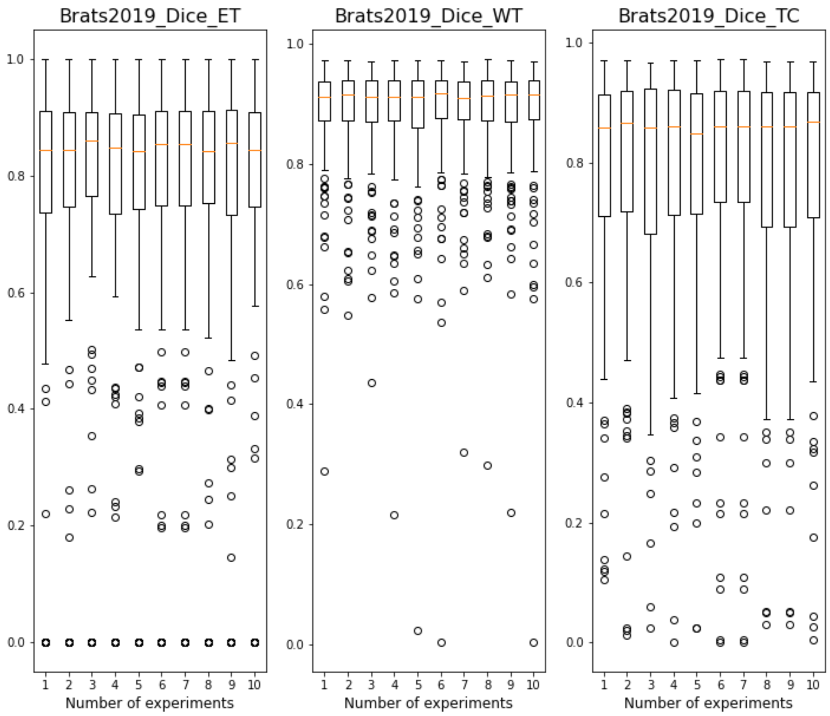

Figure 4.3 Boxplots of the test results obtained in ten experiments on BraTS 2019 validation dataset.

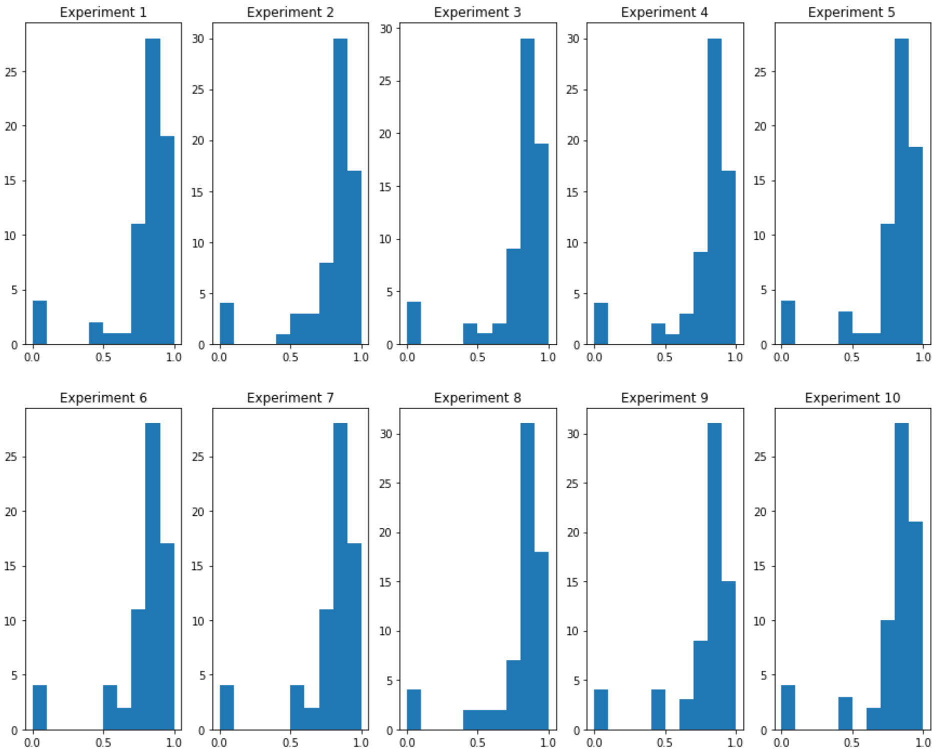

Figure 4.4 Distributions of the mean Dice scores of ET with BraTS 2018 validation samples.

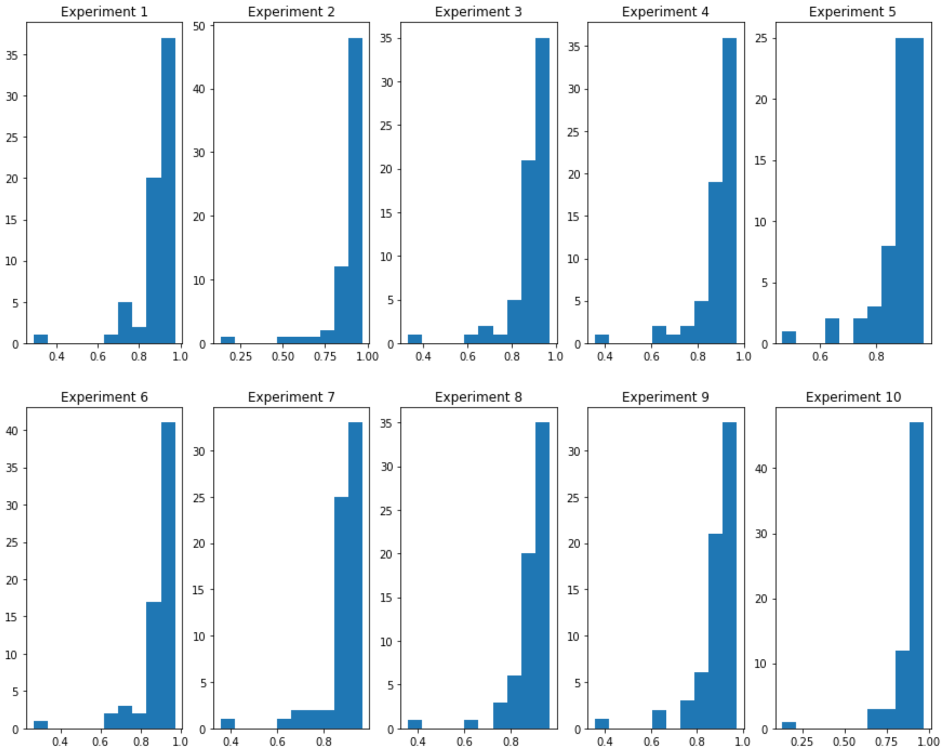

Figure 4.5 Distributions of the mean Dice scores of WT with BraTS 2018 validation samples.

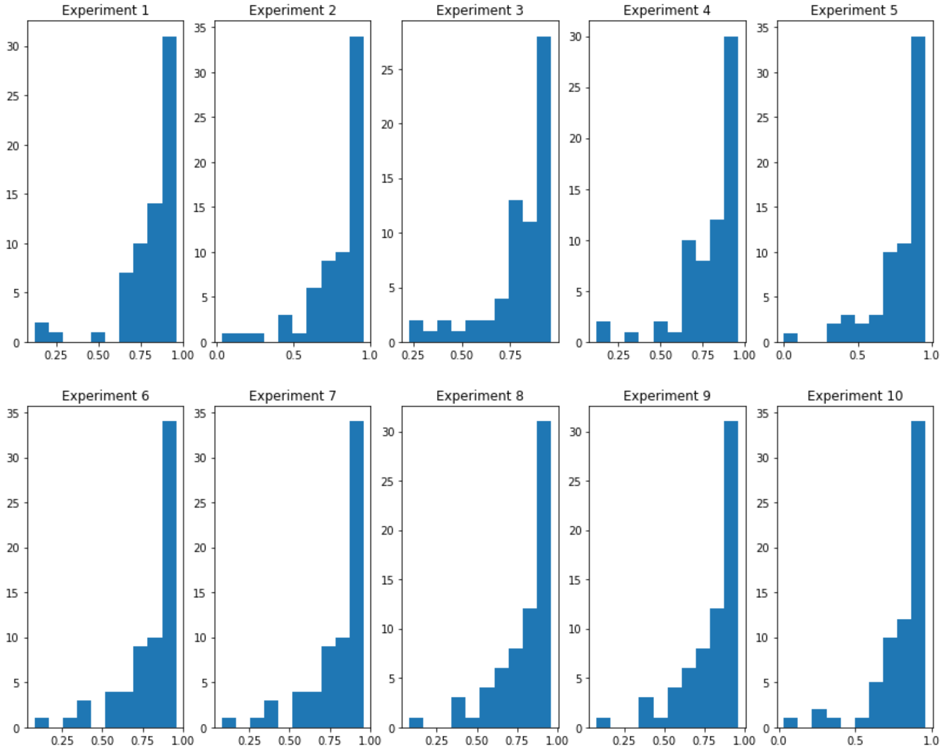

Figure 4.6 Distributions of the mean Dice scores of TC with BraTS 2018 validation samples.

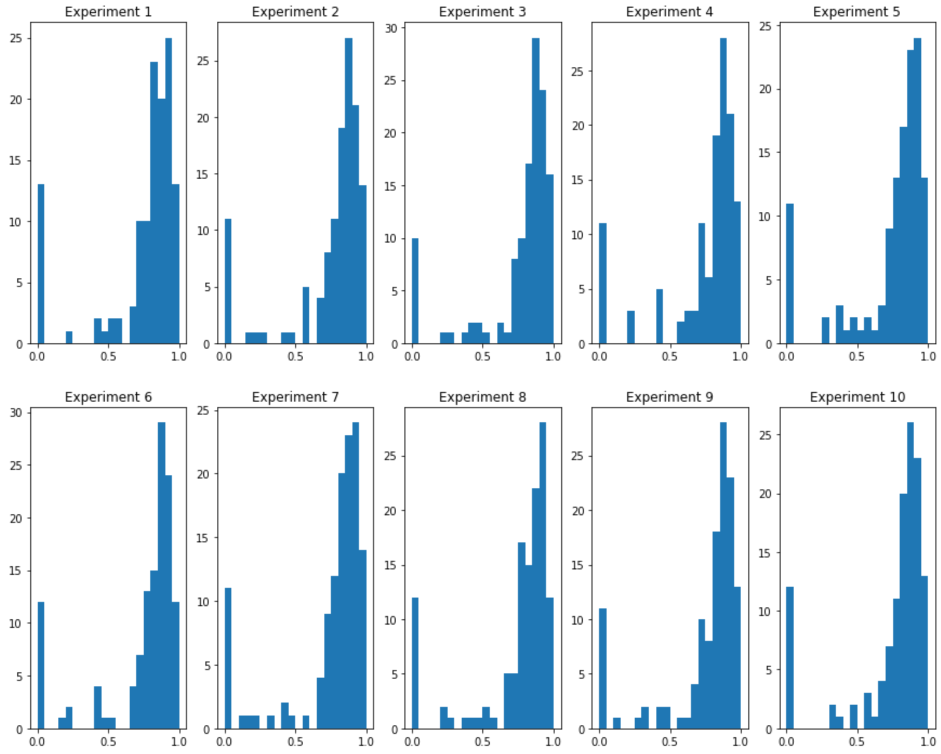
Experiment 1
Experiment 2
Experiment 3
Experiment 4
Experiment 5
Experiment 6
Experiment 7
Experiment 8
Experiment 9
Experiment 10

Figure 4.7 Distributions of the mean Dice scores of ET with BraTS 2019 validation samples.

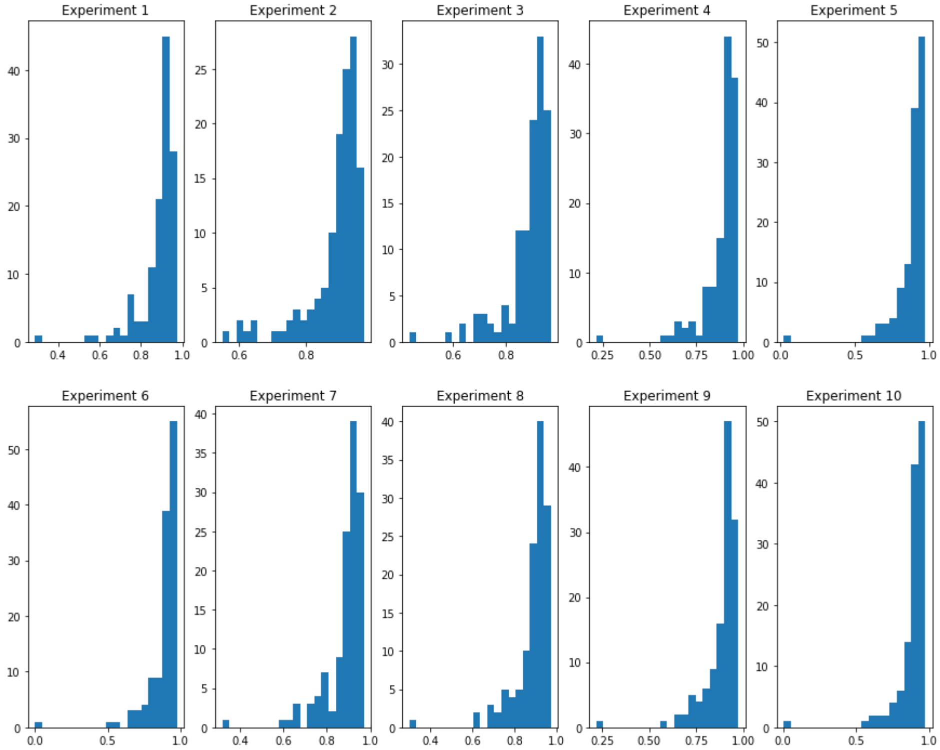
Experiment 1
Experiment 2
Experiment 3
Experiment 4
Experiment 5
Experiment 6
Experiment 7
Experiment 8
Experiment 9
Experiment 10

Figure 4.8 Distributions of the mean Dice scores of WT with BraTS 2019 validation samples.

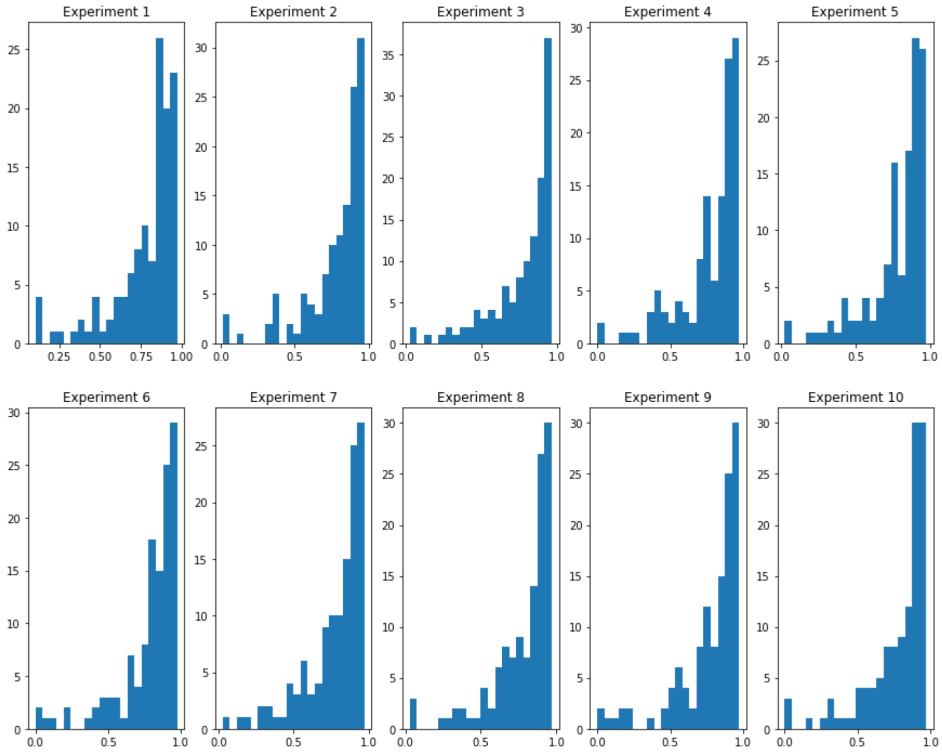

Figure 4.9 Distributions of the mean Dice scores of TC with BraTS 2019 validation samples.

The data visualization of the distribution of average Dice scores of three tumor areas are given from Figure 4.2 to Figure 4.9. One can observe that, in the ten experiments, the distribution of the average Dice scores of each tumor area also has a high degree of similarity, which further demonstrates that the high processing quality of the proposed system is not achieved by accident, but in a consistent manner.

It is also important to measure the reproducibility of the proposed system on processing individual patient case. A reliable system should be able to reproduce similar results on individual patient case after each retraining. Five patient cases, out of 125, from BraTS 2019 validation dataset are selected. Their mean ET Dice scores are around 0.930, 0.858, 753, 730, 460, which are located in 75th percentile, median, average and 25th percentile and below-25^{-}percentile, respectively, of the 125 cases.

Table 4.4 Test results of patient case BraTS19_CBICA_AAM_1 in the ten experiments.

Exp	Dice			Sensitivity		
	ET	WT	TC	ET	WT	TC
1	0.923	0.887	0.933	0.993	0.823	0.963
2	0.932	0.883	0.932	0.992	0.818	0.957
3	0.932	0.887	0.937	0.992	0.824	0.965
4	0.929	0.885	0.929	0.991	0.825	0.962
5	0.927	0.896	0.934	0.990	0.843	0.965
6	0.933	0.883	0.937	0.991	0.816	0.969
7	0.931	0.889	0.935	0.988	0.837	0.959
8	0.936	0.880	0.936	0.987	0.808	0.953
9	0.929	0.891	0.935	0.991	0.837	0.962
10	0.925	0.884	0.925	0.993	0.825	0.973
Average	**0.930**	**0.887**	**0.933**	**0.991**	**0.826**	**0.963**
STDDEV	**0.004**	**0.004**	**0.004**	**0.002**	**0.010**	**0.005**

Table 4.5 Test results of patient case BraTS19_TCIA13_611_1 in the ten experiments.

Exp	Dice			Sensitivity		
	ET	WT	TC	ET	WT	TC
1	0.859	0.857	0.711	0.907	0.928	0.896
2	0.854	0.871	0.767	0.896	0.920	0.886
3	0.861	0.851	0.725	0.898	0.909	0.879
4	0.855	0.862	0.831	0.899	0.916	0.884
5	0.850	0.861	0.850	0.899	0.909	0.884
6	0.867	0.867	0.790	0.902	0.915	0.887
7	0.850	0.860	0.732	0.899	0.906	0.894
8	0.861	0.869	0.857	0.904	0.922	0.890
9	0.857	0.858	0.737	0.924	0.950	0.918
10	0.862	0.874	0.880	0.898	0.925	0.884
Average	**0.858**	**0.863**	**0.788**	**0.903**	**0.920**	**0.890**
STDDEV	**0.005**	**0.007**	**0.059**	**0.008**	**0.012**	**0.010**

Table 4.6 Test results of patient case BraTS19_WashU_W047_1 in the ten experiments.

Exp	Dice			Sensitivity		
	ET	WT	TC	ET	WT	TC
1	0.747	0.715	0.645	0.974	0.911	0.645
2	0.765	0.708	0.649	0.972	0.912	0.635
3	0.780	0.715	0.636	0.963	0.883	0.609
4	0.738	0.686	0.498	0.972	0.930	0.650
5	0.770	0.712	0.543	0.969	0.908	0.514
6	0.776	0.696	0.487	0.972	0.916	0.644
7	0.756	0.718	0.586	0.957	0.910	0.629
8	0.672	0.754	0.684	0.974	0.897	0.736
9	0.747	0.713	0.614	0.973	0.931	0.630
10	0.783	0.703	0.605	0.974	0.937	0.630
Average	**0.753**	**0.712**	**0.595**	**0.970**	**0.914**	**0.632**
STDDEV	**0.031**	**0.017**	**0.063**	**0.005**	**0.015**	**0.051**

Table 4.7 Test results of patient case BraTS19_MDA_958_1 in the ten experiments.

Exp	Dice			Sensitivity		
	ET	WT	TC	ET	WT	TC
1	0.737	0.678	0.754	0.674	0.617	0.674
2	0.730	0.623	0.726	0.658	0.584	0.653
3	0.724	0.677	0.741	0.644	0.566	0.646
4	0.739	0.646	0.757	0.687	0.650	0.691
5	0.744	0.695	0.759	0.705	0.660	0.703
6	0.729	0.675	0.725	0.667	0.634	0.680
7	0.692	0.634	0.703	0.601	0.532	0.597
8	0.699	0.678	0.709	0.602	0.569	0.600
9	0.760	0.643	0.770	0.715	0.707	0.719
10	0.743	0.636	0.756	0.685	0.647	0.702
Average	**0.730**	**0.659**	**0.740**	**0.664**	**0.617**	**0.667**
STDDEV	**0.020**	**0.023**	**0.021**	**0.037**	**0.050**	**0.040**

Table 4.8 Test results of patient case BraTS19_TCIA10_172_1 in the ten experiments.

Exp	Dice			Sensitivity		
	ET	WT	TC	ET	WT	TC
1	0.514	0.969	0.918	0.376	0.975	0.951
2	0.589	0.968	0.924	0.502	0.978	0.945
3	0.501	0.967	0.903	0.385	0.977	0.936
4	0.401	0.970	0.929	0.281	0.980	0.956
5	0.384	0.968	0.918	0.259	0.977	0.938
6	0.439	0.964	0.900	0.339	0.977	0.936
7	0.342	0.962	0.912	0.245	0.971	0.955
8	0.401	0.965	0.917	0.303	0.972	0.933
9	0.416	0.963	0.911	0.294	0.976	0.933
10	0.612	0.962	0.918	0.583	0.976	0.932
Average	**0.460**	**0.966**	**0.915**	**0.357**	**0.976**	**0.942**
STDDEV	**0.086**	**0.003**	**0.008**	**0.104**	**0.003**	**0.009**

From the 5 tables above, one can observe that the proposed system is also able to reproduce almost the same segmentation result on the individual patient case as well. In terms of patient case BraTS19_CBICA_AAM_1, the average Dice scores of ET, WT and TC, obtained in 10 experiments, are 0.930, 0.887,0.933, which are higher than the average Dice scores of 125 patient cases. The standard deviations of the three tumor areas are all 0.004, which are very low. But for some difficult cases, i.e., BraTS19_MDA_958_1, whose Dice scores of ET, WT and TC are 0.730, 0.659, 0.740, the standard deviations are all about 2% which are higher than the average standard deviations of the overall performance, but still much lower compared to that of many other existing CNN systems. In general, due to the custom-designed simple config, the proposed system is able to deliver high-quality segmentation results in a consistent manner, in terms of both overall performance and individual case.

4.6. Ablation Study

The proposed system is designed specifically for brain tumor segmentation and has a number of specific characters, in network configuration and in model parameters, which differentiates it from the other CNN systems for the same task. An ablation study is carried out to assess their effectiveness in processing quality.

The CNN in the proposed system has 4 characters:

1. three-path extraction of mono-modality, paired-modality and cross-modality feature data,
2. classification of 4-class pixels by 3 convolution branches allowing separated training of the filtering coefficients,
3. identification of WT, NET and ET pixels, instead of WT, TC and ET, and
4. minimized numbers of kernels in convolution layers.

Two sets of 4 tests are performed on BraTS 2018 and BraTS 2019 datasets, respectively, to measure the effectiveness of each of the 4 characters in the segmentation quality. The results are presented in Tables 4.9 and 4.10, and each value presented is a mean score obtained in 3 experiments.

In the first test, the 3-path FE block is replaced by a single path involving 3 layers of 48 kernels per layer, and the rest of the network remains unchanged as shown in Figure 4.10. This replacement, while increasing the total number of parameters from 61.843K to 95.971K, results in a decrease of Dice scores in all the tumor categories. In particular, the ET score is reduced from 0.787 to 0.775 with the data samples of BraTS 2018. With the data samples of BraTS 2019, the decrease is even more significant, i.e., from 0.751 to 0.708. It has been confirmed that the 3-path convolution block effectively optimizes the feature extraction from the 4-modality brain images, and it is thus able to produce high-density features information that leads to a better segmentation.

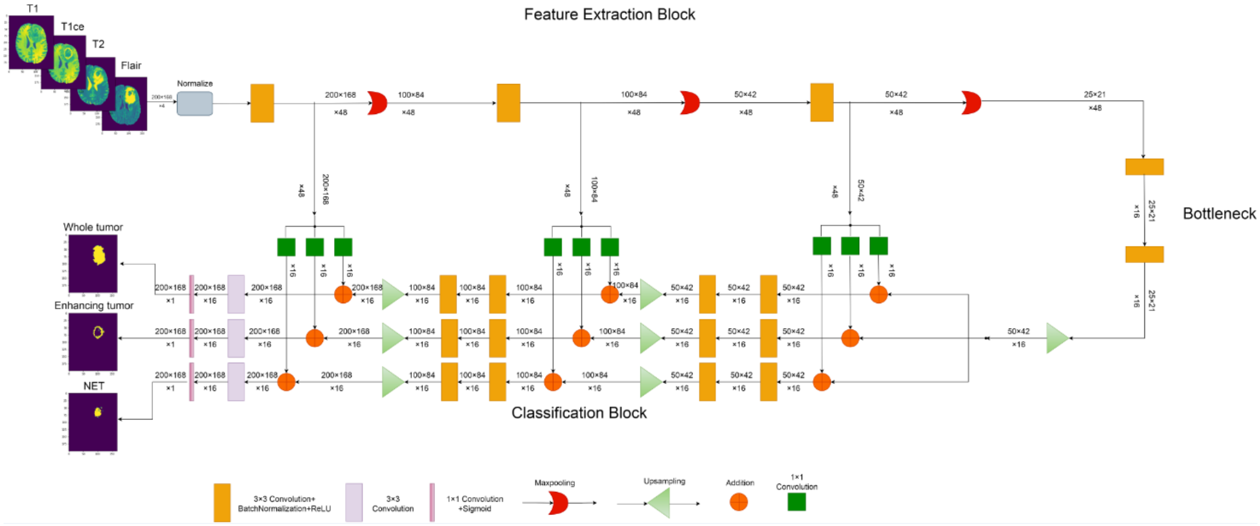

Figure 4.10 Diagram of single-path FE block.

The second test is carried out to measure the effectiveness of the 3 branches in the classification block. For this measurement, the system is modified in such a way that the 3 branches are combined into single one and the number of kernels in each layer is tripled to be 48 as shown in Figure 4.11, which increases considerably the number of parameters from 61.843K to 115.667K. This modification makes the segmentation less accurate, as shown by the data presented in the second row in Tables 4.9 and 4.10, and the most noticeable is the decrease of 2% in ET Dice score, on both BraTS 2018 and 2019 datasets. This test proves that in the proposed system, the 3-branch structure for the classification of the pixels in 3 different tumor categories is effectively advantageous in terms of processing quality and computation cost. While classification of multi-class is considered as a complex task, this structure permits each branch to be trained separately and specifically to handle a binary classification of a particular tumor category, which simplifies the task and improves the processing quality.

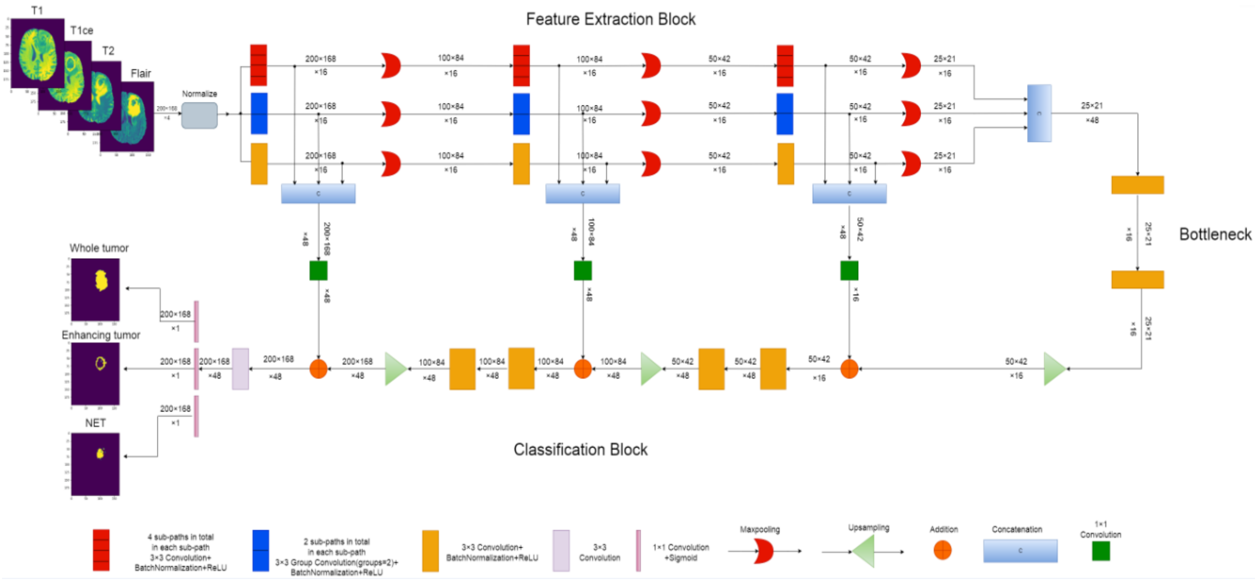

Figure 4.11 Diagram of CNN with single-branch classification block.

In the third test, the three classification branches are dedicated to identifying pixels of WT, TC and ET instead of WT, NET and ET. In light of the results from Table 4.14, the dice score of ET in BraTS 2018 is decreased by 3%. One can infer from this observation that the identification of different tumor areas will also influence the network's performance.

The fourth test is to illustrate that the proposed system, despite its very small number of parameters, is just-sufficient for the brain tumor segmentation task and able to perform high-quality segmentation. The test has been conducted in 2 phases. In phase 1, the proposed system is modified in a such way: the number of kernels in each of the 2 layers of the bottleneck is tripled from 16 to

48, and the number of kernels in each of the 3 branches of the classification block decreases from 48 to 16 over the 3 layers instead of constant 16. This modification tends to improve the processing quality while the number of parameters is increased from 61.843 k to 280.419k. The results of this test are presented in the 4th row in Tables 4.9 and 4.10. One can observe that, despite that the number parameters are quadrupled, the segmentation quality, measured by Dice and sensitivity scores, is not improved. In Phase 2, the number of kernels of each convolution is doubled from 16 to 32, making the total number of parameters increase to 241.187K. The test results, illustrated in the 5th row, do not show any benefit generated from more filtering kernels. All the data produced in these 2 phases demonstrate that the configuration and the usage of parameters in the proposed system are the most appropriate to optimize the segmentation quality under the current circumstances.

Table 4.9 Test results of the ablation studies on BraTS 2018 validation dataset.

Tests	Ablation type	Dice ET	Dice WT	Dice TC	Sensitivity ET	Sensitivity WT	Sensitivity TC	Number of parameters
Test1	Multi-path FE block	0.775	0.882	0.791	0.822	0.898	0.804	95.971K
Test2	Multi-branch classification	0.767	0.878	0.792	0.827	0.895	0.818	115.667K
Test3	Identification areas	0.757	0.883	0.799	0.824	0.904	0.833	61.843K
Test4.1	Number of parameters	0.782	0.887	0.794	0.843	0.903	0.817	280.419K
Test4.2	Number of parameters	0.782	0.883	0.803	0.832	0.892	0.827	241.187K
	Proposed CNN	**0.787**	**0.886**	**0.801**	**0.837**	**0.900**	**0.816**	**61.843K**

Table 4.10 Test results of the ablation studies on BraTS 2019 validation dataset.

Tests	**Ablation type**	**Dice ET**	**Dice WT**	**Dice TC**	**Sensitivity ET**	**Sensitivity WT**	**Sensitivity TC**	**Number of parameters**
Test1	Multi-path FE block	0.708	0.881	0.771	0.731	0.877	0.779	95.971K
Test2	Multi-branch classification	0.732	0.882	0.773	0.762	0.889	0.784	115.667K
Test3	Identification areas	0.742	0.885	0.777	0.787	0.888	0.772	61.843K
Test4.1	Number of parameters	0.752	0.886	0.768	0.791	0.888	0.761	280.419K
Test4.2	Number of parameters	0.750	0.883	0.777	0.792	0.892	0.773	241.187K
	Proposed CNN	**0.751**	**0.885**	**0.776**	**0.796**	**0.891**	**0.783**	**61.843K**

4.7. Performance Comparison

The performance of the proposed system is compared, in terms of segmentation quality and computation complexity, with that of those recently reported in reputed journals. The comparison results are presented in Tables 4.11 and 4.12.

In the two tables, the results of the proposed system are presented in the bottom rows. The Dice and Sensitivity scores are generated by the ten experiments specified in Sub-chapter 4.5. One can have the following two observations.

In terms of segmentation quality, the performance of the proposed system is among the best. Its Dice and Sensitivity scores are higher than that of many other CNN systems reported in research journals, with validation samples of both BraTS 2018 and 2019 datasets.

In terms of computation cost, the proposed system is the best, standing ahead of the others by a considerable distance. The entire convolutional network has only 61,843 parameters, while most existing systems need multi-millions. If the number of voxels of each patient case is 240×240×155×4, the number of Flops to complete the test is merely 146G.

Table 4.11 Comparison of the proposed system with other CNN systems recently reported in reputed journals on BraTS 2018 validation dataset.

	Dice			Sensitivity			Number of parameters
	ET	WT	TC	ET	WT	TC	
Huang *et al.* 2021 [39]	0.717	0.801	0.759	0.829	0.962	0.800	N.A.
Akil *et al.* 2020 [24]	0.732	0.860	0.733	0.740	0.838	0.702	181k
Liu *et al.* 2021 [40]	0.767	0.898	0.834	N.A.	N.A.	N.A.	3.34M
Sun *et al.* 2021 [41] *	0.771	0.900	0.795	0.769	0.904	0.751	27.42M
Zhang *et al.* 2020 [27] *	0.772	0.872	0.808	N.A.	N.A.	N.A.	7.5M
Rehman *et al.* 2021 [42] *	0.773	0.894	0.826	N.A.	N.A.	N.A.	31.4M
Zhang *et al.* 2020[43]	0.782	0.896	0.824	N.A.	N.A.	N.A.	363k
Best score of the proposed system	**0.792**	**0.887**	**0.813**	**0.851**	**0.901**	**0.820**	**61.834k**
Average score of the proposed system	**0.787**	**0.886**	**0.801**	**0.837**	**0.900**	**0.816**	**61.834k**
STDDEV of the proposed system	**0.003**	**0.002**	**0.007**	**0.007**	**0.003**	**0.005**	**61.834k**

* Indicates that the number of parameters is noy given in original paper, it’s an estimation based on the network config from the original paper.

Table 4.12 Comparison of the proposed system with other CNN systems recently reported in reputed journals on BraTS 2019 validation dataset.

	Dice			Sensitivity			Number of parameters
	ET	WT	TC	ET	WT	TC	
Di Ieva, *et al.* 2021 [44] *	0.675	0.870	0.711	0.702	0.823	0.670	28.03M
Rehman *et al.* 2021 [42] *	0.708	0.869	0.775	N.A.	N.A.	N.A.	31.4M
Zhang *et al.* 2020 [27] *	0.709	0.870	0.777	N.A.	N.A.	N.A.	7.5M
Huang *et al.*2021 [39]	0.730	0.827	0.788	0.798	0.967	0.801	N.A.
Ali *et al.* 2021 [45]	0.740	0.880	0.810	0.770	0.890	0.790	N.A.
Liu *et al.* 2021 [40]	0.759	0.885	0.851	N.A.	N.A.	N.A.	3.34M
Sun *et al.* 2020 [41] *	0.761	0.890	0.779	0.767	0.883	0.762	27.42M
Best Score of the proposed system	**0.767**	**0.886**	**0.774**	**0.800**	**0.889**	**0.784**	**61.843K**
Average score of the proposed system	**0.751**	**0.885**	**0.776**	**0.796**	**0.891**	**0.783**	**61.834K**
STDDEV of the proposed system	**0.007**	**0.002**	**0.004**	**0.006**	**0.002**	**0.007**	**61.834K**

* Indicates that the number of parameters is noy given in original paper, it’s an estimation based on the network config from the original paper.

The test results demonstrates that the proposed system is able to perform brain tumor segmentation of excellent quality in a consistent manner, and this performance is achieved at a computation cost that is only a small fraction of the average amount of the others. It confirms the effectiveness of the specific characters in the design. They altogether make the proposed system operate with a high computation efficiency, low randomness, and reliable performance.

4.8. Summary

The focus in designing the proposed system is on the high processing quality, high performance consistency and low computation cost. To assess the performance of the system in these aspects, a good number of experiments have been carried out with BraTS 2018 and 2019 datasets.

In terms of BraTS 2018 validation dataset, the average Dice scores, obtained from ten experiments, are 0.787, 0.886 and 0.801 for ET, WT and TC, respectively. In terms of BraTS 2019 dataset, the average Dice scores are 0.751, 0.885, 0.776 for ET, WT and TC. The standard deviations of the Dice scores of three tumor areas are all less than 1% on both datasets. Those statistic numbers prove that the proposed system is able to perform high-quality segmentation in a consistent manner. Furthermore, in order to test the reproducibility of the processing quality on individual patient case, five patient cases are selected from BraTS 2019 validation dataset and the statistic results demonstrates a good reproducibility of the proposed system on individual patient case as well.

The results of the proposed system are compared with those of other CNN systems recently reported in reputed journals. In terms of the segmentation quality, the proposed system is among the best. In terms of the computation cost, the proposed system is the lowest, whose number of parameters in only 61,843, while most CNN systems require multi-millions.

The comprehensive performance evaluation of the proposed system confirms the effectiveness of the specific characters in the design. They altogether make the proposed system operate with a high computation efficiency, low randomness, and reliable performance.

5. Conclusion

Brain tumors cause serious diseases and brain tumor detection is important for the diagnosis. Due to the complexity of brain structures and a huge amount of variations of tumor patterns, manual brain tumor segmentation is a time-consuming and challenging task, which may lead to a lengthy wait in many patient cases. It's imperative to develop fully automated brain segmentation systems to accelerate the diagnosis process.

For medical applications, the system needs to be reliable and implementable in the computation resources restricted environment. Thus, the objective of the work presented in this thesis is to develop a performance-consistent, computation-efficient CNN system for high-quality brain tumor segmentation.

To achieve this objective, the CNN system has been custom-designed based on the analysis of the input/output. The proposed system consists of three blocks, for the pre-processing, CNN segmentation and refinement. The pre-processing block is to reduce the data volume by cropping the excessive margins in each slice and removing the evident tumor-free slices, and the refinement block is to correct the falsely classified ET voxels. The main processing is done by the CNN block.

The CNN consists of three parts, (i) a multi-path FE block to extract comprehensive feature data of mono-modality, paired-modality and cross-modality, (ii) a bottleneck to perform tumor localization and coarse segmentation, and, (iii) a multi-branch classification block where each is branch trained separately in order to adjust its coefficients for the precise identification of one specific kind of tumor area. Also, the number of kernels per layer and the number of layers are just-sufficient for the accurate brain tumor segmentation in order to reduce randomness and computation complexity.

In order to evaluate the performance of the proposed system, a good number of experiments have been carried out on BraTS 2018 and BraTS 2019 datasets. In terms of the 10 experiments on BraTS 2018 validation samples, the average Dice scores and their standard deviations are 0.787±0.003, 0.886±0.002, 0.801±0.007, respectively, for enhancing tumor, whole tumor and tumor core. For the validation samples of BraTS 2019 in 10 experiments, the average Dice scores and standard deviations of enhancing tumor, whole tumor and tumor core are 0.751±0.007, 0.885±0.002, 0.776±0.004, respectively. The test results give a strong proof that the proposed system is able to perform high-quality segmentation in a consistent manner at very low computation cost.

The performance of the proposed system is compared with other CNN systems recently reported in reputed journals. The comparison results show that the proposed system is among the best in terms of the segmentation quality. In terms of the computation complexity, the proposed system is the best, standing ahead of the others by a considerable distance. The entire CNN block has only 61,843 parameters, while most existing systems need multi-millions. If the number of voxels of each patient case is 240×240×155×4, the number of Flops to complete the test is merely 146G.

The high processing quality and low computation complexity of the proposed system make it implementable in various environments. It can be expected that such system will have wide applications in medical image processing.

Reference

[1] https://www.tfri.ca/about-cancer/cancer-types/cancer-type/brain

[2] https://www.mayfieldclinic.com/pe-anatbrain.htm

[3] https://www.newscientist.com/article/dn9969-introduction-the-human-brain/

[4] https://www.hopkinsmedicine.org/health/conditions-and-diseases/brain-tumor

[5] Menze, B. H. *et al*. (2014). The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE transactions on medical imaging*, *34*(10), 1993-2024.

[6] Gibbs, P., Buckley, D. L., Blackband, S. J., & Horsman, A. (1996). Tumour volume determination from MR images by morphological segmentation. Physics in Medicine & Biology, 41(11), 2437.

[7] Praveen, G. B., & Agrawal, A. (2015, November). Hybrid approach for brain tumor detection and classification in magnetic resonance images. In 2015 Communication, Control and Intelligent Systems (CCIS) (pp. 162-166). IEEE.

[8] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

[9] Matsugu, M., Mori, K., Mitari, Y., & Kaneda, Y. (2003). Subject independent facial expression recognition with robust face detection using a convolutional neural network. Neural Networks, 16(5-6), 555-559.

[10] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[11] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[12] https://medium.com/unpackai/batch-normalization-25905f889723.

[13] https://en.wikipedia.org/wiki/Bilinear_interpolation.

[14] Ng, D., & Feng, M. (2020). Medical Image Recognition: An Explanation and Hands-On Example of Convolutional Networks. In Leveraging Data Science for Global Health (pp. 263-284). Springer, Cham.

[15] Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.

[16] Li, X., Chen, H., Qi, X., Dou, Q., Fu, C. W., & Heng, P. A. (2018). H-DenseUNet: hybrid densely connected UNet for liver and tumor segmentation from CT volumes. IEEE

transactions on medical imaging, 37(12), 2663-2674.

[17] Salehi, S. S. M., Erdogmus, D., & Gholipour, A. (2017). Auto-context convolutional neural network (auto-net) for brain extraction in magnetic resonance imaging. IEEE transactions on medical imaging, 36(11), 2319-2330.

[18] Rosas González, S., Birgui Sekou, T., Hidane, M., Zemmoura, I., & Tauber, C. (2021). Asymmetric Ensemble of Asymmetric U-Net Models for Brain Tumor Segmentation With Uncertainty Estimation. Frontiers in Neurology, 1421.

[19] Luo, Z., Jia, Z., Yuan, Z., & Peng, J. (2020). Hdc-net: Hierarchical decoupled convolution network for brain tumor segmentation. IEEE Journal of Biomedical and Health Informatics, 25(3), 737-745.

[20] Maji, D., Sigedar, P., & Singh, M. (2022). Attention Res-UNet with Guided Decoder for semantic segmentation of brain tumors. Biomedical Signal Processing and Control, 71, 103077.

[21] Zhou, T., Ruan, S., Vera, P., & Canu, S. (2021). A Tri-Attention fusion guided multi-modal segmentation network. Pattern Recognition, 108417.

[22] Chen, Y., Cao, Z., Cao, C., Yang, J., & Zhang, J. (2018, June). A modified U-Net for brain Mr image segmentation. In International Conference on Cloud Computing and Security (pp. 233-242). Springer, Cham.

[23] Cahall, D. E., Rasool, G., Bouaynaya, N. C., & Fathallah-Shaykh, H. M. (2019). Inception modules enhance brain tumor segmentation. Frontiers in computational neuroscience, 13, 44.

[24] Akil, M., Saouli, R., & Kachouri, R. (2020). Fully automatic brain tumor segmentation with deep learning-based selective attention using overlapping patches and multi-class weighted cross-entropy. Medical image analysis, 63, 101692.

[25] Yogananda, C. G. B., Shah, B. R., Vejdani-Jahromi, M., Nalawade, S. S., Murugesan, G. K., Yu, F. F., ... & Maldjian, J. A. (2020). A fully automated deep learning network for brain tumor segmentation. Tomography, 6(2), 186-193.

[26] Wang, L., Wang, S., Chen, R., Qu, X., Chen, Y., Huang, S., & Liu, C. (2019). Nested dilation networks for brain tumor segmentation based on magnetic resonance imaging. Frontiers in Neuroscience, 13, 285.

[27] Zhang, J., Jiang, Z., Dong, J., Hou, Y., & Liu, B. (2020). Attention gate resU-Net for automatic MRI brain tumor segmentation. IEEE Access, 8, 58533-58545.

[28] Huang, Z., Zhao, Y., Liu, Y., & Song, G. (2021). GCAUNet: A group cross-channel attention residual UNet for slice based brain tumor segmentation. Biomedical Signal Processing and

Control, 70, 102958.

[29] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2019). Unet++: Redesigning skip connections to exploit multiscale features in image segmentation. IEEE transactions on medical imaging, 39(6), 1856-1867.

[30] Zhou, T., Canu, S., Vera, P., & Ruan, S. (2021). Latent correlation representation learning for brain tumor segmentation with missing MRI modalities. IEEE Transactions on Image Processing, 30, 4263-4274.

[31] Wang, B., Yang, J., Peng, H., Ai, J., An, L., Yang, B., ... & Ma, L. (2021). Brain Tumor Segmentation via Multi-Modalities Interactive Feature Learning. Frontiers in Medicine, 8.

[32] Wang, G., Li, W., Ourselin, S., & Vercauteren, T. (2019). Automatic brain tumor segmentation based on cascaded convolutional neural networks with uncertainty estimation. Frontiers in computational neuroscience, 13, 56.

[33] Jiang, Z., Ding, C., Liu, M., & Tao, D. (2019, October). Two-stage cascaded U-Net: 1st place solution to BraTS challenge 2019 segmentation task. In International MICCAI brainlesion workshop (pp. 231-241). Springer, Cham.

[34] Zhou, C., Ding, C., Wang, X., Lu, Z., & Tao, D. (2020). One-pass multi-task networks with cross-task guided attention for brain tumor segmentation. IEEE Transactions on Image Processing, 29, 4516-4529.

[35] Li, X., Luo, G., & Wang, K. (2019, October). Multi-step cascaded networks for brain tumor segmentation. In International MICCAI Brainlesion Workshop (pp. 163-173). Springer, Cham.

[36] https://www.med.upenn.edu/sbia/brats2018/evaluation.html

[37] https://www.med.upenn.edu/cbica/brats2019/evaluation.html\

[38] Salehi, S. S. M., Erdogmus, D., & Gholipour, A. (2017, September). Tversky loss function for image segmentation using 3D fully convolutional deep networks. In *International workshop on machine learning in medical imaging* (pp. 379-387). Springer, Cham.

[39] Huang, H., Yang, G., Zhang, W., Xu, X., Yang, W., Jiang, W., & Lai, X. (2021). A deep multi-task learning framework for brain tumor segmentation. Frontiers in Oncology, 11.

[40] Liu, Z., Tong, L., Chen, L., Zhou, F., Jiang, Z., Zhang, Q., ... & Zhou, H. (2021). CANet: Context aware network for brain glioma segmentation. IEEE Transactions on Medical Imaging, 40(7), 1763-1777.

[41] Sun, J., Peng, Y., Guo, Y., & Li, D. (2021). Segmentation of the multimodal brain tumor image used the multi-pathway architecture method based on 3D FCN. Neurocomputing, 423, 34-45.

[42] Rehman, M. U., Cho, S., Kim, J., & Chong, K. T. (2021). BrainSeg-net: Brain tumor MR

image segmentation via enhanced encoder–decoder network. Diagnostics, 11(2), 169.

[43] Zhang, D., Huang, G., Zhang, Q., Han, J., Han, J., Wang, Y., & Yu, Y. (2020). Exploring task structure for brain tumor segmentation from multi-modality MR images. IEEE Transactions on Image Processing, 29, 9032-9043.

[44] Di Ieva, A., Russo, C., Liu, S., Jian, A., Bai, M. Y., Qian, Y., & Magnussen, J. S. (2021). Application of deep learning for automatic segmentation of brain tumors on magnetic resonance imaging: a heuristic approach in the clinical scenario. Neuroradiology, 63(8), 1253-1262.

[45] Ali, M. J., Raza, B., & Shahid, A. R. (2021). Multi-level Kronecker Convolutional Neural Network (ML-KCNN) for Glioma Segmentation from Multi-modal MRI Volumetric Data. Journal of Digital Imaging, 34(4), 905-921.

[46] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[47] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision* (pp. 1026-1034).