

# **Development of Computation-Efficient Computer Vision Systems for High- Quality Brain Tumor Segmentation**

Yanming Sun

A Thesis

In the Department of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of

Doctor of Philosophy (Electrical and Computer Engineering)

at Concordia University

Montreal, Quebec, Canada

March 2025

© Yanming Sun, 2025

**CONCORDIA UNIVERSITY**  
**SCHOOL OF GRADUATE STUDIES**

This is to certify that the thesis prepared

By: Yanming Sun

Entitled: Development of Computation-Efficient Computer Vision Systems for High-Quality Brain Tumor Segmentation

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

**Doctor of Philosophy (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. Ferhat Khendek	
_____	External Examiner
Dr. Yong Zhao	
_____	Arms-Length Examiner
Dr. Xinxin Zou	
_____	Examiner
Dr. Omair Ahmad	
_____	Examiner
Dr. Weiping Zhu	
_____	Thesis Supervisor
Dr. Chunyan Wang	

Approved by:

\_\_\_\_\_  
Dr. Jun Cai, Graduate Program Director

\_\_\_\_\_  
20

\_\_\_\_\_  
Dr. Mourad Debbabi, Dean,  
Gina Cody School of Engineering and Computer Science

# ABSTRACT

## Development of Computation-Efficient Computer Vision Systems for High-Quality Brain Tumor Segmentation

**Yanming Sun, Ph.D.**

**Concordia University, 2025**

In this thesis, two design methodologies are proposed, and also applied in the development of 2 computer-vision systems for computation-efficient and high-quality brain-tumor detection.

The first methodology aims at developing systems to detect, by conventional image processing procedures, 3D-object locations with a pixel-wise precision. The main operations of the detection are predicting gray-level distribution of the pixels in the object region and, based the prediction result, identifying/removing regions of non-interest. As 3D inputs can be sliced into axial, coronal or sagittal slice series, the prediction/identification/removal operations are performed step-by-step to the 3 series, respectively. Each removal increases the density of the object-information, facilitating the next prediction. To comprehend the pixel distributions with their locations, a 2D histogram presentation is proposed. In the design of the brain-tumor detection system, it is used to highlight the left-right asymmetry of a brain structure. Since the asymmetry is caused by tumors and non-pathological elements, an adaptive histogram modulation method is proposed to enhance the former by attenuating the latter. The prediction/identification/removal operations transform a 3D brain image into a tumoral minimum bounding box, which is then transformed into a tumor mask using simple morphological operations. The test results, on 1251 samples, have confirmed the high quality of the prediction of the tumor data distributions and the tumor detection.

The second methodology is proposed to design CNN (convolutional neural network) systems handling a complex task of brain-tumor segmentation, i.e., classifying the pixels of a brain image into 4 classes of intra-tumoral regions and the background. The methodology is to decompose this complex task into simple subtasks and each of them is performed by a simply-configured and independently-trained CNN. By doing so, one can optimize the use of computing power and minimize the gradient conflict in training. The 4-class classification is decomposed into 3 binary classifications. Each of them is further decomposed into 2: first locating the object region and then identifying the pixels inside the region, performed by 2 independent-and-simple modules. The overall system, requiring only 0.75M trainable parameters, has been trained/tested with BraTS datasets, and its processing quality is among the best reported recently.

# Acknowledgements

I am extremely grateful to my supervisor, Dr. Chunyan Wang, for her guidance throughout the research during my PhD program. Her expertise, encouragement, and patience were crucial to the completion of this work. I am also grateful to my thesis committee members for their thoughtful insights and helpful suggestions.

I extend my deepest gratitude to my family for their unwavering support. I would also like to thank my friends, colleagues and collaborators, including but not limited to Yunlong Ma, Mingze Ni and Yijian Zhao.

Finally, thank Natural Sciences and Engineering Research Council (NSERC) of Canada, as well as Digital Research Alliance of Canada, for supporting this work.

# Table of Contents

List of Figures.....	vii
List of Tables.....	x
Table of Acronyms and Abbreviations.....	xi
Table of Symbols.....	xii
Chapter 1 Introduction.....	1
1.1 Background and Challenges.....	1
1.2 Motivation and Objectives.....	3
1.3 Scope and Organization.....	4
Chapter 2 Background and Relevant Work.....	6
2.1 Basics of Filtering and CNN for Image Processing.....	6
2.1.1 Basics of Conventional Filtering.....	6
2.1.2 Basics of CNN.....	7
2.2 Related Work of Conventional Non-CNN System for Brain Tumor Detection.....	10
2.3 Related Work of CNN System for Brain Tumor Segmentation.....	12
2.4 Summary.....	17
Chapter 3 Knowledge-Based System for Whole Tumor Detection.....	18
3.1 Methodology.....	19
3.2 Two-D Gray Level Distribution of 3D data.....	19
3.3 Proposed Knowledge-Based system.....	23
3.3.1 System Overview.....	23
3.3.2 Brain Image Asymmetry Measure to Extract Tumor Information.....	26
3.3.3 Generation of the Modulation Function.....	28
3.3.4 Prediction of the Tumor Pixel Distribution.....	31
3.3.4.1 First Two Coarse Predictions and Cropping Operations.....	32

3.3.4.2 Third Coarse Prediction & Cropping and the Final Prediction.....	35
3.3.5 Whole Tumor Detection in Pixel-Wise Precision .....	37
3.4 Performance Evaluation.....	39
3.4.1 Dataset.....	39
3.4.2 Visualization of Prediction and Detection Results .....	40
3.4.3 Performance Measurements .....	43
3.4.3.1 Performance Metrics .....	44
3.4.3.2 Prediction Results.....	44
3.4.3.3 Whole Tumor Detection Results .....	45
3.4.4 Performance Comparison.....	47
3.5 Summary.....	49
Chapter 4    CNN for Multi-Class Brain Tumor Segmentation .....	50
4.1 CNN Problem Statement and Design Methodology .....	50
4.2 Detailed Design.....	53
4.3 Performance Evaluation.....	58
4.3.1 Dataset and Performance Metrics.....	58
4.3.2 Training and Implementation Details.....	59
4.3.3 Test Results and Comparison .....	60
4.4 Summary.....	62
Chapter 5    Conclusion .....	64
Reference.....	66

# List of Figures

Figure 1.1 (a)(b)(c)(d) Center slice of the brain images, acquired by MRI of Flair, T2, T1 and T1c modalities, respectively, from a patient. (e) Tumor mask obtained from the 4 slices. The non-zero pixels are found in the tumor region that is further segmented into 3 kinds of intra-tumoral regions, namely ED (gray), NET (darker gray), and ET (brightest). .....2

Figure 1.2 (a) Example 2D slices of Flair, T2, T1 and T1c modalities, and the corresponding expected output. (b) Those of another patient case. ....3

Figure 2.1 Example kernels of Gabor filter. (a) Wavelength  $\lambda = 5$ , orientation  $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$ . (b) Wavelength  $\lambda = 10$ , orientation  $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$ . ....7

Figure 2.2 (a) Convolution operation. (b) Dilation convolution,  $dilation=2$ . (c) Convolution with  $stride=2$ . ....8

Figure 2.3 Widely used normalization methods [7], where  $C$  is the number of channels,  $N$  is the batch size, and  $H$  &  $W$  are the height and width of an image. ....9

Figure 2.4 (a) Example of slice of brain image with tumor. Tumor reduces the degree of the natural left-right symmetry of the brain. (b) Tumor mask. (c) Gray level distributions of the left and right halves of the slice.  $H_n$  is the histogram of the half without tumor.  $H_p$  is the histogram of the half with tumor.  $H_s$  is the histogram difference between  $H_n$  and  $H_p$  [22]. ....11

Figure 2.5 Architecture of U-Net [24]. ....12

Figure 2.6 Structure of residual cyclic unpaired encoder-decoder network (RescueNet) [32]. ....13

Figure 2.7 Block diagram of One-pass Multi-task Network (OM-Net) [33]. ....13

Figure 2.8 Block diagram of task-structured brain tumor segmentation network (TSBTS net) [34]. ....14

Figure 2.9 Structure of multi-view dynamic fusion framework (MVFusFra) [35]. ....14

Figure 2.10 CNN system with 3 different dimensional networks [36]. ....15

Figure 2.11 Block diagram of Dual-force U-Net (DF-U-Net) [37]. ....15

Figure 2.12 Block diagram of multi-scale context and attention mechanisms network (MsANet) [38]. ....16

Figure 2.13 Block diagram of 3D Parallel Shifted Window-based Transformer module (3D PSwinBTS) [39]. ....16

Figure 2.14 Block diagram of Aligned Cross-Modality Interaction Network (ACMINet) [40]. ....16

Figure 3.1 (a)(c) Slices sampled from two 3D brain images of Flair modality. (b)(d) Binary tumor masks of the slices in (a) and (c). ....18

Figure 3.2 Gray level distributions of 3D brain images given by the patient cases 01417 and 01438 from BraTS2021 dataset [43]. The pixels outside the brain regions are excluded. ....20

Figure 3.3 (a) Two-D histogram of the 3D brain region of the patient case 01417 from BraTS2021 dataset. It is composed of a series of gray level distributions of the 155 2D axial slices, excluding the pixels outside the brain region. The x-axis specifies the gray levels, normalized to  $[0, 1]$ , the y-axis the index of axial slices, the z-axis the number of pixels. (b) Distribution of the 80th axial slice. (c) Distribution of the 142nd axial slice. ....21

Figure 3.4 (a) Two-D histogram of a 3D brain region in a series of axial slices, given by the case 01417 from BraTS2021 dataset. (b) Two-D histogram of the tumor region (ground truth data). (c)(d) Two-D histograms given by the case 01438 from BraTS2021 dataset. ....22

Figure 3.5 Two-D histograms of a brain region and that of the tumor region, given by (a)(b) the axial slices, (c)(d) the coronal slices, and (e)(f) the sagittal slices. The original data sample is from the case 01417 from BraTS2021 dataset. ....23

Figure 3.6 Block diagram of the proposed system. It receives a 3D brain image input that can be sliced into a series of  $n_A$  axial slices, or a series of  $n_C$  coronal slices or  $n_S$  sagittal slices, and generates the predicted 2D histogram of the brain tumor region for each of the 3 series of slices and a 3D tumor mask. ....25

Figure 3.7 (a) Slice of brain image without tumor. The left-right asymmetry in image details is referred to as natural asymmetry. (b)(c) Slice of brain image with tumor and its binary tumor mask. The asymmetry is more noticeable. 26

Figure 3.8 Four 2-D histograms obtained from the 155 axial slices of a low-passed 3D Flair image sample. The X-axis is the normalized gray scale and the zero point corresponds to the mean value of the 3D brain region, excluding the pixels of gray level values below the mean. The data sample is from the case 01412 of BraTS2021 dataset. (a)  $H_{left}(i,j)$ , the 2D histogram of the left half of the 3D image, (b)  $H_{right}(i,j)$ , the 2D histogram of the right half of the 3D image, (c)  $\Delta H(i,j) = |H_{left}(i,j) - H_{right}(i,j)|$ , and (d)  $H_T(i,j)$ , the ground truth of the 2D gray level distribution of the tumor region. ....27

Figure 3.9 Graph of the modulation function  $f_m(i)$ , plotted in magenta, that matches well  $h_{tumor-free}(i)$  and  $h_{tumor}(i)$ , the gray level distribution of the tumor-free region and that of the tumor region, respectively. Ideally, its low (L), mid (M) and high (H) sections should adapt to each individual patient case. The range of  $f_m(i)$  is set to be  $[\alpha, 1]$  and  $\alpha \ll 1.28$

Figure 3.10 Gray level distributions given by patient cases 01412 and 01414 from BraTS 2021 dataset. In the 2 graphs,  $h_{tumor-free-half}$ , the distribution of the pixels from the tumor-free half is compared with  $h_{tumor-half}$ , that of the tumor half and  $h_{tumor-free}$ , that of the true tumor-free region inside the entire brain region. ....29

Figure 3.11 (a) Block diagram of the procedure to transform  $h_{ij}(i)$  to  $f_m(i)$ . (b) Graph of  $h_{ij}(i)$  of a 3D Flair image,  $h_T(i)$ , truncated  $h_{ij}(i)$  with  $max_T \approx 0.5$ ,  $min_T \approx 0.05$ ,  $1/h_T(i)$  and  $f_m(i)$  given by Equation (9) with  $\gamma = 1.8$  and  $\alpha = 0.02$ ....30

Figure 3.12 Two characteristics, plotted in magenta, of the modulation function  $f_m(i)$  with  $\gamma = 1.8$  and  $\alpha = 0.02$ , generated with the input data from the Flair modality of the patient cases 01412 and 01414, respectively, of BraTS2021 dataset. In each graph, the curve of  $f_m(i)$  is superimposed with  $h_{tumor}(i)$ , the distribution of the pixels in the tumor region and  $h_{tumor-free}(i)$ , that of the tumor-free part of brain region. The scale of  $f_m(i)$  is  $[0,1]$ . ....31

Figure 3.13 (a) Asymmetry map  $\Delta H_a(i,j)$ , generated with the axial slice series of the patient case 01412 from BraTS2021 dataset. (b) Modulated 2D histogram of the axial slice series,  $H_{ma}(i,j)$ , to be used as a coarsely predicted 2D histogram of the tumor region. (c) Two-D histogram of the pixels in the tumor region given by the ground truth,  $H_{Ta}(i,j)$ , over the axial slices. ....32

Figure 3.14 (a) Coarsely predicted 2D histogram of the tumor region,  $H_{ma}(i,j)$ , over the axial slices. (b) Predicted locational distribution of the tumor pixels over the series of axial slices,  $h_{la}(j)$ , plotted in blue. The 2 local minima define the 2 boundaries of the predicted set of consecutive tumor slices, specified by the green frame, in comparison with the ground truth framed in red. The data sample is from the patient case 01412 of BraTS2021.....33

Figure 3.15 Coronal slices sampled from a 3D image, after the cropping operation to remove the tumor-free axial slices. (a)(e) Cropped coronal slice without tumor. The left half is somehow symmetrical to the right half. (b)(c)(d) Those with tumor. The left-right symmetry is much less than that in (a) or (e). ....34

Figure 3.16 (a) Asymmetry map  $\Delta H_c(i,j)$ , generated with the cropped coronal slice series of the patient case 01412 in BraTS2021. (b) Modulated 2D histogram of the coronal slice series,  $H_{mc}(i,j)$ , to be used as a coarsely predicted 2D histogram of the tumor region. (c) Two-D histogram of the pixels in the tumor region given by the ground truth,  $H_{Tc}(i,j)$ , over the coronal slices. ....34

Figure 3.17 (a) Sagittal slices sampled from an original series, before the 2 cropping operations. (b) Sagittal slices after the cropping operations applied to the axial and coronal slices. The first and the last sagittal slices are tumor-free, whereas the other 3 involve tumor regions. ....	35
Figure 3.18 (a) Two-D histogram $H_s(i,j)$ , given by the pixels in the cropped sagittal slice series of the patient case 01412 in BraTS2021. (b) Modulated 2D histogram of the sagittal slice series, $H_{ms}(i,j)$ , to be used as a coarsely predicted 2D histogram of the tumor region. (c) Two-D histogram of the pixels in the tumor region given by the ground truth, $H_{Ts}(i,j)$ , over the sagittal slices. ....	35
Figure 3.19 (a)(d)(g) Two-D histograms of minimum bounding box of axial, coronal and sagittal slice series. (b)(e)(h) Predicted 2D histograms of tumor pixels in axial, coronal and sagittal slice series. (c)(f)(i) Ground truth of the 2D histograms in (b)(e)(h). The data sample is from the patient case 01412 in BraTS2021. ....	37
Figure 3.20 (a) True gray level distributions of the pixels inside the tumor region, plotted in red, and that of outside the tumor region, plotted in green, of a 3D minimum bounding box. The blue dashed line indicates an assumed gray level threshold. The data sample is from the case 01452 of BraTS2021 dataset. (b) Predicted gray level distributions, obtained from the original 3D Flair image of the same case. (c) CDF derived from the predicted distribution of the tumor pixels shown in (b). The gray level threshold is 0.44, corresponding to the level of CDF = 20%. ....	38
Figure 3.21 (a) Slice from a 3D minimum bounding box generated from the data of the patient case 01418 of BraTS2021 dataset. (b) Slice of the coarse mask after the binarization. (c) Slice of the final binary mask after the morphological operation. (d) Slice of the true tumor mask. ....	39
Figure 3.22 (a)(b)(c) Predicted 2D gray level distributions of tumor pixels of axial, coronal and sagittal slice series. (d)(e)(f) True 2D gray level distributions of the case. (g) Predicted and true 1D gray level distribution of tumor pixels. (h) Predicted and true CDF of (g). The data sample is from patient case 01268 of BraTS2021 dataset. ....	41
Figure 3.23 (a)(b)(c) Predicted 2D gray level distributions of tumor pixels of axial, coronal and sagittal slice series. (d)(e)(f) True 2D gray level distributions of the case. (g) Predicted and true 1D gray level distribution of tumor pixels. (h) Predicted and true CDF of (g). The data sample is from patient case 01414 of BraTS2021 dataset. ....	42
Figure 3.24 Four examples of brain tumor detection by the proposed system. The 4 original MRI slices of Flair modality are found in the left column, the 4 detected binary tumor masks in the middle column and the true ones in the right column. The original data are from patient cases 01268, 01414, 01417, and 01420 of BraTS2021 dataset. ....	43
Figure 3.25 Boxplots of the results in Table 3.3. The tests have been done on 7 different sets of data samples. ....	47
Figure 4.1 Decomposing the 4-class classification task into 3 binary classification subtasks for WT, TC and ET, respectively. ....	52
Figure 4.2 Decomposing an image segmentation task into 2, for coarse location and for precise classification, respectively. ....	53
Figure 4.3 Detailed CNN structure of Module 1 for low-resolution location mask generation. ....	54
Figure 4.4 Block diagram of Module 2 of the CNN. ....	55
Figure 4.5 Detailed CNN structure of Module 2 for precise pixel-wise classification. ....	56

# List of Tables

Table 3.1 Numbers of patient cases in BraTS dataset.....	40
Table 3.2 Similarity between the predicted and true gray level distributions. The data are generated by testing the 1251 patient cases of BraTS2021 dataset .....	45
Table 3.3 Dice score, Sensitivity and FDR of tumor detected by the proposed system .....	46
Table 3.4 Comparison of the results of the proposed system with those of other conventional non-CNN systems....	47
Table 3.5 Comparison of the results of the proposed system with those of CNN systems .....	48
Table 4.1 Details of the CNN configuration of Module 1 .....	57
Table 4.2 Details of the CNN configuration of Module 2 .....	57
Table 4.3 Number of patient cases in different versions of BraTS dataset .....	59
Table 4.4 Weight values of loss function for the two CNN modules .....	60
Table 4.5 Test results on BraTS validation sets, assessed by online platforms. ....	61
Table 4.6 Computation cost of the proposed CNN system .....	61
Table 4.7 Performance comparison Dice and Sensitivity scores resulting from the tests on BraTS validation sets...	62

# Table of Acronyms and Abbreviations

ADAM	Adaptive moment estimation
ASCNN	Application specific convolutional neural network
BN	Batch normalization
BraTS	Brain tumor segmentation
CBICA IPP	Center for Biomedical Image Computing and Analytics Image Processing Portal
CDF	Cumulative distribution function
CNN	Convolutional neural network
ED	Edema
ET	Enhancing tumor
FDR	False discovery rate
FN	False negative
FP	False positive
GN	Group normalization
HF	High-frequency
IN	Instance normalization
LF	Low-frequency
LN	Layer normalization
MF	Mid-frequency
MRI	Magnetic Resonance Imaging
MAC	Multiply-accumulate operation (MAC)
NET	Non-enhancing/necrotic core
ReLU	Rectified linear unit
SNR	Signal-to-noise ratio
TC	Tumor core
TP	True positive
WT	Whole tumor

# Table of Symbols

$Dice$	Dice score
$FDR$	False discovery rate
$FN, FP, TP$	False negative, false positive, true positive
$f_{m1}(i), f_{m2}(i)$	Modulation functions for coarse and fine prediction
$H(i,j)$	Two-D histogram
$H_{ba}(i,j), H_{bc}(i,j), H_{bs}(i,j)$	Two-D histograms of minimum bounding box sliced along axial, coronal and sagittal planes
$H_{pa}(i,j), H_{pc}(i,j), H_{ps}(i,j)$	Fine predicted 2D histogram of axial, coronal and sagittal slices
$H_s(i,j)$	Two-D histogram of sagittal slices series after the first and second cropping operations
$H_T(i,j)$	Two-D histograms of ground truth of tumor
$H_{Ta}(i,j), H_{Tc}(i,j), H_{Ts}(i,j)$	Two-D histograms of ground truth of tumor from axial, coronal and sagittal slice series
$H_{left}(i,j), H_{right}(i,j)$	Two-D histogram of left and right halves of a brain
$H_{ma}(i,j), H_{mc}(i,j), H_{ms}(i,j)$	Modulated 2D histograms, also named coarsely predicted 2D histograms, of axial, coronal and sagittal slice series
$h_{la}(j)$	One-D predicted locational distribution of the tumor pixels in axial slice series
$h_{tumor-free}(i), h_{tumor}(i)$	One-D gray level distribution of pixels in tumor-free region and tumor region
$h_{tumor-free-half}(i), h_{tumor-half}(i)$	One-D gray level distribution of pixels in the tumor-free half of a brain, that in the other half with tumor
$h_{yf}(i), h_T(i)$	One-D gray level distribution of the identified half that is presumed tumor-free, Truncation of $h_{yf}(i)$
$i$	Ranged (0, 1), representing the gray levels
$j$	Index of slice
$Loss_1, loss_2$	Loss functions of the two CNN modules
$loss_{Dice}$	Dice loss
$loss_{FN}, loss_{FP}$	Losses focus on false negative and false positive
$max_T, min_T$	Pre-determined highest and lowest bin-heights of truncation
$N_{left}, N_{right}$	Number of high-gray-level pixels in the left half of a brain, that in the right half of a brain
$N_s$	Number of slices
$Sens$	Sensitivity or true positive rate
$S_n \times C_n \times A_n$	Size of 3D brain image

$SSIM(x,y)$	Structural similarity index measure
$w_1, w_2$	Weight coefficients of loss function
$\Delta H_a(i,j), \Delta H_c(i,j)$	Asymmetry map of axial and coronal slice series
$\mu$	Mean value
$\sigma$	Standard deviation

# Chapter 1

## Introduction

The work presented in this thesis is about developing computer vision systems for brain tumor detection, a difficult task performed by medical professionals. Such systems do not intend to replace them, but reduce their workload to facilitate timely diagnosis and treatments.

In Subchapter 1.1, the task of brain tumor detection is defined in an elaborated manner, the other terminologies used for the same task are explained. The challenges in this particular topic area are described. In Subchapter 1.2, the motivation and objective of the work presented in this thesis are specified. In Subchapter 1.3, the scope of the work is described methodologically, and the organization of the materials of the thesis is presented.

### 1.1 Background and Challenges

Brain tumors cause serious health problems. Timely discovery of such tumors is very critical for diagnosis and treatment. While 3D brain scanning by magnetic resonance imaging (MRI) has been widely used in clinical practice, analyzing brain images to detect tumors and their development is usually performed by highly trained medical professionals. Limited resources may lead to a lengthy waiting for detection results. Developing computer vision systems for fully automatic brain tumor detection can help to reduce the work load of the medical specialists and improve the chance of timely diagnosis and treatments.

By MRI scanning, each patient case can have 3D brain images of four modalities, namely Flair, T2, T1 and T1c. Multi-modality MRI allows a better display of different developments inside tumor regions, for example, edema (ED), non-enhancing/necrotic core (NET) and enhancing core (ET). Each 3D brain image is usually presented as a series of 2D slices. As an example, the center slices from the images of the 4 modalities, acquired from one case, are presented in Figure 1.1 (a) ~ (d).

The function of a brain tumor detection system is, in general, to specify the tumor regions with pixel-wise precision in a 3D image. In terms of image processing, it is to segment precisely the tumor regions with respect to the background. Hence, it can be seen, on one hand, as an image segmentation problem. On the other hand, some may consider the detection as a classification problem, as one needs to classify each pixel, whether it belongs to the tumor region or not. For example, in the case shown in Figure 1.1, the detection results in a mask shown in Figure 1.1 (e). In this mask, all the pixels classified as background, i.e., outside the tumor region, are given the gray level value of zero, and the other pixels, classified as inside ones, are given non-zero values. If a whole tumor region is further segmented into 3 different intra-tumoral regions, namely ED,

NET, and ET, instead of binary classification of background versus object region, the system will need to perform a classification of multi-classes.

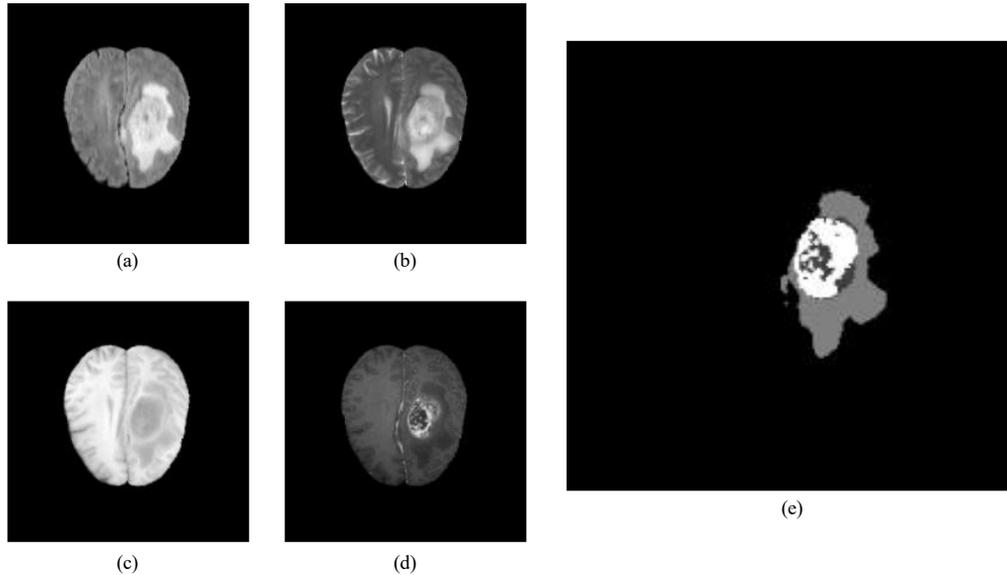


Figure 1.1 (a)(b)(c)(d) Center slice of the brain images, acquired by MRI of Flair, T2, T1 and T1c modalities, respectively, from a patient.

(e) Tumor mask obtained from the 4 slices. The non-zero pixels are found in the tumor region that is further segmented into 3 kinds of intra-tumoral regions, namely ED (gray), NET (darker gray), and ET (brightest).

Tumors can be found almost everywhere in a brain, and their appearances and sizes can vary enormously, as 2 examples shown in Figure 1.2, making the tumor detection a very challenging task. In case of computer vision systems, there are 2 approaches to the tumor detection.

- *Conventional filtering approach.* This approach requires very low level of computing power to do the task. Moreover, the performance consistency is excellent, i.e., the results are completely reproducible. However, to handle a wide range of feature variations in brain images, one needs a very large number of filters and it is unlikely to determine the parameters of those filters manually and in an optimal manner. Thus, the processing quality of such filtering systems for brain tumor detection is usually limited.
- *Machine learning approach by convolutional neural networks (CNN).* In this case, the filtering parameters are determined by means of progressive update in a training process. Thus, one can apply a sufficient number of filters to handle various kinds of feature data. However, to design an efficient CNN system, one should address particular problems in neural networks. Firstly, such a system requires a huge amount of computing power to train it and then to perform the functions. Secondly, the quality of the system training may be affected by many elements, e.g., limitation on accessible data resources and employable training methods. Also, as there are many random elements in a training process, repeatedly re-training a CNN without modifying its config will result in different systems. Hence results produced by a CNN can not be exactly reproduced.

It is evident that a brain tumor detection system by conventional filtering can operate at a low computational cost, but the detection accuracy is its weak point. On the contrary, it is possible to achieve a high accuracy by means of a CNN system, but its design is not less challenging, due to the complexity in its structure and training process.

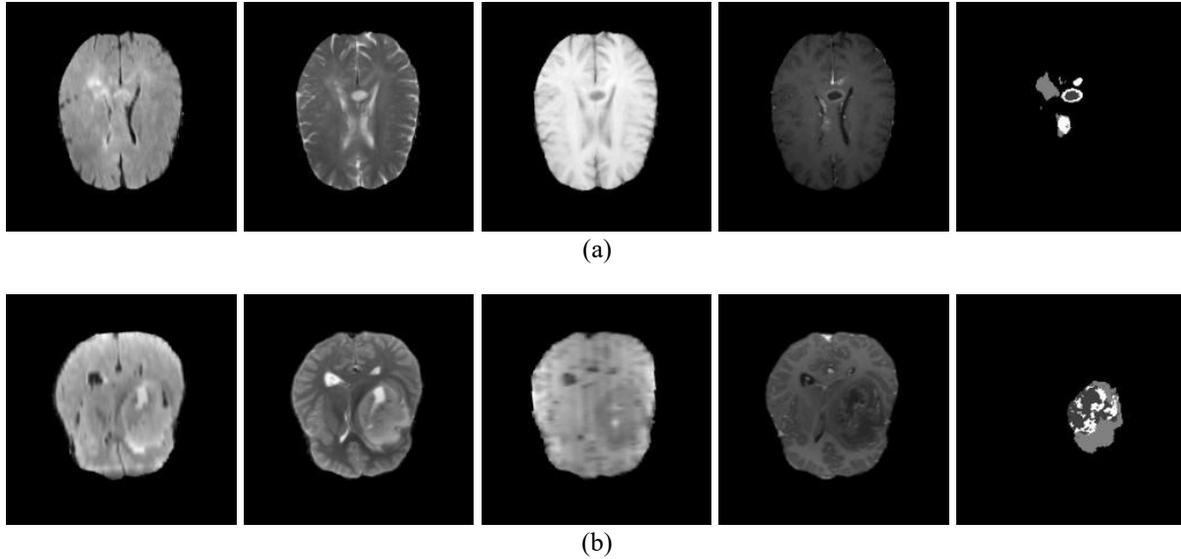


Figure 1.2 (a) Example 2D slices of Flair, T2, T1 and T1c modalities, and the corresponding expected output. (b) Those of another patient case.

## 1.2 Motivation and Objectives

The 2 approaches specified in Subchapter 1.1 seem to be complement each other, in terms of processing quality and computation cost. In general, a good detection system is expected to have both good quality and low cost, because a poor accuracy makes it useless and high cost limits its implementation & applications. Hence, we are motivated to explore all possible avenues in this topic area in order to find effective design methods to achieve both.

The objective of the research work presented in this thesis is to develop computer vision systems for brain tumor detection in 3D brain images. The emphasis is on a high computation efficiency, i.e., achieving a good processing quality at a very low computation cost to enable an easy implementation and a wide range of applicability of the systems.

To achieve the objective, we will propose 2 different methodologies for brain image data processing. The work includes their applications in designing 2 systems performing, respectively, 2 tasks specified as follows.

- The first system is able to detect brain tumor locations, sizes and shapes in 3D brain images, delivering binary tumor masks. It should require a very small volume of computation to make it possible to operate in an ordinary desktop or laptop, so that such a detection function can be performed very easily, accessible everywhere.

- The second one is expected to perform a precise brain image segmentation to identify not only the tumor regions, but also intra-tumoral segments. In other words, every pixel should be classified into one of the 4 categories, namely ED, NET, ET and the background, resulting in 4-class tumor masks. Though it needs more computations than the first system, all the research efforts will be made to minimise the computational complexity with a view to optimizing the processing and the training. Also, such a system will be implementable and functionable in a computational resource restricted environment.

The 2 methodologies should be different to target the design of the systems performing tasks of different complexity. The scope of the work for their development and applications is presented in the following subchapter.

### 1.3 Scope and Organization

The 2 methodologies for brain data processing will be proposed to design effectively brain tumor detection systems performing tasks of different complexity. Though both aim to achieving a good processing quality at the lowest computational cost, they should be different to suit the complexities of the tasks of the systems.

The first methodology is proposed to develop a tumor detection system performing a binary classification of the pixels in a 3D brain image and delivering a binary brain tumor mask. To make it possible to implement the system in an ordinary personal computer, the system will be designed with deterministic models, no need for network training nor a large number of filtering kernels. As tumors are related to some abnormalities in brains, statistical characters of 3D data of brain scans will be analyzed and the symmetric properties of brain structures be explored to determine an effective method to detect any abnormalities. Moreover, as the detailed structures of human brains may vary from case to case, techniques of adaptive processing will be developed and employed in the design process to obtain a good detection quality.

The second methodology is for CNN design. The second system is expected to perform a precise multi-class classification of the pixels of a 3D brain image, which is hardly achievable by a conventional filtering system, and a CNN framework is thus the core of the data processing. To maximize the computation efficiency, we will explore the following avenues.

- Design paradigm of application specific convolutional neural network (ASCNN). If the system is designed specifically for a particular task, without any redundancy, the computational complexity will be minimized.
- Decomposition of a multi-classification task into simpler subtasks. By doing so, one can develop multiple simple CNNs for each of the subtasks.
- Optimization of network training and optimization of computing structure. Let the 2 help and complement each other.
- Maximum use of available data resources. It can also help to improve the training.

The thesis is organized as follows. The background of image processing, by conventional filtering and CNN, and related work on brain tumor detection are presented in Chapter 2. In

Chapter 3, the methodology for designing knowledge-based brain tumor detection systems is proposed and the design/test process of such a system is presented in detail. Chapter 4 is dedicated to the presentation of our CNN design methodology and its application of a CNN system for brain tumor segmentation. The conclusion of the work presented in this thesis is found in Chapter 5.

# Chapter 2

## Background and Relevant Work

Many researchers have made significant efforts to design systems for automated brain tumor segmentation. This chapter is dedicated to describe the basics of the methods for medical image processing and the state-of-the-art systems for brain tumor segmentation.

The basics of filtering and CNN are found in Subchapter 2.1. Subchapter 2.2 and 2.3 are to present the related works of conventional non-CNN systems and CNN systems. Subchapter 2.4 is a summary of the chapter.

### 2.1 Basics of Filtering and CNN for Image Processing

The conventional filtering method and the CNN method are widely used to design systems for image processing. The basics of the two methods are found in the following two subchapters.

#### 2.1.1 Basics of Conventional Filtering

Conventional filtering system is able to extract various features under extremely low computation cost. The convolution operation of the filtering is to apply a kernel to the input image, transforming the input into the desired features. The parameters of the kernel are deterministic and are set by the designer, without any training process.

There are two kinds of filter commonly used in practice. i) Low-pass filter. It is to extract features of low-frequency signals, or to attenuate the noise so that increase the signal-to-noise ratio (SNR). Gaussian filter, average filter and median filter are widely used low-pass filters. ii) High-pass filter. It is to extract features of high-frequency signals and to detect the edges of the object. Some typical high-pass filters, e.g., Sobel filter, Prewitt filter and Laplacian filter, are discrete differentiation operator, computing the gradient of the image intensity.

Beside the two kinds of typical filter, Gabor filter [1] is more powerful for texture analysis, edge detection and feature extraction. It is able to extract any specific frequency features in specific orientations, and is defined as a sinusoidal wave multiplied by a Gaussian function, as found in the following equation.

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \sin\left(2\pi \frac{x'^2}{\lambda} + \psi\right) \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \quad (1)$$

where  $x' = x\cos(\theta) + y\sin(\theta)$  and  $y' = -x\sin(\theta) + y\cos(\theta)$ .  $\lambda$  and  $\theta$  are the wavelength and orientation.  $\psi$ ,  $\sigma$  and  $\gamma$  are the phase offset, sigma of Gaussian function and spatial aspect ratio. Example kernels with different  $\lambda$  and  $\theta$  are illustrated in Figure 2.1.

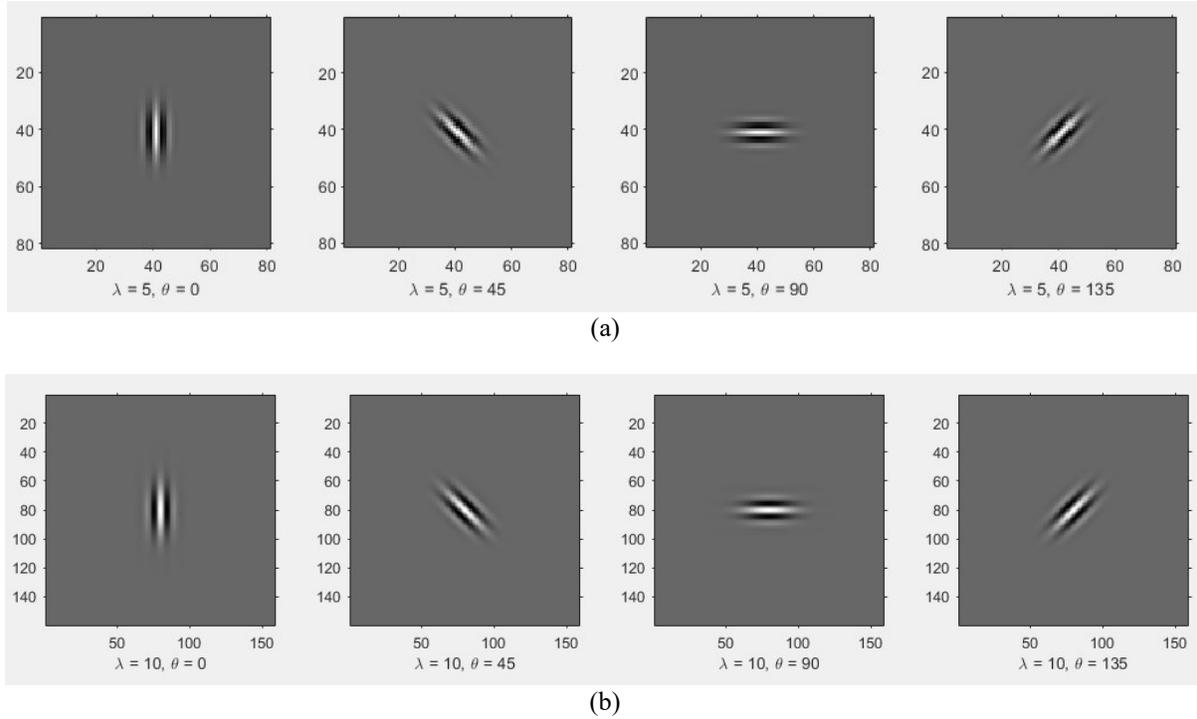


Figure 2.1 Example kernels of Gabor filter.

- (a) Wavelength  $\lambda = 5$ , orientation  $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$ .
- (b) Wavelength  $\lambda = 10$ , orientation  $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$ .

Besides the conventional filtering method with deterministic kernels, the CNN method with trainable kernels has become much more popular in recent years. The basics of the CNN are found in the following subchapter.

## 2.1.2 Basics of CNN

In general, large scale CNN is able to handle complex tasks, but requires a huge number of computation resources. The CNN involves multiple convolution layers, and other critical components e.g., normalization, activation function, pooling (down-sampling), up-sampling and so on [2][3]. The parameters of the convolution kernels are initialized randomly and then updated iteratively by a backpropagation operation of the training process. The components relevant to the work of this thesis are as follows.

### 1) Convolution layer

In a convolution layer, assume the convolution kernel size is  $3 \times 3$ , each element of the output map is generated from the corresponding input element and its 8 neighbors, as an example presented in Figure 2.2 (a). In some case, the convolution mode can be modified to acquire better

output. For instance, i) dilation convolution illustrated in Figure 2.2 (b), its receptive field is extended significantly, which means that each output is generated from a larger neighborhood, ii) convolution with larger stride, e.g.,  $stride=2$  presented in Figure 2.2 (c), the input data is pooled into a smaller size, so the computation cost is reduced significantly and the signal density is increased.

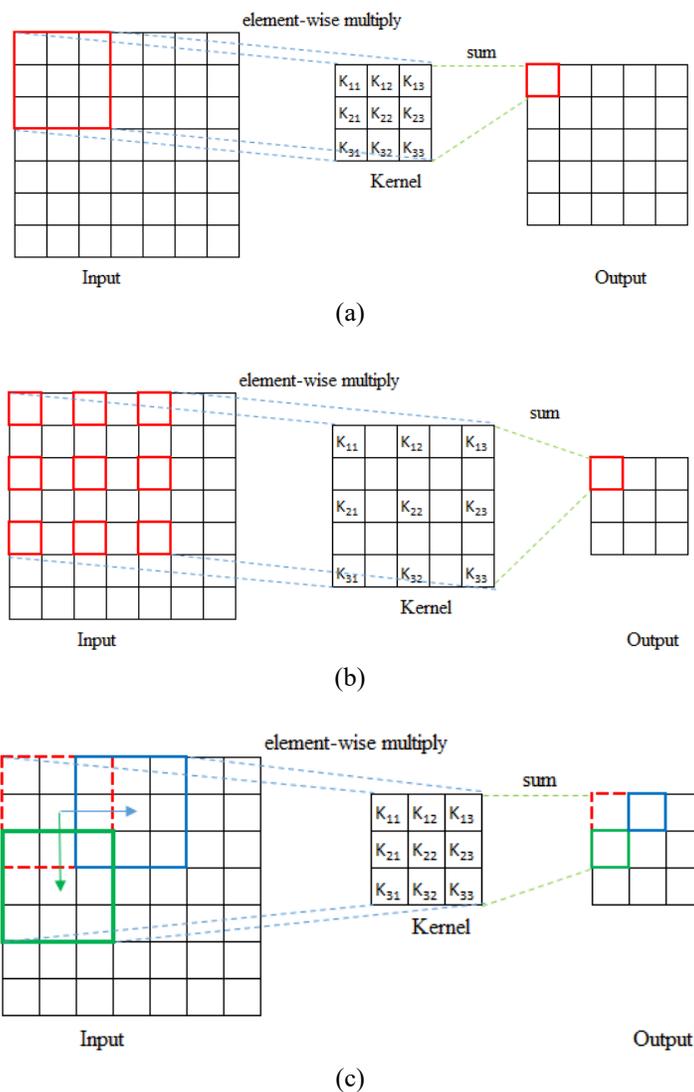


Figure 2.2 (a) Convolution operation. (b) Dilation convolution,  $dilation=2$ . (c) Convolution with  $stride=2$ .

## 2) Normalization

Normalization is performed to the input data and the internal convolutional data of the CNN, in order to uniform the gray-level range of the data and facilitate the convergence of the training process. Its calculation is expressed as  $y = (x - \mu) / \sigma$ , where  $\mu$  and  $\sigma$  are the mean and standard deviation of a set of data. The main difference among the widely used normalization methods, i.e., batch normalization (BN) [4], layer normalization (LN) [5], instance normalization (IN) [6] and group normalization (GN) [7], is the acquisition of  $\mu$  and  $\sigma$ , which is presented in Figure 2.3. As

for the batch normalization, the  $\mu$  and  $\sigma$  is channel-wise, i.e., each channel has its own  $\mu$  and  $\sigma$  generated from the whole mini-batch.

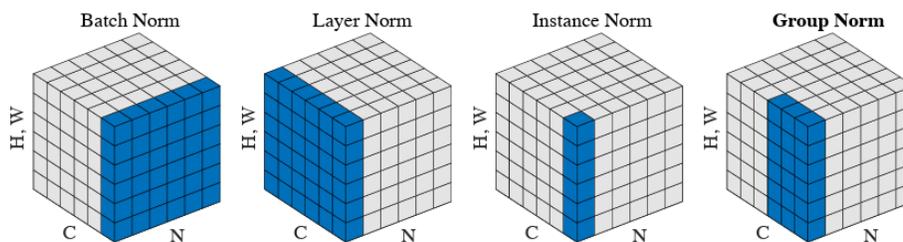


Figure 2.3 Widely used normalization methods [7], where  $C$  is the number of channels,  $N$  is the batch size, and  $H$  &  $W$  are the height and width of an image.

### 3) Non-linear activation function

In general, the expected transfer function of a CNN system for image segmentation is a non-linear function, because the output of the CNN is usually a set of simple binary or multi-class labels, but the input image has a large number of intensity levels, e.g., 255. There is no linear function to generate the simple output label from such a complicated input image. Since convolution operation is linear function, the CNN requires additional non-linear component. A non-linear activation function should be continuously differentiable, as the training process of the CNN is gradient-based optimization.

Rectified linear unit activation function (ReLU) [8] is one of the most computation-efficient activation function. It is defined as  $f(x) = \max(0, x)$ . Although it is non-differentiability at  $x=0$ , the derivative value at this point can be simply set to be 0 or 1.

Sigmoid function is another widely used activation function. It is defined as  $f(x) = 1 / (1 + e^{-x})$ , bounds the range of the data into  $(0, 1)$ . Thus, it is often performed to the data of the last layer of a CNN in order to generate the probabilities of each class.

### 4) Pooling and up-sampling

Pooling operation is to down-sample the image, thereby increasing the signal density and reducing the computation cost. The most commonly used pooling methods include max pooling and average pooling, as well as the trainable convolution with  $stride=2$ .

For certain medical image processing tasks, e.g., image segmentation, the dimension of the output should be identical to that of the input. Therefore, up-sampling and pooling operations are performed in pairs. Commonly used up-sampling methods include non-trainable nearest neighbor and bilinear interpolation, as well as the trainable deconvolution (transposed convolution).

### 5) Loss function

A loss function, such as cross-entropy loss [9] or Dice loss [10], is used to measure the degree of dissimilarity between the predicted results and the ground truth. Cross-entropy loss is widely

used for classification problem, as well as image segmentation problem, which is regarded as pixel-wise classification task. Dice loss is typically used to measure the dissimilarity between two regions. They are defined as follows.

Cross-entropy loss:

$$L_{cross-entropy} = -\sum_i^C y_i \log \hat{y}_i \quad (2)$$

where  $y$  and  $\hat{y}$  are the label of ground truth and the predicted probability, respectively, e.g.,  $y \in \{p, 1 - p\}$  and  $\hat{y} \in \{q, 1 - q\}$ .  $C$  is the number of classes.

Dice loss:

$$L_{Dice} = 1 - \frac{2(|X| \cap |Y|)}{|X| + |Y|} \quad (3)$$

where  $|X|$  and  $|Y|$  are the number of elements, e.g., pixels, of the two sets.

## 6) Backpropagation

Backpropagation [11][12] is a gradient-based method to update the trainable parameters in the training process of a CNN. A very simple parameter updating by backpropagation operation is defined as follows.

$$\omega_{new} = \omega_{old} - \eta \nabla Q(\omega) \quad (4)$$

where  $\omega$  is trainable parameter.  $Q(\omega)$  is loss value.  $\nabla Q(\omega)$  is gradient.  $\eta$  is learning rate.

The components of the CNN mentioned above can be assembled into various complex systems. These systems are then trained using various strategies and achieve impressive performance in image processing.

Many conventional non-CNN systems and CNN systems have been developed for brain tumor segmentation. The state-of-the-art systems related to the work of this thesis are found in the following subchapters.

## 2.2 Related Work of Conventional Non-CNN System for Brain Tumor Detection

Conventional non-CNN systems require little computational cost, allowing fast object detection. A good number of such systems have been developed and reported for brain tumor detection.

In general, a conventional non-CNN system performs 2 functions, feature extraction from the input data and classification operation applied to the extracted features. As the 2 functions can be performed in different ways, there are varieties systems reported in this topic area. Feature extraction can be performed by means of filters. Gabor filters are commonly used for texture analysis and feature extraction in conventional non-CNN systems. The extracted features are then applied to various classifiers. For example, the method of Extremely Randomized Trees can be used for this purpose [13][14]. In some systems, Gabor filtering method is combined with Support

Vector Machine (SVM) to detect brain tumors [15][16]. One can also combine Gabor filtering and K-means clustering methods for feature extraction and SVM together with Random Forest (RF) for classification to improve the detection result [17]. Gabor filtering and Walsh-Hadamard transform (WHT) can be used for feature extraction, and Fuzzy C-Means clustering for classification [18].

Some region-based image segmentation methods are used to detect brain tumor, e.g., homogeneity- and object-feature based Random Walks (HORW) [19], and multi-agent adaptive region grow [20]. In these methods, initial seed points should be selected, and the neighboring pixels are examined and determined whether they belong to the same region of the seed.

The feature information concerning brain tumors can also be extracted by measuring asymmetry of a brain structure, as a tumor can make its left-right halves less symmetrical. The degree of asymmetry can be measured by calculating, for example, the pixel-by-pixel difference of the two 3D halves. Then, the 3D data resulting from such a calculation are used as feature data to be applied to a classifier of Random Forest [21].

It should be noted that the 3D data produced by the dissimilarity measures represent all the asymmetry caused not only by the tumors but also by the differences of texture details in healthy parts. To make the latter less pronounced, one can measure the degree of asymmetry of the 2 halves based on their statistical presentations, e.g., gray level distributions, instead of their 3D data. For example, the difference between the gray level histograms of the normal hemisphere and the pathological hemisphere of a brain are calculated by a very simple subtraction operation, as illustrated in Figure 2.4 [22]. One can also generate multiple pairs of histograms, each of which given by 2 subregions located symmetrically in the 2 halves, and calculate the degree of dissimilarity by Bhattacharya coefficient method to find the likely tumor location [23]. By these measures, the dissimilarity of healthy parts in image details may be less pronounced, but it can still be more visible than that caused by tumors.

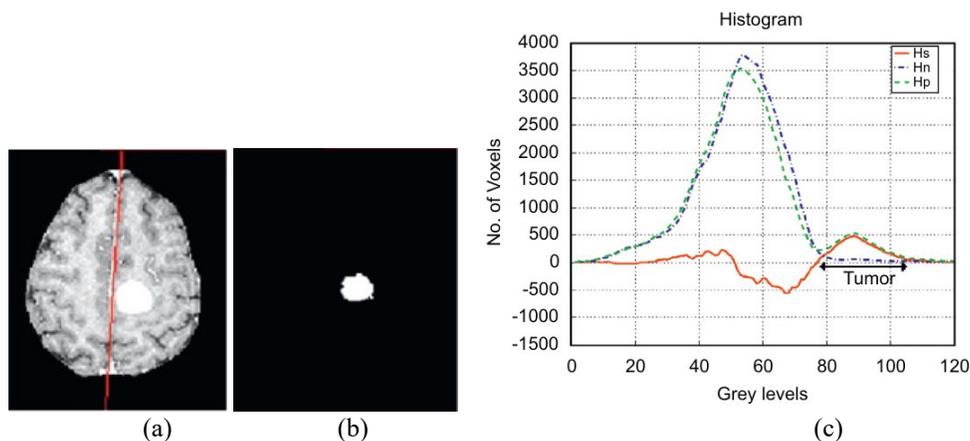


Figure 2.4 (a) Example of slice of brain image with tumor. Tumor reduces the degree of the natural left-right symmetry of the brain. (b) Tumor mask. (c) Gray level distributions of the left and right halves of the slice.  $H_n$  is the histogram of the half without tumor.  $H_p$  is the histogram of the half with tumor.  $H_s$  is the histogram difference between  $H_n$  and  $H_p$  [22].

Besides the conventional non-CNN systems for fast brain tumor detection, many CNN systems have been developed for high-quality brain tumor segmentation. The details of the CNN systems published in recent years and related to the work of this thesis are found in the next subchapter.

## 2.3 Related Work of CNN System for Brain Tumor Segmentation

A CNN system has the potential to handle challenging problems, as it can be very large scale to extract and classify complex features. Many CNN systems have been developed and reported for brain tumor segmentation.

U-Net [24] is the most popular CNN for medical image processing. It composed a contracting path for feature extraction and an expanding path for precise segmentation, as presented in Figure 2.5. In the contracting path, the size of the feature map is reduced gradually through pooling operations, while the number of channels increases accordingly. The expanding path is a symmetrical version of the contracting path, where the size of the feature map is gradually increased while the number of channels decreased. Several skip connections provide the feature maps generated by the contracting path for the expanding path. Many state-of-the-art CNN systems for brain tumor segmentation are designed on the basic of U-Net [25][26][27][28]. In clinical routine, some of the MRI modalities can be missing, because of time constraints and/or image artifacts (such as patient motion), so many U-Net based CNNs are designed for brain tumor segmentation with missing MRI modalities [29][30][31].

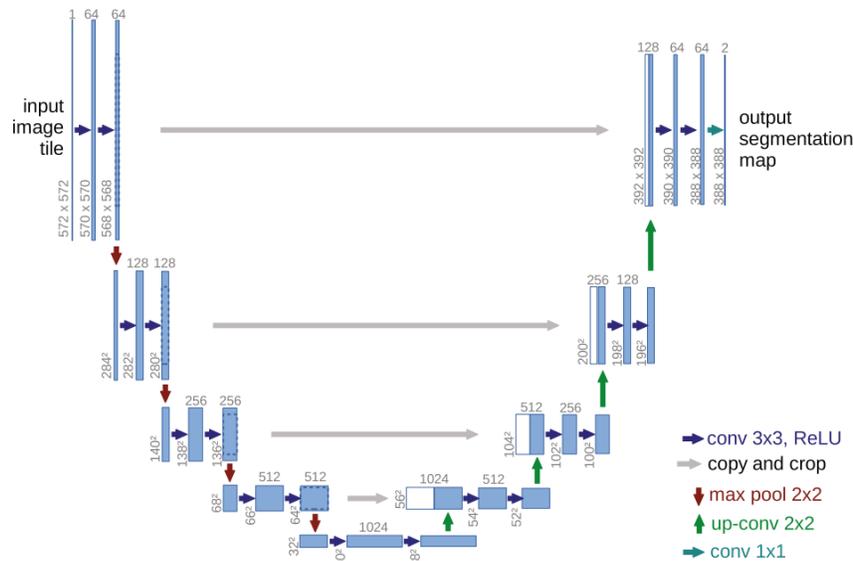


Figure 2.5 Architecture of U-Net [24].

The task of brain tumor segmentation can be decomposed into several subtasks, as it is a multi-task problem, i.e., detecting the whole tumor (WT) region and segment the intra-tumoral regions of tumor core (TC) and enhancing tumor (ET) inside the WT region. Residual cyclic unpaired

encoder-decoder network (RescueNet) [32] is a CNN system involving 3 networks, as presented in Figure 2.6. Network I is responsible for detecting the WT region, while Network II and Network III are to segment the TC and ET regions.

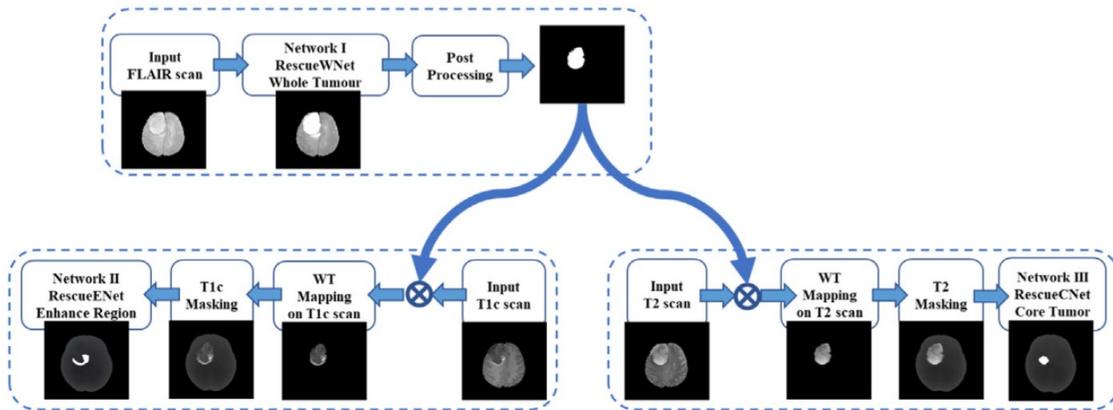


Figure 2.6 Structure of residual cyclic unpaired encoder-decoder network (RescueNet) [32].

The three subtasks can be performed by a single CNN with three blocks in parallel or in a cascade. One-pass Multi-task Network (OM-Net) [33] is a U-Net based 3D CNN with 3 parallel pathways for three subtasks. Its block diagram is found in Figure 2.7. The three sets of input data are applied to a shared model, then the feature maps generated by the shared model are split into three sections and applied to the three classifiers. The output of the former classifier is utilized as the cross-task guidance for the later classifier. The subtasks can also be performed by multiple blocks structured in a complex architecture, e.g., task-structured brain tumor segmentation network (TSBTS net) [34]. The block diagram of TSBTS net is presented in Figure 2.8. Module I of the green block is responsible for WT detection, while Module II of the red block and Module III of the blue block are for TC and ET segmentation, respectively.

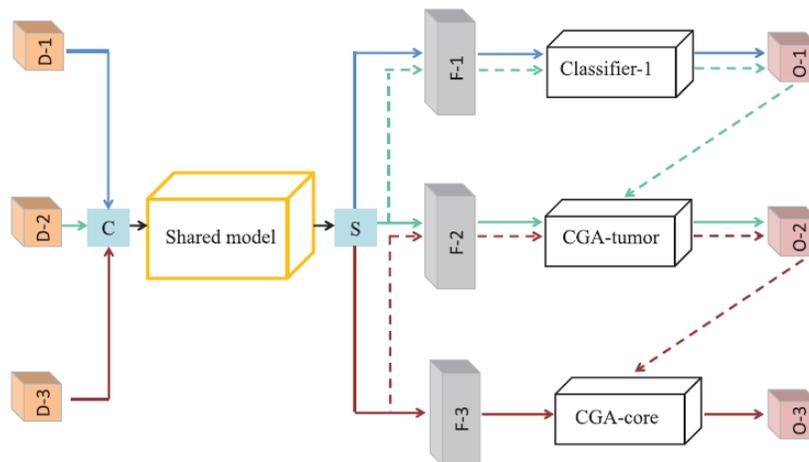


Figure 2.7 Block diagram of One-pass Multi-task Network (OM-Net) [33].

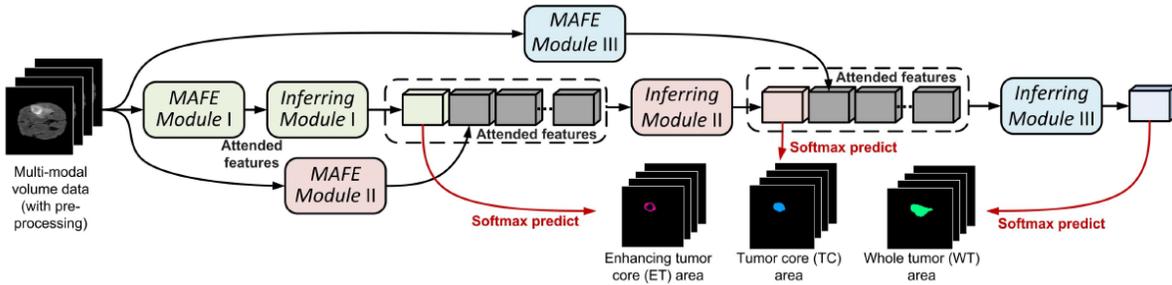


Figure 2.8 Block diagram of task-structured brain tumor segmentation network (TSBTS net) [34].

Besides using multiple CNNs for several subtasks, many CNN systems involve several networks in order to make a good use of the 3D input data. A 3D brain image can be sliced into three series of 2D slices in three directions, namely axial, coronal and sagittal. The 3 sets of the input data can be fed into three 2D networks, respectively, as an example of multi-view dynamic fusion framework (MVFusFra) [35], which structure is illustrated in Figure 2.9. Then, the 3 output are fused into 3D output. A CNN system can also incorporate 3 networks of 3D, 2D and 2.5D for high-quality segmentation [36], as an example presented in Figure 2.10. Unlike the plain 2D or 3D network, the input of the 2.5D is a group of adjacent slices, while the output is a 2D slice.

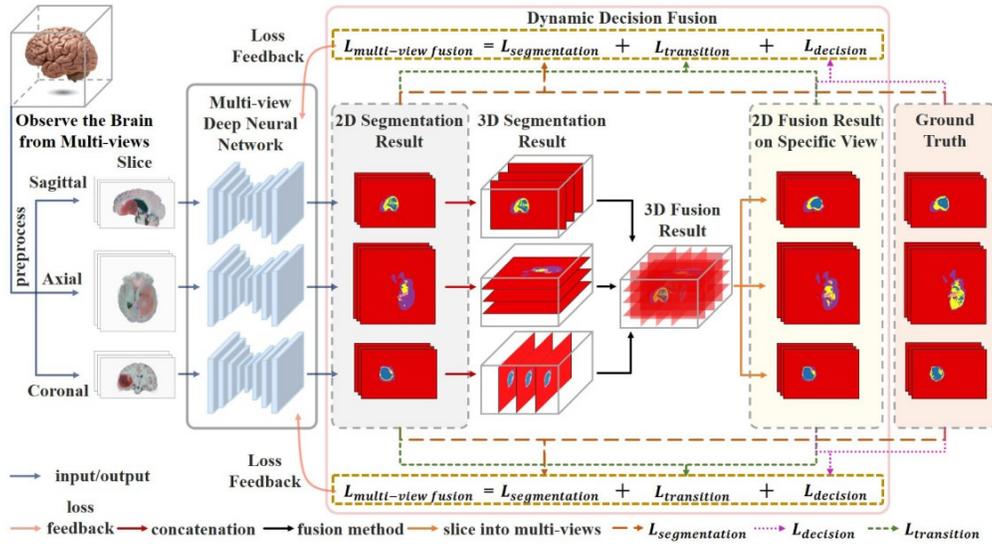


Figure 2.9 Structure of multi-view dynamic fusion framework (MVFusFra) [35].

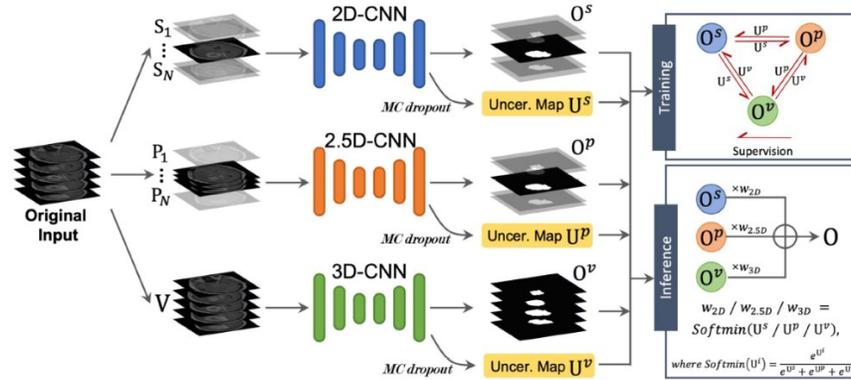


Figure 2.10 CNN system with 3 different dimensional networks [36].

In addition to optimizing the CNN structure, much research has focused on enhancing the training process of CNN systems. Deep supervision is a widely used technique for this purpose. It is to calculate the loss values not only for the final output layer but also for several intermediate layers of a CNN. In the U-Net based CNNs, the deep supervision can be performed to various spots of the network, e.g., i) the bottom layer [37], as presented in Figure 2.11, ii) the expanding path [38], as found in Figure 2.12, iii) the contracting path[39], as presented in Figure 2.13, and iv) both of the bottom layer and the expanding paths [40], as presented in Figure 2.14. Acquiring appropriate ground truth data for a specific supervision spot in the hidden layers of a CNN is challenging. The feature maps of the supervision spot are not directly comparable to the final ground truth, because they differ both in data content and data dimensions. Thus, additional convolution layers along with up-sampling operations can be performed to the feature maps to make them comparable to the final ground truth [38][39][40][41]. Besides, one can also down-sample the final ground truth to generate the ground truth data for the specific supervision spot, and additional convolution layers are still required for the feature maps [37].

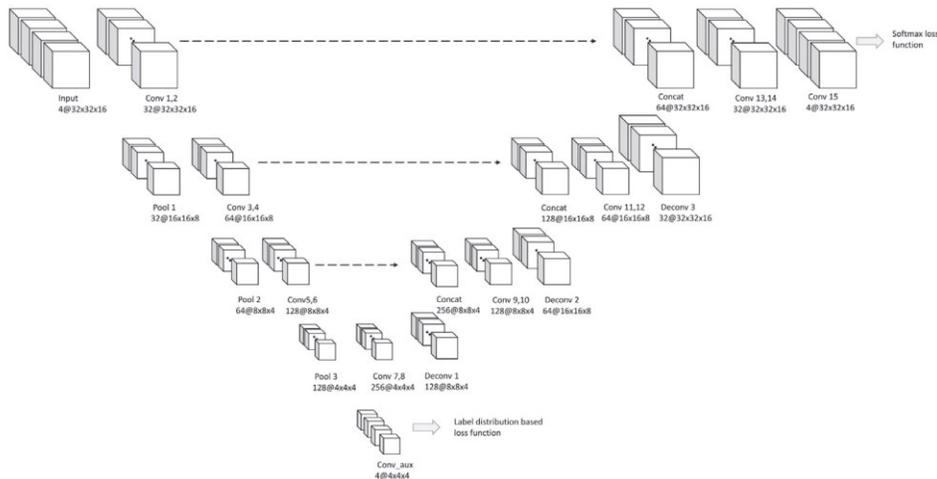


Figure 2.11 Block diagram of Dual-force U-Net (DF-U-Net) [37].

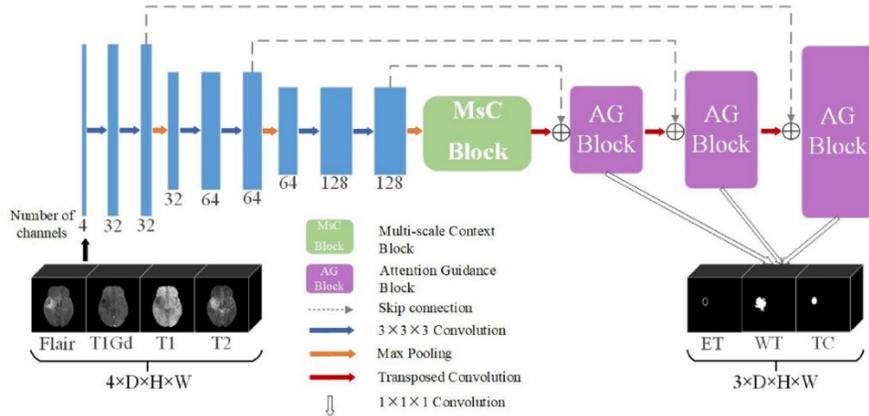


Figure 2.12 Block diagram of multi-scale context and attention mechanisms network (MsANet) [38].

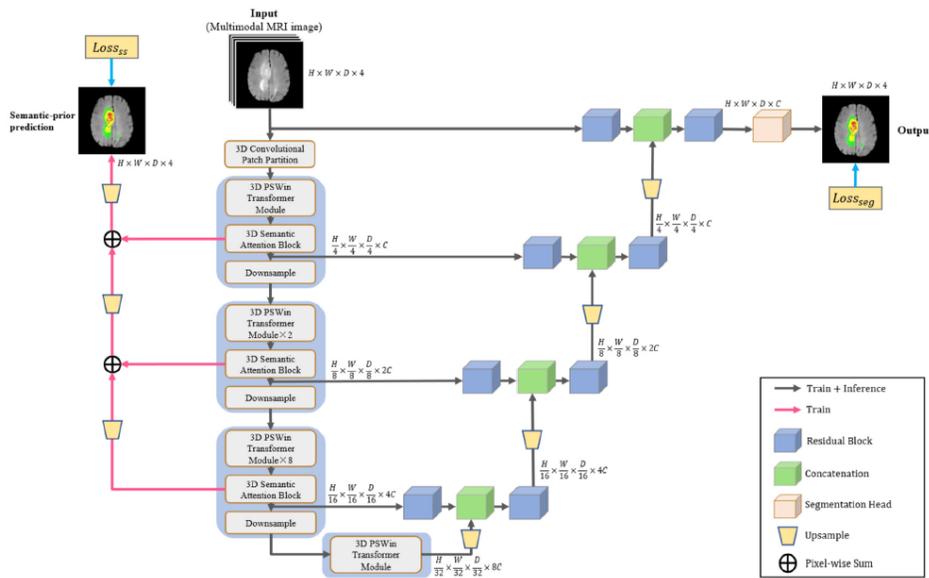


Figure 2.13 Block diagram of 3D Parallel Shifted Window-based Transformer module (3D PSwinBTS) [39].

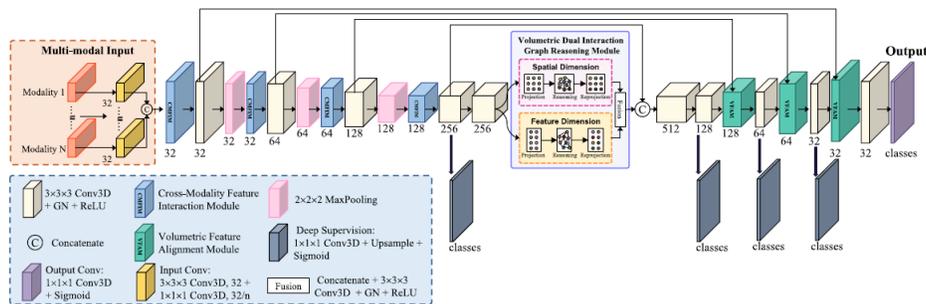


Figure 2.14 Block diagram of Aligned Cross-Modality Interaction Network (ACMINet) [40].

The systems mentioned above show that significant efforts have been made for automated brain tumor segmentation. However, they can still be optimized for higher processing quality and lower computational cost.

## **2.4 Summary**

In this chapter, the basics of conventional filtering and CNN, which are relevant to the work of this thesis, are introduced. Moreover, the methods of the state-of-the-art systems, conventional non-CNN systems and CNN systems, for brain tumor segmentation are presented. In general, conventional non-CNN systems are able to achieve fast object detection with very low computational cost, but they struggle to segment intra-tumoral regions with high precision. In contrast, CNN-based systems have the potential to achieve high processing quality, but require extensive computational resources and large amounts of training data.

In this work, we propose methodologies to develop high-efficient systems for high-quality brain tumor segmentation. Both conventional non-CNN methods and CNN methods are employed to design the high-performance systems.

## Chapter 3

# Knowledge-Based System for Whole Tumor Detection

The objective of this part of the work is to develop a system detecting brain tumor locations, sizes and shapes in 3D brain images, delivering binary tumor masks. It aims at achieving a good processing quality at the lowest computation cost so that the system can operate in an ordinary desktop or laptop, making such a brain tumor detection accessible everywhere.

Tumors can be found everywhere in a brain and appear in very different sizes, shapes and texture patterns. They can change from case to case with little coherence, as two examples of brain image slices illustrated in Figure 3.1 (a) and (c). A brain tumor detection system is expected to generate a 3D tumor mask covering, in a pixel-wise precision, the entire tumor region in all the slices. Hence, the detection by computer vision is a very challenging task. It becomes even more challenging under the condition of very limited computing power.

This chapter is organized as follows. The methodology is presented in Subchapter 3.1. A new presentation of the pixels distribution and the design of the proposed system are found in Subchapters 3.2 and 3.3. Subchapter 3.4 is dedicated to the performance evaluation. A summary is found in Subchapter 3.5.

The methodology and the design of the system are also presented in the research paper titled “Brain tumor detection based on a novel and high-quality prediction of the tumor pixel distributions” [42]. The paper has been published on the journal of “Computers in Biology and Medicine”.

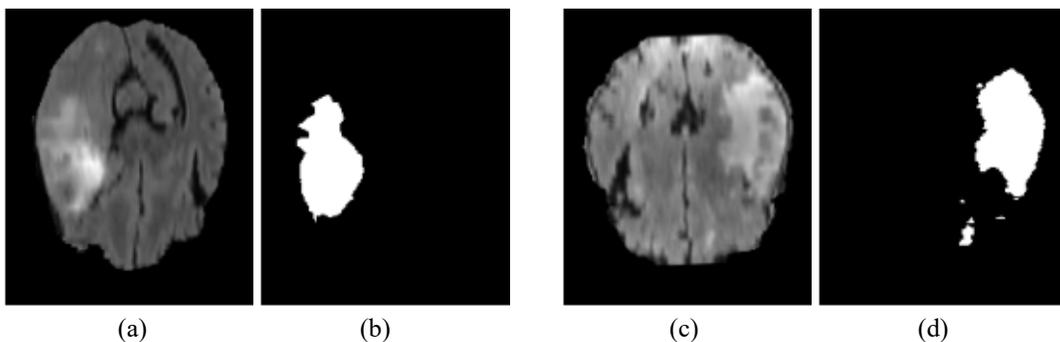


Figure 3.1 (a)(c) Slices sampled from two 3D brain images of Flair modality.  
(b)(d) Binary tumor masks of the slices in (a) and (c).

## 3.1 Methodology

To achieve a high processing quality at the lowest computation cost, we propose to develop the system having 2 characters: mono-modality MRI data input and knowledge-based computing structure. Each brain image is of 3D data. Handling multiple 3D images, acquired by multi-modality scanning, for each patient case is a heavy load for data storage/transfer/processing. We choose the images scanned with Flair modality as the input of the proposed system, since they illustrate better the whole tumor regions, with respect to those with other modalities. The knowledge-based computing structure of the system is designed specifically to address the challenge.

The proposed system has a very specific structure. The main processing in the system is to predict, step-by-step, the gray level distribution of the pixels in the tumor regions of a 3D brain image. The prediction result of each step is used to identify and to remove regions of non-interest, i.e., tumor-free regions, from the 3D image. Each removal reduces the data volume and improves the density of the tumor information, facilitating the prediction in the succeeding step. The final prediction result is then applied to the remaining 3D data to detect, by means of very simple operations, the tumor locations precisely. In the design of this system, the following ideas and methods have been proposed and implemented, as positive contributions to this topic area.

- Presentation of 2D histogram of 3D data. It encompasses the gray level distribution of the data and their locational distribution. As the pixels of a 3D image can be presented in a series of 2D slices, the 2D histogram illustrates how the pixels at a particular gray level, or in a given gray level range, are distributed over the slices.
- Histogram modulation function to attenuate the presence of tumor-free elements. It transforms a histogram representing the gray level distribution of the elements in both tumoral and tumor-free regions to a histogram representing mainly the distribution of the tumoral elements. The modulation function is generated with the original data of each patient case so that its characteristic can adapt to the data distribution of the particular case.
- Method to interleave a step-by-step processing to predict the gray level distribution of pixels in the object region and that to identify/remove non-object region in the same 3D image. The two interact with each other and complement each other: The result of each prediction step is used to identify and to remove non-object regions, improving the density of the object information and benefiting the prediction in the following step.

The details of the new presentation of the 2D histogram and the design of the system on the based of the 2D histogram are found in the following subchapters.

## 3.2 Two-D Gray Level Distribution of 3D data

A tumor can appear in any location in a 3D brain structure and tumor regions can have various gray level distributions. The histograms illustrated in Figure 3.2 (a) and (b) are the gray level distributions of two brain images and those of their tumor regions. Such histograms provide us with important statistical characters of a 3D image data, but without locational information about

the tumor regions. In this subchapter, we propose a 2D histogram presentation, bringing the locational information to the gray level distributions.

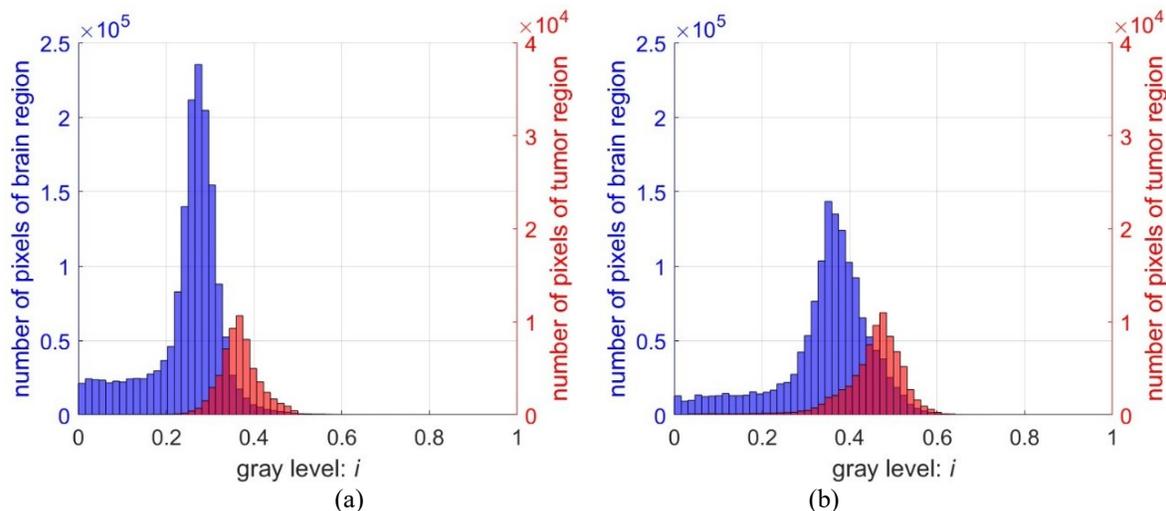
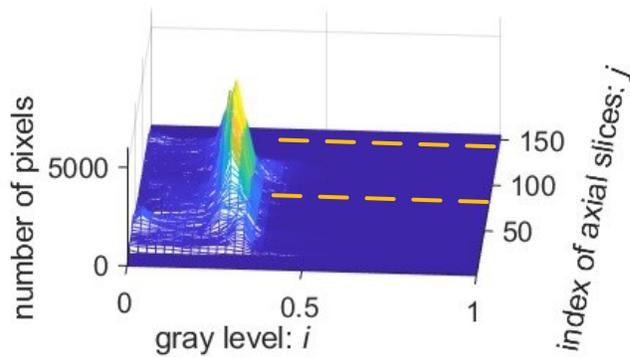


Figure 3.2 Gray level distributions of 3D brain images given by the patient cases 01417 and 01438 from BraTS2021 dataset [43]. The pixels outside the brain regions are excluded.

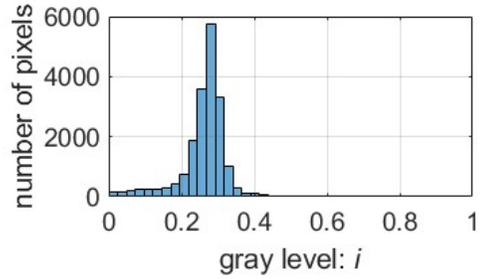
It is known that a 3D brain image can be presented as a series of axial, coronal or sagittal slices, and each slice has a gray level distribution. A 2D histogram presents collectively a series of distributions given by a series of slices, as one example shown in Figure 3.3 (a). Let  $H(i, j)$  denote such a histogram, the  $i$ -axis specifies the gray level, normalized to  $[0, 1]$ , and the  $j$ -axis is the slice index, i.e., one of the 3 coordinates in the 3D structure.

If  $j_0$  is given,  $H(i, j_0)$  is the gray level distribution of the pixels in the  $j_0^{\text{th}}$  slice, whereas if  $i = i_0$ ,  $H(i_0, j)$  represents the locational distribution of the pixels at the gray level  $i_0$  over the slices in the series. Hence, a 2D histogram  $H(i, j)$  encompasses gray level distribution and locational distribution of the pixels.

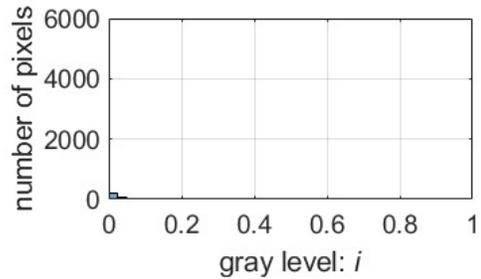
The 2D histogram illustrated in Figure 3.3 (a) is made of the pixels inside the brain region of the 155 slices from a 3D Flair brain image. It demonstrates that a vast majority of the pixels are found (i) in the gray level range (0.2, 0.4) and (ii) in the slices indexed 15 to 142. In other words, the first 14 slices and the last 13 slices (indexed 143 ~ 155) are outside the effective brain region. Hence the coordinates in the  $y$ -axis define the location of the brain region in the direction perpendicular to the slices.



(a) Two-D histogram of a 3D brain region



(b) Gray-level distribution of 80<sup>th</sup> slice



(c) Gray-level distribution of 142<sup>nd</sup> slice

Figure 3.3 (a) Two-D histogram of the 3D brain region of the patient case 01417 from BraTS2021 dataset. It is composed of a series of gray level distributions of the 155 2D axial slices, excluding the pixels outside the brain region. The x-axis specifies the gray levels, normalized to  $[0, 1]$ , the y-axis the index of axial slices, the z-axis the number of pixels.  
 (b) Distribution of the 80th axial slice.  
 (c) Distribution of the 142nd axial slice.

The 2D histogram shown in Figure 3.4 (a) is given by the pixels in a 3D brain region whereas that in Figure 3.4 (b) by the tumor pixels, i.e., the pixels in the 3D tumor regions inside the brain. The latter illustrates not only what the gray level distribution of the tumor pixels looks like, but also which slices contain the tumor pixels and which slices are tumor-free. The 2D histogram shown in Figure 3.4 (c) and (d) are of another patient case and plotted in the same manner. Comparing the two cases, one can get the following observation.

- The gray level distributions can vary a lot from case to case. The tumor regions in different cases have different brightness.
- The difference in tumor locations in the 2 cases are illustrated in Figure 3.4 (b) and (d). Figure 3.4 (b) shows the tumor appears in the higher index-numbered section. As the axial slices are index-numbered from the bottom up, the tumor is found in the very top of the brain. The tumor in the 2nd case, shown in Figure 3.4 (d), is visibly in the lower section of the brain.

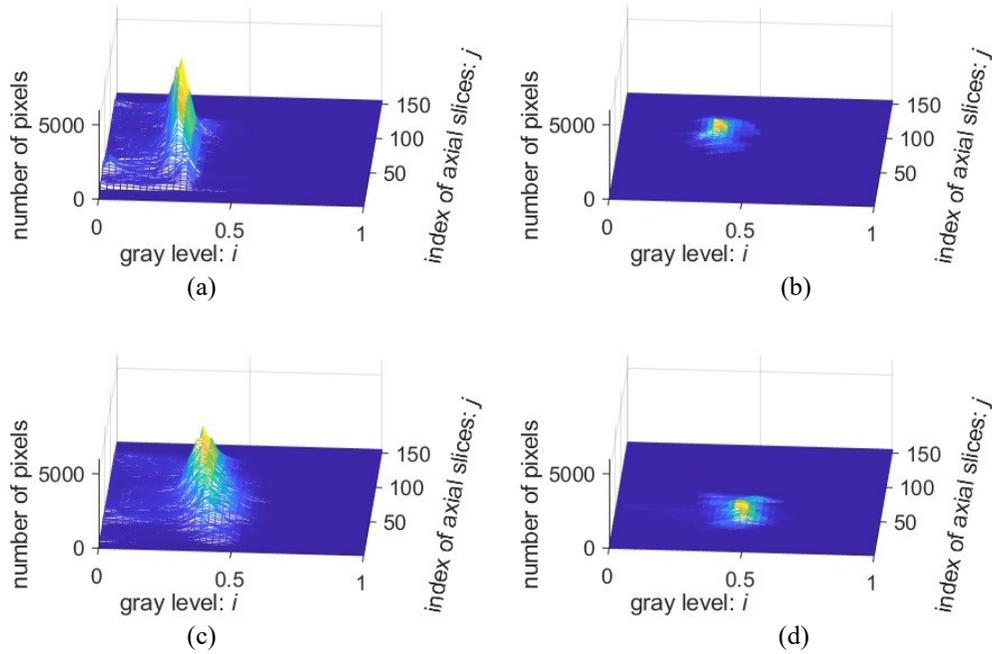


Figure 3.4 (a) Two-D histogram of a 3D brain region in a series of axial slices, given by the case 01417 from BraTS2021 dataset.  
 (b) Two-D histogram of the tumor region (ground truth data).  
 (c)(d) Two-D histograms given by the case 01438 from BraTS2021 dataset.

If a 3D brain image is sliced 3 times, resulting in axial, coronal, and sagittal slice series, one will have three 2D histograms representing the gray level distributions of the pixels over the 3 series, respectively. Figure 3.5 illustrates such a case. The 2D histograms shown in Figure 3.5 (b) (d) and (f) represent the gray level distributions of tumor regions in the three series, respectively. They indicate, on one hand, the tumor location in the 3D brain image, and on the other hand, the tumor-free axial, coronal or sagittal slices.

It should, however, be noted that, in a real detection case, 2D histograms of tumor pixels are not available. Nevertheless, they are predictable. We propose a method to use the information from 2D histograms of a brain image to predict the gray level distribution of the tumor pixels inside the image. Based on the results of the prediction, the task of the brain tumor detection can be done easily and effectively to achieve a good processing quality. The procedure of the prediction and the detection is described in the following subchapter.

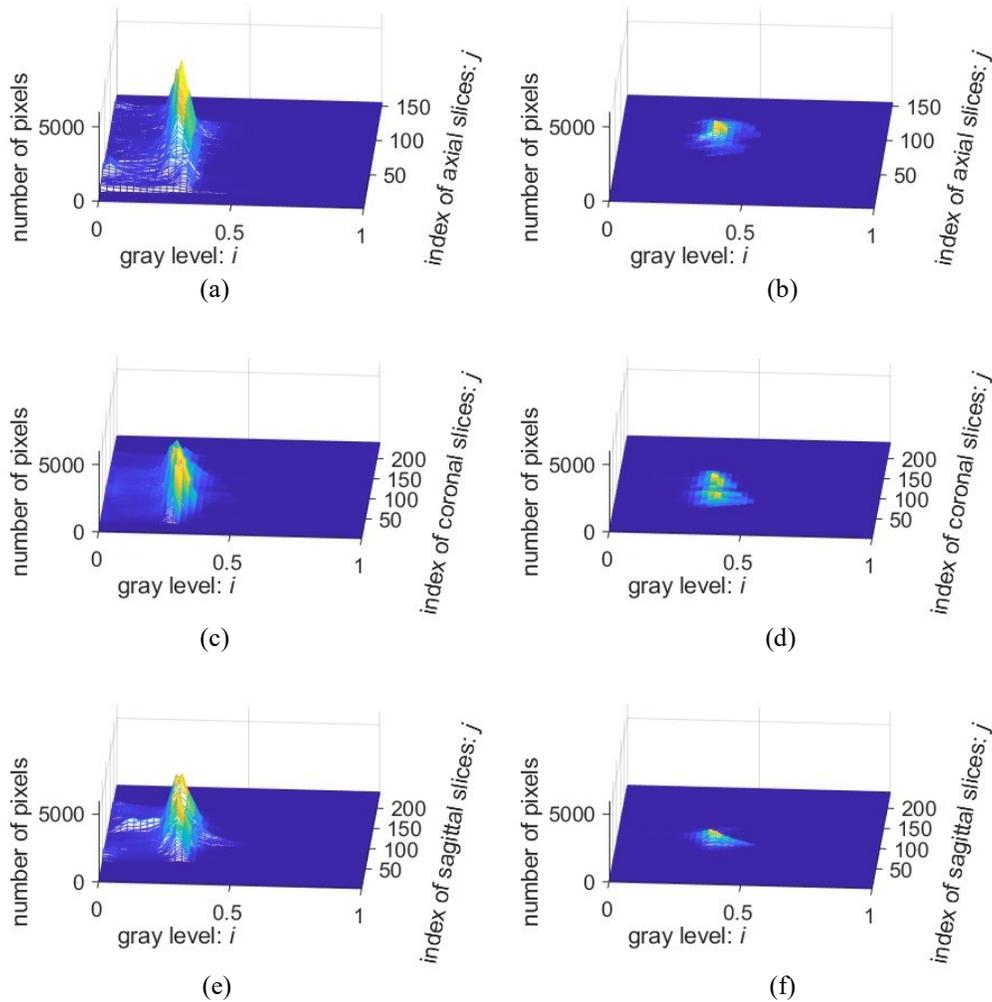


Figure 3.5 Two-D histograms of a brain region and that of the tumor region, given by (a)(b) the axial slices, (c)(d) the coronal slices, and (e)(f) the sagittal slices. The original data sample is from the case 01417 from BraTS2021 dataset.

### 3.3 Proposed Knowledge-Based system

The proposed system is designed to predict 2D histograms of the pixels in the tumor region in a 3D brain image of Flair modality, and the prediction results are used to detect brain tumor in pixel-wise precision. The details are as follows.

#### 3.3.1 System Overview

Of a 3D input image, the object region takes, in general, only a very small percentage of the space and thus the density of the object information is extremely low in the input data. In case of brain

tumor detection and there is a thick tumor-free margin in each of the 6 sides of the 3D input. In other words, in each of the 3 series of slices, namely axial, coronal and sagittal series, only a small number of slices contains tumor pixels, and the other slices are tumor-free. However, as a tumor can be found in any place in the 3D brain, it is not easy to localize these slices in the series. Moreover, though we know that the gray levels of the tumor are mainly found in an upper section of the range of the brain region, there is no model relating the gray level distribution of the pixels inside the tumor space to that of the entire brain. Hence, it is very challenging task to predict the 2D histograms with a good precision.

The proposed system is designed to explore 3 commonly known points.

- Though the object location is unknown, some object-free regions can be localized with some certainty. One can identify/remove object-free regions in multiple steps, starting from the most obvious ones, and each step results in a higher density of object information.
- A higher object information density in the input data leads to a better processing quality.
- Since a 3D input image can be sliced three times in the three different directions, i.e., x, y, z axis, resulting in 3 different series of slices, one can design a 3-step process and each step can be performed with a different series of the same 3D data.

The processing scheme in the proposed system is shown in Figure 3.6. It has 3 prediction steps interleaved with 3 cropping operations. In each step, the 3D data is sliced in one of the 3 directions, the 2D histogram of the tumor pixels of this series is predicted, and the result is then applied to crop out object-free margins, i.e., tumor-free slices. The cropped 3D data is expected to have a higher object information density, with respect to that in the preceding step, and are then used for the prediction in the following step. In this way, the prediction result can be improved step by step.

The proposed system also involves the generation of asymmetry maps and a modulation function, as shown in Figure 3.6. The modulation function is generated from the input data and used to modulate the asymmetry map, or pixel distribution, in each step to produce the prediction result. In the present design, the same modulation function is applied to all the 3 coarse prediction steps, and a modified version to the finalization of the prediction.

The progressive removals of tumor-free regions in the proposed scheme transforms the input 3D image into a 3D minimum bounding box, in which most of the pixels are inside the tumor region. This minimum bounding box is then used to finalize the 3 predicted 2D histograms, as shown in Figure 3.6.

The proposed system also includes a simple procedure of brain tumor detection, in which the predicted gray level distribution of the tumor pixels is used to localize them in the minimum bounding box to segment the brain tumor region in a pixelwise precision.

The quality of the final results is related to the data processing quality in each of the prediction steps. In particular, as the operations are performed sequentially, the first prediction and cropping are very critical. The design of the blocks is presented in the following subchapters.

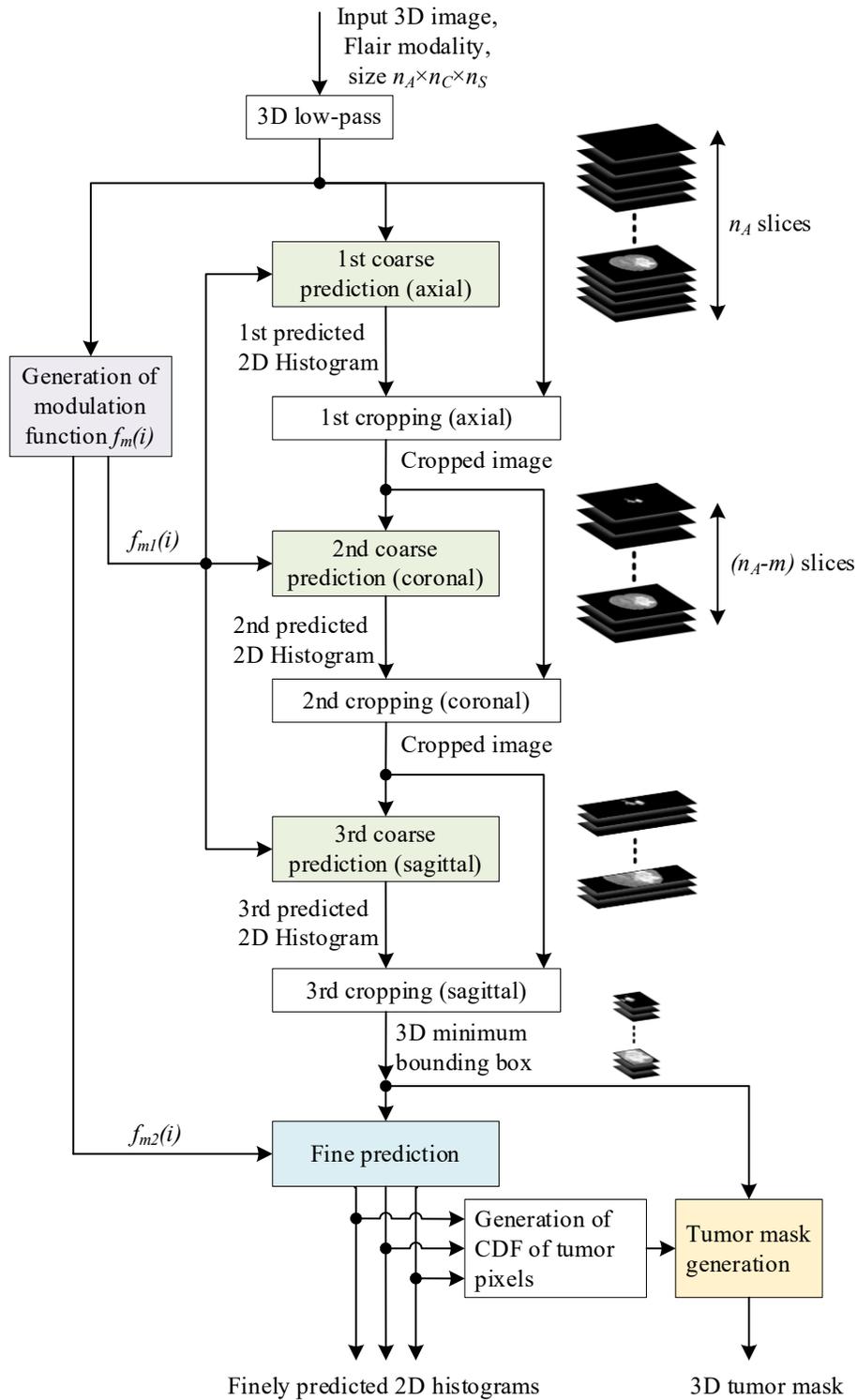


Figure 3.6 Block diagram of the proposed system. It receives a 3D brain image input that can be sliced into a series of  $n_A$  axial slices, or a series of  $n_C$  coronal slices or  $n_S$  sagittal slices, and generates the predicted 2D histogram of the brain tumor region for each of the 3 series of slices and a 3D tumor mask.

### 3.3.2 Brain Image Asymmetry Measure to Extract Tumor Information

In the proposed system, the brain asymmetry is measured for the extraction of the brain tumor information. The results are presented in 2D histograms to indicate how the asymmetry evolves from slice to slice. In this subchapter, the details of the measurement are described and the analysis of the data is presented.

A healthy human brain looks left-right symmetrical, though its details are not really left-right mirrored [44]. A tumor-free axial slice shown in Figure 3.7 (a) is an example. The presence of a brain tumor causes a more noticeable asymmetry in its structure, as shown in Figure 3.7 (b) and (c). Hence, the asymmetry measures in brain images have been used to detect brain tumors [22][23][45]. It should, nevertheless, be noted that, though the tumor-related asymmetry is salient for trained human eyes, it is not prominent in an asymmetry measurement in computer vision. The results of the measurement can be more dominated by the elements representing the natural asymmetry in brain image details than those of the asymmetry caused by tumor, referred to as tumoral asymmetry.

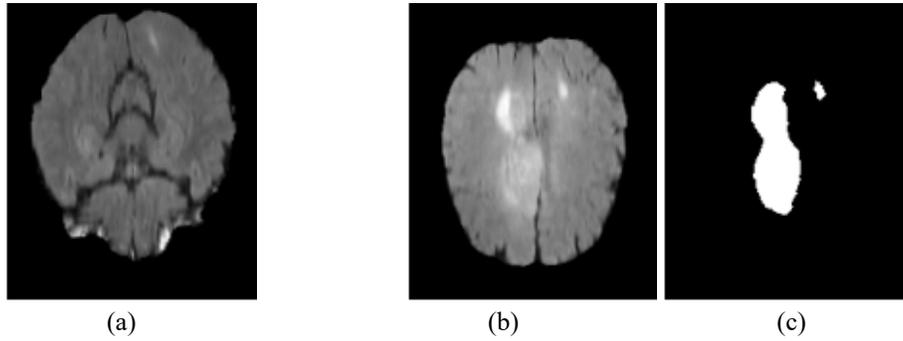


Figure 3.7 (a) Slice of brain image without tumor. The left-right asymmetry in image details is referred to as natural asymmetry.  
 (b)(c) Slice of brain image with tumor and its binary tumor mask. The asymmetry is more noticeable.

The natural asymmetry in brain images is in image details, reflecting different tissues and fluid, whereas the tumoral asymmetry is more in brain structure. Before all the measures, a 3D low-pass filtering is applied to the input and then each slice is down-sampled to erase some image details so that the elements of natural asymmetry are less dominant in the asymmetry measures.

In the proposed prediction process, the left-right asymmetry of a 3D brain image is measured simply by means of the difference between the 2 histograms given by the left and right halves, respectively, so that the natural asymmetry in fine image patterns is less counted. It should be noted that, in this measure, all the histograms are 2D so that each of them indicates the gray level distributions with the coordinates in one of the 3 dimensions. Let  $\Delta H(i,j)$  denote the unsigned histogram difference of the 2 halves, and it is expressed as follows.

$$\Delta H(i,j) = |H_{left}(i,j) - H_{right}(i,j)| \quad (5)$$

where  $H_{left}(i,j)$  is the 2D histogram of the left half and  $H_{right}(i,j)$  is that of the right half,  $i$  representing the gray level, scaled between 0 and 1, and  $j$  the slice index in the axial or coronal series. As the 2D histogram  $\Delta H(i,j)$  represents the gray level distribution of the asymmetry elements over the series, it is referred to as asymmetry map.

As the gray level range of a tumor space is in an upper-level section of that of the brain region, the pixels having their gray levels below the mean level of the 3D brain region are not included in  $\Delta H(i,j)$ . In other words, in the asymmetry maps presented in this section, the gray scale is normalized to the range of  $[0,1]$  with  $i = 0$  corresponding to the mean level of the 3D brain region.

Figure 3.8 illustrates an example of  $H_{left}(i,j)$ ,  $H_{right}(i,j)$  and  $\Delta H(i,j)$  obtained from a 3D Flair image of a typical patient case, in comparison with  $H_T(i,j)$ , the 2D gray level distribution of the true tumor region, referred to as the ground truth. Comparing  $H_{left}(i,j)$  and  $H_{right}(i,j)$ , one can see the right half has more pixels in the upper gray levels, indicating the presence of a tumor, which is also reflected in  $\Delta H(i,j)$ . Comparing  $\Delta H(i,j)$  and the ground truth  $H_T(i,j)$ , one can clearly see that the distribution in the upper level range in  $\Delta H(i,j)$  is highly correlated to that of  $H_T(i,j)$ , but that in the lower level range is not.

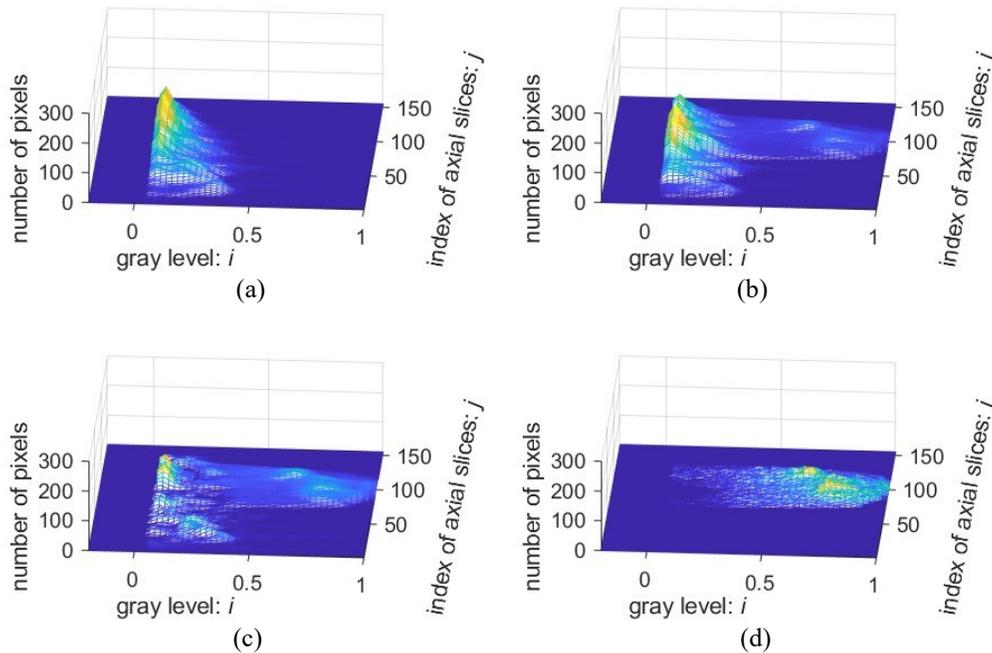


Figure 3.8 Four 2-D histograms obtained from the 155 axial slices of a low-passed 3D Flair image sample.

The X-axis is the normalized gray scale and the zero point corresponds to the mean value of the 3D brain region, excluding the pixels of gray level values below the mean. The data sample is from the case 01412 of BraTS2021 dataset.

(a)  $H_{left}(i,j)$ , the 2D histogram of the left half of the 3D image,

(b)  $H_{right}(i,j)$ , the 2D histogram of the right half of the 3D image,

(c)  $\Delta H(i,j) = |H_{left}(i,j) - H_{right}(i,j)|$ , and

(d)  $H_T(i,j)$ , the ground truth of the 2D gray level distribution of the tumor region.

Evidently, the upper-gray-level section of  $\Delta H(i,j)$  is dominated by the pixels in the tumor region, representing more the tumoral asymmetry. The section of the lower gray levels in  $\Delta H(i,j)$  is, however, more relevant to the natural asymmetry. With a view to obtaining a good prediction of the gray level distribution of the tumor region, the data of  $\Delta H(i,j)$  needs to be modulated so that the elements related to the natural asymmetry will be attenuated. The development of the modulation function is presented in the next subchapter.

### 3.3.3 Generation of the Modulation Function

The asymmetry measurement results in a 2D histogram  $\Delta H(i,j)$  representing the natural and tumoral asymmetries in the consecutive slices of a 3D brain image. To generate, from  $\Delta H(i,j)$ , a 2D histogram  $H_m(i,j)$  resembling the true gray level distribution of the tumor pixels, one needs to attenuate the elements of natural asymmetry in  $\Delta H(i,j)$ . As such elements are found in the lower part of the gray level range of  $\Delta H(i,j)$ , we propose a modulation function  $f_m(i)$ , of which the characteristic is shown in Figure 3.9, and  $H_m(i,j)$  will simply be the product of  $\Delta H(i,j) \cdot f_m(i)$ .

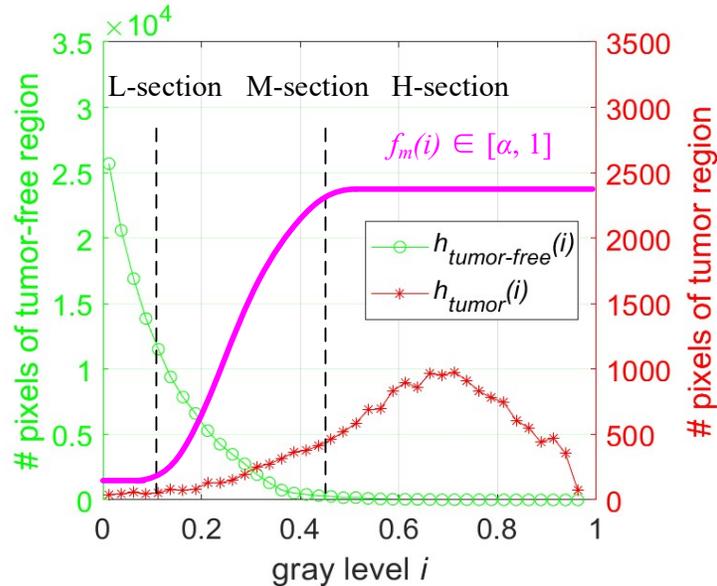


Figure 3.9 Graph of the modulation function  $f_m(i)$ , plotted in magenta, that matches well  $h_{tumor-free}(i)$  and  $h_{tumor}(i)$ , the gray level distribution of the tumor-free region and that of the tumor region, respectively. Ideally, its low (L), mid (M) and high (H) sections should adapt to each individual patient case. The range of  $f_m(i)$  is set to be  $[\alpha, 1]$  and  $\alpha \ll 1$ .

Let us divide the gray level range of the modulation function  $f_m(i)$  into L-section, M-section and H-section, as shown in Figure 3.9. L-section, in which  $f_m(i)$  has its lowest value, covers the gray level range where most pixels are in the tumor-free region and H-section covers that of the tumor region, whereas M-section should cover the range where each gray level bin contains pixels of both tumor and tumor-free regions. Ideally, the 3 sections of  $f_m(i)$  should match the gray level distribution of the tumor region and that of the tumor-free region in each patient case, but neither of them is available. Hence, the data of the input brain image is the only information source to be used to establish the modulation function  $f_m(i)$ .

In case of brain tumor detection, the data of the left or right half of a brain is available to generate its 1D and 2D histograms. Since a tumor region usually appears in the left or right half, the half involving the tumor will have its histogram more populated in the upper gray levels than the other half. Let us call the first half *tumor-half* and the other *tumor-free-half*, and  $h_{tumor-half}$  and  $h_{tumor-free-half}$  denote their 1D histograms, respectively. Figure 3.10 (a) and (b) illustrate  $h_{tumor-half}$  and  $h_{tumor-free-half}$  given by 2 very different patient cases, and each pair is superimposed with the ground truth  $h_{tumor-free}$ , the normalized gray level distribution of the pixels outside the tumor space in the entire 3D brain region. One can find, in each of these 2 cases, a high degree of similarity between  $h_{tumor-free-half}$  and  $h_{tumor-free}$ . It indicates that, in the half that is less affected by the tumor, the statistical characters of the data are not much different from those of the entire tumor-free regions of the brain. Thus,  $h_{tumor-free-half}$  can be used to emulate  $h_{tumor-free}$  to determine M-section and H-section of the modulation function  $f_m(i)$ .

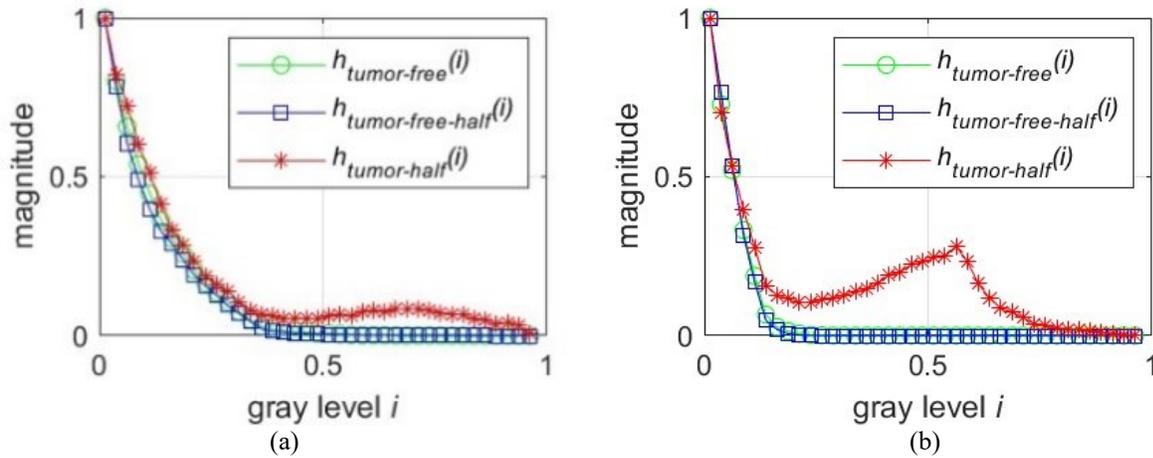


Figure 3.10 Gray level distributions given by patient cases 01412 and 01414 from BraTS 2021 dataset. In the 2 graphs,  $h_{tumor-free-half}$ , the distribution of the pixels from the tumor-free half is compared with  $h_{tumor-half}$ , that of the tumor half and  $h_{tumor-free}$ , that of the true tumor-free region inside the entire brain region.

Of the 1251 patient cases available in BraTS2021 datasets, approximately 94% have tumors developed in either left or right half, and the above observation/analysis is valid for a vast majority of patient cases. Even though a tumor grows in the middle, its region can hardly straddle the left and right halves symmetrically. Hence, the histogram of the half having fewer tumor pixels bears a similitude of  $h_{tumor-free}$  and thus can substitute it to determine  $f_m(i)$ .

The procedure to generate the modulation function  $f_m(i)$  from the distribution of the pixels in the tumor-free half has 2 steps. The first step is to identify which of the 2 halves is more likely to be tumor-free, and the second step is to transform the distribution of the pixels of the identified half into a desired  $f_m(i)$ .

As the pixels in tumor regions are in the upper-gray-level section, the tumor-free half of the 3D brain image should have a smaller number of high-gray-level pixels with respect to the other half. Hence, the identification is done by simply counting the number of pixels in the upper-gray-level section. In case of samples from BraTS2021, this section is defined as  $[0.55, 1]$  in the

normalized gray scale, in which the point  $i = 0$  corresponds to the mean value of the pixels in the brain region. Let  $N_{left}$  denote the number of the pixels in the upper-gray-level section of the left half, and  $N_{right}$  that of the right half. Let  $h_{yf}(i)$  denote the 1D gray level distribution of the identified half that is presumed tumor-free, and it is calculated as follows.

$$N_{left} = \sum_{i=0.55}^1 \sum_{j=1}^{N_s} H_{left}(i, j), \quad N_{right} = \sum_{i=0.55}^1 \sum_{j=1}^{N_s} H_{right}(i, j) \quad (6)$$

$$h_{tf}(i) = \begin{cases} \sum_{j=1}^{N_s} H_{left}(i, j), & \text{if } N_{left} \leq N_{right} \\ \sum_{j=1}^{N_s} H_{right}(i, j), & \text{otherwise} \end{cases} \quad (7)$$

where  $N_s$  is the number of the slices.

Transforming  $h_{yf}(i)$ , the distribution of the pixels of the identified half, into a desired  $f_m(i)$  is done mainly by truncation and inversion. A block diagram of the transformation presented in Figure 3.11 (a), and the curves of the data in this process is visualized in Figure 3.11 (b). The 1D histogram  $h_{yf}(i)$  of the tumor-free half, plotted in blue, is the input of the process. It is truncated to limit the heights of its bins, resulting in  $h_T(i)$  plotted in cyan. The curve of  $1/h_T(i)$ , plotted in black, can be adjusted to approach the expected  $f_m(i)$ , plotted in magenta. In this process, low-pass filtering operations are applied before and after the inversion to remove the discontinuity in the curves. The mathematic expressions used in the transformation process are as follows.

$$h_T(i) = \begin{cases} \max_T, & h_{tf}(i) > \max_T \\ h_{tf}(i), & \min_T \leq h_{tf}(i) \leq \max_T \\ \min_T, & h_{tf}(i) < \min_T \end{cases} \quad (8)$$

$$f_m(i) = \left[ \frac{1}{h_T(i)} \right]^\gamma + \alpha \quad (9)$$

where  $\max_T$  and  $\min_T$  are the pre-determined highest and lowest bin-heights,  $\gamma$  is a correction factor, and  $\alpha$  is a constant.

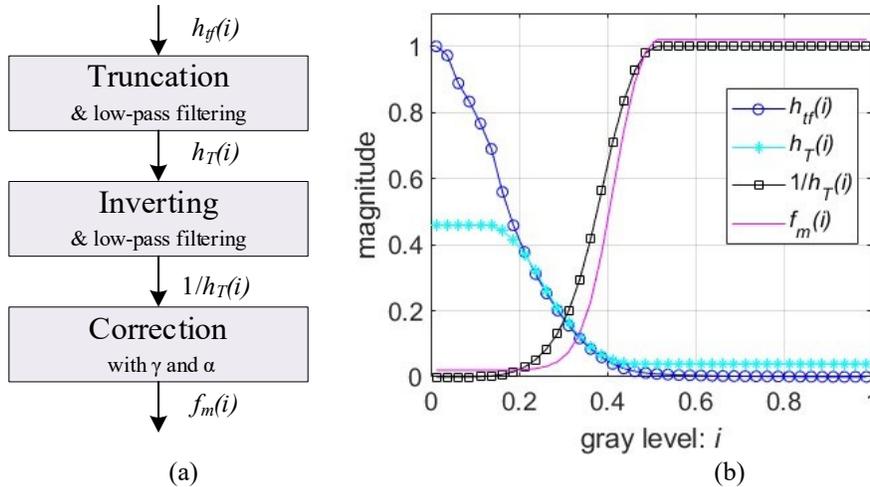


Figure 3.11 (a) Block diagram of the procedure to transform  $h_{yf}(i)$  to  $f_m(i)$ .

(b) Graph of  $h_{yf}(i)$  of a 3D Flair image,  $h_T(i)$ , truncated  $h_{yf}(i)$  with  $\max_T \approx 0.5$ ,  $\min_T \approx 0.05$ ,  $1/h_T(i)$  and  $f_m(i)$  given by Equation (9) with  $\gamma = 1.8$  and  $\alpha = 0.02$ .

The modulation function  $f_m(i)$  can be adjusted by means of the four parameters,  $max_T$  and  $min_T$ ,  $\gamma$ , and  $\alpha$ . One can use  $max_T$  and  $min_T$  to fine-tune, respectively, the 2 particular points where  $df_m(i)/di = 1$ , and these 2 points define M-section of  $f_m(i)$  curve. The parameter  $\gamma$  can be used to modify  $df_m(i)/di$  in this section, and  $\alpha \ll 1$  to maintain a minimum value of  $f_m(i)$ . For example, increasing the values of  $max_T$  and  $min_T$  shifts the M-section slightly left-wards, making the modulation "milder", i.e., attenuating less the elements in mid gray level range.

Figure 3.12 (a) and (b) illustrate the curves of  $f_m(i)$  generated from the Flair images of 2 patient cases, respectively. Each of them is superimposed with the ground truth, i.e., the gray level distribution of the tumor-free region and that of the tumor region. One can see that  $f_m(i)$  can be made to adapt to the distributions of the tumor and tumor-free pixels in each case, although it is generated independently without them. Its high-value section covers the gray level range of a large majority of tumor pixels, whereas its low-value section covers that of most pixels in the tumor-free region.

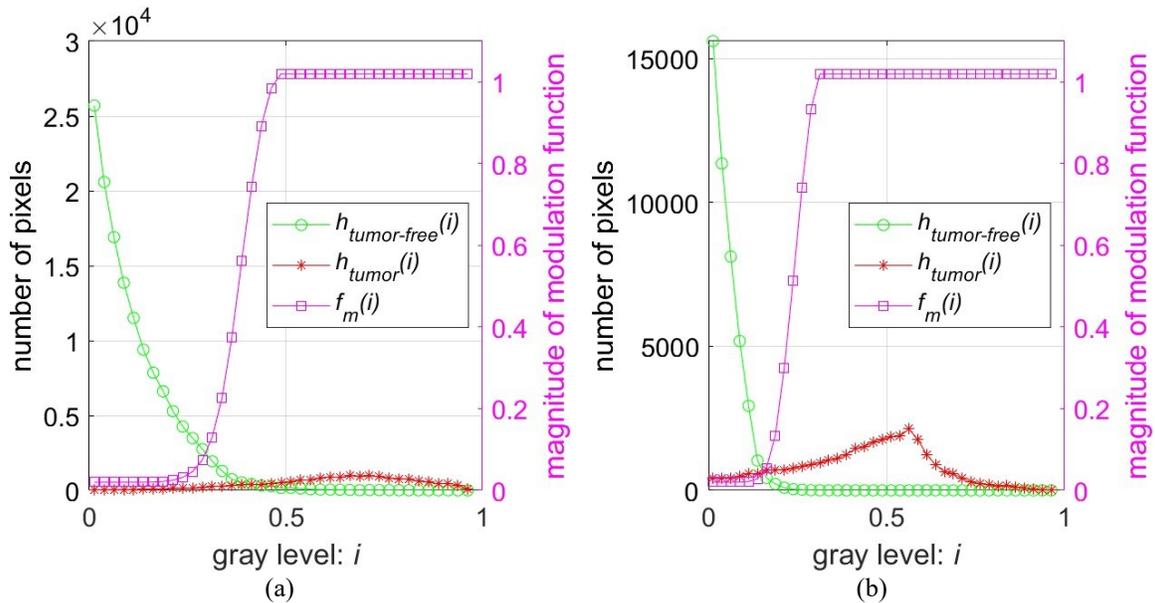


Figure 3.12 Two characteristics, plotted in magenta, of the modulation function  $f_m(i)$  with  $\gamma = 1.8$  and  $\alpha = 0.02$ , generated with the input data from the Flair modality of the patient cases 01412 and 01414, respectively, of BraTS2021 dataset. In each graph, the curve of  $f_m(i)$  is superimposed with  $h_{tumor}(i)$ , the distribution of the pixels in the tumor region and  $h_{tumor-free}(i)$ , that of the tumor-free part of brain region. The scale of  $f_m(i)$  is  $[0,1]$ .

### 3.3.4 Prediction of the Tumor Pixel Distribution

The core of the proposed system is the procedure of 3 coarse predictions of the 2D histograms of the tumor region interleaved with 3 cropping operations, as shown in Figure 3.6. The 3 coarse predictions are performed with axial, coronal and sagittal slice series, respectively. The objective of each prediction is to find the concentration of likely tumor pixels in order to identify the likely tumor-free slices that are then removed, i.e., cropped out, from the slice series.

The processing in the first 2 steps are performed with the axial and coronal slices, respectively, as each of them reflects the left-right symmetry of brain structure, allowing to generate an asymmetry map  $\Delta H(i,j)$ , whereas that in the third step is with a series of cropped sagittal slices. After the 3-step prediction and cropping, the input 3D image is reduced to a minimum bounding box, from which the predicted distribution of the tumor region is refined. The details of the prediction and cropping operations are presented in the following subchapters.

### 3.3.4.1 First Two Coarse Predictions and Cropping Operations

In the proposed system, the first coarse prediction is performed on the axial slices. Let  $\Delta H_a(i,j)$  denote the asymmetry map generated from the axial slices and  $H_{ma}(i,j)$  denote the coarsely predicted distribution of the tumor pixels over the axial slices, we have  $H_{ma}(i,j) = \Delta H_a(i,j) \cdot f_{m1}(i)$ . The modulation function  $f_{m1}(i)$  is defined by Equations (8) and (9), described in Subchapter 3.3.3, generated from the data of the tumor-free half of the original 3D input.

The modulated 2D histogram  $H_{ma}(i,j)$  represents the asymmetry in the upper gray levels, where most of the pixels in the tumor region are found. Thus, it is highly correlated with  $H_{Ta}(i,j)$ , the 2D histogram of the pixels in the tumor region given by the ground truth.

Figure 3.13 (a) and (b) illustrates an example of the first prediction results, in which  $\Delta H_a(i,j)$  and  $H_{ma}(i,j)$  are obtained from the same patient case shown in Figure 3.8. Comparing  $H_{ma}(i,j)$  with  $H_{Ta}(i,j)$  shown in Figure 3.13 (c), one can observe that i)  $H_{ma}(i,j)$  emulates well the distribution of most pixels in the tumor region and ii) it indicates a slice-index range, very similar to that in  $H_{Ta}(i,j)$ , where the tumor pixels are located. The same degree of similarity is also observed in the prediction results of a vast majority of the 1251 patient cases in BraTS2021 dataset. Thus, in the absence of  $H_{Ta}(i,j)$ ,  $H_{ma}(i,j)$  can be considered as a coarsely predicted gray level distribution of the tumor pixels in the consecutive axial slices.

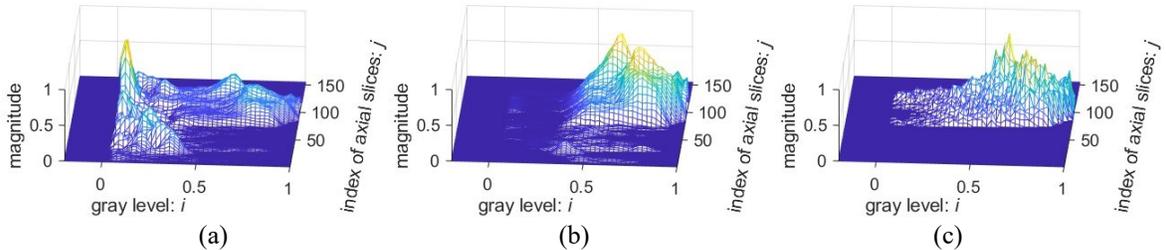


Figure 3.13 (a) Asymmetry map  $\Delta H_a(i,j)$ , generated with the axial slice series of the patient case 01412 from BraTS2021 dataset.  
 (b) Modulated 2D histogram of the axial slice series,  $H_{ma}(i,j)$ , to be used as a coarsely predicted 2D histogram of the tumor region.  
 (c) Two-D histogram of the pixels in the tumor region given by the ground truth,  $H_{Ta}(i,j)$ , over the axial slices.

To identify the tumor-free axial slices, the  $H_{ma}(i,j)$ , obtained from the axial series, is transformed into 1D histogram  $h_{1a}(j)$  to represent the locational distribution of the tumor pixels in the axial slice, which is done as follows.

$$h_{1a}(j) = \sum_i H_{ma}(i,j) \quad (10)$$

where  $j = 1 \sim N_s$  and  $N_s$  is the number of slices in the series. An example of  $h_{la}(j)$  shown in Figure 3.14 (b) is obtained from  $H_{ma}(i,j)$  shown in Figure 3.14 (a). High magnitudes in  $h_{la}(j)$  indicate the concentration of pixels of interest, i.e., tumor pixels, in the corresponding slices. The index range of these slices is determined by the two local minima in  $h_{la}(j)$  curve. The slices indexed between them are considered slices with tumor, and the others tumor-free. One can see in Figure 3.14 (b) that the set of the axial slices identified as tumor slices are almost identical to that in the ground truth. The tumor-free slices, found on the top and bottom of the input 3D image, constitute two tumor-free margins and are then effectively cropped out.

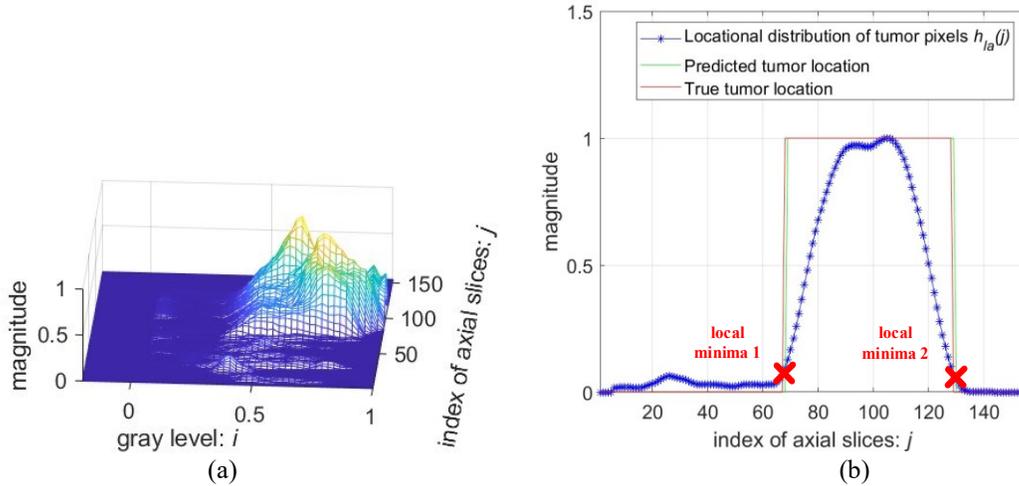


Figure 3.14 (a) Coarsely predicted 2D histogram of the tumor region,  $H_{ma}(i,j)$ , over the axial slices. (b) Predicted locational distribution of the tumor pixels over the series of axial slices,  $h_{la}(j)$ , plotted in blue. The 2 local minima define the 2 boundaries of the predicted set of consecutive tumor slices, specified by the green frame, in comparison with the ground truth framed in red. The data sample is from the patient case 01412 of BraTS2021.

By the first cropping operation, the size of the 3D image is reduced significantly, whereas the loss of the tumor pixels is insignificant. Of the 1251 patient cases in BraTS2021, it results in a removal of more than 60% slices from the 3D brain region, while losing less than 4% of the tumor pixels.

The second coarse prediction is then applied to the cropped 3D image presented as a series of coronal slices. The number of slices in this series is the same as that of the original Flair image, but the number of non-zero pixels per slice is much smaller because the predicted tumor-free top and bottom margins of the input 3D image have been cropped out in the first step of prediction/cropping, as examples shown in Figure 3.15. Nevertheless, the overall left-right symmetry is preserved in the cropped coronal slices that are not affected by tumor, and a coronal asymmetry map  $\Delta H_c(i,j)$  can be calculated to represent the distribution of asymmetrical elements from the cropped coronal slices.

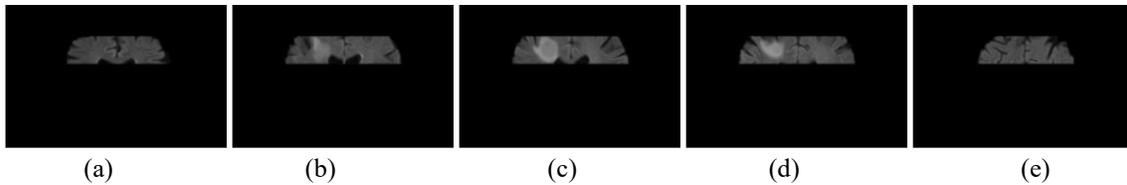


Figure 3.15 Coronal slices sampled from a 3D image, after the cropping operation to remove the tumor-free axial slices.

(a)(e) Cropped coronal slice without tumor. The left half is somehow symmetrical to the right half.

(b)(c)(d) Those with tumor. The left-right symmetry is much less than that in (a) or (e).

The procedure of the second prediction is the same as the first one. The same modulation function  $f_{m1}(i)$  is applied to the coronal asymmetry map  $\Delta H_c(i,j)$  obtained from the coronal slices to generate the second predicted tumor pixel distribution  $H_{mc}(i,j)$ . Figure 3.16 illustrates the predicted distribution over the coronal slices, in comparison with the ground truth.

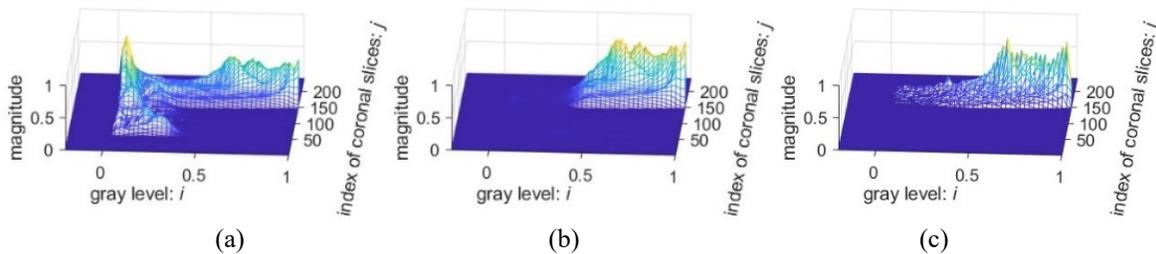


Figure 3.16 (a) Asymmetry map  $\Delta H_c(i,j)$ , generated with the cropped coronal slice series of the patient case 01412 in BraTS2021.

(b) Modulated 2D histogram of the coronal slice series,  $H_{mc}(i,j)$ , to be used as a coarsely predicted 2D histogram of the tumor region.

(c) Two-D histogram of the pixels in the tumor region given by the ground truth,  $H_{Tc}(i,j)$ , over the coronal slices.

The cropping operation following the second coarse prediction is identical to that in the first step. It results in the removal of 2 sets of coronal slices that are considered tumor-free.

After the 2 cropping operations in the first 2 steps, the predicted tumor-free margins in the top, bottom, back and front sides of the original 3D input have been removed. The result of these removals can be seen in sagittal slices. Figure 3.17 illustrates a few examples of sagittal slices cropped twice, in comparison with the original ones. The series of cropped sagittal slices is then ready for the next step of prediction and cropping.

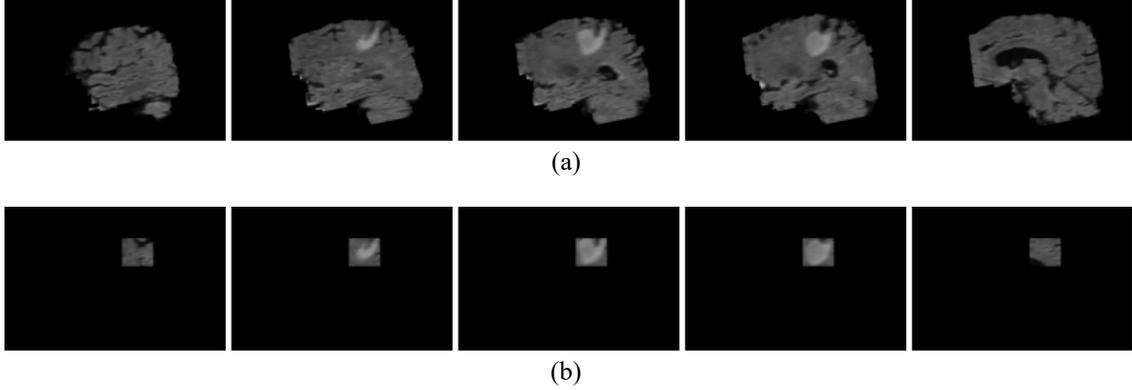


Figure 3.17 (a) Sagittal slices sampled from an original series, before the 2 cropping operations.  
 (b) Sagittal slices after the cropping operations applied to the axial and coronal slices. The first and the last sagittal slices are tumor-free, whereas the other 3 involve tumor regions.

### 3.3.4.2 Third Coarse Prediction & Cropping and the Final Prediction

The objective of the 3rd coarse prediction and cropping is to identify tumor-free sagittal slices and to remove them. As a sagittal slice does not feature left-right symmetry, no asymmetry map can be generated in this step. It should, however, be noted that the percentage of tumor pixels in this sagittal slice series is evidently much higher than that of the original 3D input. In particular, most pixels in the upper gray levels are found in the tumor region. The prediction in this 3rd step is done by modulating  $H_s(i,j)$ , the 2D histogram of the cropped sagittal slices. The coarsely predicted 2D histogram of the tumor pixels over the sagittal slices is denoted as  $H_{ms}(i,j) = H_s(i,j) \cdot f_{m1}(i)$ , with the same  $f_{m1}(i)$  used in the 2 previous coarse prediction steps. This modulation attenuates the elements in the lower-gray-level section, resulting in a coarsely predicted 2D histogram of the tumor pixels in the sagittal slices, as an example shown in Figure 3.18.

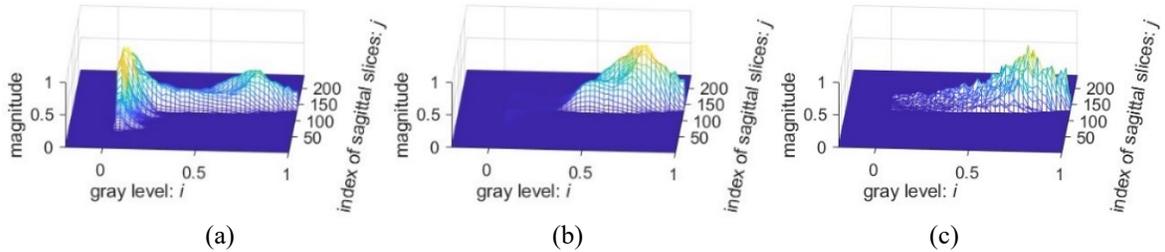


Figure 3.18 (a) Two-D histogram  $H_s(i,j)$ , given by the pixels in the cropped sagittal slice series of the patient case 01412 in BraTS2021.  
 (b) Modulated 2D histogram of the sagittal slice series,  $H_{ms}(i,j)$ , to be used as a coarsely predicted 2D histogram of the tumor region.  
 (c) Two-D histogram of the pixels in the tumor region given by the ground truth,  $H_{Ts}(i,j)$ , over the sagittal slices.

The cropping operation in this step is identical to that in the other steps. It results in removal of the 2 sets of tumor-free sagittal slices, i.e., the tumor-free margins in the left and right sides of the 3D input. By the 3-step cropping, the original 3D input is reduced to a 3D minimum bounding box, in which most of the pixels are found in the tumor region.

The operation of the fine prediction is applied to the data of the 3D minimum bounding box. Its axial, coronal and sagittal slice series give three 2D histograms, denoted by  $H_{ba}(i,j)$ ,  $H_{bc}(i,j)$  and  $H_{bs}(i,j)$ , respectively. An example is illustrated in Figure 3.19 (a) (d) and (g). Comparing the 3 histograms with the ground truth of the tumor pixel distributions illustrated in Figure 3.19 (c) (f) and (i), one can notice that A) the 2 sets are very similar in the upper-gray-level section and B) their differences are found in the lower and mid gray level sections, as the minimum bounding box involves tumor-free regions. Hence, like the coarse predictions, the operation for the fine prediction is to attenuate the elements of  $H_{ba}(i,j)$ ,  $H_{bc}(i,j)$  and  $H_{bs}(i,j)$  in the mid and lower-gray-level sections by a simple modulation.

It should, however, be noticed that the gray levels of the pixels in tumor regions can cover a wide range. Though a majority of the tumor pixels is found in the upper-gray-level section, a non-negligible minority is found in the middle and lower sections, which should be taken into consideration in the fine prediction. Hence, the modulation function in this stage is adjusted to attenuate less elements in the mid and lower-gray-level ranges, with respect to that in the coarse prediction steps.

The final prediction results in the three 2D histograms, denoted as  $H_{pa}(i,j)$ ,  $H_{pc}(i,j)$  and  $H_{ps}(i,j)$ , indicating the gray level distribution of the tumor pixels over the axial, coronal, and sagittal slice series, respectively. They are expressed as follows.

$$\begin{cases} H_{pa}(i,j) = H_{ba}(i,j) \cdot f_{m2}(i) \\ H_{pc}(i,j) = H_{bc}(i,j) \cdot f_{m2}(i) \\ H_{ps}(i,j) = H_{bs}(i,j) \cdot f_{m2}(i) \end{cases} \quad (11)$$

where  $f_{m2}(i)$  is defined by Equations (8) and (9). Compared to  $f_{m1}(i)$ , the characteristic of  $f_{m2}$  is slightly left-wards shifted, the slope,  $df_{m2}(i)/di$ , in M-section is gentler, and the minimum value of  $f_{m2}(i)$  is increased to preserve more pixels in the low-and-mid gray level range. It is done by i) slightly increasing  $max_T$  and  $min_T$ , ii) reducing  $\gamma$  and iii) increasing  $\alpha$ .

An example of the 3 predicted 2D histograms are presented in Figure 3.19 (b)(e)(h). The ground truth data of the tumor distributions are presented in Figure 3.19 (c)(f)(i). One can find that, the predicted tumor distributions are very similar to the ground truth. They can be used, for example, to detect the whole tumor regions in pixel-wise precision in the brain image.

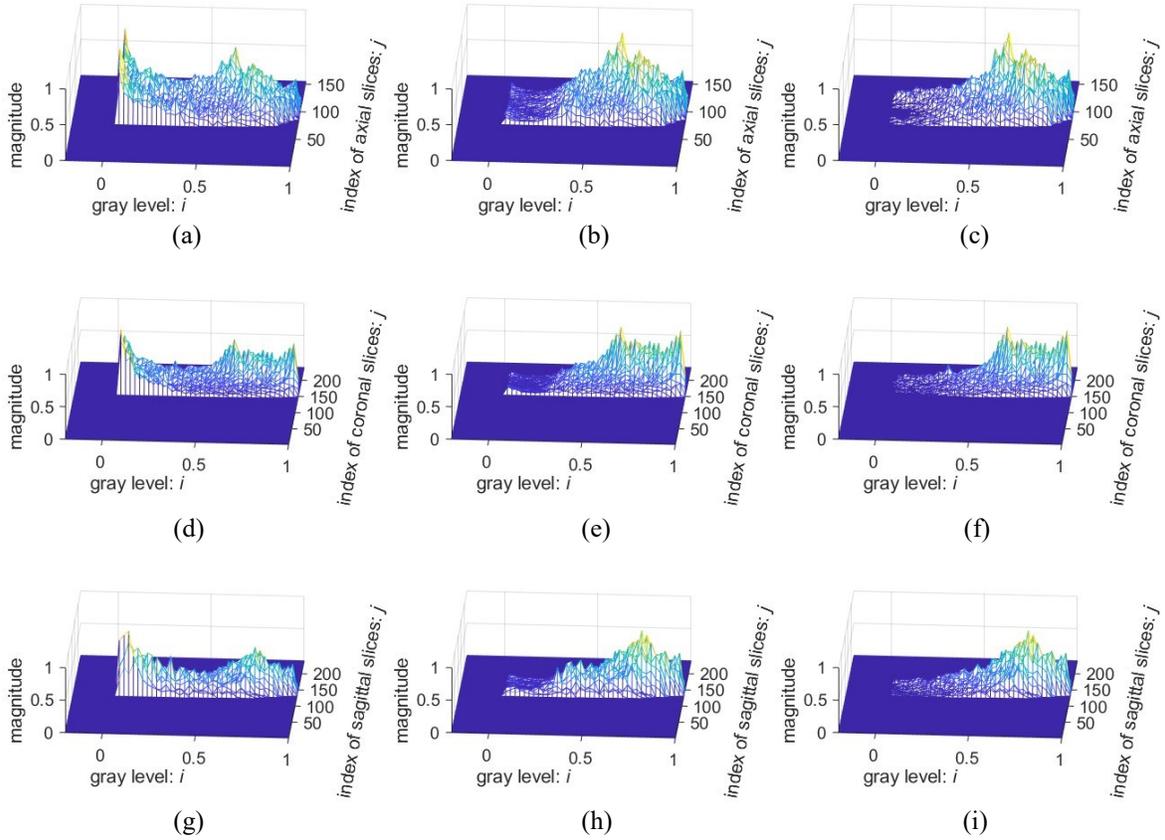


Figure 3.19 (a)(d)(g) Two-D histograms of minimum bounding box of axial, coronal and sagittal slice series.

(b)(e)(h) Predicted 2D histograms of tumor pixels in axial, coronal and sagittal slice series.

(c)(f)(i) Ground truth of the 2D histograms in (b)(e)(h).

The data sample is from the patient case 01412 in BraTS2021.

### 3.3.5 Whole Tumor Detection in Pixel-Wise Precision

In the proposed system, the input data of the brain tumor detection block is a 3D bounding box after the 6 tumor-free margins are cropped out from the original 3D brain image of Flair modality. The gray level distribution of the tumor pixels has been predicted, but the locations of the pixels in this bounding box are not specified. The process in this detection block is to transform the bounding box into a 3D binary tumor mask with pixel-wise precision. The transformation is done by 2 very simple operations, i.e., pixel binarization by gray level thresholding and morphological processing by low-pass filtering.

The binarization is to divide, coarsely by a simple gray level threshold, the pixels in the bounding box into 2 groups, those inside the tumor region and those outside. The threshold should be determined with 2 issues taken into consideration.

- It should be variable to adapt to the gray level distribution of the pixels in individual cases.

- The gray level range of the pixels in the tumor region can extend to a very low point, as the example shown in Figure 3.20 (a). If the threshold is determined in such a way that most of the tumor-free pixels are put in one side and most of tumor pixels on the other side, as the graphs in Figure 3.20 (a) shows, a small portion of the true tumor pixels will unavoidably be misplaced.

In practice, in order to separate a vast majority of tumor-free pixels from the tumor pixels, it is reasonable to allow, e.g., 20% of tumor pixels to be misplaced if their misplacement is insignificant enough to be corrected in the following processing. In this case, the threshold is found at the gray level point corresponding to 20% in the cumulative distribution function (CDF) of the tumor pixel population, as shown in Figure 3.20 (c). The thresholds defined in this manner can be adaptive to the various distributions of individual cases. Since the true distribution of the tumor region is not available, the predicted one is used to determine the threshold, as Figure 3.20 (c) shows.

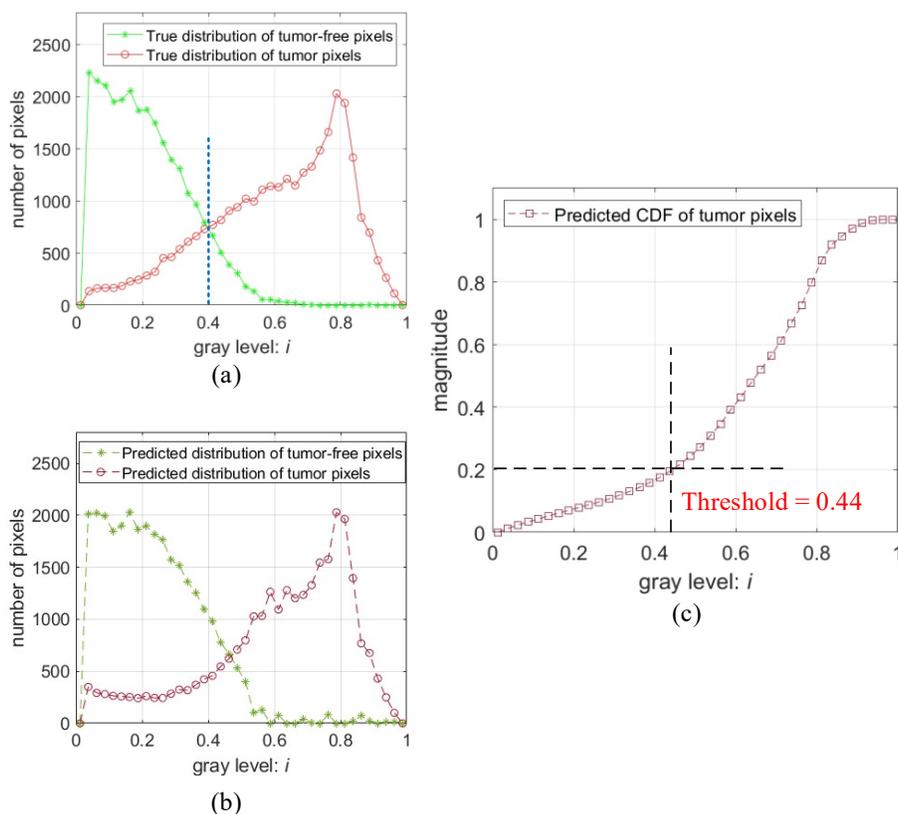


Figure 3.20 (a) True gray level distributions of the pixels inside the tumor region, plotted in red, and that of outside the tumor region, plotted in green, of a 3D minimum bounding box. The blue dashed line indicates an assumed gray level threshold.

The data sample is from the case 01452 of BraTS2021 dataset.

(b) Predicted gray level distributions, obtained from the original 3D Flair image of the same case.

(c) CDF derived from the predicted distribution of the tumor pixels shown in (b). The gray level threshold is 0.44, corresponding to the level of CDF = 20%.

The binarization by means of a simple thresholding results in a coarse binary tumor mask with a minority of tumor pixels misplaced in the tumor-free group and vice versa. A slice of such a 3D mask is illustrated in Figure 3.21 (b). To correct the misplacement, a morphological operation is applied. In this design, it is done by (i) a convolution with a simple 3D averaging kernel of  $5 \times 5 \times 5$  pixels and (ii) assigning the logic-1 value to all the pixels having their gray levels greater than a pre-determined floor and logic-0 to the others. Figure 3.21 (c) illustrates the slice generated by such a morphological operation, in comparison with the ground truth illustrated in Figure 3.21 (d).

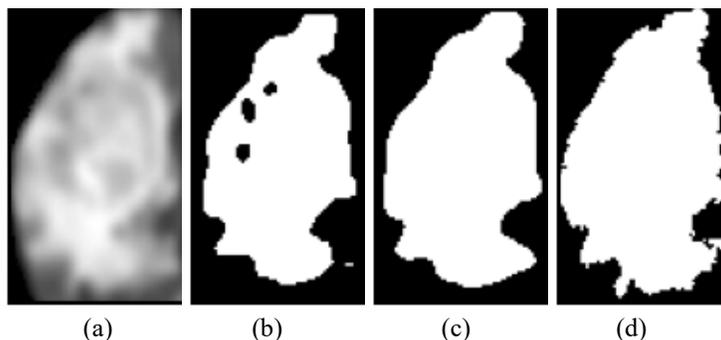


Figure 3.21 (a) Slice from a 3D minimum bounding box generated from the data of the patient case 01418 of BraTS2021 dataset.  
 (b) Slice of the coarse mask after the binarization.  
 (c) Slice of the final binary mask after the morphological operation.  
 (d) Slice of the true tumor mask.

This brain tumor detection is done by the 2 very simple operations, as it is performed on the data of the predicted 3D minimum bounding boxes of tumors and the predicted tumor pixel distribution. Hence it is an application of the prediction results. The quality of the detection depends very much on the quality of the prediction. Various quality measures have been conducted to evaluate the performance of the proposed system. The results are presented in the next subchapter.

## 3.4 Performance Evaluation

The performance of the proposed system has been evaluated with the available patient cases in BraTS datasets. The quality of the prediction of the tumor pixel distribution in a 3D brain image has been assessed, so has the quality of the brain tumor detection. The information about the datasets is found in Subchapter 3.4.1. The prediction and detection results are visualized in Subchapter 3.4.2. The performance measurements are found in Subchapter 3.4.3, and the performance comparison is presented in Subchapter 3.4.4.

### 3.4.1 Dataset

The processing quality of the proposed system has been measured with the data of BraTS2021 [43]. There are 1251 patient cases of MRI scanning and each is accompanied by a ground-truth tumor mask approved by medical specialists. As the system does not need training, the data of all

the 1251 patient cases have been used to measure the processing quality of both the prediction of the tumor pixel distributions and the brain tumor detection.

In BraTS2021 dataset there are additional 219 patient cases, referred to as the validation samples, of which the ground truth data is not accessible for public. They have also been used to evaluate the tumor detection quality of the proposed system, by means of the online platform Synapse [46] where the assessment is a standard process with data from the Cancer Imaging Archive [47][48][49][50].

In order to compare the performance of the proposed system, in terms of brain tumor detection, with those published in recent years, earlier versions of BraTS datasets, namely BraTS2013, BraTS2017, BraTS2018 [51], BraTS2019 [52] and BraTS2020 [53] have also been used for the evaluation. The number of patient cases in each of the datasets is specified in Table 3.1. In case of testing on the validation samples of these datasets, the tumor detection results have been evaluated by the online platform, Center for Biomedical Image Computing and Analytics Image Processing Portal (CBICA IPP) [54].

**Table 3.1 Numbers of patient cases in BraTS dataset**

	BraTS 2013	BraTS 2017	BraTS 2018	BraTS 2019	BraTS 2020	<b>BraTS 2021/2023</b>
Training set	30	285	285	335	369	<b>1251</b>
Validation set	N.A.	N.A.	66	125	125	<b>219</b>

### 3.4.2 Visualization of Prediction and Detection Results

The first step of the performance assessment of the proposed system is to observe (i) the predicted histograms and (ii) the detected 3D tumor masks, in comparison with the ground truth of the datasets. The histograms include three 2D gray level distributions of tumor pixels in the axial, coronal and sagittal series, respectively.

The prediction results of 2 patient cases are illustrated in Figure 3.22 and Figure 3.23. The distributions of tumor pixels in these cases are very different, as shown in Figure 3.22 (d)(e)(f) and Figure 3.23 (d)(e)(f), which is very common in practice. One can see that the predicted histograms are highly similar to the true ones, demonstrating that the proposed system is able to predict the distributions of different cases. The predicted 1D histograms are derived from the 2D ones and are also highly similar to the true ones, as shown in Figure 3.22 (g)(h) and Figure 3.23 (g)(h).

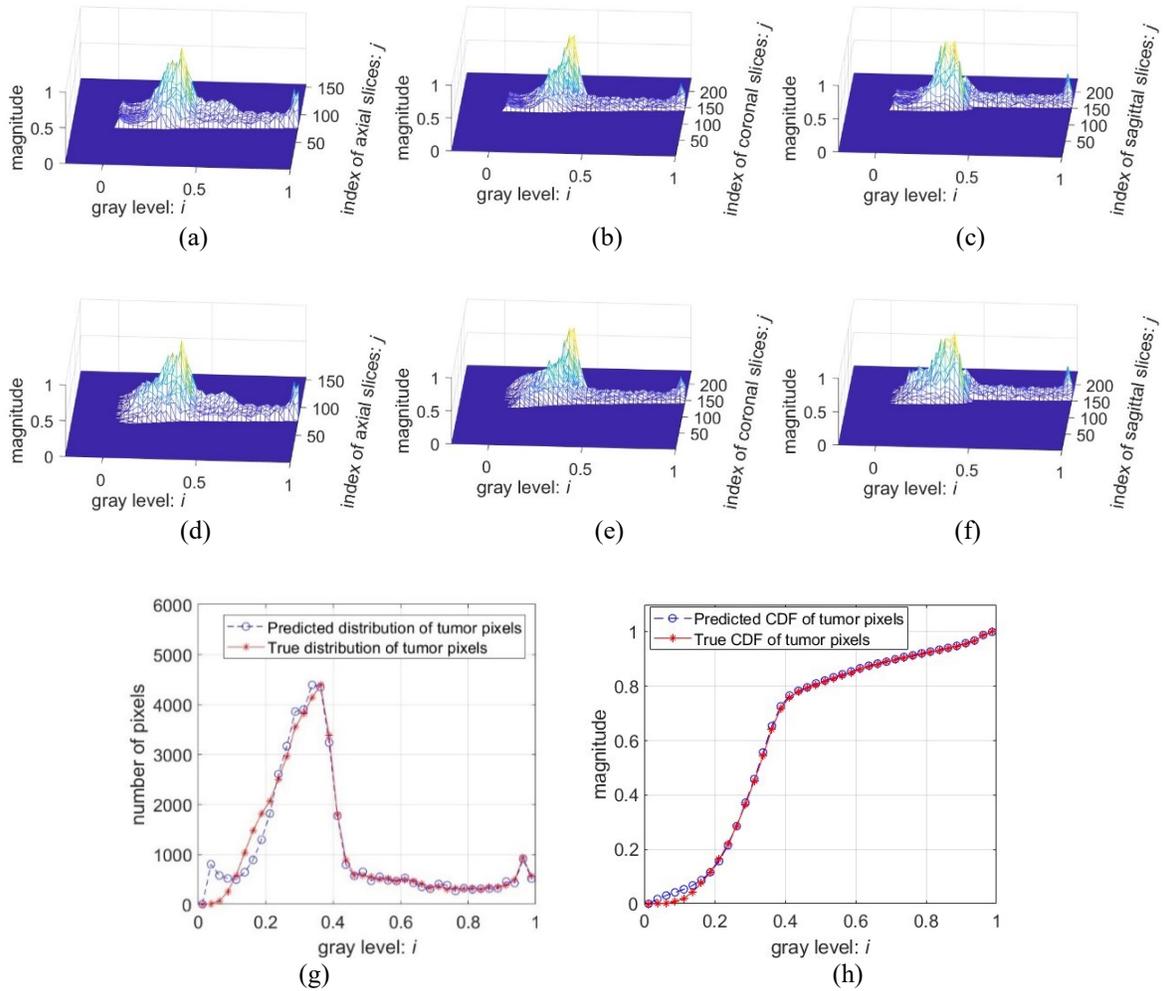


Figure 3.22 (a)(b)(c) Predicted 2D gray level distributions of tumor pixels of axial, coronal and sagittal slice series.

(d)(e)(f) True 2D gray level distributions of the case.

(g) Predicted and true 1D gray level distribution of tumor pixels.

(h) Predicted and true CDF of (g).

The data sample is from patient case 01268 of BraTS2021 dataset.

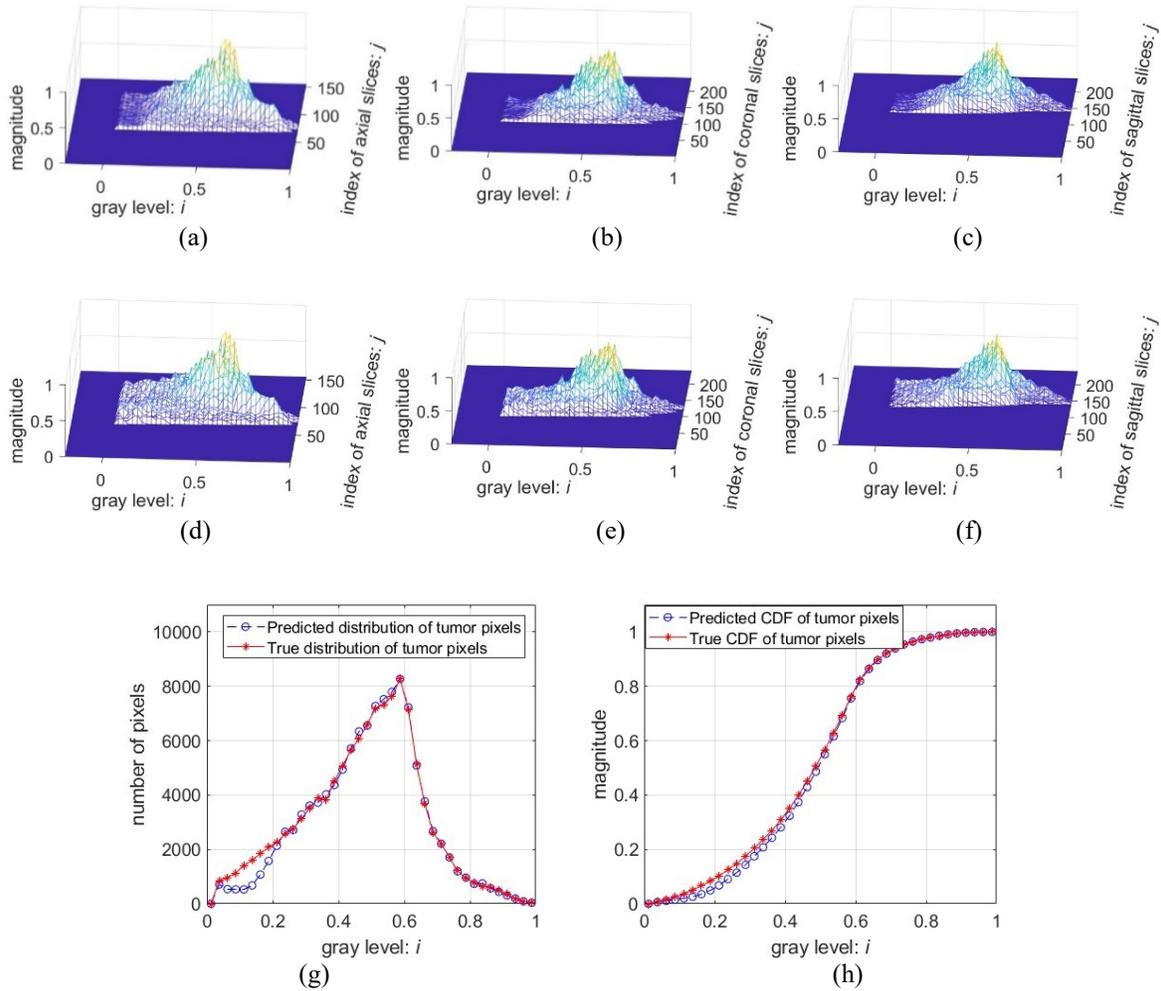


Figure 3.23 (a)(b)(c) Predicted 2D gray level distributions of tumor pixels of axial, coronal and sagittal slice series.

(d)(e)(f) True 2D gray level distributions of the case.

(g) Predicted and true 1D gray level distribution of tumor pixels.

(h) Predicted and true CDF of (g).

The data sample is from patient case 01414 of BraTS2021 dataset.

In Figure 3.24, four examples of tumor detection results generated by the proposed system are illustrated. The gray level ranges and variations inside and outside the tumor regions are very different in these cases. As described previously, the detection process is a very simple thresholding operation followed by a low-pass-based morphological operation. Finding an appropriate threshold for each individual case is the key to achieve a good detection. A good prediction of the tumor pixel distribution results in a suitable threshold. The 4 detected tumor masks presented in the central column are very similar to the true ones shown the right column, demonstrating that the proposed method is effective to detect varieties of tumors. Though the detected masks look like a bit over-smoothed, compared to the true ones, they can be used to localize the brain tumors with a good accuracy.

Besides the visualization, quantitative measurements have been performed on a very large number of samples to assess more objectively the performance of the proposed system. The details are presented in the following subchapter.

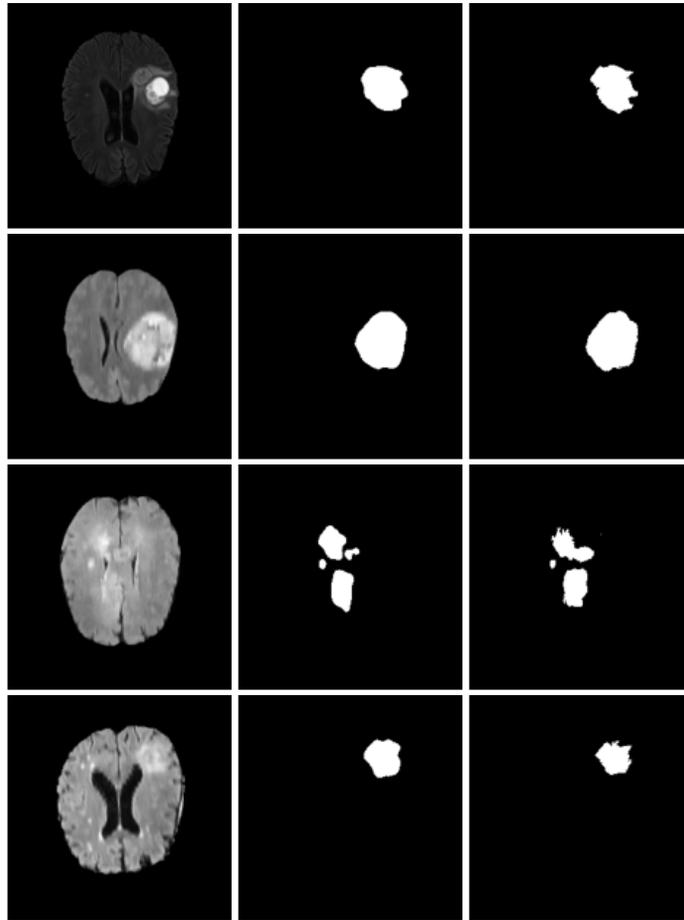


Figure 3.24 Four examples of brain tumor detection by the proposed system. The 4 original MRI slices of Flair modality are found in the left column, the 4 detected binary tumor masks in the middle column and the true ones in the right column.

The original data are from patient cases 01268, 01414, 01417, and 01420 of BraTS2021 dataset.

### 3.4.3 Performance Measurements

The quantitative measurements of the performance have been done mainly with the data of the 1251 patient cases of MRI scanning, including the ground truth data, in BraTS2021 dataset. In this subchapter, the description of the performance metrics is found in Subchapter 3.4.3.1, the results of the comprehensive tests of the prediction and detection in Subchapter 3.4.3.2 and Subchapter 3.4.3.3, respectively.

### 3.4.3.1 Performance Metrics

As the proposed system is designed to predict the gray level distributions of tumor pixels and to detect tumors, the performance metrics involve 2 kinds of measures, for the prediction and the detection, respectively.

The prediction quality can be measured by the degree of similarity between the predicted and true histograms. Correlation coefficient (CC), mean squared error (MSE) and structural similarity index measure (SSIM) [55] are commonly used for similarity measurement. SSIM is defined as follows.

$$SSIM(\mathbf{x}, \mathbf{y}) = [l(\mathbf{x}, \mathbf{y})]^\alpha \cdot [c(\mathbf{x}, \mathbf{y})]^\beta \cdot [s(\mathbf{x}, \mathbf{y})]^\gamma \quad (12)$$

where

$$l(\mathbf{x}, \mathbf{y}) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \quad c(\mathbf{x}, \mathbf{y}) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \quad s(\mathbf{x}, \mathbf{y}) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}$$

$(\mathbf{x}, \mathbf{y})$  are two sets of data,  $(\mu_x, \mu_y)$  denote the mean values of  $\mathbf{x}$  and  $\mathbf{y}$ ,  $(\sigma_x, \sigma_y)$  are the standard deviations,  $\sigma_{xy}$  is the correlation coefficient,  $(\alpha, \beta, \gamma)$  are set to be (1, 1, 1), and  $(C_1, C_2, C_3)$  are small non-zero constants to stabilize the division with weak denominator.

The detection quality can be measured in Dice score indicating how much a predicted object mask and the true object mask are overlapped. It is defined as

$$Dice = \frac{2TP}{(TP+FN)+(TP+FP)} \quad (13)$$

where  $TP$  (true positive) is the overlapped part of the predicted and true object masks,  $FN$  (false negative) is the part of the true object mask that is not covered by the predicted mask. The entire true object mask is represented by  $(TP + FN)$  and the predicted one by  $(TP + FP)$ .

Sensitivity ( $Sens$ ) and false discovery rate ( $FDR$ ) [56], defined as

$$Sens = \frac{TP}{TP+FN} \quad (14)$$

$$FDR = \frac{FP}{TP+FP} = 1 - \frac{1}{2/Dice - 1/Sensitivity} \quad (15)$$

can also be used to measure the detection quality, as complements to Dice score.

### 3.4.3.2 Prediction Results

To evaluate the prediction quality of the proposed system, the similarity between the predicted 2D histogram of the tumor pixels in each of the axial, coronal and sagittal series and its ground truth has been measured on the 1251 patient cases from BraTS 2021 dataset. Large varieties of tumors appear in these 1251 cases, and some are more difficult to detect than others. The test results, presented as statistic values of SSIM, CC and MSE measures, are shown in Table 3.2. The following two points are observed.

- Overall, the proposed system is able to deliver predicted 2D histograms of good quality, confirmed by the overall average SSIM of 84.3% and MSE of 0.004 on the 1251 cases.

- The median SSIM value of 2D histograms, obtained from axial, coronal or sagittal series, is visibly higher than the mean value, and even the 25-quantile is around 80%. It is confirmed that the proposed prediction method yields a good result for a large majority of patient cases.

**Table 3.2 Similarity between the predicted and true gray level distributions. The data are generated by testing the 1251 patient cases of BraTS2021 dataset**

		Predicted 2D gray level distributions of tumor pixels			Predicted CDF of tumor pixels
		Axial	Coronal	Sagittal	
SSIM	<b>mean</b>	<b>0.841</b>	<b>0.837</b>	<b>0.851</b>	<b>0.944</b>
	median	0.944	0.942	0.947	0.967
	25quantile	0.798	0.796	0.818	0.945
	75quantile	0.979	0.978	0.980	0.973
CC	<b>mean</b>	<b>0.878</b>	<b>0.872</b>	<b>0.887</b>	<b>0.958</b>
	median	0.956	0.954	0.958	0.971
	25quantile	0.856	0.855	0.874	0.960
	75quantile	0.982	0.981	0.983	0.974
MSE	<b>mean</b>	<b>0.0049</b>	<b>0.0038</b>	<b>0.0028</b>	<b>0.0113</b>
	median	0.0016	0.0013	0.0009	0.0027
	25quantile	0.0006	0.0005	0.0004	0.0007
	75quantile	0.0054	0.0041	0.0031	0.0101

The prediction results have been applied to detect brain tumor by means of a block with very simple thresholding and filtering operations. The test results of the tumor detection are presented in the following subchapter.

### 3.4.3.3 Whole Tumor Detection Results

As the proposed system does not need training, the data samples in both validation and training sets have been used to assess its quality of the brain tumor detection delivered by the proposed system. It is done in the following 2 approaches.

- Testing on the 3 validation sets from BraTS 2018, 2019, 2020 and 2021 datasets, respectively, as the validation sets of BraTS2019 and BraTS2020 are identical. Then using the online validation tools, namely CBICA IPP [54] and Synapse [46], to get the results.
- Testing on the 4 training sets from the BraTS datasets and measuring the detection quality with the available ground truth data. The advantage of this approach is that the test is done on a large number of data samples. For example, one can test on the 1251 patient cases in the

training set, instead of 219 in the validation set, of BraTS2021 dataset. Thus, the test has been done quite comprehensively.

In total, there are 7 tests, each on a different set of patient cases. The test results, measured in Dice score, Sensitivity and FDR, are summarized in Table 3.3, presented in 7 columns. One can see that, in all the columns, the mean Dice scores are higher than 0.80 and the median values are higher than 0.86. In the right-most column, given by the test performed extensively on the 1251 patient cases, the Dice scores of 25 quantile is 0.767, indicating that the Dice scores of 75% of the cases are 0.767 or more. It has been confirmed that the proposed detection method, applying the results of the predicted tumor pixel distributions, is very effective to detect most brain tumors, despite the vast variations in their locations, shapes, sizes and texture patterns in the tumor regions. The same test results are also visualized in Figure 3.25 by means of boxplots.

It should be underlined that, as the parameters of the proposed system are not determined by training, neither the problem of randomness in training nor the problem of reproducibility could arise. Hence the performance is robust and reliable.

**Table 3.3 Dice score, Sensitivity and FDR of tumor detected by the proposed system**

Flair mono-modality input	Test on validation set (assessed by online portal)			Test on training set				
	BraTS dataset	2018	2019/2020	2021	2018	2019	2020	2021
Number of patient cases	66	125	219	285	335	369	1251	
<i>Dice</i>	mean	<b>0.843</b>	<b>0.816</b>	<b>0.812</b>	<b>0.814</b>	<b>0.816</b>	<b>0.818</b>	<b>0.802</b>
	median	0.884	0.881	0.874	0.869	0.873	0.872	0.876
	25quantile	0.807	0.771	0.769	0.772	0.786	0.786	0.767
	75quantile	0.914	0.917	0.915	0.913	0.914	0.914	0.920
<i>Sens</i>	mean	<b>0.850</b>	<b>0.819</b>	<b>0.824</b>	<b>0.825</b>	<b>0.835</b>	<b>0.834</b>	<b>0.827</b>
	median	0.896	0.877	0.885	0.881	0.896	0.893	0.904
	25quantile	0.815	0.756	0.777	0.777	0.787	0.778	0.785
	75quantile	0.940	0.928	0.941	0.944	0.949	0.948	0.955
<i>FDR</i>	mean	<b>0.127</b>	<b>0.138</b>	<b>0.154</b>	<b>0.150</b>	<b>0.154</b>	<b>0.150</b>	<b>0.162</b>
	median	0.102	0.097	0.102	0.091	0.100	0.092	0.101
	25quantile	0.175	0.176	0.188	0.190	0.190	0.183	0.190
	75quantile	0.044	0.033	0.048	0.037	0.042	0.042	0.049

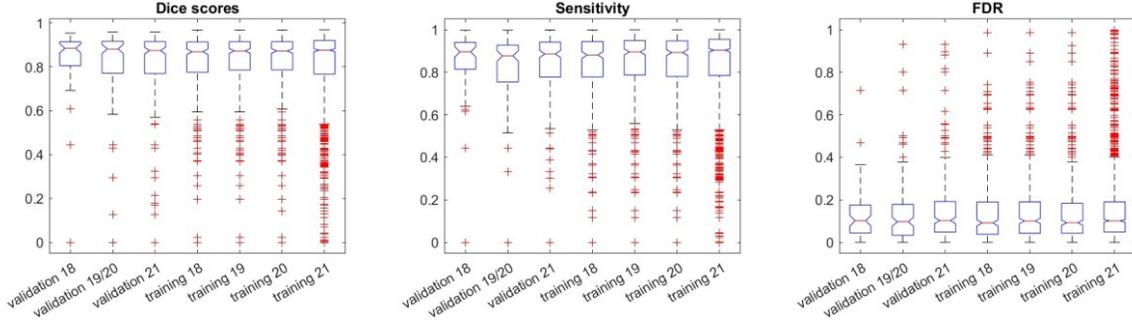


Figure 3.25 Boxplots of the results in Table 3.3. The tests have been done on 7 different sets of data samples.

### 3.4.4 Performance Comparison

The performance of the proposed system, in terms of detection quality, has been compared with that of other systems reported in reputed journals. As no data about the prediction of tumor pixel distributions are available, this performance comparison is in the aspect of brain tumor detection. To make the comparison meaningful, the test results of these systems should be produced, in principle, under the same conditions, i.e., testing on the same data samples and using the same performance metrics.

The comparison results of the proposed system with 2 conventional non-CNN systems are presented in Table 3.4. These 2 systems were chosen for the comparison because their detection quality was assessed with BraTS datasets and the test conditions were the same as those of the proposed system.

**Table 3.4 Comparison of the results of the proposed system with those of other conventional non-CNN systems**

Dataset	Systems	# cases for testing	Modality	<i>Dice</i>
BraTS2013	Lim and Mandava 2018 [19]	20 <sup>H</sup>	T1c, T2	0.701
	Lim and Mandava 2018 [19]	10 <sup>L</sup>	T1c, T2	0.692
	<b>Proposed</b>	<b>20<sup>H</sup></b>	<b>Flair</b>	<b>0.777</b>
	<b>Proposed</b>	<b>10<sup>L</sup></b>	<b>Flair</b>	<b>0.709</b>
BraTS2017	Bonte et al. 2018 [21]	210 <sup>H</sup>	Flair, T1c	0.762
	Bonte et al. 2018 [21]	75 <sup>L</sup>	Flair, T1c	0.656
	<b>Proposed</b>	<b>210<sup>H</sup></b>	<b>Flair</b>	<b>0.829</b>
	<b>Proposed</b>	<b>75<sup>L</sup></b>	<b>Flair</b>	<b>0.772</b>
BraTS2021	<b>Proposed</b>	<b>1251</b>	<b>Flair</b>	<b>0.802</b>

H: High-grade glioma (HGG), L: Low-grade glioma (LGG)

Table 3.5 summarizes the comparison of the detection quality between the proposed system and 4 CNNs reported recently. In order to make the test results comparable to these CNN systems,

the proposed system has been tested on BraTS 2018 and 2019 datasets, besides BraTS 2021 dataset. It should, however, be mentioned that, the test conditions of the proposed systems are much more rigorous, as the test has been done on all the available patient cases of each dataset, instead of a small number of cases sampled from the training pools of the datasets. Also, the test has been done on the BraTS validation sets and the Dice scores have been generated by a 3rd-party platform CBICA IPP [54], whereas the corresponding data of the other systems listed in the table are not available.

**Table 3.5 Comparison of the results of the proposed system with those of CNN systems**

Dataset	Systems	# cases for testing	Modality	Dice
BraTS2018Training set	Wu et al. 2021 [25]	143 out of the 285	T2	0.619
	Zhou et al. 2021 [29]	57 out of the 285	Flair	0.737
	Yang et al. 2022 [30]	95 out of the 285	Flair	0.842
	Rahimpour et al. 2021 [31]	57 out of the 285	T1	0.790
	<b>Proposed</b>	<b>All the 285 cases</b>	<b>Flair</b>	<b>0.814</b>
BraTS2019Training set	Zhou et al. 2021 [29]	67 out of 335	Flair	0.743
	<b>Proposed</b>	<b>All the 335</b>	<b>Flair</b>	<b>0.816</b>
BraTS2018Validation set	<b>Proposed</b>	<b>All the 66*</b>	<b>Flair</b>	<b>0.843</b>
BraTS2019Validation set	<b>Proposed</b>	<b>All the 125*</b>	<b>Flair</b>	<b>0.816</b>
BraTS2021Training set	<b>Proposed</b>	<b>All the 1251</b>	<b>Flair</b>	<b>0.802</b>

\* Testing on the official validation set, and the results are assessed by CBICA IPP [54]

The data presented in Table 3.4 and Table 3.5 demonstrate the high performance of the proposed system, specified in the following aspects.

- Processing quality measured in Dice scores. It outperforms the other systems, conventional non-CNN systems or CNN systems, under the same or more tougher test conditions.
- Performance robustness and reproducibility. On one hand, the results of performance evaluation are obtained by testing extensively with a very large number of patient cases. They are much more comprehensive and reliable than those given by the other systems. On the other hand, as it is a deterministic system, its results are completely reproduceable.
- Computation cost. It is so low that one can run the computation procedure for the prediction and detection in an ordinary laptop or desktop. In case of laptop of i7-11800H CPU with clock of 4.6 GHz, it takes only 0.85 seconds to process a patient case.

It should be mentioned that the proposed system is developed analytically and the processing are modeled mathematically. The functions implemented in the system, such as the 2D histogram modulation or the gray level thresholding for the mask generation, are adaptive to the statistical image features of individual patient cases. Very little tuning is needed. Thus, users can be applied it easily, without "trial and error".

## 3.5 Summary

The challenges in brain tumor detection, like in all kinds of object detections, are often related to the extremely low density of object information in the input data and the enormous variations of the objects. In this chapter, we have proposed a system that predicts the gray level distributions of tumor pixels, i.e., pixels in the tumor regions, of a 3D MRI brain scan of Flair modality, and detects precisely tumor locations in the 3D scan. We have proposed (i) 2D histogram presentations of the data in the axial, coronal and sagittal slice series of a 3D image, comprehending the distributions of the gray levels of the pixels with their locations, (ii) extraction of brain tumor information by exploiting the left-right asymmetry of a brain structure, (iii) histogram modulation, automatically on a case-by-case basis, to enhance the structural asymmetry related to the presence of tumors and to attenuate that due to non-pathological causes, (iv) step-by-step prediction of tumor pixel distribution, accompanied by step-by-step cropping out the regions of non-interest to improve the signal density, and (v) tumor mask generation process consisting of a simple thresholding, based on the prediction results, and a low-pass filtering for morphological purpose.

The proposed system does not need training. It has been tested extensively with the data of more than one thousand patient cases in BraTS 2018~2021 datasets. The test results demonstrate that, with the input data of only Flair modality, the predicted 2D histograms have a high degree of similarity with respect to the true ones. Also, the tumor detection performed by the system is also of high-quality. Moreover, as the system parameters are determined without randomness, its performance is completely reproduceable. It is worth mentioning that the good performance of the proposed system has been achieved at an extremely low computation cost that may be negligible with respect to those of other state-of-the-art systems.

Though the system has been designed to process the data of MRI brain scanning, it can also be used if the 3D data are from CT scanning. The design principle can also be applied to develop systems detecting 3D objects in a symmetrical environment.

Needless to say, the proposed system is not perfect. As it takes the input data produced by MRI Flair scanning to minimize the data volume in the process, its ability to identify some types of tumor regions, in particular in some low-grade glioma (LGG) cases, may be limited. Also, like many other exiting systems, the performance may be reduced if the input images are of poor quality. The future work can be in 2 avenues. The first avenue is the input data pre-processing to improve the image quality. The other avenue is to incorporate CNN and knowledge-based processing blocks to further improve the processing quality. In this case, there will be new challenges of limitation of data samples and the risk of randomness in system training.

## Chapter 4

# CNN for Multi-Class Brain Tumor Segmentation

The objective of the research work presented in this chapter is to develop a CNN system. It receives a set of four 3D images, acquired by MRI brain scanning of the 4 modalities, i.e., Flair, T2, T1 and T1c. It is expected to deliver a tumor mask indicating 3 different intra-tumoral regions, namely ED, NET, and ET, inside the whole tumor (WT) region, with respect to the background. As mentioned previously, it is to segment, with pixel-wise precision, the 3 intra-tumoral regions from the input, which is often done by classifying each of the pixels into one of the 4 classes. Hence, this segmentation problem is, in fact, a complex multi-level classification problem.

To design a CNN system handling such a complex classification problem, we propose a particular design methodology. It aims to achieve a high computational efficiency by optimization of both computing structures and training processes.

This chapter are organized as follows. In Subchapter 4.1, we present the challenge in designing & training convolutional neural networks, and present the proposed methodology. The detailed design of the computing structure and training process of the brain tumor segmentation system is found in Subchapter 4.2. Subchapter 4.3 is dedicated to the performance evaluation of the system. The work is summarized in Subchapter 4.4.

## 4.1 CNN Problem Statement and Design Methodology

A CNN can have a large number of filtering kernels and layers, to perform complex feature extraction and classification operations. The complexity of the brain tumor segmentation task prompts us to take this approach to design our system. One should, however, notice that the performance of a CNN is not only related to its computing structure but also its training process, as the latter determines the values of the trainable parameters in the structure. A commonly used backpropagation-based training method enables an effective parameter update, but not necessarily results in an optimal state of the system to give the best processing result. Hence, improving the training quality is critically important to improve the performance. When the network gets deeper and wider, and its units are more intricately combined, the backpropagation will be more complicated and it will be more difficult to optimize the training processes.

To improve the training quality, it is reasonable to add more supervision spots in a network so that the information captured in hidden layers can be involved in the process of the back propagation. To implement this idea, one may note the following 2 challenges.

- If the ground truth data at a supervised spot of a hidden layer is not available, it is a challenge to find an appropriate way to calculate the loss at that spot.

- In the training process, the loss generated in the  $n^{\text{th}}$  layer will effectively be involved in updating the parameters from the first layer to the  $n^{\text{th}}$  layer. If there is another loss generated in-between and it attempts to interfere the parameter updating differently, the gradient conflict may get worse.

It may not be easy to handle these 2 challenges. Without having to deal with them, the methodology proposed here allows to supervise, effectively, at multiple spots in the process of generating precise classification results from the 3D brain scan data. It involves the following elements.

- Decomposing a complex task into simpler subtasks and each is performed by a simpler CNN that is easy to configure and to train.
- Custom-designing the structures of these CNNs to optimize, on one hand, the computation efficiency and, on the other hand, the training process.
- Multiple networks supervised at their respective output spots, instead of multiple supervisions in a single network.
- Emphasizing on a particular performance aspect in designing each of the CNNs, and making them complement each other, with a view to solving certain design conflicts (e.g., false positive and false negative) and to achieving the best overall performance.
- Exploring the available data resources to generate reasonable “ground truth” data usable to train a network performing a subtask or sub-subtask.

We have applied this methodology to design the brain tumor segmentation system. The above-mentioned elements are used to determine the basic frame of the design plan.

First of all, the task of the pixel classification of the 4 classes is decomposed in 2 steps. In the first step, it is decomposed into 3 binary classifications, each of which is to check if a pixel is inside the object region or not, as illustrated in Figure 4.1. It should be mentioned that the segmentation accuracy of brain tumors is conventionally assessed in 3 categories, i.e., enhancing tumor (ET), tumor core (TC) that combines NET and ET regions, and whole tumor (WT) covering all the 3 intra-tumoral regions. Hence, the 3 binary classifications are defined as follows.

- ET vs. outside ET. The ET regions are the object regions.
- TC vs. non-TC. The object regions cover all the ET and NET regions combined.
- WT vs. outside all tumor regions. The object regions cover all ED, ET and NET regions combined.

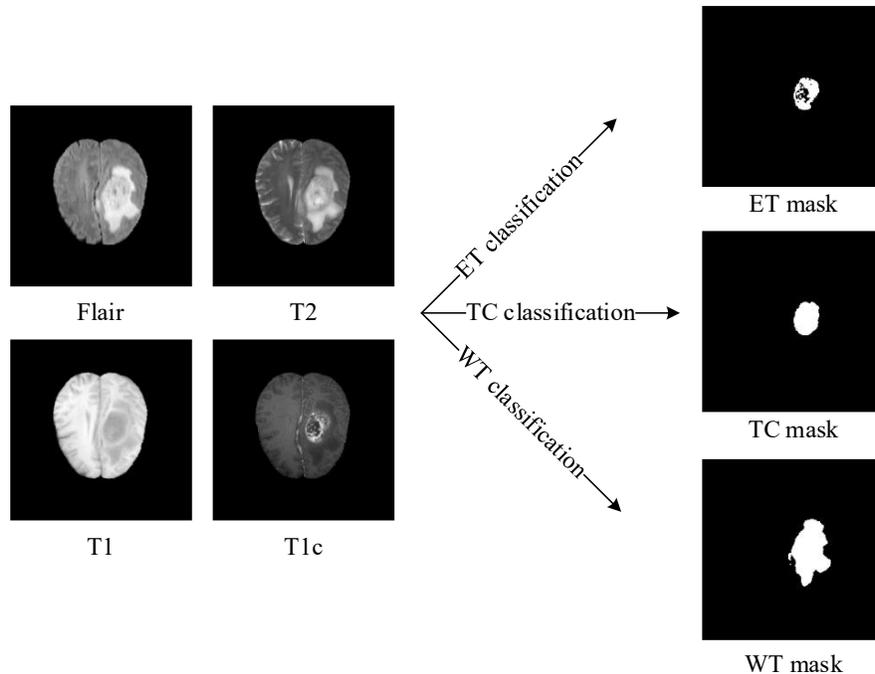


Figure 4.1 Decomposing the 4-class classification task into 3 binary classification subtasks for WT, TC and ET, respectively.

In the second step, each of the above-mentioned binary classification is decomposed into 2 sub-functions so that they can be performed, respectively, by 2 separate functional CNN modules, instead of single CNN. The sub-functions should be defined in such way that (i) the sub-functions can be easily performed by simple modules and (ii) the ground truth data for the module training can be obtained from the available data sources.

The process to segment a tumor region involves usually 2 kinds of operations, feature extraction and pixel classification. The features representing different patterns/textures are used (i) to locate the object region in the entire input image and (ii) to identify the pixels inside the region. In general, the location needs features extracted from a very large neighborhoods, whereas the identification needs features of fine and coarse details. Nevertheless, if the object is localized, even though in a low resolution, the regions of interest are detected, the pixel identification can be done more easily.

We propose to decompose an image segmentation task into 2: approximately locating the object region and then precisely identifying the pixels inside the region. For example, to segment a whole tumor (WT) region from a brain image, we use the first CNN module to generate a low-resolution WT location mask and then another module to produce the final high-resolution WT mask, as shown in Figure 4.2. The reasons for this decomposition are as follows.

- Firstly, one can use a simple series of convolution layers to generate the features specifically for the object-locating purpose and to generate low-resolution mask to indicate approximately the location in the entire image. Such a straight forward CNN structure can be easily trained.

- One can produce very easily, from the ground-truth masks of full-size image, the ground truth data of low-resolution location masks to train the first module.
- The location mask produce by the first module will be used in the second module to enhance the signal in the region of interest. The feature extraction in the second module has then only one focus: optimizing the pixel classification. There will be less risk of the gradient conflict in the training.

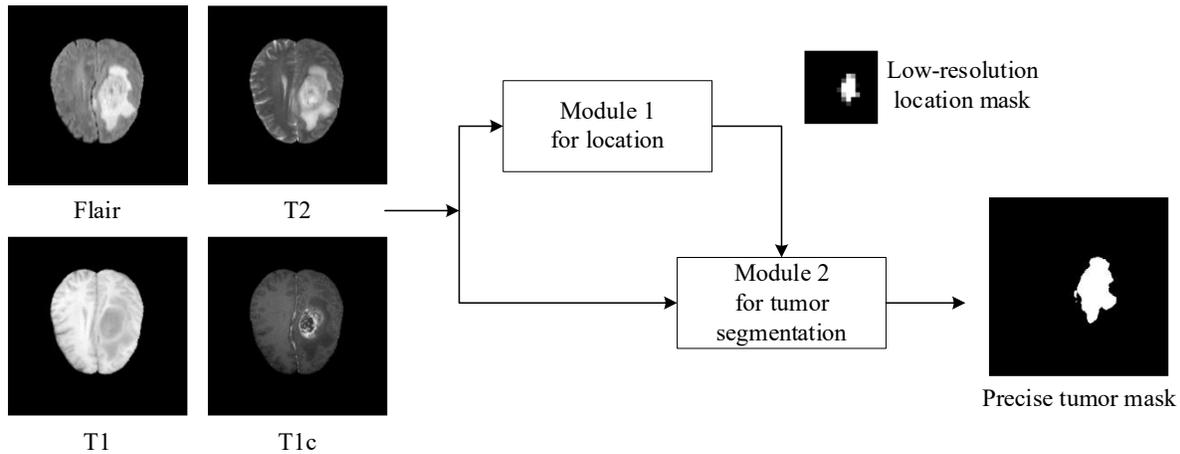


Figure 4.2 Decomposing an image segmentation task into 2, for coarse location and for precise classification, respectively.

In summary, by the decomposition in the first step, the task of classifying the pixels of a 3D brain image in 4 classes is turned into 3 binary classifications. The decomposition in the second step makes it possible to perform each binary classification by 2 CNN modules trained separately. In each of the modules, the features extracted from its input data are made to serve exclusively for one purpose, minimizing the risk of gradient conflict in the training process. The detailed design of the CNN structures is described in the following subchapter.

## 4.2 Detailed Design

The proposed CNN system consists of 3 subsystems to perform the 3 binary classifications to identify the pixels in WT, TC and ET regions, respectively. Each subsystem can be configured and trained independently so that the operations in each of them can be optimized to deal with the particular challenges in detecting its designated object regions.

As each binary classification is further decomposed into 2 subfunctions, in each of the 3 subsystems, there are 2 CNN modules that can also be configured and trained independently. They operate in 2 phases. In Phase 1, Module 1 is used to generate, from brain image data acquired with the four modalities, a low-resolution mask to locate approximately the object region. In Phase 2, the same image data, together with the low-resolution mask, are processed by Module 2 to produce a precise mask delineating the object region in the brain image in its original resolution.

The four 3D brain images of the four MRI modalities are pre-processed in 2 steps before applied to each of the 2 modules. The first step is to remove tumor-free axial slices, using the method presented in Chapter 3, reducing approximately 30% ~ 50% of the data volume and increasing the density of the object information. The data in each of the 4 images are then normalized with the mean and standard deviation calculated only from the pixels inside the brain region.

Module 1 is a simple series of 9 convolution layers, as shown in Figure 4.3. The processing is mainly on 2D maps for the sake of simplicity. To collect information of consecutive slices, each input sample has series of 5 consecutive slices, and 2 layers of 3D convolution (without padding) convert series of slices into 2D data maps. To gather feature information from a large neighborhood, down-sampling and convolution operations with dilated kernels ( $d=2$ ) are employed. As it focuses on the object location in a low-resolution image, instead of object details, the decrease of the image resolution caused by the down-samplings is tolerated.

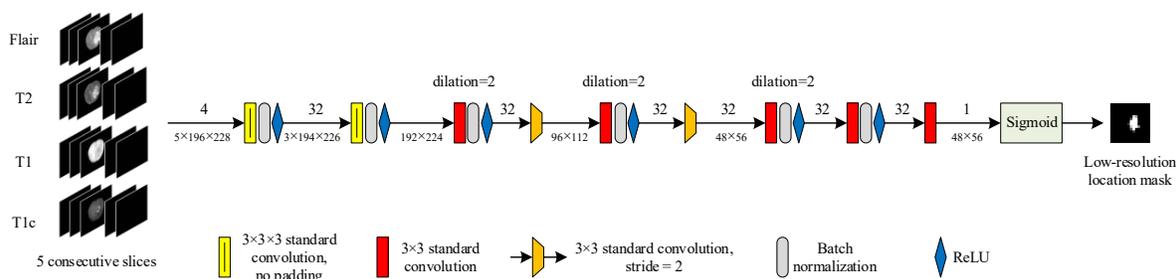


Figure 4.3 Detailed CNN structure of Module 1 for low-resolution location mask generation.

To train Module 1, one can obtain the ground truth data of the location mask easily. The full-size tumor masks can be generated from the brain images with the tumor labels, and one can generate the low-resolution location masks by simply down-sampling the tumor masks.

Module 2 is expected to identify all the pixels in the designated object region to produce a high-resolution mask. The block diagram is shown in Figure 4.4. Its input, i.e., the pre-processed brain image data acquired with the 4 MRI modalities, provides the module with ample image features all over the space. The location mask helps to locate approximately the regions of interest so that the data processing for the feature extraction and pixel classification can be more focused. On one hand, the feature extraction is focused on a sole purpose, i.e., getting features useful only for the pixel classification, no more need for object location, reducing the risk of the gradient conflict in the training process. On the other hand, the pixel classification is focused on the feature data around and inside the object regions, as they are highlighted by means of the location mask.

It should be noted that, to recognize every pixel in the object region, Module 2 needs the feature data representing image details “seen” in different receptive fields. Thus, the module involves 3 branches to extract, respectively, high-frequency (HF), mid-frequency (MF) and low-frequency (LF) features, as shown in the block diagram illustrated in Figure 4.4.

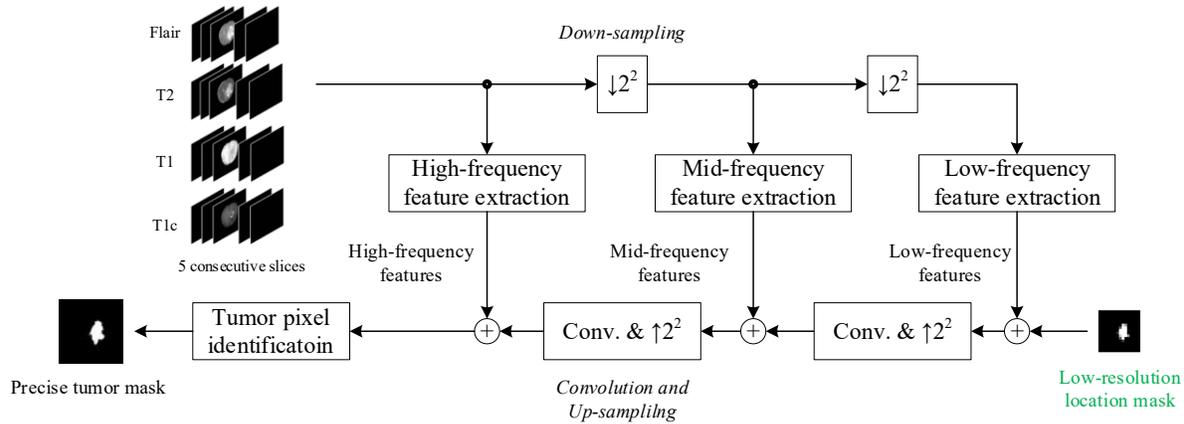


Figure 4.4 Block diagram of Module 2 of the CNN.

The detailed structure of Module 2 is illustrated in Figure 4.5. The particular elements are as follows.

- In the first two layers, 3D convolutions are employed to collect information of consecutive slices, which is similar to Module 1. But the module trainings will make them different to serve their respective purposes.
- Average pooling operations are used to remove image details and reduce the resolution gradually, in view of extracting features in different levels of receptive field and applying the low-resolution location mask.
- Three feature extraction units are employed and each consists of 3 convolution layers. It should be mentioned that there is neither BN nor ReLU applied to the output data of the third layer.
- The low-frequency feature extraction is performed on low-resolution data maps, minimizing the computation volume, while generating the feature maps of the same resolution as that of the location mask produced by Module 1.
- Two layers of convolutions are applied to the location mask to generate data maps compatible to the LF feature maps.
- Simple bilinear interpolations are used to raise the image resolution. When the map size is expanded steps by step, more and more detailed image data, represented by MF and HF features, are added successively to get involved in the pixel classification operations.

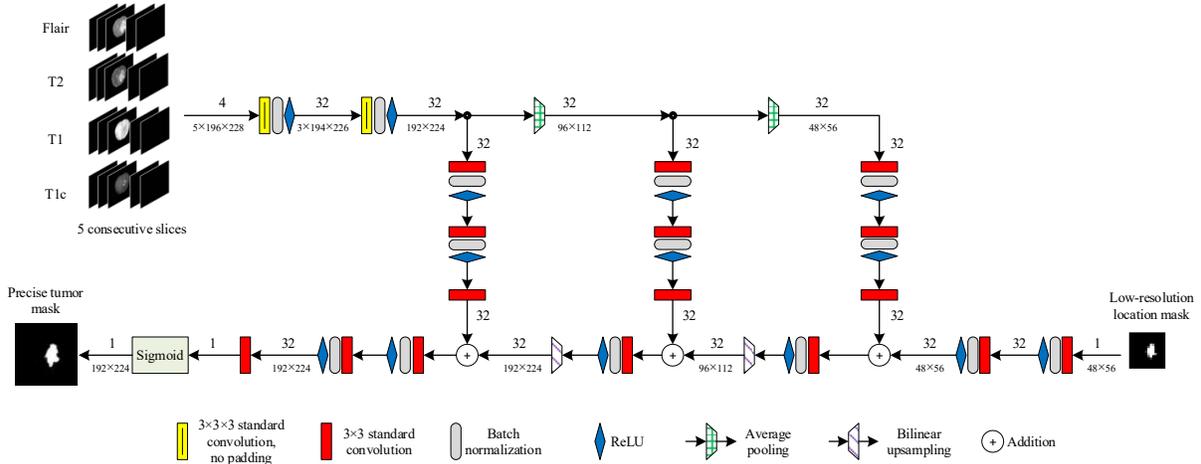


Figure 4.5 Detailed CNN structure of Module 2 for precise pixel-wise classification.

It is understood that neither of the 2 modules can be perfect. The location masks produced by Module 1 can not be identical to the true ones. If a segment of the true object region is not covered by the mask, the pixels in this segment will not be highlighted in the processing of Module 2 and can then be irrecoverably missed out on the classification, causing more false negatives (FN). Contrarily, the mask covering too much may cause more false positives (FP). In the design of Module 2, one has also to deal with the FN-FP conflict.

The strategy to achieve the best overall outcome is to let Module 1 produce location masks of good coverage, minimizing the false negatives, and Module 2 focus on minimizing the false positives. Though this strategy may lead to producing a location mask overcovering the object location, the filtering functions in Module 2 can be enhanced to better distinguish the pixels outside the object region from those inside it. An example of applying this strategy is detailed in Subchapter 4.3.2.

This strategy can be implemented by training the 2 modules separately with different emphases. The loss calculation in each of the 2 training processes involves multiple error components. For Module 1, the component related to the error of false negatives is more weighted than that of false positives. For Module 2, the loss value is made to be more sensitive to the error of false positives. By doing so, the 2 modules are made to complement each other to yield, overall, a good result.

In this design, for the sake of simplicity, the 3 subsystems, segmenting WT, TC and ET regions, respectively, are configured identically, though they can have different structures if needed. The configuration details of Modules 1 and 2 are presented in Table 4.1 and Table 4.2, respectively. Each subsystem for ET, TC or WT segmentation has 249,474 trainable parameters and the total number for the complete system is 748,422, which is a tiny fraction of what one can find in a modest CNN system for the same task.

**Table 4.1 Details of the CNN configuration of Module 1**

Layer	Kernel size	# input channels	# output channels	Input dimension	Output dimension	# parameters
1	3×3×3	4	32	5×196×228	3×194×226	3,552
2	3×3×3	32	32	3×194×226	192×224	27,744
3*	3×3	32	32	192×224	192×224	9,312
4**	3×3	32	32	192×224	96×112	9,248
5*	3×3	32	32	96×112	96×112	9,312
6**	3×3	32	32	96×112	48×56	9,248
7*	3×3	32	32	48×56	48×56	9,312
8	3×3	32	32	48×56	48×56	9,312
9	3×3	32	1	48×56	48×56	289
Total						87,329

\* Convolution with dilated kernels ( $d=2$ )

\*\* Down-sampling by convolution ( $stride=2$ )

**Table 4.2 Details of the CNN configuration of Module 2**

Block	Kernel size	# input channels	# output channels	Input dimension	Output dimension	# parameters
First 3D Conv. layer	3×3×3	4	32	5×196×228	3×194×226	3,552
Second 3D Conv. layer	3×3×3	32	32	3×194×226	192×224	27,744
Low-frequency feature extraction	3×3	32	32	48×56	48×56	27,872
Mid-frequency feature extraction	3×3	32	32	96×112	96×112	27,872
High-frequency feature extraction	3×3	32	32	192×224	192×224	27,872
2 layers performed to locational mask	3×3	1	32	48×56	48×56	9,696
Conv. & bilinear up-sampling	3×3	32	32	48×56	96×112	9,312
Conv. & bilinear up-sampling	3×3	32	32	96×112	192×224	9,312
3 layers to generated final mask	3×3	32	1	192×224	192×224	18,913
Total						162,145

There is a 2-step post-processing applied to the masks delivered by the 3 CNN subsystems designated for ET and TC, and WT segmentation.

- The first step is to generate a single multi-class mask delineating 4 different regions, i.e., 3 intra-tumoral regions, namely ED, NET, and ET, and the background. As mentioned previously, a TC region covers both NET and ET regions, whereas a WT region includes ED and TC. Thus, the 3 intra-tumoral regions are specified by the pixel values produced by the 3 subsystems. There could be a small number of cases of classification conflict. For example, at a pixel position, the subsystems for TC tend to identify it in the TC region by giving a high probability value, whereas the subsystem for WT gives a low probability value. In such a case, the final decision is made, by means of the Euclidean distance calculation, based on the 3 probability values given by the 3 subsystems.
- The second step is to remove isolated regions. It can be a tumoral region surrounded by the background, or a small piece of background inside a tumoral region. Each of them consists of a very small number of pixels, and are too small to be real. Thus, such pixels are more likely misclassified and to be removed. In this design, a morphological operation, by means of low-pass filtering, is done to remove such isolated regions.

With respect to the 2 CNN modules, the computation volume of the post processing unit is negligible. The performance evaluation of the proposed system is presented in the next subchapter.

## 4.3 Performance Evaluation

The proposed system has been trained and test with the patient cases of BraTS datasets. In this subchapter, the details of the datasets, the training parameters and loss functions are described. The test results and performance comparison are also presented.

### 4.3.1 Dataset and Performance Metrics

BraTS dataset has been used in developing CNN systems for brain tumor detections. This dataset has been evolved over the past years. In each version, there are 2 sets of samples, one for training and the other exclusively for validation. As the patient cases in the latter are not labelled, the quality of the output masks needs to be assessed by online platforms [57].

Consider that many brain tumor segmentation systems reported in recent publications were tested with the validation sets of earlier versions, in order to make a fair comparison with those system, the proposed system has also been trained and tested on the datasets of BraTS2021 [43], BraTS2020 [53] and BraTS2019 [52]. The numbers of the samples in these samples are found in Table 4.3.

The 3 scores concerning the processing quality of the system are the Dice, Sensitivity and false discovery rate (*FDR*). Their general definitions are presented in Subchapter 3.4.3.1, in which the performance metrics of whole tumor detection are described. The system presented in this chapter

performs the segmentations of the 3 kinds of tumor regions, namely WT, TC and ET. In each of the 3 segmentations, the processing results are measured to generate the 3 scores.

The computation complexity of the system is also part of the performance metrics. It is measured in the number of trainable parameters and the number of multiply-accumulate (MAC) operations required to complete the test of a patient case, i.e., generating the WT, TC and ET masks from the images of the 4 MRI modalities. The processing time, with a particular computing hardware system, is also part of the performance metrics.

**Table 4.3 Number of patient cases in different versions of BraTS dataset**

	BraTS2023/ BraTS2021	BraTS2020	BraTS2019
Training set	1251	369	335
Validation set (online assessment)	219	125	125

### 4.3.2 Training and Implementation Details

The proposed system has three 2-module subsystems, and each module is independently trained. The filtering parameters of each of the modules are determined in a training process. Applying the training strategy described in Subchapter 4.2 is part of the implementation of the design methodology in the system design. The following training hyperparameters are commonly used to train all the modules.

- The number of data samples in each batch is 128. Each data sample is a series of 5 consecutive axial slices.
- The optimizer is chosen to be Adaptive Moment Estimation (ADAM) [58].
- Each module is trained in 50 epochs.
- The learning rate is variable cosine decay [59], changing from  $10^{-3} \sim 10^{-6}$  over the 50 epochs.
- The trainable parameters are randomly initialized using a uniform distribution [60].

As mentioned previously, for each subsystem to deliver a good classification result, the training strategy is to train Module 1 sensitive to the error of false negatives (*FN*) and Module 2 to that of false positives (*FP*). This strategy can be implemented by means of loss calculations involving elements related to the false negatives or false positives. To this end, the two loss functions,  $loss_1$  and  $loss_2$  to train the Modul 1 and Module 2, respectively, are defined as follows.

$$loss_1 = loss_{Dice} + w_1 \cdot loss_{FN} \quad (16)$$

$$loss_2 = loss_{Dice} + w_2 \cdot loss_{FP} \quad (17)$$

where  $w_1$  and  $w_2$  are weight coefficients. In each of the loss function,  $loss_{Dice}$  is the basic loss component and the other component is added to place more weight on *FP* or *FN* errors. We define  $loss_{FN}$  and  $loss_{FP}$  as follows.

$$loss_{FN} = \frac{\sum_{n=1}^N -y_n \cdot \ln(\hat{y}_n)}{\varepsilon + \sum_{n=1}^N y_n} \quad (18)$$

$$loss_{FP} = \frac{\sum_{n=1}^N -(1-y_n) \cdot \ln(1-\hat{y}_n)}{\varepsilon + \sum_{n=1}^N (1-y_n)} \quad (19)$$

where  $N$  is the number of pixels in a mini-batch, while  $y_n$  and  $\hat{y}_n$  represent, respectively, the true label and the predicted probability value of the  $n^{th}$  pixel, and  $\varepsilon$  is a small constant to avoid division by zero.

The values of  $w_1$  and  $w_2$  can be determined independently for the 3 subsystems. The values chosen in our experiments are found in Table 4.4.

**Table 4.4 Weight values of loss function for the two CNN modules**

	WT	TC	ET
$w_1$	0.2	1.0	0.1
$w_2$	0.1	0.2	0.1

### 4.3.3 Test Results and Comparison

The results of the processing quality assessment are presented in Table 4.5. Each experiment, i.e., a procedure of initializing, training and then testing, has been repeated 3 times under the exactly same conditions. One can see that the proposed system delivers tumor masks of a good quality and the data resulting from 3 experiments have very small deviations. In other words, its results are almost reproducible.

Table 4.6 illustrates the main specifications concerning the computing power requirement. It includes the measures for computation complexity, training time and inference time.

- The computation complexity is indicated by 2 numbers: the total number of the trainable parameters of the entire system and the total number of multiply-accumulate (MAC) operations needed for the system to generate the final mask delineating 3 different tumoral regions. The latter is also related to the data volume of the input. In this test, a patient case consists of four 3D images and each has  $240 \times 240 \times 155$  pixels.
- The training time is related to the batch size, the total number of training samples, the data volume per sample, the number of epochs, the computing system used for the training, etc. As the modules in the proposed system are trained independently, in Table 4.6, their training times are listed.
- The inference time is defined as the time required for a trained system to complete all the data processing for a patient case. It is highly dependent on the type of computing hardware employed. Moreover, as the proposed system has 3 subsystems that can operate concurrently or sequentially, the inference time also depends on the way that the system operates. The inference time presented in the table is in case of the subsystems operating sequentially on an ordinary laptop with GPU NVIDIA A3000 and CPU i7-11800H 2.3 GHz.

**Table 4.5 Test results on BraTS validation sets, assessed by online platforms.**

Dataset		Dice			Sensitivity			FDR		
		ET	WT	TC	ET	WT	TC	ET	WT	TC
<b>BraTS2023/ BraTS2021</b> Training set: 1251 samples Validation set: 219 samples	Expt. 1	0.825	0.918	0.840	0.826	0.925	0.879	0.176	0.090	0.195
	Expt. 2	0.827	0.918	0.834	0.829	0.927	0.879	0.175	0.091	0.207
	Expt. 3	0.830	0.918	0.841	0.830	0.926	0.884	0.170	0.090	0.199
	<b>Mean</b>	<b>0.827</b>	<b>0.918</b>	<b>0.838</b>	<b>0.828</b>	<b>0.926</b>	<b>0.881</b>	<b>0.174</b>	<b>0.090</b>	<b>0.200</b>
<b>BraTS2020</b> Training set: 369 samples Validation set: 125 samples	Expt. 1	0.763	0.897	0.782	0.761	0.907	0.828	0.235	0.112	0.259
	Expt. 2	0.773	0.899	0.778	0.773	0.913	0.833	0.226	0.115	0.271
	Expt. 3	0.770	0.896	0.773	0.770	0.910	0.831	0.231	0.118	0.277
	<b>Mean</b>	<b>0.769</b>	<b>0.897</b>	<b>0.778</b>	<b>0.768</b>	<b>0.910</b>	<b>0.831</b>	<b>0.230</b>	<b>0.115</b>	<b>0.269</b>
<b>BraTS2019</b> Training set: 335 samples Validation set: 125 samples	Expt. 1	0.765	0.893	0.769	0.765	0.910	0.828	0.235	0.125	0.281
	Expt. 2	0.771	0.894	0.767	0.770	0.903	0.835	0.227	0.115	0.291
	Expt. 3	0.777	0.893	0.763	0.772	0.907	0.829	0.217	0.121	0.293
	<b>Mean</b>	<b>0.771</b>	<b>0.893</b>	<b>0.767</b>	<b>0.769</b>	<b>0.907</b>	<b>0.831</b>	<b>0.226</b>	<b>0.120</b>	<b>0.288</b>

**Table 4.6 Computation cost of the proposed CNN system**

Specifications	Values	Notes
Number of trainable parameters	748,422	The total number of the entire system
Number of MACs per patient case	1,956 G	Input data: 4 MR images, each having 240×240×155 pixels
Training time (1251 samples, 50 epochs)	Module 1 ≈ 6 h Module 2 ≈ 6 h	Module 1: GPU 1 × NVIDIA A100 (40 GB memory), CPU 16 cores of AMD EPYC 7413 (Zen 3) @ 2.65 GHz Module 2: GPU 2 × NVIDIA A100 (40 GB memory), CPU 16 cores of AMD EPYC 7413 (Zen 3) @ 2.65 GHz
Inference time per patient case	≈ 2.5 s	Laptop with GPU NVIDIA A3000, CPU i7-11800H @ 2.3 GHz, 3 subsystems operating in series

The performance of the proposed system has also compared with 5 other systems. The criteria used to select these systems for comparison are as follows.

- Using the same input data and performing the same brain tumor segmentation task
- Published recently in reputed research journals in the topic area
- Tested on the same BraTS validation sets and the results assessed by the same online platform

The comparison results are presented in Table 4.7. The data scores are grouped according to the versions of BraTS dataset used in the training/testing. One can see that the number of parameters of the proposed system is significantly smaller than the others, requiring far less

computing power to operate. Its processing quality is, however, among the best reported in literature. Hence, it has no doubt a very high computation efficiency.

It is evident that the proposed system delivers excellent performance with high processing quality and low requirement for computing power. Moreover, thanks to its multi-module structure and independency of each module in training and testing, one can train the modules one by one and operate them one after another. In this way, the minimum computing power required to operate the system is related to its largest module, i.e., Module 2 of 162,145 trainable parameters. Hence, it can be easily implemented for various applications.

**Table 4.7 Performance comparison**  
**Dice and Sensitivity scores resulting from the tests on BraTS validation sets**

Datasets	Systems	Number of parameters	Dice			Sensitivity		
			ET	WT	TC	ET	WT	TC
BraTS2019	Liu et al., 2021 [41]	33.4 M	0.759	0.885	0.851	N.A.	N.A.	N.A.
	An et al., 2024 [26]	5.8 M + 0.02 M	0.694	0.875	0.763	N.A.	N.A.	N.A.
	An et al., 2024 [26]	19 M + 0.06 M	0.730	0.887	0.790	N.A.	N.A.	N.A.
	Liu et al., 2024 [27]	1.90 M	0.771	0.896	0.833	0.780	0.930	0.822
	<b>Proposed</b>	<b>0.75 M</b>	<b>0.771</b>	<b>0.893</b>	<b>0.767</b>	<b>0.769</b>	<b>0.907</b>	<b>0.831</b>
BraTS2020	An et al., 2024 [26]	5.8 M + 0.02 M	0.692	0.873	0.776	N.A.	N.A.	N.A.
	An et al., 2024 [26]	19 M + 0.06 M	0.721	0.888	0.794	N.A.	N.A.	N.A.
	Liu et al., 2024 [27]	1.90 M	0.778	0.900	0.832	0.791	0.920	0.823
	<b>Proposed</b>	<b>0.75 M</b>	<b>0.769</b>	<b>0.897</b>	<b>0.778</b>	<b>0.768</b>	<b>0.910</b>	<b>0.831</b>
BraTS2021	Liang et al., 2022 [39]	20.4 M	0.826	0.926	0.867	N.A.	N.A.	N.A.
	An et al., 2024 [26]	5.8 M + 0.02 M	0.757	0.884	0.803	N.A.	N.A.	N.A.
	An et al., 2024 [26]	19 M + 0.06 M	0.782	0.903	0.839	N.A.	N.A.	N.A.
	Liu et al., 2024 [27]	1.90 M	0.835	0.907	0.889	0.824	0.930	0.895
	Liu and Xia, 2024 [28]	3.65 M	0.787	0.899	0.865	0.848	0.855	0.848
	<b>Proposed</b>	<b>0.75 M</b>	<b>0.827</b>	<b>0.918</b>	<b>0.838</b>	<b>0.828</b>	<b>0.926</b>	<b>0.881</b>

## 4.4 Summary

To handle more and more challenging processing tasks, such as brain tumor segmentation, CNN systems become more and more complex, requiring more and more computing and data resources. In this chapter, we have proposed a CNN design methodology, aiming at develop CNN systems of high computation efficiency, and also proposed a CNN system, applying the methodology, for high quality brain tumor segmentation.

The methodology is to decompose a complex task, step by step, into sub-subtasks processed concurrently or sequentially. Each of them can be performed by a simple CNN module that can be configured and trained independently. For the purpose of training the modules, 2 issues can determine how the task should be decomposed, or how the subtasks or sub-subtasks are defined. One is the feasibility to generate the ground truth data from available datasets, and the other is the facility to build a simple CNN module for a particular task.

The proposed methodology implies a scheme of multi-module CNN system. The particular character in this scheme is that each module can be configured and trained to enhance a particular aspect of performance. If the module is designed to have a simple structure and to emphasize on one sole issue in its operations, there will be little risk of the gradient conflict in its training process. All the modules can be made to operate in such a way that they complement each other to yield overall the best results.

Applying the methodology, we propose a CNN system for brain tumor segmentation. It consists of 3 subsystems for WT, TC and ET segmentations, respectively. Each subsystem has 2 simple custom-design modules operating sequentially. The first generates a low-resolution object-location mask that is used in the second to produce a high-resolution object mask. We have applied a particular training strategy that the first module is made to have a good location coverage, minimizing the false negatives, and the second module is trained to minimize the false positives.

Despite that the proposed system consists of 3 subsystems, it has, in total, only 0.75 M trainable parameters, an insignificant number with respect to those of other systems performing the same function. Moreover, as the modules, of which the largest has only 0.162 M parameters, can be trained separately one after another, such a system can be developed without need for high computing power resources.

The proposed CNN system has been trained with the data from different versions of BRATS dataset. The test results, on the BRATS validation sets, are among the best reported recently in reputed research journals in the topic area, though the system operates with a very tiny fraction of the computing power needed by others. The effectiveness of the design methodology and the high computation efficiency of the system have been confirmed.

# Chapter 5

## Conclusion

Brain tumor detection, or brain tumor segmentation in terms of image processing, is a complex task and there are CNN systems developed to handle it automatically. To achieve a good processing quality, one tends to build complex CNN systems that require a lot of computing power to train the systems and to operate them, which may limit their implementations, their performance in terms of reliability, and their applications.

The objective of the work presented in this thesis is to develop brain tumor segmentation systems with the emphasis on a high computation efficiency, i.e., achieving a good processing quality at a very low computation cost to enable an easy implementation and a wide applicability of the systems. To this end, two different methodologies have been proposed, and also applied in the development of 2 brain tumor detection systems, respectively.

In the first part of the work presented in the thesis, we have proposed a design methodology that uses conventional knowledge-based models to develop a system capable to detect object locations, sizes and shapes in a 3D image. The detection has 2 main operations interleaved each other and performed step-by-step: predicting the gray level distribution of the pixels in the object regions and employing the prediction results to identify/remove regions of non-interest. Each removal increases the density of the object information in the remaining part of the 3D image, improving the prediction and identification/removal in the following step. In the design of the system for whole tumor detection, as each input 3D brain image can be sliced into series of axial, coronal or sagittal slices, the prediction/identification/removal operations are performed in 3 steps to the 3 series of slices, respectively. To comprehend the distributions of the gray levels of the pixels with their locations, we have proposed a 2D histogram presentations of the data from each of the 3 series. Also, the left-right asymmetry of a brain structure is explored to extract brain tumor information and a novel adaptive histogram modulation method is applied to enhance the structural asymmetry related to the presence of tumors and to attenuate that due to non-pathological causes. The 3-step operations turn an input 3D brain image into a minimum 3D bounding box covering the tumor region that is then transformed into a tumor mask by means of simple morphological operations. The system has been tested extensively with the samples of more than 1000 patient cases. The test results have confirmed the high quality of the prediction of the distributions. The average Dice scores of the whole tumor masks is above 80%, which is achieved at a computation cost that is completely negligible with respect to those of the other systems for the same task.

The second part of the work is to develop a new CNN system performing a brain tumor segmentation. It needs not only to detect the whole tumor (WT) regions, which the system presented in the first part of the thesis does, but also to classify each pixel into 4 classes, namely ED, NET, ET and the background. To develop a system for such a complex task, we have proposed

a methodology aiming at designing CNN systems for high-computation-efficient image segmentation. It is to decompose a complex task into several simple subtasks in such a way that each of them can be performed by a simple CNN module configured and trained independently. In this way, one can optimize the use of computing power for processing, on one hand, and minimize the gradient conflict in training, on the other hand. For the design of the CNN system, the task of classifying each pixel into one of the 4 classes is decomposed into 3 binary classifications, each of which is performed in 2 steps: first locating the object region and then identifying the pixels in the region. Thus, we have developed 3 CNN subsystems and each consists of 2 independent and simple modules. One generates low-resolution location maps and the other the final masks delineating its designated tumor regions. The 2 modules are made to complement each other to achieve an overall good performance at a low computation cost. The entire CNN system has been trained and tested with different versions of BraTS dataset, and its average Dice score, obtained on the BraTS2023 validation set, is 82.7% for ET, 91.8% for WT and 83.8% for TC, which is among the best reported recently, whereas the system requires only 0.75 M trainable parameters, a tiny fraction of the others.

This research work provides, to the topic area of computer vision for medical image processing, with the design methodologies to develop high-quality object detection/segmentation systems. It demonstrates that one can choose to use knowledge-based structures to perform a difficult task. To do it, the key is to understand the characters/features of the input images and to make good use of them, i.e., finding the hidden correlation between the input features and the output data to be generated. The advantage of this kind of knowledge-based systems is, obviously, its extremely low-computation cost, and little dependency on data resources as no training is needed. More importantly, it yields consistent performance and the results are always reproducible. However, such a system has its limitation, and may not be able to handle a very complex task, such as multi-class segmentation with more variations in objects. In this case, the proposed methodology aiming at CNN design can be useful. Applying this methodology, we have developed the CNN system of multiple independently-trained modules. It demonstrates that a CNN system can be made computation-efficient, i.e., good processing quality achieved without complex networks. The keys in this design are (i) appropriately decomposing the task and well defining the subtasks that one can easily implement in simple and independently-trained CNN modules, (ii) custom-design each module and understanding its design constrains, and (iii) using a good training strategy to optimize the parameter updating.

Brain tumor detection is a specific kind of object detections. We will apply the methodologies proposed in this thesis to develop systems for other object detection/segmentation tasks. Hopefully, our work will have a positive effect in the area of object detection/recognition and let everybody pay more attention to the computation efficiency. In this way, even with limited available computation and data resources, more developments in designing efficient processing systems can be expected in the important area.

# Reference

- [1] Gabor, D. (1946). Theory of communication. Part 1: the analysis of information. *J Inst Electr Eng-Part III: Radio Commun Eng* 93 (26): 429–441.
- [2] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.
- [3] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
- [4] Ioffe, S., & Szegedy, C. (2015, June). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, 448-456.
- [5] Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.
- [6] Ulyanov, D., Vedaldi, A., & Lempitsky, V. (2016). Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*.
- [7] Wu, Y., & He, K. (2018). Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, 3-19.
- [8] Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, 807-814.
- [9] Good, I. J. (1963). Maximum entropy for hypothesis formulation, especially for multidimensional contingency tables. *The Annals of Mathematical Statistics*, 911-934.
- [10] Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3), 297-302.
- [11] Kelley, H. J. (1960). Gradient theory of optimal flight paths. *Ars Journal*, 30(10), 947-954.
- [12] Lillicrap, T. P., Santoro, A., Marris, L., Akerman, C. J., & Hinton, G. (2020). Backpropagation and the brain. *Nature Reviews Neuroscience*, 21(6), 335-346.
- [13] Soltaninejad, M., Yang, G., Lambrou, T., Allinson, N., Jones, T. L., Barrick, T. R., Howe, F. A. & Ye, X. (2017). Automated brain tumour detection and segmentation using superpixel-based extremely randomized trees in FLAIR MRI. *International journal of computer assisted radiology and surgery*, 12, 183-203.
- [14] Imtiaz, T., Rifat, S., Fattah, S. A., & Wahid, K. A. (2019). Automated brain tumor segmentation based on multi-planar superpixel level features extracted from 3D MR images. *IEEE Access*, 8, 25335-25349.
- [15] Gilanie, G., Bajwa, U. I., Waraich, M. M., Habib, Z., Ullah, H., & Nasir, M. (2018). Classification of normal and abnormal brain MRI slices using Gabor texture and support vector machines. *Signal, Image and Video Processing*, 12, 479-487.
- [16] Wu, W., Chen, A. Y., Zhao, L., & Corso, J. J. (2014). Brain tumor detection and segmentation in a CRF (conditional random fields) framework with pixel-pairwise affinity and superpixel-level features. *International journal of computer assisted radiology and surgery*, 9, 241-253.
- [17] Rehman, Z. U., Zia, M. S., Bojja, G. R., Yaqub, M., Jinchao, F., & Arshid, K. (2020). Texture based localization of a brain tumor from MR-images by using a machine learning approach. *Medical hypotheses*, 141, 109705.
- [18] Venkatachalam, K., Siuly, S., Bacanin, N., Hubálovský, S., & Trojovský, P. (2021). An efficient Gabor Walsh-Hadamard transform based approach for retrieving brain tumor images from MRI. *IEEE Access*, 9, 119078-119089.
- [19] Lim, K. Y., & Mandava, R. (2018). A multi-phase semi-automatic approach for multisequence brain tumor image segmentation. *Expert systems with applications*, 112, 288-300.

- [20] Bennai, M. T., Guessoum, Z., Mazouzi, S., Cormier, S., & Mezghiche, M. (2020). A stochastic multi-agent approach for medical-image segmentation: application to tumor segmentation in brain MR images. *Artificial Intelligence in Medicine*, 110, 101980.
- [21] Bonte, S., Goethals, I., & Van Holen, R. (2018). Machine learning based brain tumour segmentation on limited data using local texture and abnormality. *Computers in biology and medicine*, 98, 39-47.
- [22] Khotanlou, H., Colliot, O., Atif, J., & Bloch, I. (2009). 3D brain tumor segmentation in MRI using fuzzy classification, symmetry analysis and spatially constrained deformable models. *Fuzzy sets and systems*, 160(10), 1457-1473.
- [23] Saha, B. N., Ray, N., Greiner, R., Murtha, A., & Zhang, H. (2012). Quick detection of brain tumors and edemas: A bounding box method using symmetry. *Computerized medical imaging and graphics*, 36(2), 95-107.
- [24] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention—MICCAI 2015: 18th international conference*, 234-241.
- [25] Wu, X., Bi, L., Fulham, M., Feng, D. D., Zhou, L., & Kim, J. (2021). Unsupervised brain tumor segmentation using a symmetric-driven adversarial network. *Neurocomputing*, 455, 242-254.
- [26] An, D., Liu, P., Feng, Y., Ding, P., Zhou, W., & Yu, B. (2024). Dynamic weighted knowledge distillation for brain tumor segmentation. *Pattern Recognition*, 155, 110731.
- [27] Liu, H., Huang, J., Li, Q., Guan, X., & Tseng, M. (2024). A deep convolutional neural network for the automatic segmentation of glioblastoma brain tumor: Joint spatial pyramid module and attention mechanism network. *Artificial Intelligence in Medicine*, 148, 102776.
- [28] Liu, L., & Xia, K. (2024). BTIS-Net: Efficient 3D U-Net for Brain Tumor Image Segmentation. *IEEE Access*, 12, 133392 – 133405.
- [29] 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.
- [30] Yang, Q., Guo, X., Chen, Z., Woo, P. Y., & Yuan, Y. (2022). D2-Net: Dual disentanglement network for brain tumor segmentation with missing modalities. *IEEE Transactions on Medical Imaging*, 41(10), 2953-2964.
- [31] Rahimpour, M., Bertels, J., Radwan, A., Vandermeulen, H., Sunaert, S., Vandermeulen, D., Maes, F., Goffin, K., & Koole, M. (2021). Cross-modal distillation to improve MRI-based brain tumor segmentation with missing MRI sequences. *IEEE Transactions on Biomedical Engineering*, 69(7), 2153-2164.
- [32] Nema, S., Dudhane, A., Murala, S., & Naidu, S. (2020). RescueNet: An unpaired GAN for brain tumor segmentation. *Biomedical Signal Processing and Control*, 55, 101641.
- [33] 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.
- [34] 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.
- [35] Ding, Y., Zheng, W., Geng, J., Qin, Z., Choo, K. K. R., Qin, Z., & Hou, X. (2021). MVFusFra: A multi-view dynamic fusion framework for multimodal brain tumor segmentation. *IEEE Journal of Biomedical and Health Informatics*, 26(4), 1570-1581.
- [36] Zhao, J., Xing, Z., Chen, Z., Wan, L., Han, T., Fu, H., & Zhu, L. (2023). Uncertainty-Aware multi-dimensional mutual learning for brain and brain tumor segmentation. *IEEE Journal of Biomedical and Health Informatics*, 27(9), 4362-4372.
- [37] Chen, S., Ding, C., & Liu, M. (2019). Dual-force convolutional neural networks for accurate brain tumor segmentation. *Pattern Recognition*, 88, 90-100.

- [38] Wang, Z., Zou, Y., Chen, H., Liu, P. X., & Chen, J. (2024). Multi-scale features and attention guided for brain tumor segmentation. *Journal of Visual Communication and Image Representation*, 100, 104141.
- [39] Liang, J., Yang, C., & Zeng, L. (2022). 3D PSwinBTS: an efficient transformer-based Unet using 3D parallel shifted windows for brain tumor segmentation. *Digital Signal Processing*, 131, 103784.
- [40] Zhuang, Y., Liu, H., Song, E., & Hung, C. C. (2022). A 3D cross-modality feature interaction network with volumetric feature alignment for brain tumor and tissue segmentation. *IEEE Journal of Biomedical and Health Informatics*, 27(1), 75-86.
- [41] Liu, Z., Tong, L., Chen, L., Zhou, F., Jiang, Z., Zhang, Q., Wang, Y., Shan, C., Li, L., & Zhou, H. (2021). Canet: Context aware network for brain glioma segmentation. *IEEE Transactions on Medical Imaging*, 40(7), 1763-1777.
- [42] Sun, Y., & Wang, C. (2024). Brain tumor detection based on a novel and high-quality prediction of the tumor pixel distributions. *Computers in Biology and Medicine*, 172, 108196.
- [43] RSNA-ASNR-MICCAI, Multimodal brain tumor segmentation challenge 2021 (BraTS2021). <http://braintumorsegmentation.org/>, 2021 (accessed 1st July 2022).
- [44] Geschwind, N., & Levitsky, W. (1968). Human brain: left-right asymmetries in temporal speech region. *Science*, 161(3837), 186-187.
- [45] Sun, Y., & Wang, C. (2022). A computation-efficient CNN system for high-quality brain tumor segmentation. *Biomedical Signal Processing and Control*, 74, 103475.
- [46] RSNA-ASNR-MICCAI, Synapse platform. <https://www.synapse.org/#!/Synapse:syn25829067/wiki/610863>, 2021 (accessed 6th September 2023).
- [47] Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., ... & Van Leemput, K. (2015). The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE transactions on medical imaging*, 34(10), 1993-2024.
- [48] Baid, U., Ghodasara, S., Mohan, S., Bilello, M., Calabrese, E., Colak, E., ... & Bakas, S. (2021). The RSNA-ASNR-MICCAI BraTS 2021 benchmark on brain tumor segmentation and radiogenomic classification, [arXiv:2107.02314](https://arxiv.org/abs/2107.02314).
- [49] Bakas, S., Akbari, H., Sotiras, A., Bilello, M., Rozycki, M., Kirby, J. S., Freymann, J. B., Farahani, K., & Davatzikos, C. (2017). Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features. *Scientific data*, 4, 170117.
- [50] Bakas, S., Reyes, M., Jakab, A., Bauer, S., Rempfler, M., Crimi, A., ... & Jambawalikar, S. R. (2018). Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge. [arXiv preprint arXiv:1811.02629](https://arxiv.org/abs/1811.02629).
- [51] MICCAI, Multimodal brain tumor segmentation challenge 2018 (BraTS2018). <https://www.med.upenn.edu/sbia/brats2018/data.html>, 2018 (accessed 30th April 2018).
- [52] MICCAI, Multimodal brain tumor segmentation challenge 2019 (BraTS2019). <https://www.med.upenn.edu/cbica/brats2019/data.html>, 2019 (accessed 10th June 2019).
- [53] MICCAI, Multimodal brain tumor segmentation challenge 2020 (BraTS2020). <https://www.med.upenn.edu/cbica/brats2020/data.html>, 2020 (accessed 18th May 2020).
- [54] CBICA, Center for biomedical image computing and analytics image processing portal (CBICA IPP), <https://ipp.cbica.upenn.edu/>, 2017 (accessed 1st January 2018).
- [55] Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4), 600-612.
- [56] Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1), 289-300.

- [57] RSNA-ASNR-MICCAI, Brain Tumor Segmentation 2023 (BraTS2023). <https://www.synapse.org/Synapse:syn51156910/wiki/621282>, 2023 (accessed 14th November 2023).
- [58] Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization, 3rd International Conference for Learning Representations, San Diego.
- [59] Loshchilov, I., & Hutter F. (2017). Sgdr: stochastic gradient descent with warm restarts. International Conference on Learning Representations 2017.
- [60] 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, 1026-1034.