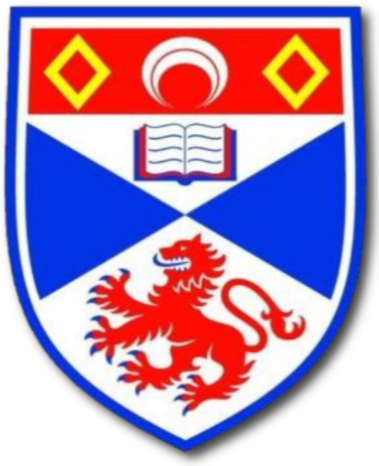


University of St Andrews

School of Computer Science

Brain Tumour Segmentation (in MRI) using Deep Learning Project

Daria Sergeevna Tkachova

Matriculation Number: 190024486

MSc in Artificial Intelligence

CS5098 - GROUP PROJECT AND DISSERTATION IN COMPUTER SCIENCE

19 August 2020

Declaration

I declare that the material submitted for assessment is my own work except where credit is explicitly given to others by citation or acknowledgement. This work was performed during the current academic year except where otherwise stated.

This project was completed in collaboration with two group members, Carina Norregaard (matriculation number: 150004238) and Neha Ghatia (matriculation number: 190024130) under the supervision of Dr. David Harris-Birtill and Dr. Lewis McMillan together with the assistance of an external advisor Dr Roushanak Rahmat.

The main text of this project report is approximately 14,000 words long, including project specification and objectives.

In submitting this project report to the University of St Andrews, I give permission for it to be made available for use in accordance with the regulations of the University Library. I also give permission for the title and abstract to be published and for copies of the report to be made and supplied at cost to any bona fide library or research worker, and to be made available on the World Wide Web. I retain the copyright in this work.

Signature	**Date**
Daria Sergeevna Tkachova	19/08/2020

Acknowledgements

I would like to thank my supervisor, Dr. David Harris-Birtill for his valuable recommendations, explanations, guidance and suggestions throughout this project.

I would like to thank my second supervisor Dr. Lewis McMillan for all his help, suggestions and encouragement with this project.

I would like to give special thanks to my parents, Dr. Viktoria Tkachova and Sergi Tkachov, without whom all this would not have been possible. And finally I would like to thank my sister, Marina Lofthouse for all her support throughout this year.

Abstract

Manual brain tumour segmentation is a difficult and time consuming process which needs to be completed by an experienced radiologist as zero error is allowed to occur in this process. This task is especially difficult as brain tumours vary in size and location and have a tentacle-like structure with which they infiltrate the surrounding healthy tissue. The aim of automatic brain tumour segmentation is to assist certified clinicians in the brain tumour segmentation process and the consequent patient diagnosis and treatment. Even though no fully automated segmentation procedure is deployed in industry, there is progress in the application of deep neural networks for brain tumour segmentation. This project presents three models which were able to achieve brain tumour segmentation with their performance being measure with a Dice score metric per tumour region. These are, a 2D U-Net developed by the group, an individually developed 2D U-Net and a 3D U-Net. Dice scores of the best overall performing model, the 3D U-Net model, are: 0.7960, 0.6001, 0.4368 and 0.6396, 0.8581, 0.8181 for whole tumour, tumour core and enhancing tumour on low-grade gliomas (LGG) and high-grade gliomas (HGG) data respectively.

Contents

List of Figures

List of Tables

1 Introduction

This is a group project which aims to implement deep learning to perform brain tumour segmentation of Magnetic Resonance Imaging (MRI). This project will focus on the analysis of a publicly available, open source Multimodal Brain Tumour Image Segmentation Benchmark (BraTS) [35], [3], [4], [5], [6] benchmark data set [22]. Ethical approval has been received for this project as described in Section 1.2. The School of Computer Science GPU lab client machines were used to develop both the group model and my individual model implementations. The main results of this project are a literature review, given in Section 2, and the consequent development of three deep neural network algorithms for the segmentation of brain tumour MRI images. The fist neural network has been developed by the group consisting of Carina Norregaard (matriculation number: 150004238) and Neha Ghatia (matriculation number: 190024130) and I. We implemented a 2D U-Net adaptation of [48]. The two consecutive deep neural network algorithms we developed as part of my individual work. These consist of the convolutional neural network implementations, namely a two-dimensional U-Net implementation based on the work of [46] and the subsequent three-dimensional U-Net implementation which was also based [46]. This model uses two slices of volumetric information for neural network training and evaluation. The performance of each of the algorithms was measured using the Dice score metric. Furthermore, the Dice scores were mearured per each individual tumour region (the multi-class case) [55]. As can be seen from Table 15 in Section 5.4, the result Dice scores achieved by these models are comparable to state-of-the-art techniques. Furthermore, the [46] based 3D U-Net implementation seems give the best performance results due to its the best overall performance for both the 210 high grade glioma (HGG) [55] and 75 low grade glioma (LGG) [55] data when evaluated on the BraTS 2018 Challenge test set as described in Section 1.3. The final results of the three U-Net implementations are provided in Table 16 and illustrated in Figure 16 of Section 5.5. Sample images of brain tumour segmentation per one patient are provided in Section 5. Figure 13 shows low grade glioma (LGG) [55] data while Figure 15 shows samples of high grade glioma (HGG) [55]. Even though these are sample images and the segmentation is performed across the data set, these figures help to visualise and illustrate the performance of the models that has been measured in Dice score. In the clinical context, using convolutional neural networks to accomplish brain tumour segmentation may assist radiologist by performing automatic brain tumour segmentation, thereby saving time [15] and assisting to circumvent any possible brain tumour segmentation variability [64] that may occur in manual human based brain tumour segmentation [64].

1.1 Project Objectives

The aim of this project was the application of deep learning techniques to the Multimodal Brain Tumour Image Segmentation Benchmark (BraTS) [35], [3], [4], [5], [6] benchmark data set [22], in order to survey and improve the classification and segmentation of brain tumours within magnetic resonance imaging (MRI) brain images.

This is a group project where the primary objectives include an initial pipeline implementation and secondary objectives entail the subsequent deep learning algorithm implementations as shown in detail in Figure 11 in Section 3.

The implementation of the primary objectives which entail the initial pipeline was completed as a group and consisted of a BraTS brain tumour magnetic resonance imaging (MRI) image data segmentation implementation and entailed the loading in of the

data, data pre-processing together with the implementation and evaluation of a brain tumour segmentation algorithm. This initial pipeline was completed as a group, while the consequent implementation of deep learning algorithms including their specific machine learning architectures were completed individually. This is shown in Figure 11in Section 3. The details of the segmentation algorithm used for the initial pipeline implementation and the individual tasks were identified and guided by the literature review as presented in Section 2.

The following project objectives were identified:

1.1.1 Primary Objectives

- Complete a literature review which surveys the state of the art on the topic of Brain Tumour Segmentation (in MRI) using deep learning and formulate research questions.
- Implement a pipeline that will read in and pre-process the data from the open source data set (BraTS) [35], [3], [4], [5], [6].
- Create a segmentation of brain tumour images with an implementation of a segmentation algorithm.

1.1.2 Secondary Objectives

- Create a segmentation of brain tumour images with a re-implementation of an existing algorithm as identified in the literature review.
- Implement more advanced algorithms or optimise existing deep learning algorithms to possibly find a novel solution for Brain Tumour Segmentation.
- Compare the performance and results of algorithms and the deep learning technique implementation from both my own model analysis and the results achieved by the team.

1.2 Ethical Considerations

Ethical approval for this project has been received with the following corresponding Approval Code CS14954. The full ethics approval document has been appended to the appendix and is found in Appendix A.1 at the end of this document. The Multimodal Brain Tumour Image Segmentation Benchmark (BraTS) [35], [3], [4], [5], [6] benchmark data set [22], is an open source publicly available data set which has been provided on the website of the Perelman School of Medicine at the University of Pennsylvania [43]. This data is available from:

http://braintumorsegmentation.org/.

The BraTS data was obtained by completing a request for access through a registration on the CBICA Image Processing Portal (available at: ipp.cbica.upenn.edu) and consequently acquiring the required permission to download the BraTS data set as per the instructions provided on [44].

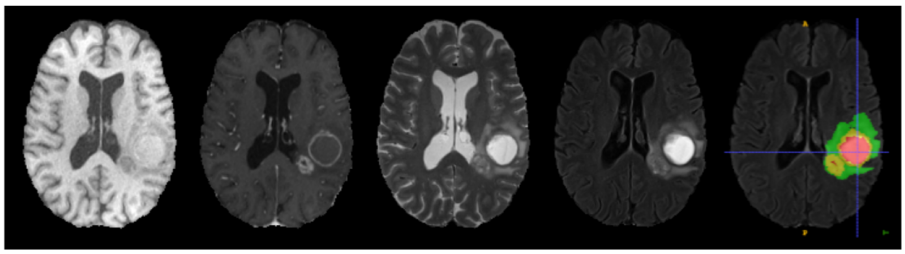


Figure 1: An illustration of the BraTS 2018 data set MRI modalities. The structural modalities left to right are: T1, T1c, T2, FLAIR, together with the ground truth (GT) segmentation. Tumour sub-regions are displayed per colour, with green: edema, yellow: enhancing tumour, and red: necrotic and non-enhancing tumour core. [55].

1.3 Data

The Multimodal Brain Tumour Image Segmentation Benchmark (BraTS) [35], [3], [4], [5], [6] benchmark data set [22] includes the brain tumour segmentation BraTS 2018 Challenge data set [43]. The BraTS 2018 Challenge data was chosen to be used for this project as this is the BraTS 2018 data set used by [55] as identified in the literature review in Section 2. The BraTS 2018 data set contains glioma brain tumour MRI images [55]. Gliomas are discussed in further detail in Section 2.1. The BraTS 2018 data set contains the MRI images of 210 high grade glioma (HGG) patient cases and 75 low grade glioma (LGG) patient cases [55]. For each patient in the BraTS 2018 data set, the data includes for each subject the four magnetic resonance imaging (MRI) structural modalities, namely T1, T1c, T2, and FLAIR, together with the segmentation files containing pixel-wise annotated, complete tumour ground truths (GT) images [55] as shown in Figure 1.

As described in [33] the images in of the BraTS 2018 data set are skull-stripped and were manually segmented with the approval of experienced neuro-radiologists [33]. Labels were assigned per target class with tumour core, peritumoral edema and enhancing tumour assigned labels 1, 2, 4 respectively [33]. While each of these four target classes are independent, there is an encompassing relationship between them as shown in Figure 2 [70].

All the BraTS multimodal brain scan files are provided in the format of NIfTI (.nii.gz) [43]. NIfTI (Neuroimaging Informatics Technology Initiative) [38] files store MRI data with the correct brain image orientation which allows the correct identification of both sides of the brain [38]. Figure 3 displays an in-depth visualisation of a FLAIR image using the ITK-SNAP tool developed in [66].

These modalities are considered to be the standard MRI image classifications used in the diagnosis of glioma, where T1-weighted MRI (T1) are used for the identification of healthy brain tissue [22]; followed by T2-weighted MRI (T2) images used to delineate the edema tumour regions which produce a bright signal on the MRI image [22] together with the Fluid Attenuated Inversion Recovery (FLAIR) MRI images which assist in highlighting the difference between the tumour edema region and Cerebrospinal Fluid (CSF) [22] as these images are taken in such a way that the signal produced by water molecules are suppressed thereby creating this contrast on the image [22]. The development of the BRATS benchmark data set, which is used for automatic brain tumour segmentation, has allowed for an objective comparison of various glioma seg-

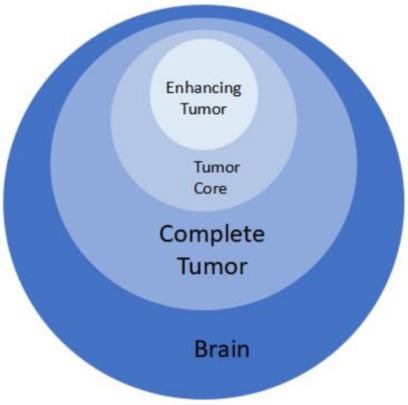


Figure 2: The relationship between the four tumour target classes as illustrated in [70]. The labeled classes of tumour are independent of each other with a containing relationship between four of the target classes [70]. These relationships are as follows: tumour core contains enhancing tumour, non-enhancing tumour and necrosis, complete tumour contains tumour core and edema [70].

mentation methods to take place on a set of common data [22].

The BraTS 2018 Challenge data is a training data set [43] and furthermore, according to [22], the testing data of this data set contains brain tumour scans of undetermined ground truths, with the evaluation on the BraTS test data only being possible through an online evaluation tool [22]. It is for this reason that in the implementation of this project our group solution implementation and later my two individual model implementations used a splitting of the BraTS 2018 training data set, containing the data of 285 patients [55], into further separate validation and testing sets. The BraTS test data presents results using metrics in the form of a Dice Score, Sensitivity and Specificity for the evaluation of the principle tumour regions [22]; namely the whole tumour (comprising of all tumour components) [22], the core tumour (comprising of all tumour components except for the edema) and the active tumour (comprising of only active cells) [22]. For each of the three tumour regions the Dice score is given by the following equation [22]:

$$Dice(P,T) = \frac{|P1 \wedge T1|}{(|P1| + |T1|)/2}$$

where P1 represents voxels of the predicted tumour area (which is the result of an applied segmentation algorithm) and T1 is the actual tumour area as indicated in the provided ground truth MRI images [22]. The dice score is then calculated for each tumour region where the intersect symbol represents the logical AND symbol and the absolute symbol shows the set size as the number of voxels that belong to that set [22]. Furthermore, the Dice score is identical to the F measure [17]. The F measure is also referred to as the F1 score [49] which is also known as the dice coefficient (DSC) [18]. The Dice coefficient is metric which measures the overlap between what has been predicted and the ground truth [25] as illustrated in Figure 4. This metric has a range of 0 to 1 where a complete overlap between the prediction and ground truth is represented by 1 [25]. Specificity is a true negative rate or a ratio of negative class instances which

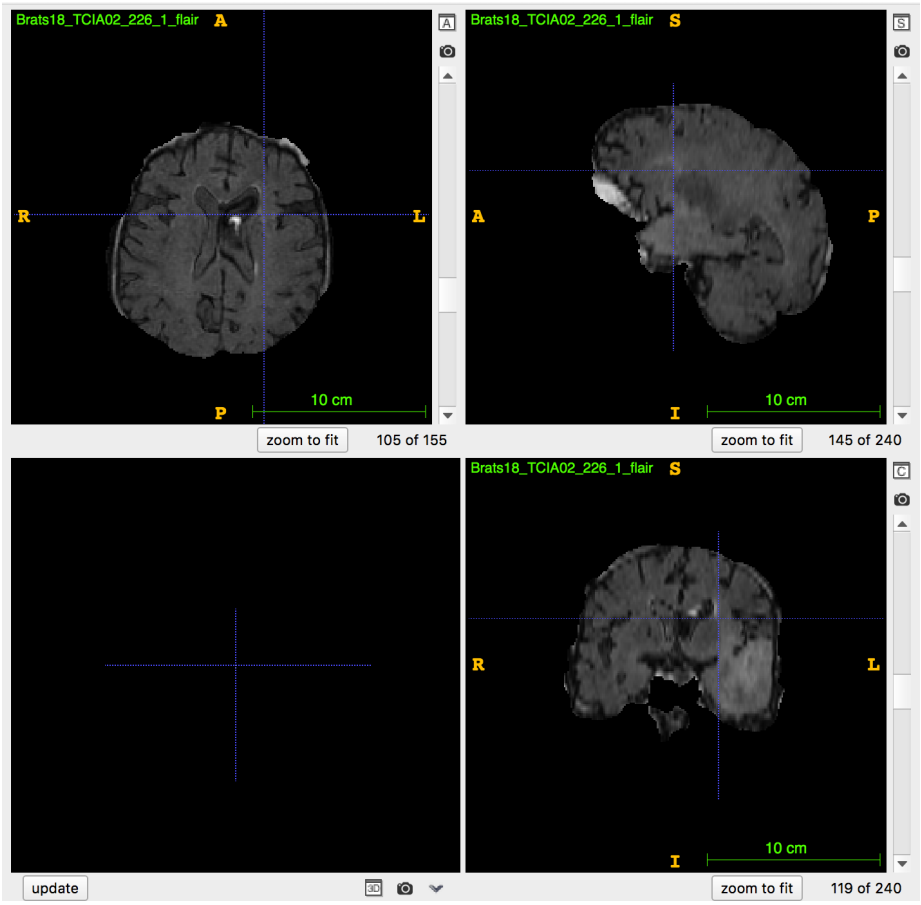


Figure 3: BraTS 2018 data set FLAIR MRI modality image examined from the ITK-SNAP tool developed by [66] to assist with exploring the data set images.

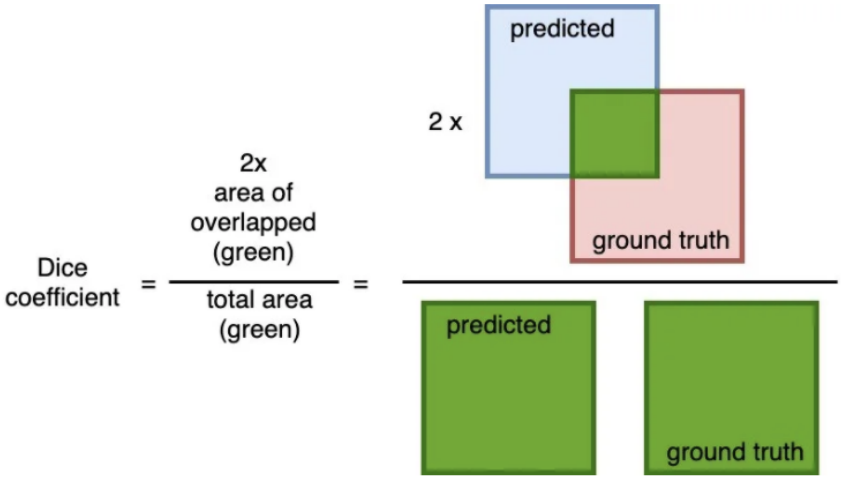


Figure 4: Dice coefficient illustration provided by [25].

are correctly classified as negative, while Sensitivity is the metric describing the true positive rate or a ratio of positive instances which have been correctly identified by the classifier [13].

1.4 Background

1.4.1 Image Segmentation

Image Segmentation requires an algorithm to identify both the location and shape of the object which a model has been trained to recognise [25]. Labels are required to be assigned to every pixel of the image, such that pixels with the same label belong to one object [25]. Image segmentation models are able to accurately identify an object found in an image [25]. The differences between Image Classification, Object Detection and Image Segmentation are illustrated in Figure 5 by [25]. Image Classification performs image content classification by identifying whether an type of object is present within an image [25]. This would be, for example, done by predicted whether a cat object is present in an image [25]. Object Detection identifies where an object is location within an image [25] and this would be, for example, done by predicted where a cat object is present in the image [25]. In Image Segmentation, however, a pixel-wise mask is created for each object that is present within an image in order to identify each object [25]. In this way every pixel of an image is classified with the corresponding labels in order to find both the outline and the location of each object present within the image [25].

1.4.2 Deep Learning Methods

Deep Neural Networks (DNN) are an example of deep learning methods used for brain tumour segmentation [17], in particular the use of Convolutional Neural Networks (CNN), which are a specifically designed type of DNN that is suitable for image data analysis [17].

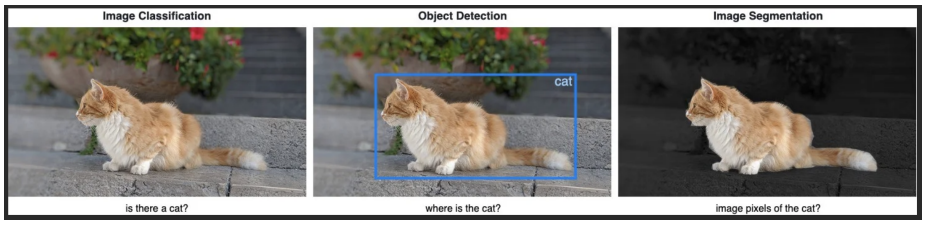


Figure 5: Image illustrating Image Classification, Object Detection and Image Segmentation as given in [25].

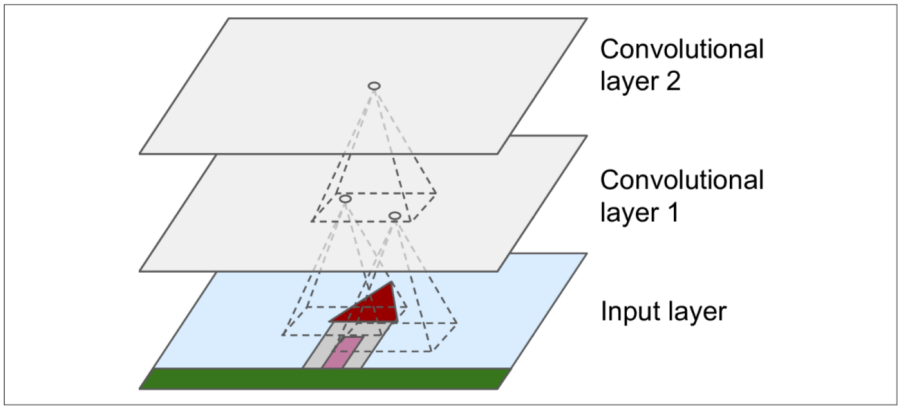


Figure 6: Illustration of the different CNN layers with their respective receptive fields as presented in [13].

1.4.3 Convolutional Neural Networks (CNNs)

CNNs complete complex visual tasks such as image searches and video classification [13]. Furthermore, these neural networks are also able to perform voice recognition and natural language processing [13]. As shown in Figure 6, CNNs consist of a number of convolutional layers with each layer processing a section of the input image, namely the pixels of the input image found within their receptive fields [13]. In order for each convolutional layer to have uniform width and height dimensions, zero padding is used [13]. This is the process of adding zeros added around the input [13]. These layers detect patterns found within an image by sliding across the image with convolution kernels or filters [13]. Filters are matrices of numbers [13]. An example is a matrix of zeros with a central column of ones which represents a black square that has a white line along the middle [13]. Each convolutional layer that uses a filter will output a feature map which will show features of the original image which had the most effect on the activation of that filter [13]. The most suitable filters for a given task are identified by a convolutional layer durring training [13]. The convolutional layer within CNN models allows the neural net architecture to stack several layers with each layer extracting features from preceding layers thereby creating a feature map [17]. Figure 7 illustrates the combined effect of two feature maps [13].

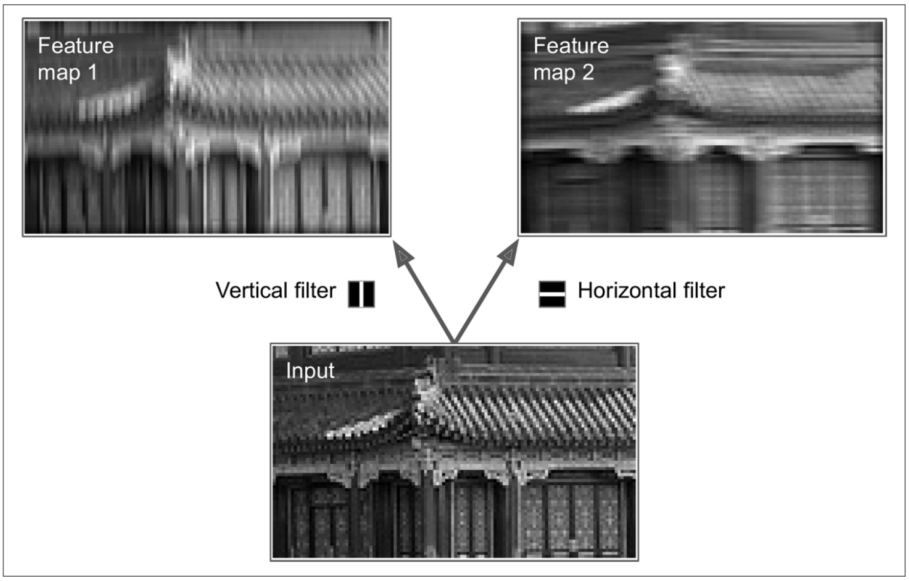


Figure 7: Illustration of the combined effect of two feature maps as given in [13].

2 Literature Review

2.1 Clinical Background

A brain tumour may be described as a set of abnormal cells which grow in the brain and are characterised by the way they reproduce in an uncontrolled manner [7]. There are several types of brain tumours which occur in adult patients, these may be classified as either benign tumours or malignant tumours [7]. Benign tumours are noncancerous, and are characterised by their less aggressive nature, forming slowly and staying isolated from the normal brain tissues surrounding them as benign tumours do not spread to other regions of the brain or other parts of the body[7]. The malignant type of brain tumours, however are cancerous and these tumour types are the more dangerous cell structures as they infiltrate the surrounding tissue regions [7]. The most frequently occurring malignant or cancerous type of tumours occurring in adult patients are Gliomas [33].

These types of tumours begin to occur within cell structures known as glial cells, which surround nerve cells and assist nervous cell functionality by providing both structural support and insulation [23]. There exist different types of glial cells and tumours which occur within each type of glial cell are each given a unique classification according to their cell of origin [16]. The first type of glial cells are called astrocytes and tumours which occur in these cells are classified as astrocytoma, anaplastic astrocytoma and glioblastoma [16]. The second type of glial cells are known as oligodendrocytes [23] and tumours which begin in this type of cells are called oligodendrogliomas [16]. The third type of glial cells found in the brain are ependymal cells [59] and tumours which occur within these structures are known as ependymomas [16]. Glioblastoma Multiforme (GBM) is the term used to describe the most malignant and prevalent types of primary astrocytomas [17]. In order to further assist with the classification of gliomas, the World Health Organisation (WHO) applies an international

standard nomenclature for the diagnosis of gliomas [16]. Gliomas are thus classified into grade I to IV according to their rate of growth and malignancy level as determined per a histopathological criteria [16]. Grade I gliomas are slow growing and it is possible to remove these tumours with surgical procedure, while fast growing tumours are classified as grade II to IV gliomas and are both invasive and highly malignant with the most aggressive and invasive type of glioma tumours classified as grade IV astrocytomas and referred to as Glioblastoma Multiforme (GBM) [16]. It has been found that patients with the slower growing, less aggressive low grade tumours such the astrocytomas or oligodendrogliomas may have a further life expectancy of several years [33], while those patients with more aggressive, high grade glioma tumours have a life expectancy of approximately fourteen to fifteen months from the time of diagnosis [17]. In both groups of patients, intensive neuroimaging protocols are used both prior to and after treatment in order to evaluate whether the cancer continues to grow or spread and establish the effectiveness of the chosen treatment strategy [33].

Being able to occur within any region of the brain, glioma tumours vary in size and shape, contain tentacle-like structures with which they are able to infiltrate surrounding tissue the extension of which is difficult to identify and locate, and having a low level of contrast with surrounding cells makes them difficult to localise and segment from the healthy brain tissue and other brain artefacts [33]. Magnetic Resonance Imaging or MRI is a noninvasive medical imaging technique used for clinical diagnosis which produces high spatial resolution images that are able to identify a high level of contrast between soft tissues [7]. For this reason the majority of brain tumour diagnosis and the precise identification of their exact position and boundaries is done using MRI images. [7]. Furthermore, glioma tumours are difficult to distinguish from the normal brain tissue surrounding the tumours as the tumour intensities overlap with the intensities of their surrounding tissues and thereby tumours delineation becomes particularly difficult along the tumour borders [7]. Additionally, the edges of tumours are identified by the intensity of changes as seen between the healthy tissues and tumour effected cells which is conducted as accurately as possible with manual studies of brain tumour magnetic resonance imaging (MRI) scans by expert radiologists [33]. Brain tumour segmentation is the process of detection and extraction of the tumour region from healthy brain tissues [7]. Accurate and effective segmentation of brain tumours is a challenging task as brain tumours appear in different locations in the brain, their shape varying in size, appearing as nonrigid, complex structures with various appearance properties.[7].

The brain tumour segmentation methods are currently classified into the categories of manual segmentation, semiautomatic segmentation and fully automatic segmentation according to the degree of manual human intervention required for each segmentation task [15]. Due to the high level of accountability and responsibility required from the accuracy of segmentation, the current methods used in industry rely mainly on manual or semi automated techniques as zero degree of error in image analysis interpretation is acceptable [15]. This is both a difficult and time consuming process which needs to be completed by a radiologist [22]. Furthermore, in manual segmentation human experts such as radiologists, anatomists and trained technologists are required to not only make use of the information presented in each image but also rely on additional knowledge of the anatomy and the selection of the tumour region, which is referred to as the region of interest (ROI) [15]. This is not only a tedious and time consuming task as the human expert needs to manually draw tumour regions on each slice [15], but the images presented to the human expert view show the data as a series of two-dimensional slices thereby limiting their ability in using three-dimensional information for the segmentation task as three-dimensional based modeling of struc-

tures for the extensive range of views of these anatomical objects is not available [15]. Additionally manual segmentation is dependent on the individual radiologist and the segmentation results vary [64].

Therefore fully automated and highly accurate analysis of these types of brain tumour scans which would be able to reproduce the measurements of tumour structures would greatly assist doctors in the diagnosis and treatment of patients [33]. The task of brain tumour segmentation consists of separating the brain tumour tissue modalities including the solid or active tumour, edema, and necrosis from the other, normal brain tissues which consist of gray matter (GM), white matter (WM) and cerebrospinal fluid (CSF) [15]. This task of fully automatic segmentation is difficult due to the fact that abnormal tissue of tumours are structures with a complex three-dimensional shape which are not uniform and have a high degree of variation in both size, position, density [15]; while the image segmentation process requires an image to be partitioned in a uniform way such that each image region is related to each other region in such away as not to coincide [15]; with each region being homogenous and adhering to a certain set criteria [15]. Which means that few features such as tumour anatomical knowledge, expected location, size, contour appearance and shape, or possibly bilateral symmetry are available that would be able to serve as a basis for the facilitation of automatic segmentation methods [15]. Furthermore automatic brain tumour segmentation models are required to not only have the ability to accurately describes the size, shape, location and appearance of each brain tumour but also be able to account for variations of these characteristics that are expected in practice [15]. According to [15] currently the is no completely automatic segmentation method which has been adopted for use in the clinic environment [15].

2.2 Segmentation Methods and Algorithms

Several methods and approaches have been explored and implemented in order to achieve this type of fully automatic brain tumour segmentation [15]. These include region-based techniques such as Watershed and pixel classification methods which, in brain tumour segmentation, use either supervised or unsupervised classifiers to group the pixels within a given feature space [15]. Supervised segmentation methods require a set of manually labelled training data on which the algorithm will run the initial training phase during which the model learns the features of the data set from the provided labels [15]. This process will consequently enable the model to apply this knowledge to unseen data during final model performance testing [15]. As the BraTS data set specifically contains labelled ground truth data per patient [33], the focus in this literature survey will be on supervised segmentation methods. Unlike supervised segmentation methods, unsupervised methods do not require manually labelled training data and implement algorithms that automatically group together similar pixels [15]. In this way image-based features such as intensities and textures are identified and the original image becomes divided into homogeneous regions [15]. In this way the number of required classes is found as each homogeneous region falls into one of these classes [15]. However, there exist several disadvantages when using unsupervised techniques for image segmentation [47]. These include the need to pre-specify the number of regions, the possibility of tumours being divided into further multiple regions and the lack of textural boundaries or clearly defined intensities within tumours [47]. Supervised segmentation methods include deep learning methods which implement convolutional neural networks [40]. These have proven to work more effectively than watershed methods [40] and have become the state-of-the-art segmentation meth-

ods in automatic segmentation [62]. An analysis of the state-of-the-art approaches is provided in further detail below.

2.3 State-of-the-Art

2.3.1 U-Net

An example of a successful deep neural network is the U-Net [46]. The U-Net is a Fully Convolutional Network (FCN) which was specifically developed for the segmentation of biomedical images within the field of image recognition [46]. In this filed a network is required to assign class labels per pixel while a large number (thousands [25]) of training images may not be available for network training [46]. The convolutional computation within a CNN reduces the spacial dimensions of the original image and becomes a problem as these dimensions are required for object segmentation [46]. In order to solve this problem a fully convolutional neural network is required to preserve input dimensions by using upsampling and downsampling techniques to replace pooling [29].

The U-Net was developed as an improvement on the work of [12] who used a neural network for the segmentation of electron microscopy images [25]. The approach by [12] used a sliding-window to make per-pixel class label predictions by choosing as input the local region (called a patch) found around each pixel [25]. This approach has limited network speed as the sliding window needed to scan every patch and created redundancy due to overlapping of patches [25]. The U-Net implementation was able to modify the segmentation process [46]. As illustrated in Figure 8, the U-Net implements a U-shaped architecture of a convolutional neural network created by the number of feature channels being used in the upsampling operations such that the expansive and contracting paths are symmetrical [46]. This means that the U-Net, a fully convolutional network architecture, contains two paths that form the U-shape [28]. The first path is the contraction path (which is called the encoder) and the second path is a symmetric expanding path (which is called the decoder) [28]. Consisting of stacked convolutional and max pooling layers the encoder is used to process the context of an image [28]. Each convolution process is activated by a ReLU activation function [46]. In this way all features are mapped to a single output vector [46]. The Decoder is used to create a high resolution segmentation map [28]. Intuitively in the encoder the model identifies what is present in the image, but it loses the information of where it is present, while the decoder recovers the where information (using precise localization) [28]. The U-Net will output whole, high resolution images with fully classified pixels [28]. Additionally this neural network has no fully connected layers and the segmentation map consists of only the pixels for which the input image has full context [46]. This technique is able to perform clear-cut image segmentation using few training samples [25]. This is advantageous in brain tumour segmentation as the acquisition of annotated medical images is difficult [25].

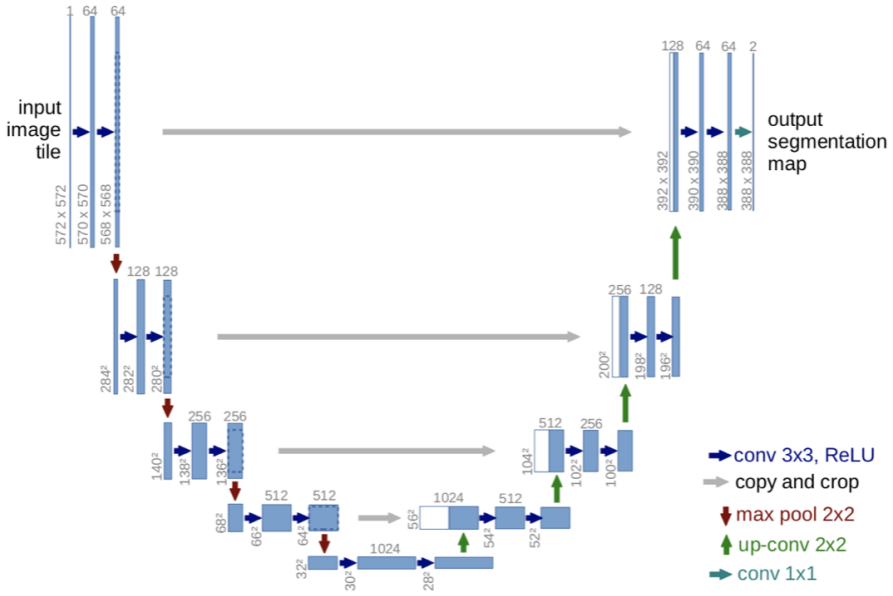


Figure 8: An illustration of the U-Net network architecture as given in [46].

2.3.2 V-Net

Similar to the U-Net neural network, the V-Net is a fully connected convolutional neural networks which was especially developed for 3D volumetric medical image segmentation [36]. In Figure 9, the V-Net neural network architecture is provided [36]. While other approaches process three dimensional input images by analysing two dimensional images slice-by-slice and then stacking these to form 3D volumes, the V-Net uses 3D convolutions to segment 3D volumetric data holistically [36]. In medical volumes such as the tumour images, the anatomy of the region being investigated occupies a region of the scan with the rest consisting of the background [36]. This may cause the neural network to reach a local minima where the network predictions are biased towards the image background [36]. In this situation the foreground region of interest will either be missing or be partially detected [36]. An advantage of this model is this model's dice loss layer, which does not need samples to be re-weighing when the number of background and foreground pixels is unbalanced and is useful for application in binary segmentation tasks [36].

2.3.3 InputCascadeCNN

Another example of a successful application of the Convolutional Neural Networks (CNN) to the open source BRATS dataset has been implemented by [17]. Their model, the InputCascadeCNN network implemented a fully automated method the results of which had been ranked second on the BRATS 2013 scoreboard giving Dice score values of 0.84, 0.71 and 0.57 for whole tumour, tumour core and enhancing tumour respectively [17]. This work included a novel implementation of a network architecture where two pathways are implemented allowing the model to learn about both the details of the brain and the larger context of the dataset [17]. According to [17] machine

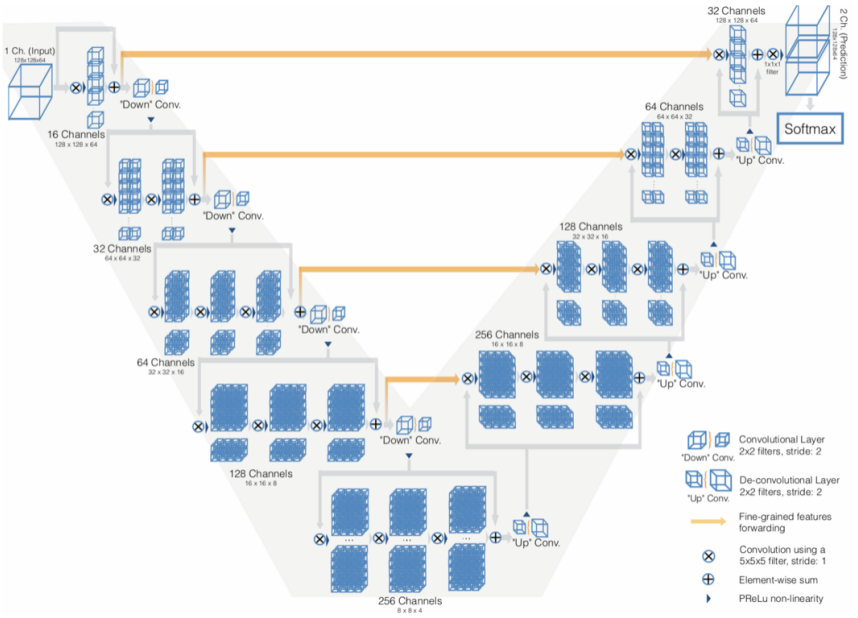


Figure 9: An illustration of the V-Net network architecture as given in [36].

learning methods are constrained by the fact that they perform pixel classification while not taking into account the local dependencies of labels [17]. An example of this can be illustrated by how segmentation labels are conditionally independent for a given input image [17]. This may be overcome by using pixel-wise probability estimates of the initial CNN as input to a second DNN creating a cascaded architecture [17]. The InputCascadeCNN neural network implemented by [17] used this principle to create a faster model which completed segmentations at a speed of between twenty five seconds and three minutes [17].

2.3.4 First Position in the Brain Tumour Segmentation Challenge 2013

A novel CNN-based approach by [42] achieved the first position in the Brain Tumour Segmentation Challenge 2013 (BRATS 2013) with Dice Similarity Coefficient values of 0.88, 0.83, 0.77 for whole tumour, tumour core, and enhancing tumour regions respectively [42]. The same model achieved second place in the BRATS 2015 Challenge, with Dice Similarity Coefficient values of 0.78, 0.65, and 0.75 for whole tumour, tumour core, and enhancing tumour regions respectively [42]. Data augmentation was used to address the variability present in brain tumour spatial localisation and structural composition [42]. Furthermore, in order to address the challenge of data variability, the patches of MRI input images were normalised to obtain zero mean and unit variance [42]. As the appearance and composition of LGG and HGG tumours differ, to address the variability present in intra-tumoural structures the designed CNN modified the normalisation of each tumour grade, LGG and HGG, was completed independently [42]. A deeper CNN architecture was used for HGG than LGG as the results for LGG were not improved by a deeper network implementation [42]. According to [42], deeper architectures have shown to increase over-fitting on a small LGG dataset ([42] reported

to have 10 and 54 patient scans in the training set of BraTS 2013 and BraTS 2015 respectively) due to the larger number of layers with weights [42]. Furthermore, LGG data requires the use of Dropout [52] regularisation to reduce over-fitting which is not the case for HGG data [42].

As stated in [42], 3D filters make use of the information present in three-dimentional (3D) images at the cost of an increase in computational load [42]. This approach used pooling to combines features of feature maps which are close together creating a compact image representation and decreases the computational load [42].

2.3.5 DeepMedic

As stated by [68], both [42] and [17] implemented CNNs that predict the central voxel labels of a patch. Due to the larger number of parameters and three-dimensional (3D) convolutions present in three-dimensional (3D) CNNs, there is an increase in both memory and computational load in these neural networks [26]. However [72] has found that an increase in parameters had made an improvement in neural network performance as illustrated by the difference between the segmentation performance results of the VNet implementation by [72] when compared to their implementations of neural networks called VNet+ and VNet++ [72]. The performance on lung nodule segmentation was measured as Intersection over Union (IoU), giving results of 71.17, 75.93 and 76.24 for each network, which had 22.6 million, 25.3 million and 26.2 million parameters respectively [72].

In order to reduce this computational burden [26] developed the DeepMedic model [68]. As described by [26], DeepMedic is a dual path three-dimensional (3D) CNN which has a depth of eleven layers.

2.3.6 Two-stage Cascaded U-Net

The current top performing CNN, which received first place in the Multimodal Brain Tumour Segmentation Challenge (BraTS) 2019 segmentation task challenge, is a novel two-stage cascaded U-Net implementation developed by [24]. This segmentation method was trained on the BraTS 2019 training dataset and achieved average Dice scores of 0.88796, 0.83697, 0.83267 for whole tumour, tumour core and enhancing tumour respectively, when evaluated on the BraTS 2019 testing set [24].

The first stage of this two-stage cascaded U-Net uses a version of a 3D U-Net implementation to train the network to produce a coarse prediction [24]. In the second stage a 3D U-Net of increased width is used with the addition of two decoders [24]. The role of the second stage is to refine the prediction map and produce a segmentation map of higher accuracy [24].

An advantage of this approach includes the prevention of model over-fitting by implementing data augmentation [24]. This was done by first implementing a per channel, random intensity shift of the channel standard deviation [24], which was accompanied by a random scaling intensity for the network input [24]. The input images where then cropped and a random flipping with a fifty percent probability was completed along the three dimensional axes [24].

The disadvantage of this two-stage cascaded U-Net is the amount of memory this network requires [24] as the input images needed to be cropped from a size of 240 x 240 x 155 voxels down to 128 x 128 x 128 voxels in order to overcome memory limitation [24]. Furthermore training of this method requires more than 12 Gigabytes of memory when completed on a Nvidia Titan V GPU [24]. Additionally [24] report

that performance variability was present between the performance of each individual model [24].

2.3.7 Deep Convolution Neural Network (DCNN) Techniques

The second place on the BraTS 2019 challenge was achieved by [67] who applied a combination of Deep Convolution Neural Network (DCNN) techniques to three-dimensional brain tumour segmentation [67]. This DCNN achieved the following Dice score results on the BraTS2019 online testing set: 0.883 for whole tumour, 0.861 for tumour core and 0.810 for enhanced tumour core [67]. The methods used to produce the overall method which was submitted for the challenge are a combination of four techniques which where consecutively applied to a U-Net [67].

An in depth analysis and explanations of each of these techniques and the combination used to produce this overall DCNN brain tumour segmentation method is further described in [67].

Each individual technique is designed and aimed at solving the challenges found in three-dimensional brain tumour segmentation [67]. These challenges may be divided into three main categories: data processing due to data imbalance, model architecture and model optimisation [67]. The respective techniques for these challenges, are described in detail in [67].

As discussed in Section 2.3.5, in brain tumour segmentation a significant number of background voxels is present in training data [67]. These are easily predicted by the classifier, therefore in order to reduce their effect a heuristic method is used in which patches of the input MRI image are randomly cropped and the image containing the most foreground voxels is used as input to the model [67]. Another approach to this challenge is using large image patches in order to include more contextual information [67]. This approach requires the use of small batch size that lead to increase in stochastic gradient variance which reduces optimisation, while larger batches reduce the patch size [67]. However by altering the number of padding and cropping layers present between the convolution layer, a model is able to learn the comprehensive information from both the largest patch and the texture from small patches with identical parameters [67].

Another solution presented by [67] addresses the improvement of model accuracy through model architecture design. While multiple prediction results may be combined to improve accuracy, [67] present a novel architecture which takes predictions at every scale of a U-Net and joins these into a single output [67]. Furthermore, while this process may become memory intensive as prediction tensors are up-sampled to the largest size, [67] combines the predictions recursively.

While becoming complex as a result of the implementation of all these techniques into a single model, this model combines their benefits of addressing data imbalance challenges, model architecture development and optimisation methods [67].

As reported by [67], the false predictions which this model makes on small anatomical regions is a limitation of this model.

2.3.8 DeepSCAN

The third place on the BraTS 2019 challenge was achieved by [32] who implemented a modified version of the DeepSCAN CNN presented in [31]. On the testing set this modified version of DeepSCAN achieved mean Dice scores of 0.89 for whole tumour,

0.83 for tumour core and 0.81 for enhancing tumour [32]. The model architecture is modified by replacing Batch normalisation with Instance normalisation [32].

The DeepSCAN network architecture mainly consists of two-dimensional convolutions with three-dimensional initial network convolutions provided for 3D context [32]. The network therefore contains an anisotropic receptive field which allow it to use the symmetries present in the human brain for predictions and the CNN is trained on sagital, coronal, and axial brain views [32]. The final network results are an ensemble of predictions of these three brain views [32]. A heteroscedastic loss function as described in [31] is used for DeepSCAN training [32].

The DeepSCAN CNN successfully performs standardisation and homogenisation on input data, with nonzero intensities being standardised across individual volumes, which facilitates optimal CNN learning [32]. The imbalance between the background and foreground present in the input data is resolved with the implementation of a focal loss function in the heteroscedastic networks [32]. The results of the DeepSCAN classifiers are filtered based on tumour biology [32]. This includes addressing the challenge found predominantly in LGG data when no tumour core is detected [32]. In this case it was accepted that the whole tumour consists of tumour core [32].

The third place on the BraTS 2018 challenge was a tie between the implementation of DeepSCAN by [31] and the submission by [68]. On the BraTS 2018 testing set DeepSCAN achieved mean Dice scores of 0.88593 for whole tumour, 0.79926 for tumour core and 0.73189 for enhancing tumour [31]. This novel approach uses a shallow symmetrical U-Shaped network consisting of densely connected blocks and dilated convolutions [10]. Additionally [31] present a novel loss function used to address both uncertainty and label noise [10].

This implementation of the DeepSCAN network is an improvement in memory efficiency of an earlier version of the DeepSCAN network presented in [30]. The DeepSCAN network of [30] is based on the Densenet architecture by [19], in which dense blocks and non-densely connected convolutional layers called transition blocks are combined to restrict parameter explosion by restricting the input size per dense block [31]. The DeepSCAN network developed by [30] has no pooling operations and the network may be viewed as a single dense block as there are no network transition layers [31]. Instead, the dilated convolutions of this network by [30] expand the classifier receptive field [31]. While this is an advantage over the U-net, the limitation of this approach is memory consumption as the final segmentation image is the result of all feature maps being present [31]. The limitation is addressed by using a pooling-free dense net inside a U-Net CNN architecture [31]. This solution improves memory efficiency as the dense net operates at a lower resolution inside the U-Net shaped network [31].

Evaluation of the CNN implementation by [68] on the BraTS 2018 testing set achieved Dice scores of 0.8842 for whole tumour, for 0.7960 tumour core and 0.7775 for enhancing tumour [68]. Multiple deep neural networks were combined by [68] in order to reduce model over-fitting in segmentation [10]. The different models used for this implementation include an implementation of modified versions of the three-dimensional FusionNet by [45], the OM-Net by [69], and a three-dimensional MC-baseline network by [71] which are described in detail in [68] and in [45], [69] and [71] respectively.

2.3.9 Second Place on the BraTS 2018 Challenge

The second place on the BraTS 2018 challenge was achieved by a modified U-Net implemented by [21], giving Dice scores of 87.81 for whole tumour, 80.62 for tumour core and 77.88 for enhancing tumour, when evaluated on the BraTS 2018 testing set [21]. This result was achieved by optimising the network training process [10] of a modified three-dimensional U-Net developed by [11]. This was done by using region-based training and the addition of further training data provided to [21] by their own organisation [34].

Region based training involves using Dice loss directly to complete optimisation of the three overlapping brain tumour regions, namely the whole tumour, tumour core and enhancing tumour [21]. This method was able to improve the model segmentation Dice scores of both the tumour core and enhancing tumour regions [21].

The implementation by [21] used a batch size of 2 [34], large patch input images of 128 x 128 x 128 [10] and replaced traditional ReLU activation functions with leaky ReLU functions [21].

This U-Net implementation addressed the challenge of false positives produces in brain tumour segmentation during enhancing tumour detection by using a combination of post-processing, cross-entropy loss and Dice loss [21]. Furthermore, the challenge of memory usage was addressed by modefying the model architecture [21]. A reduction in the number of filters before upsampling together with the addition of in-place operations produced a network that had a maximum resolution of 30 feature channels [21].

2.3.10 Asymmetrical U-Net

The first place on the BraTS 2018 challenge was achieved by [34] who utilised a U-Net which had the following Dice score results on the testing dataset: Dice scores of 0.8839 for whole tumour, 0.8154 for tumour core and 0.7664 for enhancing tumour [34]. The asymmetrical U-Net used a large encoder for the extraction of image features and a smaller decoder for the reconstruction of labels [67]. An MRI volume patch size of 160 x 192 x 128 voxels was used for network input [10] and a variational autoencoder (VAE) was added to the model for regularisation [67]. In order to accommodate the large MRI volume patch size, [34] used a batch size of one due to GPU memory limits on an NVIDIA Tesla V100 32 GB GPU [34]. As described by [34], the network was implemented in Tensorflow [1].

2.3.11 Deep Learning Radiomics Algorithm for Gliomas (DRAG) Model

A novel CNN-based approach by [2] which achieved Dice score values of 0.8474, 0.7687, 0.6677 for whole tumour, core tumour, and enhancing tumour regions respectively [2]. This model implemented a three-dimentional patch based U-Net model for the Brain tumour Segmentation (BraTS) 2018 challenge [2]. While focusing on overall survival (OS) of patients prediction, this model also addressed the challenge of data pixel imbalance between tumour and normal brain tissue images which is especially prominent in intra-tumour segmentation [2]. This was done by training the network on fixed size 3D image patches [2]. No data augmentation was performed for this approach [2]. Example images of sample segmentation results of this approach are given in Figure 10 [2].

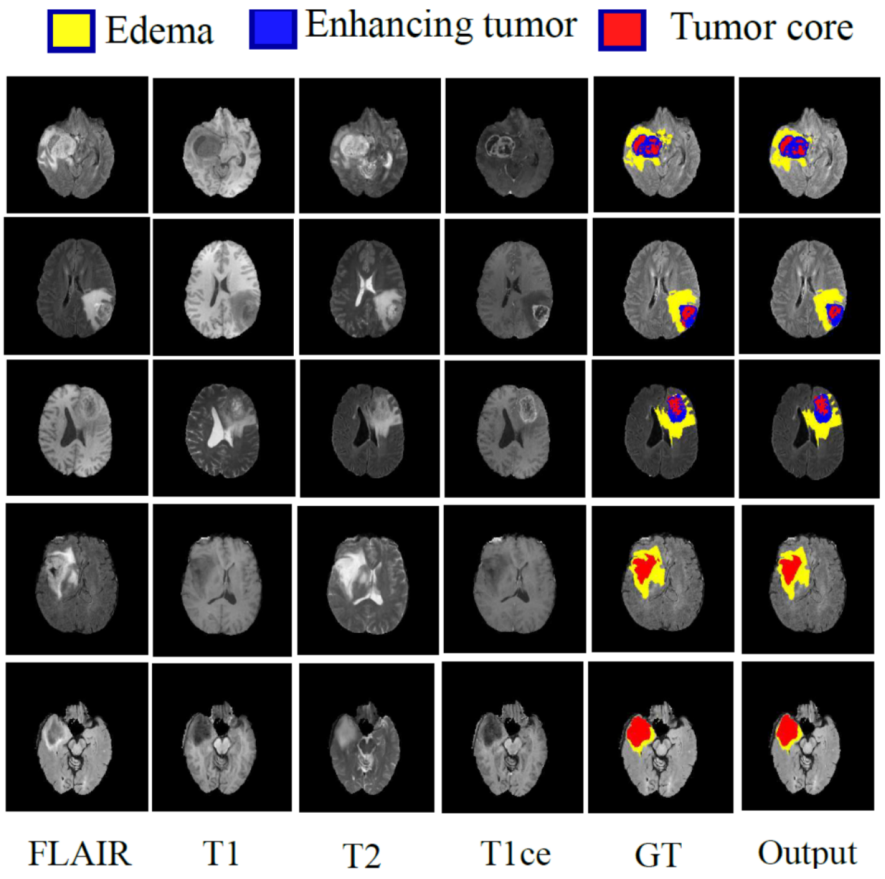


Figure 10: Segmentation results by [2] showing five rows of cases per the four BraTS 2018 Challenge data set MRI image modalities: FLAIR, T1, T2, T1ce, alongside the ground truth (GT) and result segmentation Output [2]. Segmentation labels are given as: yellow-edema, blue-enhancing tumour and red-tumour core [2].

2.3.12 WNet

Developed by [60], the WNet is one of three cascaded networks which has been trained for a multi-class segmentation of tumour sub-regions by [61]. This model was used as the base model for the Autofocus Net of [55] which had the following Dice score results on the BraTS 2018 Challenge test data set: 66.88, 55.16, 64.13 for whole tumour, core tumour and enhancing tumour respectively [55]. The two-dimensional (2D) WNet was adopted to develop the 3D WNet [55]. This approach was based on the development of the 3D U-Net, which was an adaptation of the two-dimensional (2D) U-Net [55]. The Autofocus Net performs multi-class segmentation of tumours on 3D MRI image and has shown that the model performance is improved as the number of parallel convolutions in the autofocus layers increase [55].

2.3.13 Summary

In summary, most of the state-of-the art models discussed above use an adaption of the U-Net architecture in their neural network implementation, which include the following: two-stage cascaded U-Net implementation developed by [24], the combination of Deep Convolution Neural Network (DCNN) techniques to three-dimensional brain tumour segmentation [67], the DeepSCAN models presented by [32] and [31], the modified U-Net implemented by [21], the Asymmetrical U-Net [34], and the novel CNN-based approach by [2]. The following challenge were addressed by the respective model implementations: model segmentation speed are addressed by [17]; input data and intra-tumoural variability are addressed by [42], [32], [67], [32], [2] and [67]; reduce computational load are addressed by [26]; model over-fitting are addressed by [24] and [68]; and memory limitations are addressed by [34] and [21]. While the following neural network models have the limitations of computational load in [42] and [17]; memory consumption in [24] and [31] and false predictions on small anatomical regions in the network by [67]. A summary of the top performing CNN implementations for the BraTS 2019, BraTS 2018 and the BraTS 2013 challenge test data sets are given in Table 1.

2.4 Research Questions

The review of the literature has identified several research questions.

The following questions have been identified:

- What is the segmentation performance, measured as Dice scores per each individual tumour region (the multi-class case), of a 2D U-Net CNN implementation on LGG and HGG data of the BraTS 2018 data set?
- How do the Dice scores per each individual tumour region (the multi-class case), of a 2D U-Net CNN [46] based architecture implementation compare to this 2D U-Net CNN implementation on LGG and HGG data of the BraTS 2018 data set?
- How do the Dice scores per each individual tumour region (the multi-class case), of a 2D U-Net CNN [46] based architecture implementation compare to a 3D U-Net CNN [46] based architecture implementation given the addition of volumetric data, in the form of two stacked slices being available in 3D U-Net CNN for training/learning?

Model	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
Two-stage Cascaded U-Net by [24]	0.88796	0.83697	0.83267
DCNN by [67]	0.883	0.861	0.810
DeepSCAN by [32]	0.89	0.83	0.81
Asymmetrical U-Net by [34]	0.8839	0.8154	0.7664
U-Net by [21]	0.8781	0.8062	0.7788
DeepSCAN by [31]	0.88593	0.79926	0.73189
Multiple Deep Networks by [68]	0.8842	0.7960	0.7775
CNN by [42]	0.88	0.83	0.77
CNN by [17]	0.84	0.71	0.58

Table 1: A summary of the top performing CNN implementations for the BraTS 2019, BraTS 2018 and the BraTS 2013 Challenge test data sets.

3 Implementation Methodology

3.1 Overview

In order to answer the research questions, an initial implementation of a 2D U-Net was developed as part of the initial pipeline which was completed as a group. This was followed by the development and implementation of my individual work. As part of my individual work I modified the code of the initial pipeline to implement a 2D U-Net that is based on the U-Net model by [46]. The [46] network architecture is illustrated in Figure 8. This code and 2D U-Net [46] based model was then modified and developed into a 3D U-Net neural network. The details of the initial pipeline implementation together with each individual's work is given in Figure 11. In this way while the three implemented networks have a different architecture, all three follow the same pre-processing steps with the exception of the 3D U-Net [46] based model which used a different image cropping size as described in Section 3.4.

3.2 Resources

Similar to the approach used in the methodology by [34], the code development was done in Python using the Tensorflow [57] library for the neural network implementations. The Tensorflow version 2.2 was required due to the specific use of one-hot encoding. This required a compatible version [57] of CUDA 10.1 [41]. A python virtual environment needs to be set up in which to run the code. The virtual environment was set up so that all the project specific libraries and inter dependent versions software are kept together in one place so as not to interfere with other programs on the computer. Other software libraries include seaborn [63], matplotlib [20], sklearn [50], numpy [39], SimpleITK [51], tqdm [58], and os.py libraries.

3.2.1 Lab GPU

A resource required for this project was the Computer Science Lab GPU as I needed to run machine learning algorithms which are based on the work of [46], on a GPU. The Lab GPU client which I was assigned had the following properties, which were provided as output during program run-time.

Found device 0 with properties:
pciBusID: 0000:01:00.0 name: GeForce GTX 1060 6GB computeCapability: 6.1
coreClock: 1.7085GHz coreCount: 10 deviceMemorySize: 5.94GiB
deviceMemoryBandwidth: 178.99GiB/s

This means that this machine is a GeForce GTX 1060 6GB model which has 6 Gigabytes of video ram (VRAM) required for running the deep neural network algorithms.

The CPU of this computer has four cores each having the following details:
GenuineIntel Intel(R) Core(TM) i5-6500 CPU 3.20GHz model
cpu MHz : 800.000
The RAM is given as having size: 32GiB

3.2.2 Lab GPU Limitations

Two GPU related error were encountered during model training and data processing. These will be referred to as the Core Dumped error and an Out of Memory error. The Core Dumped error occurred when an attempt was made to train on more than five epoch of any of the three implemented algorithms on the high grade glioma (HGG) [55] data from the BraTS 2018 data set. This error is characterised by the following example stack trace:

Epoch 6/10
794/794 [==============================] - ETA: 0s
Segmentation fault (core dumped)

According to [53] this error indicates that some Linux program has crashed, which would be the Python interpreter in this case. This may be a Tensorflow library specific bug, but the error was not investigated further as a work around was found and implemented (as described in Section 4). The Out of Memory (OOM) error would be encountered during, for example, the evaluation of the [46] based 3D U-Net model. This error is characterised by the following example stack trace:

tensorflow/core/framework/op_kernel.cc:1753] OP_REQUIRES failed at
concat_op.cc:161 :
Resource exhausted:
OOM when allocating tensor with shape[2538,2,192,192,4] and type float on
/job:localhost/replica:0/task:0/device:GPU:0 by allocator GPU_0_bfc

According to [14] and [54] this is an error thrown when the GPU is not able to allocate the required amount of video memory in order to complete a task.

3.3 Group 2D U-Net Implementation

Similar to the approach by [55] described in Section 2.3.12. The initial pipeline implementation is an adaptation of the Git repository of [48]. The BraTS 2018 LGG and HGG data was processed separately. The program reads in the files from each folder MRI separating the modalities and ground truth data. These are saved as individual files. The LGG and HGG is split into training/validation/test data sets, each having

50 : 17 : 8 and 141 : 48 : 21 patient cases respectively. The test set is pre-processed separately at evaluation per the steps described in Figure 11, while training and validation sets are then normalised by subtracting the mean and dividing by the standard deviation for each MRI image. These images are then transposed to comply with Tensorflow required tensor shape. The MRI images are 3D volumes consisting of 155 slices with width and depth of 240 [48]. As the tumour region is not present in all 155 slices, the images are cropped to only use the image mid portion, taking the portion of 90 slices, from slice 30 to slice 120 [48]. This reshapes the images from a size of 155 x 240 x 240 to 90 x 192 x 192. One-hot encoding, is used for ground truth image target class encoding [48] assigning a value of zero to background and a label of 3 to the enhancing tumour target class. The group deep learning model, a 2D U-Net implementation, uses a ReLu activation function, the Adam optimiser for model optimisation, BatchNormalization for model optimisation [8], a Dropout of 20 percent to prevent model overfitting [27] and MaxPooling2D for downsampling [27]. The model hyper-parameters include stride size of 2 in Conv2DTranspose for inverse convolutions [9]. Pooling is set to SAME in order to keep the padding of the input and output of the same length in Conv2D [27]. The model specific Dice coefficient and Dice coefficient loss functions used for model compilation were taken directly from [48]. Additional code was taken from [65] for the evaluation of tumour target class label classification and the confusion matrix implementation code was referenced from [56] and [37]. As stated in Section 1.3, the Dice score is identical to the F score [17], therefore the f1_score function from [49] was used in model performance evaluation per tumour target class.

3.4 Individual Work

I investigated the U-Net segmentation performance further by modifying the group 2D U-Net implementation architecture to that of the original 2D U-Net by [46] which also used BatchNormalization but no Dropout. I then further modifying this architecture to implement a [46] based 3D U-Net. I was required to keep the pre-processing unaltered within all three models in order to accommodate their comparison. The aim for the 3D model was to investigate the effect on model performance due to the additional volumetric information. Due to the Out of Memory error, this needed to be constrained to two slices. This model also used BatchNormalization, but no Dropout. This implementation was done by replacing the 2D convolutional operations with 3D convolutional operations (Conv2D, MaxPooling2D, Conv2DTranspose replaced with Conv3D, MaxPooling3D, Conv3DTranspose), adjusting the model architecture in order to accommodate the 2 x 192 x 192 size image, together with the addition of six UpSampling3D operations, four of size (2, 1, 1), one of size (4, 1, 1) and (8, 1, 1) parameters; used for adjusting tensor size in order to produce correct size tensors of corresponding size (this was deduced from [9] which discussed UpSampling2D) for concatenation within the model. This led to the need to use an additional final MaxPooling3D downsampling operation of pool size (8, 1, 1) to produce the output image. In this way I developed the 3D U-Net implementation based on the U-Net architecture of [46] which uses most of the same pre-processing steps as the other two U-Net models.

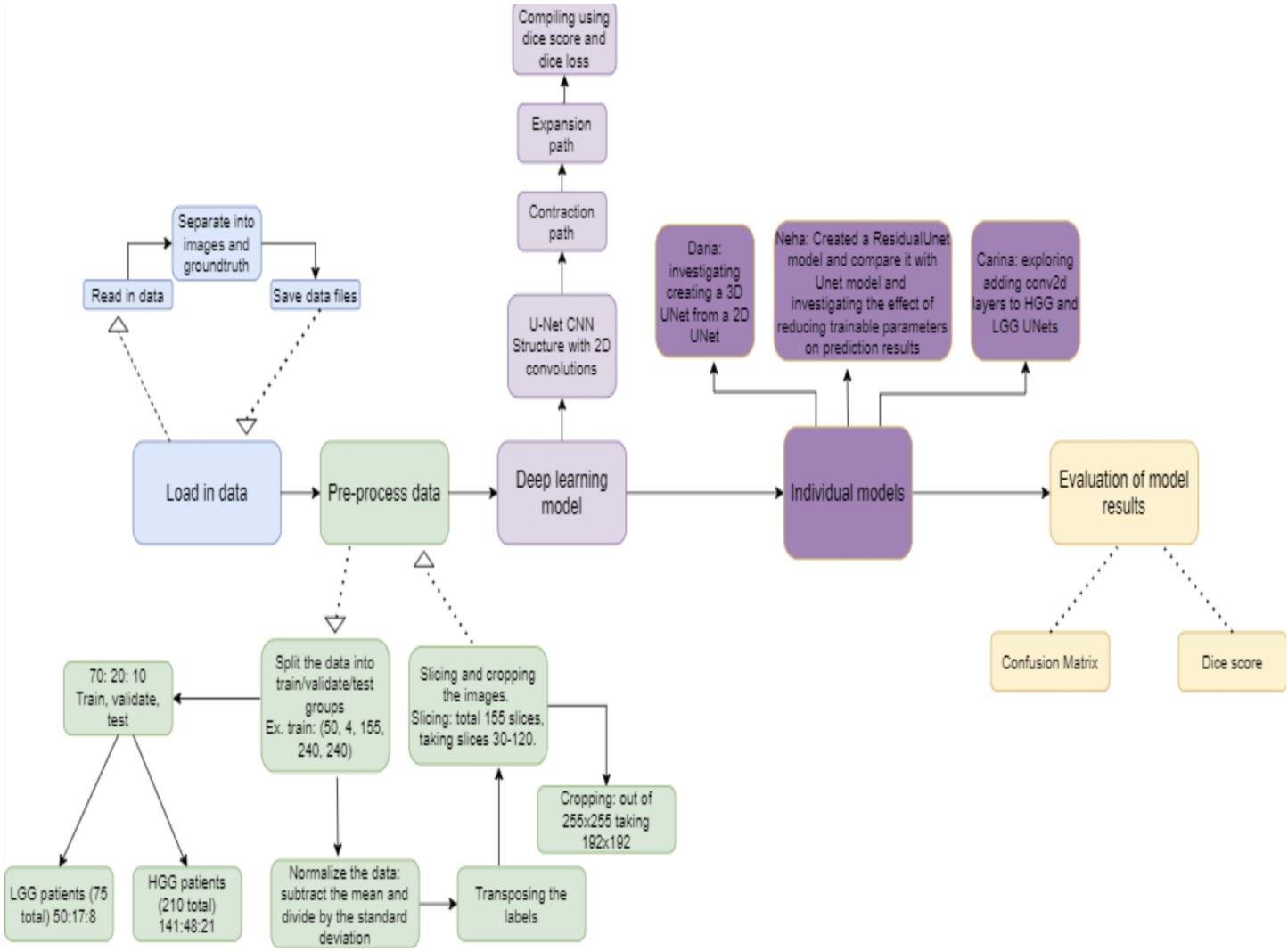


Figure 11: Group segmentation pipeline implementation and individual investigations of each team member. Flow chart created in collaboration with Carina Norregaard (matriculation number: 150004238) and Neha Ghatia (matriculation number: 190024130).

4 Results

As stated in Section 1.3, the BRATS data set consists of low grade glioma (LGG) and high grade glioma (HGG) data [55]. The results were obtained by training each model on each of these respective Glioma types separately. Similarly to the method presented in [2], each model was trained for up to 50 epochs with the exception of the [46] based 3D U-Net implementation on HGG data. This was due to the fact that an average of approximately 17 hours per 5 epochs were required to train the model on the lab machine, as seen from Table 4. All predictions on all models were completed with the assistance of the batch implementation taking a batch size of one as an Out of Memory error (as discussed in Section 3.2.2) occurs when evaluating the performance of the [46] based 3D U-Net implementation. In order to save time due to time constraints and for consistency due to the Core Dumped Error (described in Section 3.2.2), the training was done such that each model was trained for 5 epochs, then saved, then the evaluation of the model was done on the validation set, and then this same model would be trained for another five epochs, saved and evaluated. These steps were completed continuously until a full 50 epochs of training was reached and all models were saved. This number consists of 10 models for each of the separately designed and implemented 2D U-Net architectures. Thus creating a total of separate sets of 40 2D U-Net models trained on LGG and HGG data respectively, 10 trained 3D U-Net models for LGG and three trained models for the 3D U-Net implementation trained on HGG data. The multi-class case was examined in which separate Dice scores are calculated per tumour region as was done in [55]. The results below gives the full details of performance results of each model on the LGG and HGG validation data respectively.

4.1 LGG Data

The results on the LGG validation data of the BraTS 2018 data set of the 2D Group U-Net together with the 2D [46] based U-Net and the [46] based 3D U-Net implementations are given in Table 2. Results show 3D [46] based U-Net having longest training time.

Model run on LGG data set	Dice Scores				Time taken to train	Training batch size	Time taken to predict	Numb-er of epochs
	Whole Tumour	Tumour Core	Enhancing Tumour	Back-ground				
Group 2D U-Net	0.8540	0.5675	0.0	0.9956	1297 seconds	16	44 seconds	5
2D U-Net	0.8446	0.7107	0.0	0.9948	1175 seconds	16	45 seconds	5
3D U-Net	0.8142	0.6475	0.1086	0.9939	22017 seconds	1	399 seconds	5
Group 2D U-Net	0.8607	0.6371	0.0	0.9958	1303 seconds	16	44 seconds	10
2D U-Net	0.8700	0.6564	0.0	0.9959	1233 seconds	16	45 seconds	10
3D U-Net	0.7833	0.6792	0.2927	0.9924	22033 seconds	1	411 seconds	10
Group 2D U-Net	0.8629	0.6253	0.0	0.9956	1234 seconds	16	1234 seconds	15
2D U-Net	0.8726	0.6944	0.0	0.9961	1234 seconds	16	45 seconds	15
3D U-Net	0.7269	0.6322	0.1377	0.9894	21966 seconds	1	411 seconds	15
Group 2D U-Net	0.8677	0.6763	0.0	0.9959	1301 seconds	16	44 seconds	20
2D U-Net	0.8768	0.6933	0.0	0.9961	1201 seconds	16	44 seconds	20

Model run on LGG data set	Dice Scores				Time taken to train	Training batch size	Time taken to predict	Numb-er of epochs
	Whole Tumour	Tumour Core	Enhancing Tumour	Back-ground				
3D U-Net	0.7856	0.6730	0.2478	0.9925	22032 seconds	1	411 seconds	20
Group 2D U-Net	0.8628	0.6359	0.0	0.9957	1302 seconds	16	44 seconds	25
2D U-Net	0.8721	0.6626	0.0	0.9960	1204 seconds	16	44 seconds	25
3D U-Net	0.8049	0.6965	0.1861	0.9933	21996 seconds	1	411 seconds	25
Group 2D U-Net	0.8667	0.6605	0.0	0.9959	1307 seconds	16	44 seconds	30
2D U-Net	0.8747	0.6751	0.0	0.9960	1203 seconds	16	44 seconds	30
3D U-Net	0.8367	0.6744	0.3315	0.9950	21964 seconds	1	411 seconds	30
Group 2D U-Net	0.8615	0.6800	0.0	0.9956	1301 seconds	16	44 seconds	35
2D U-Net	0.8716	0.7013	0.0	0.9961	1202 seconds	16	44 seconds	35
3D U-Net	0.8050	0.6892	0.2344	0.9934	21912 seconds	1	411 seconds	35
Group 2D U-Net	0.8652	0.6533	0.0	0.9959	1302 seconds	16	44 seconds	40

Model run on LGG data set	Dice Scores				Time taken to train	Training batch size	Time taken to predict	Number of epochs
	Whole Tumour	Tumour Core	Enhancing Tumour	Background				
2D U-Net	0.8729	0.6655	0.0	0.9961	1205 seconds	16	44 seconds	40
3D U-Net	0.8021	0.7076	0.1990	0.9930	21915 seconds	1	410 seconds	40
Group 2D U-Net	0.8649	0.6655	0.0	0.9960	1294 seconds	16	44 seconds	45
2D U-Net	0.8718	0.6557	0.0	0.9961	1205 seconds	16	44 seconds	45
3D U-Net	0.8184	0.6687	0.2248	0.9940	21870 seconds	1	410 seconds	45
Group 2D U-Net	0.8695	0.6377	0.0	0.9960	1302 seconds	16	43 seconds	50
2D U-Net	0.8695	0.6689	0.0	0.9961	1203 seconds	16	44 seconds	50
3D U-Net	0.8232	0.7086	0.2779	0.9942	21900 seconds	1	410 seconds	50

Table 2: The performance results of all implemented models (2D Group U-Net, 2D U-Net [46] based and 3D U-Net [46] based implementations) up to 50 epochs on LGG validation data.

4.2 HGG Data

The results of the of the 2D Group U-Net implementation and the [46] based 2D U-Net implementation, evaluated on the HGG validation data of the BraTS 2018 data set, are given in Table 3. Similarly, the results on the HGG validation data of the BraTS 2018 data set of the [46] based 3D U-Net implementation are given in Table 4.

Model run on HGG data set	Dice Scores				Time taken to train the 5 epochs	Training batch size	Time taken to predict	Number of epochs
	Whole Tumour	Tumour Core	Enhancing Tumour	Back-ground				
Group 2D U-Net	0.8629	0.8044	0.7846	0.9960	4352 seconds	8	124 seconds	5
2D U-Net	0.8804	0.7934	0.7799	0.9963	4190 seconds	8	127 seconds	5
Group 2D U-Net	0.8828	0.8285	0.8003	0.9964	4421 seconds	8	120 seconds	10
2D U-Net	0.8688	0.7787	0.7799	0.9962	4235 seconds	8	122 seconds	10
Group 2D U-Net	0.8806	0.8180	0.7983	0.9964	4363 seconds	8	132 seconds	15
2D U-Net	0.8850	0.8259	0.8021	0.9965	4225 seconds	8	122 seconds	15
Group 2D U-Net	0.8818	0.8157	0.8028	0.9964	4313 seconds	8	124 seconds	20
2D U-Net	0.8858	0.8247	0.7998	0.9965	4239 seconds	8	122 seconds	20
Group 2D U-Net	0.8833	0.8199	0.7875	0.9965	4306 seconds	8	125 seconds	25
2D U-Net	0.8772	0.8185	0.7991	0.9964	4222 seconds	8	122 seconds	25
Group 2D U-Net	0.8840	0.8192	0.7971	0.9965	4324 seconds	8	125 seconds	30

Model run on HGG data set	Dice Scores				Time taken to train the 5 epochs	Training batch size	Time taken to predict	Number of epochs
	Whole Tumour	Tumour Core	Enhancing Tumour	Back-ground				
2D U-Net	0.8832	0.8060	0.7903	0.9965	4213 seconds	8	122 seconds	30
Group 2D U-Net	0.8800	0.8211	0.8001	0.9964	4318 seconds	8	124 seconds	35
2D U-Net	0.8871	0.8215	0.8014	0.9965	4212 seconds	8	122 seconds	35
Group 2D U-Net	0.8830	0.8189	0.8008	0.9964	4300 seconds	8	124 seconds	40
2D U-Net	0.8845	0.8342	0.8078	0.9965	4207 seconds	8	122 seconds	40
Group 2D U-Net	0.8818	0.8271	0.8009	0.9964	4314 seconds	8	125 seconds	45
2D U-Net	0.8837	0.8236	0.8006	0.9965	4261 seconds	8	122 seconds	45
Group 2D U-Net	0.8854	0.8246	0.8035	0.9965	4313 seconds	8	124 seconds	50
2D U-Net	0.8847	0.8167	0.7999	0.9965	4226 seconds	8	122 seconds	50

Table 3: The performance results of the 2D Group U-Net and the [46] based 2D U-Net implementation models up to 50 epochs on HGG validation data.

Data used for evaluation	Dice Scores				Number of epochs	Time taken to train
	Whole Tumour	Tumour Core	Enhancing Tumour	Back-ground		
First 24 patients (first half of validation data)	0.7827	0.6882	0.5874	0.9937	5	62180 seconds
Last 24 patients (second half of validation data)	0.9002	0.8621	0.8271	0.9964	5	62180 seconds
Average	0.8414	0.7751	0.7072	0.9950	5	
First 24 patients (first half of validation data)	0.8144	0.7411	0.6943	0.9951	10	62177 seconds
Last 24 patients (second half of validation data)	0.8980	0.8682	0.8336	0.9963	10	62177 seconds
Average	0.8562	0.8046	0.7639	0.9957	10	
First 24 patients (first half of validation data)	0.7538	0.6316	0.5449	0.9926	15	62183 seconds
Last 24 patients (second half of validation data)	0.9037	0.8443	0.8027	0.9965	15	62183 seconds
Average	0.8287	0.7379	0.6738	0.9945	15	

Table 4: The performance results of the [46] based 3D U-Net implementation model up to 15 epochs on HGG validation data.

5 Evaluation

The group implemented 2D U-Net is not overfitting the training data, with similar results (Dice score within 0.08) on test and validation data for LGG and HGG data (see Table 5, 6 and 7), except for the difference of 0.2419 on the whole tumour for HGG. Similarly for the 3D and 2D U-Net [46] based model (see Table 8, 9, 10, 11, 12, 13 and 14), having difference of 0.186 and 0.2166 respectively on the whole tumour for HGG. Table 16 and Figure 16 shows the performance results of all three implemented models evaluated on the BraTS 2018 test set as described in Section 1.3. The 3D U-Net [46] based model seems to have produced the best results. An illustration is provided in Figure 16. Table 15 shows that results of the three implemented models are comparable to state-of-the-art results. The confusion matrix results in Figure 12, show that the 2D U-Net models are not able to detect the enhancing tumour for LGG data, yet the diagonal shows that most classes are correctly classified. This is accompanied by Figure 13, where the sample segmentation images illustrate the segmentation on one patient to visualise how segmentation is performed across the data set which is represented as Dice scores. Similarly for the HGG data the confusion matrix in Figure 14 is accompanied by Figure 15. Figure 14 shows that all three models are able to classify all of the target tumour classes correctly.

Additionally from the program output during run-time, it was shown that the 3D U-Net [46] based model had 105,271,812 trainable hyper-parameters, while the 2D U-Net [46] based model had 36,954,116 trainable hyper-parameters and the Group 2D U-Net implementation had 31,348,868 trainable hyper-parameters. Based on the results suggested by [72] in Section 2.3.5, where models with a larger number of trainable hyper-parameters had better performance; I expected the Dice scores of the [46] based 3D U-Net model to perform better than the other two models on all three of the target classes on HGG data. This was not the case as shown in Table 16. However the [46] based 3D U-Net model used for the evaluation had only been trained on 10 epochs and further training of this neural network on more epochs may give better results.

5.1 Group 2D U-Net Implementation

5.1.1 LGG data set

Dataset	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
Training	0.9517	0.9093	0.0
Validation	0.8677	0.6763	0.0
Test	0.8460	0.6757	0.0

Table 5: Group 2D U-Net (this work) implementation performance on the BraTS 2018 test data set after 20 epochs of training on LGG data.

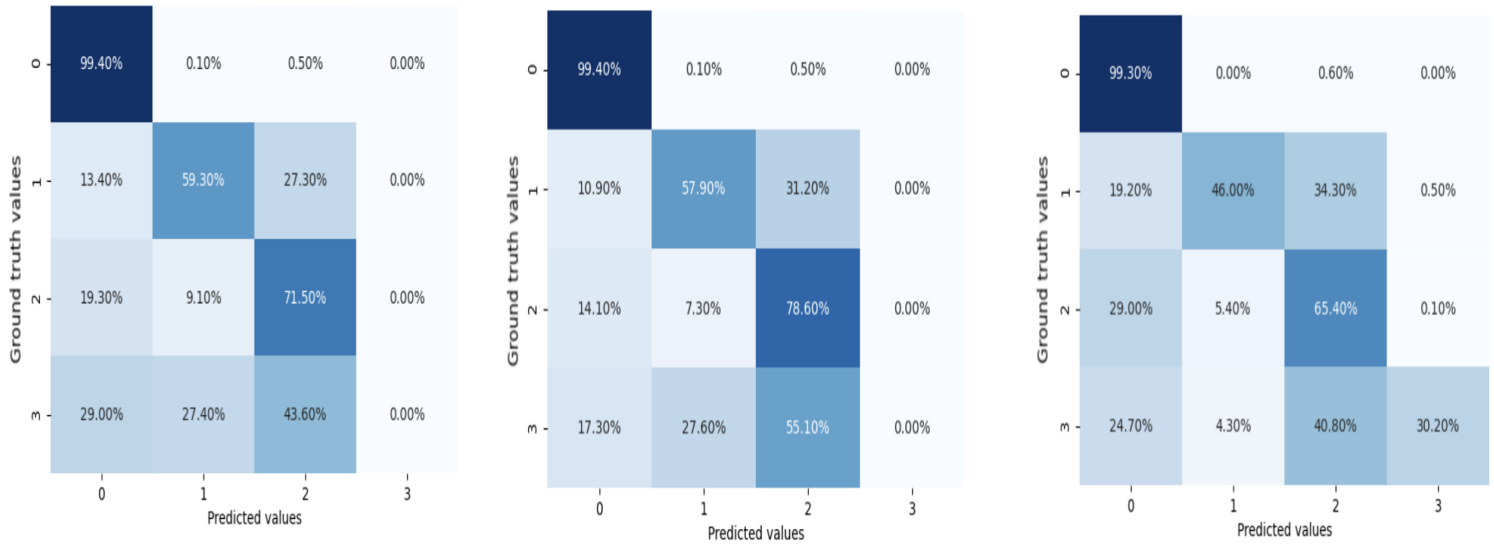


Figure 12: Confusion matrices of the final results of this work for LGG data. The performance results of the best performing models produced as evaluated on LGG data of the BraTS 2018 test data set. From left to right: the Group 2D U-Net implementation, 2D U-Net implementation based on [46] and 3D U-Net implementation based on [46]

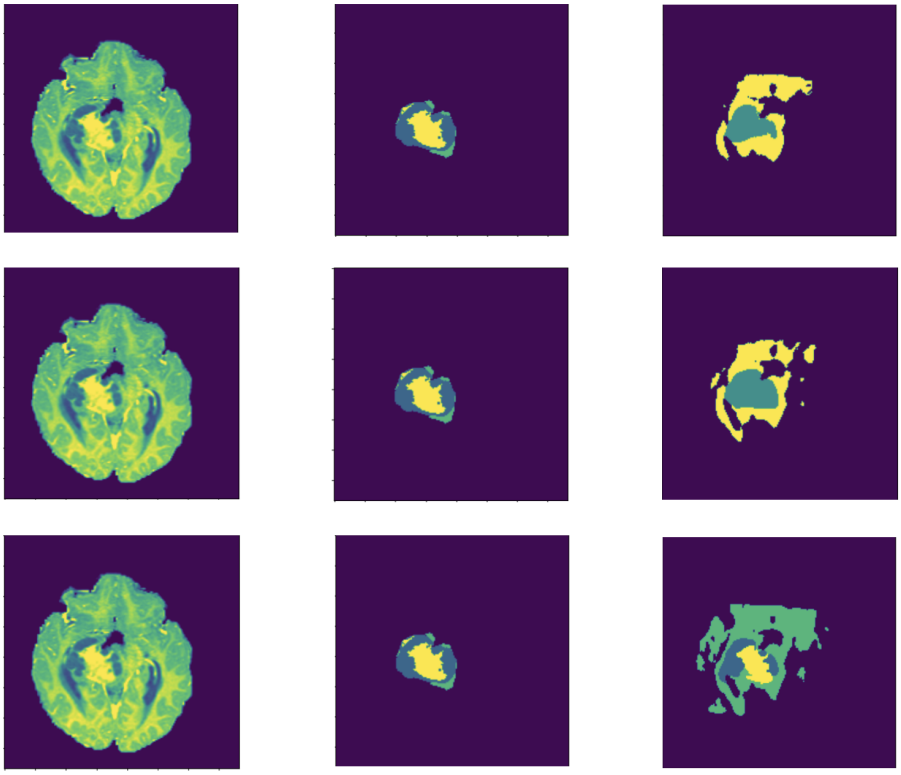


Figure 13: Sample segmentation images of the final results of this work for LGG data. The images are displayed per model and shown per column from top to bottom: the Group 2D U-Net implementation, 2D U-Net implementation based on [46] and 3D U-Net implementation based on [46]. The sample images from left to right: Brain image, Ground Truth and segmentation Output. Segmentation labels are displayed per colour, with green: edema, blue: enhancing tumour and yellow: tumour core.

5.1.2 HGG data set

Dataset	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
Training Average Scores	0.9745	0.9729	0.9346
Validation	0.8854	0.8246	0.8035
Test	0.6435	0.8320	0.8259

Table 6: Group 2D U-Net (this work) implementation performance on the BraTS 2018 test dataset after 50 epochs of training on HGG data.

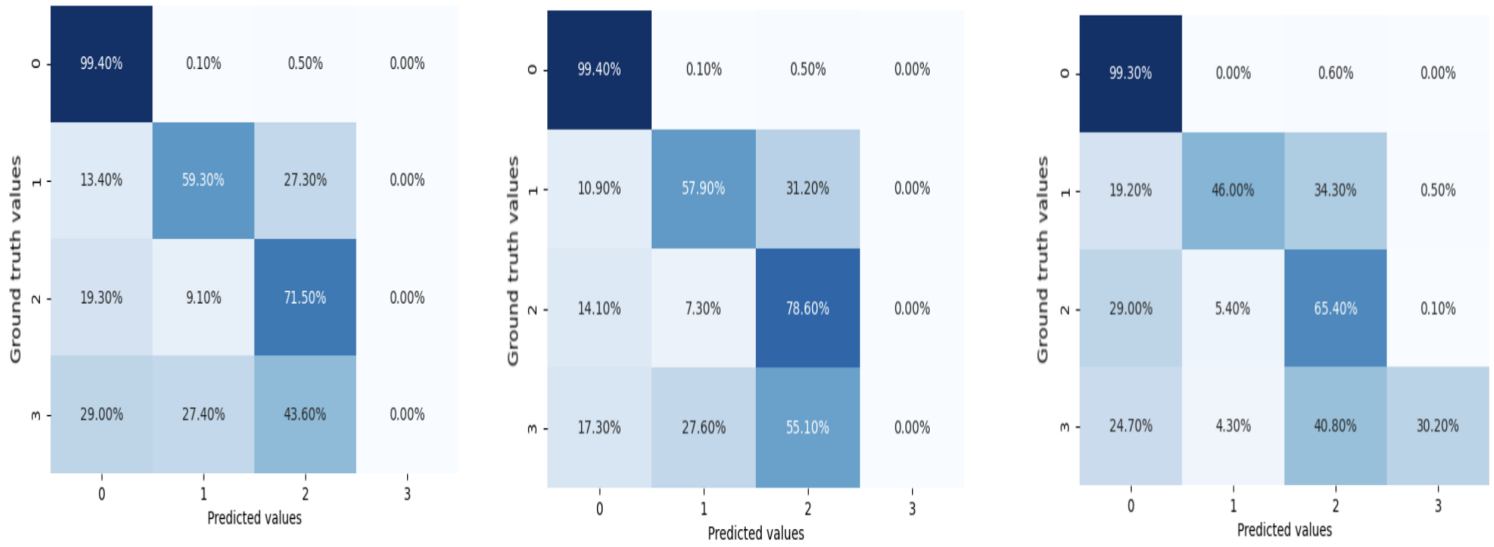


Figure 14: Confusion matrices of the final results of this work for HGG data. The performance results of the best performing models produced as evaluated on HGG data of the BraTS 2018 test data set. From left to right: the Group 2D U-Net implementation, 2D U-Net implementation based on [46], 3D U-Net implementation based on [46].

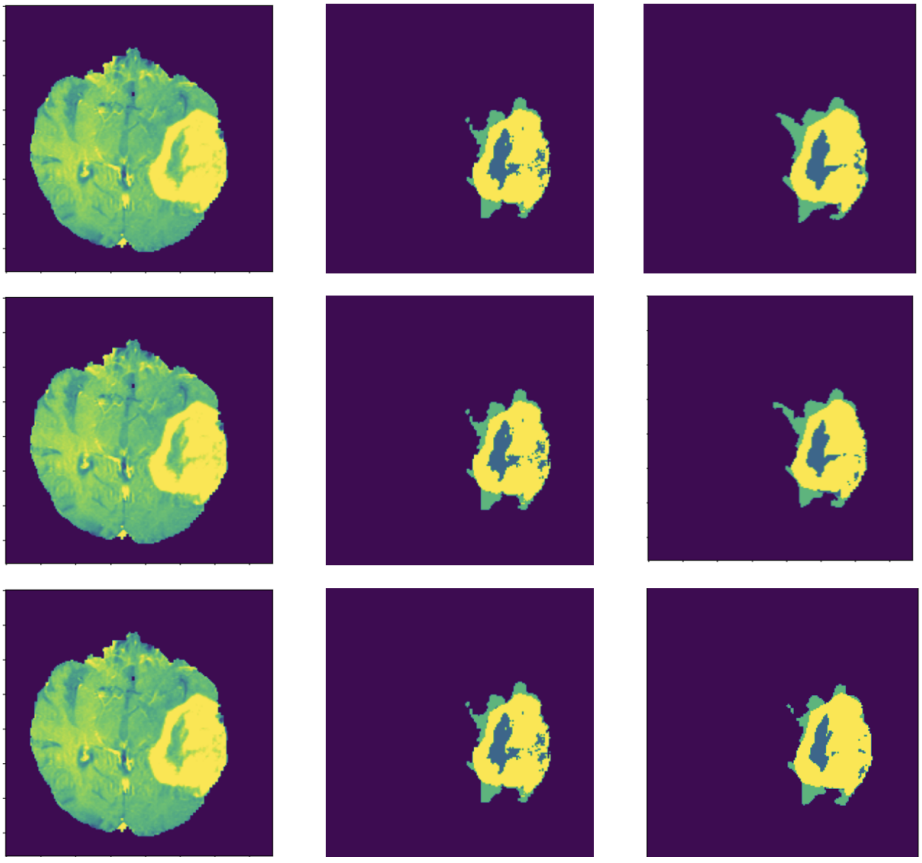


Figure 15: Sample segmentation images of the final results of this work for HGG data. The images are displayed per model and shown per column from top to bottom: the Group 2D U-Net implementation, 2D U-Net implementation based on [46] and 3D U-Net implementation based on [46]. The sample images from left to right: Brain image, Ground Truth and segmentation Output. Segmentation labels are displayed per colour, with green: edema, blue: enhancing tumour and yellow: tumour core.

Training Dataset Subsets	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
First 47 patients (first third of training data)	0.9733	0.9730	0.9313
Second 47 patients (second third of training data)	0.9761	0.9727	0.9362
Last 47 patients (last third of training data)	0.9743	0.9732	0.9364
Average	0.9745	0.9729	0.9346

Table 7: Group 2D U-Net (this work) implementation performance on the training data set after 50 epochs of training on HGG data.

5.2 Individual 2D U-Net Implementation

5.2.1 LGG data set

Dataset	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
Training	0.9560	0.9243	0.0
Validation	0.8768	0.6933	0.0
Test	0.8692	0.6714	0.0

Table 8: 2D U-Net model implementation (this work) based on [46] with performance dice scores per data set after 20 epochs of training on LGG data.

5.2.2 HGG data set

Dataset	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
Training Average Scores	0.9654	0.9639	0.9179
Validation	0.8858	0.8247	0.7998
Test	0.6998	0.8110	0.8039

Table 9: 2D U-Net based on [46] implementation performance dice scores per dataset after 20 epochs of training on HGG data.

Training Dataset Subsets	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
First 47 patients (first third of training data)	0.9638	0.9643	0.9140
Second 47 patients (second third of training data)	0.9674	0.9638	0.9209
Last 47 patients (last third of training data)	0.9652	0.9636	0.9189
Average	0.9654	0.9639	0.9179

Table 10: 2D U-Net based on [46] implementation performance dice scores on training data set after 20 epochs of training on HGG data.

5.3 Individual 3D U-Net Implementation

5.3.1 LGG data set

Dataset	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
Training Average Scores	0.9072	0.8974	0.7234
Validation	0.8367	0.6744	0.3315
Test	0.7960	0.6001	0.4368

Table 11: 3D U-Net based on [46] implementation performance dice scores per data set after 30 epochs of training on LGG data.

Training Dataset Subsets	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
First 25 patients (first half of training data)	0.8863	0.9084	0.7037
Last 25 patients (second half of training data)	0.9282	0.8864	0.7431
Average	0.9072	0.8974	0.7234

Table 12: 3D U-Net based on [46] implementation performance dice scores per training data set; subsets of 25 patients and the average after 30 epochs of training on LGG data.

5.3.2 HGG data set

Dataset	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
Training Average Scores	0.9069	0.8647	0.7874
Validation Average Scores	0.8562	0.8046	0.7639
Test	0.6396	0.8581	0.8181

Table 13: Performance dice scores of the 3D U-Net implementation (this work) based on [46] after 10 epochs of training on HGG data.

Training Dataset Subsets of patients	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
Part 1: 1269 slices	0.8576	0.7671	0.6966
Part 2: 1269 slices	0.9188	0.8851	0.8207
Part 3: 1269 slices	0.9090	0.8690	0.7816
Part 4: 1269 slices	0.9262	0.9020	0.8302
Part 5: 1269 slices	0.9233	0.9006	0.8083
Average	0.9069	0.8647	0.7874

Table 14: 3D U-Net based on [46] implementation performance dice scores per training data set; subsets of 2538 slices and their average after 10 epochs of training on HGG data.

5.4 Comparison to State-of-the-art

Model	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
Two-stage Cascaded U-Net by [24]	0.88796	0.83697	0.83267
DCNN by [67]	0.883	0.861	0.810
DeepSCAN by [32]	0.89	0.83	0.81
Asymmetrical U-Net by [34]	0.8839	0.8154	0.7664
U-Net by [21]	0.8781	0.8062	0.7788
DeepSCAN by [31]	0.88593	0.79926	0.73189
Multiple Deep Networks by [68]	0.8842	0.7960	0.7775
CNN by [42]	0.88	0.83	0.77
CNN by [17]	0.84	0.71	0.58
Group 2D U-Net (this work) for LGG data	0.8460	0.6757	0.0
2D U-Net (this work) for LGG data	0.8692	0.6714	0.0
3D U-Net (this work) for LGG data	0.7960	0.6001	0.4368
Group 2D U-Net (this work) for HGG data	0.6435	0.8320	0.8259
2D U-Net (this work) for HGG data	0.6998	0.8110	0.8039
3D U-Net (this work) for HGG data	0.6396	0.8581	0.8181

Table 15: Performance results comparison to state-of-the-art of the best performing models produced from the Group 2D U-Net; 2D U-Net implementation based on [46] and 3D U-Net implementations based on [46] per LGG and HGG data of the BraTS 2018 test data set respectively.

5.5 Summary of Final Results

Figure 16: Summary of the final results of this work. Graph showing the performance results of the best performing models produced from the Group 2D U-Net implementation alongside the 2D U-Net implementation based on [46] and the 3D U-Net implementations based on [46] per LGG and HGG data of the BraTS 2018 test data set respectively.

Model implementations evaluated on the BraTS 2018 testing dataset	Dice scores		
	Whole tumour	Tumour core	Enhancing tumour
Group 2D U-Net evaluated on LGG data	0.8460	0.6757	0.0
2D U-Net evaluated on LGG data	0.8675	0.6808	0.0
3D U-Net evaluated on LGG data	0.7960	0.6001	0.4368
Group 2D U-Net evaluated on HGG data	0.6435	0.8320	0.8259
2D U-Net evaluated on HGG data	0.6998	0.8110	0.8039
3D U-Net evaluated on HGG data	0.6396	0.8581	0.8181

Table 16: Summary of the final results of this work. Table showing the performance results of the best performing models produced from the Group 2D U-Net implementation alongside the 2D U-Net implementation based on [46] and the 3D U-Net implementations based on [46] per LGG and HGG data of the BraTS 2018 test data set respectively.

6 Conclusion

6.1 Main Results

In conclusion the main results of this project are the development of three deep neural network implementations for the segmentation of brain tumour MRI images. These are the group implemented 2D U-Net adaptation of [48], and my individual CNN implementations, namely the 2D U-Net implementation based on the work of [46] and the consequent [46] based 3D U-Net implementation. The performance of each of the algorithms measured as Dice scores per each individual tumour region (the multi-class case) [55] are provided in Table 16. As seen from Table 15 the result Dice scores achieved by these models are comparable to state-of-the-art techniques. Furthermore, the [46] based 3D U-Net implementation seems to have had the best overall performance for both the 210 high grade glioma (HGG) [55] and 75 low grade glioma (LGG) [55] data when evaluated on the BraTS 2018 Challenge test set as described in Section 1.3. In terms of the clinical context this means that using CNNs for brain tumour segmentation may assist radiologist by performing automatic brain tumour segmentation, thereby saving time [15] and assisting to circumvent any possible human based segmentation result variability [64].

6.2 Objectives Achieved

The main objectives of this project have been achieved with the solutions give per objective discussed in the Sections below.

6.2.1 Primary Objectives

- The literature review has been completed and is provided in Section 2. The consecutive research questions have been formulated and provided in Section 2.4.
- The initial pipeline to read in and pre-process the data from the open source data set (BraTS) [35], [3], [4], [5], [6] has been completed as part of the work done as a group as shown in Figure 11.
- A segmentation of brain tumour images with an implementation of a segmentation algorithm has been completed as part of the group initial pipeline implementation as shown in Figure 11 (a 2D U-Net adaptation of [48]).

6.2.2 Secondary Objectives

- The 2D U-Net [46] based convolutional neural network has been implemented to perform segmentation of tumour images as an example of a re-implementation of an existing algorithm as identified in the literature review given in in Section 2.
- The 3D U-Net [46] based convolutional neural network has been implemented as an example of more advanced algorithm for brain tumour segmentation compared to the 2D U-Net [46] based CNN implementation.
- There are two models which were implemented by me, namely the [46] based 2D U-Net implementation and [46] based 3D U-Net implementation. These were compared to the group implementation of a 2D U-Net adaptation of [48]. The performance and results of each of the three resulting CNN models has been provided in Figure 12 where my individual work is compared to the results achieved by the team.

6.3 Limitations

There were several constraints related to the GPU which were encountered during model training and data processing in the form of a Core Dumped error and an Out of Memory error (errors are described in Section 3.2.2). A Core Dumped error occurred when an attempt was made to train on more than five epoch of any of the three implemented algorithms on the 210 high grade glioma (HGG) [55] data from the BraTS 2018 data set. A work-around needed to be implemented by training each model for 5 epochs and then saving it and repeating the process. The Out of Memory error occurred for occurred on the lab GPU both during training and the evaluation of the performance of this algorithm for both the 210 high grade glioma (HGG) [55] and 75 low grade glioma (LGG) [55] patient cases. This was especially a problem for the 3D U-Net [46] based model implementation. While the 2D U-Net models used a batch size of one for evaluation on the HGG validation data and split the LGG training training data for performance evaluation, the [46] based 3D U-Net implementation needed to split the

validation data and training HGG data for evaluation together with the splitting of LGG training training data for performance evaluation. Additionally a batch size of one was used for the training of the [46] based 3D U-Net implementation in order to avoid the Out of Memory error.

6.4 Future Work

6.4.1 Combine the HGG and LGG dataset for Training and Evaluation

In this work, the training and evaluation has been done on LGG and HGG data seperately. However it would be interesting to investigate the performance of each model on the data as a whole, when compared to the state-of-the-art models as given in Table 15

6.4.2 Extend the Group U-Net Model to 3D

It would be interesting to investigate the performance of the modified 3D U-Net model which would be a modification of the group implemented 2D U-Net architecture in order to monitor any changes in the LGG Dice score results for enhancing tumour target class.

6.4.3 Use More Slices for Volumetric Information

The current [46] based 3D U-Net implementation uses two slices of each MRI image for volumetric information. It may be interesting to investigate and possibly gain an improvement in the performance of this model by increasing the number of slices.

6.4.4 Reduce Model Hyper-parameters and Model Complexity

A limitation of this 3D Model is the 105,271,812 trainable hyper-parameters which may be the reason for the model complexity and additional training time which was required compared to the other two models. It would be beneficial to experiment with the application of the exploding hyper-parameters challenge solution presented in [67] as one of the CNN technique described by [67] as discussed in Section 2.3.7.

6.4.5 Investigating the differences between 2D U-Net and 3D U-Net

It would be interesting to further investigate how the dice scores of the [46] based 2D U-Net implementation and [46] based 3D U-Net implementation described above mirror each other in performance. That is at what points in number of training epochs does performance, as measured by the dice scores per each individual tumour region (the multi-class case) [55], start to improve and/or deteriorate as changes are made in corresponding number of epochs in both of these neural networks.

References

[1] Abadi, M., et al.: TensorFlow: large-scale machine learning on heterogeneous systems (2015). https://www.tensorflow.org/, software available from tensorflow.org

[2] Baid, U., Talbar, S., Rane, S., Gupta, S., Thakur, M. H., Moiyadi, A., ... & Mahajan, A. (2018, September). Deep learning radiomics algorithm for gliomas (drag) model: a novel approach using 3d unet based deep convolutional neural network for predicting survival in gliomas. In International MICCAI Brainlesion Workshop (pp. 369-379). Springer, Cham.

[3] 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. https://doi.org/10.1038/sdata.2017.117

[4] Bakas, S., Reyes, M., Jakab, A., Bauer, S., Rempfler, M., Crimi, A., ... & Prastawa, M. (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.

[5] Bakas, S., Akbari, H., & Sotiras, A. (2017). Segmentation labels for the pre-operative scans of the TCGA-GBM collection. The Cancer Imaging Archive.

[6] Bakas, S., Akbari, H., Sotiras, A., Bilello, M., Rozycki, M., Kirby, J., ... & Davatzikos, C. (2017). Segmentation labels and radiomic features for the pre-operative scans of the TCGA-LGG collection. The cancer imaging archive, 286.

[7] Bousselham, A., Bouattane, O., Youssfi, M., & Raihani, A. (2019). Towards reinforced brain tumor segmentation on MRI images based on temperature changes on pathologic area. International journal of biomedical imaging, 2019.

[8] Brownlee, J. (2020, July 10). A Gentle Introduction to Batch Normalization for Deep Neural Networks. Retrieved from: https://machinelearningmastery.com/batch-normalization-for-training-of-deep-neural-networks/

[9] Brownlee, J. (2020, July 10). How to use the UpSampling2D and Conv2DTranspose Layers in Keras. Retrieved from: https://machinelearningmastery.com/upsampling-and-transpose-convolution-layers-for-generative-adversarial-networks/

[10] Chen, M., Wu, Y., & Wu, J. (2019, October). Aggregating Multi-scale Prediction Based on 3D U-Net in Brain Tumor Segmentation. In International MICCAI Brainlesion Workshop (pp. 142-152). Springer, Cham.

[11] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., & Ronneberger, O. (2016, October). 3D U-Net: learning dense volumetric segmentation from sparse annotation. In International conference on medical image computing and computer-assisted intervention (pp. 424-432). Springer, Cham.

[12] Ciresan, D., Giusti, A., Gambardella, L. M., & Schmidhuber, J. (2012). Deep neural networks segment neuronal membranes in electron microscopy images. In Advances in neural information processing systems (pp. 2843-2851).

[13] Géron, A. (2019). Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, (2nd Ed.). O'Reilly Media, Inc.

[14] Github. (2020, July 31). OOM when allocating tensor with shape. Retrieved from: https://github.com/tensorflow/tensorflow/issues/16768.

[15] Gordillo, N., Montseny, E., & Sobrevilla, P. (2013). State of the art survey on MRI brain tumor segmentation. Magnetic resonance imaging, 31(8), 1426-1438.

[16] Hanif, F., Muzaffar, K., Perveen, K., Malhi, S. M., & Simjee, S. U. (2017). Glioblastoma multiforme: a review of its epidemiology and pathogenesis through clinical presentation and treatment. Asian Pacific journal of cancer prevention: APJCP, 18(1), 3.

[17] Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., ... & Larochelle, H. (2017). Brain tumor segmentation with deep neural networks. Medical image analysis, 35, 18-31.

[18] Haghighi, M., Warfield, S. K., & Kurugol, S. (2018, April). Automatic renal segmentation in DCE-MRI using convolutional neural networks. In 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018) (pp. 1534-1537). IEEE.

[19] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).

[20] Hunter, J., Dale, D., Firing, E., Droettboom, M., & the Matplotlib development team. (2020, July 10). Matplotlib: Visualization with Python. Retrieved from: https://matplotlib.org.

[21] Isensee, F., Kickingereder, P., Wick, W., Bendszus, M., & Maier-Hein, K. H. (2018, September). No new-net. In International MICCAI Brainlesion Workshop (pp. 234-244). Springer, Cham.

[22] Işın, A., Direkoğlu, C., & Şah, M. (2016). Review of MRI-based brain tumor image segmentation using deep learning methods. Procedia Computer Science, 102, 317-324.

[23] Jäkel, S., & Dimou, L. (2017). Glial cells and their function in the adult brain: a journey through the history of their ablation. Frontiers in cellular neuroscience, 11, 24.

[24] Jiang, Z., Ding, C., Liu, M., & Tao, D. (2019, October). Two-Stage Cascaded U-Net: 1st Place Solution to BraTS Challenge 2019 Segmentation Task. In International MICCAI Brainlesion Workshop (pp. 231-241). Springer, Cham.

[25] Jing, H. (2020, July 31). Biomedical Image Segmentation - U-Net. GitHub. Retrieved from: https://jinglescode.github.io/2019/11/07/biomedical-image-segmentation-u-net/.

[26] Kamnitsas, K., Ledig, C., Newcombe, V. F., Simpson, J. P., Kane, A. D., Menon, D. K., ... & Glocker, B. (2017). Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. Medical image analysis, 36, 61-78.

[27] Keras. (2020, July 17). Convolution layers. Retrieved from: https://keras.io/api/layers/convolution_layers/.

[28] Lamba, H. (2019, February 17). Understanding Semantic Segmentation with UNET: A Salt Identification Case Study. Towards Data Science. Retrieved from: https://towardsdatascience.com/understanding-semantic-segmentation-with-unet-6be4f42d4b47.

[29] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3431-3440).

[30] McKinley, R., Jungo, A., Wiest, R., & Reyes, M. (2017, September). Pooling-free fully convolutional networks with dense skip connections for semantic segmentation, with application to brain tumor segmentation. In International MICCAI Brainlesion Workshop (pp. 169-177). Springer, Cham.

[31] McKinley, R., Meier, R., & Wiest, R. (2018, September). Ensembles of densely-connected CNNs with label-uncertainty for brain tumor segmentation. In International MICCAI Brainlesion Workshop (pp. 456-465). Springer, Cham.

[32] McKinley, R., Rebsamen, M., Meier, R., & Wiest, R. (2019, October). Triplanar Ensemble of 3D-to-2D CNNs with Label-Uncertainty for Brain Tumor Segmentation. In International MICCAI Brainlesion Workshop (pp. 379-387). Springer, Cham.

[33] Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., ... & Lanczi, L. (2014). The multimodal brain tumor image segmentation benchmark (BRATS). IEEE transactions on medical imaging, 34(10), 1993-2024.

[34] Myronenko, A. (2018, September). 3D MRI brain tumor segmentation using autoencoder regularization. In International MICCAI Brainlesion Workshop (pp. 311-320). Springer, Cham.

[35] Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., Lanczi, L., Gerstner, E., Weber, M. A., Arbel, T., Avants, B. B., Ayache, N., Buendia, P., Collins, D. L., Cordier, N., Corso, J. J., ... Van Leemput, K. (2015). The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). IEEE transactions on medical imaging, 34(10), 1993–2024. https://doi.org/10.1109/TMI.2014.2377694

[36] Milletari, F., Navab, N., & Ahmadi, S. A. (2016). V-net: fully convolutional neural networks for volumetric medical image segmentation 2016 Fourth International Conference on 3D Vision (3DV).

[37] Mokhtar, E. (2020, July 10). Seaborn Heatmap Tutorial (Python Data Visualization). Retrieved from: https://likegeeks.com/seaborn-heatmap-tutorial/

[38] Winkler, A. M. (23.September.2012) The NIFTI file format. Brainder. Retrieved from: https://brainder.org/2012/09/23/the-nifti-file-format/.

[39] NumPy. (2020, July 10). NumPy. Retrieved from: https://numpy.org

[40] Nurzynska, K. (2018). Deep learning as a tool for automatic segmentation of corneal endothelium images. Symmetry, 10(3), 60.

[41] NVIDIA. (2020, July 30). CUDA Zone — NVIDIA Developer. Retrieved from: https://developer.nvidia.com/cuda-zone

[42] Pereira, S., Pinto, A., Alves, V., & Silva, C. A. (2016). Brain tumor segmentation using convolutional neural networks in MRI images. IEEE transactions on medical imaging, 35(5), 1240-1251.

[43] Perelman School of Medicine, University of Pennsylvania. (2020, June 4). Multimodal Brain Tumor Segmentation Challenge 2018: Data Description Overview. Retrieved from: https://www.med.upenn.edu/sbia/brats2018/data.html.

[44] Perelman School of Medicine at the University of Pennsylvania. (2020, June 4). Multimodal Brain Tumor Segmentation Challenge 2018: BraTS 2018 Data Request. Retrieved from: https://www.med.upenn.edu/sbia/brats2018/registration.html.

[45] Quan, T. M., Hildebrand, D. G., & Jeong, W. K. (2016). Fusionnet: A deep fully residual convolutional neural network for image segmentation in connectomics. arXiv preprint arXiv:1612.05360.

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

[47] Schmidt, M. (2005). Automatic brain tumor segmentation.

[48] Sinha, A. (2020, June 30). Multimodal Brain Tumor Segmentation. GitHub. Retrieved from: https://github.com/as791/Multimodal-Brain-Tumor-Segmentation.

[49] Scikit learn. (2020, July 30). F1_score. Retrieved from: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html.

[50] Scikit-learn. (2020, July 10). Machine Learning in Python. Retrieved from: https://scikit-learn.org/stable/.

[51] SimpleITK. (2020, July 10). A simplified path to Insight. Open-source multi-dimensional image analysis in Python, R, Java, C#, Lua, Ruby, TCL and C++. Developed by the Insight Toolkit community for the biomedical sciences and beyond. Retrieved from: https://simpleitk.org/index.html

[52] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1), 1929-1958.

[53] Stackoverflow. (2020, July 15). Error: Segmentation fault (core dumped). Retrieved from: https://stackoverflow.com/questions/13654449/error-segmentation-fault-core-dumped/55954196

[54] Stackoverflow. (2020, July 31). How can I solve ŕan out of gpu memoryín TensorFlow. Retrieved from: https://stackoverflow.com/questions/36927607/how-can-i-solve-ran-out-of-gpu-memory-in-tensorflow/44102727.

[55] Stefani, A., Rahmat, R., & Harris-Birtill, D. (2020, July). Autofocus Net: Autofocused 3D CNN for Brain Tumour Segmentation. In Annual Conference on Medical Image Understanding and Analysis (pp. 43-55). Springer, Cham.

[56] T, D. (2020, July 10). Confusion Matrix Visualization. Retrieved from: https://medium.com/dtuk81/confusion-matrix-visualization-fc31e3f30fea

[57] Tensorflow. (2020, July 30). Tensorflow: Install. Retrieved from: https://www.tensorflow.org/install/source

[58] Python Software Foundation. (2020, July 10). tqdm. Retrieved from: https://pypi.org/project/tqdm/

[59] Villano, J. L., Parker, C. K., & Dolecek, T. A. (2013). Descriptive epidemiology of ependymal tumours in the United States. British journal of cancer, 108(11), 2367-2371.

[60] Wang, G., Li, W., Ourselin, S., & Vercauteren, T. (2017, September). Automatic brain tumor segmentation using cascaded anisotropic convolutional neural networks. In International MICCAI brainlesion workshop (pp. 178-190). Springer, Cham.

[61] Wang, G., Li, W., Ourselin, S., & Vercauteren, T. (2018, September). Automatic brain tumor segmentation using convolutional neural networks with test-time augmentation. In International MICCAI Brainlesion Workshop (pp. 61-72). Springer, Cham.

[62] Wang, G., Zuluaga, M. A., Li, W., Pratt, R., Patel, P. A., Aertsen, M., ... & Vercauteren, T. (2018). DeepIGeoS: a deep interactive geodesic framework for medical image segmentation. IEEE transactions on pattern analysis and machine intelligence, 41(7), 1559-1572.

[63] Waskom, M. (2020, July 10). seaborn: statistical data visualization. Retrieved from: https://seaborn.pydata.org.

[64] White, D. R., Houston, A. S., Sampson, W. F., & Wilkins, G. P. (1999). Intra- and interoperator variations in region-of-interest drawing and their effect on the measurement of glomerular filtration rates. Clinical nuclear medicine, 24(3), 177-181.

[65] Wang, W. (2020, July 3). A 3D U-Net Based Solution to BraTS 2019. GitHub. Retrieved from: https://github.com/woodywff/brats_2019.

[66] Yushkevich, P. A., Piven, J., Hazlett, H. C., Smith, R. G., Ho, S., Gee, J. C., & Gerig, G. (2006). User-guided 3D active contour segmentation of anatomical structures: significantly improved efficiency and reliability. Neuroimage, 31(3), 1116-1128.

[67] Zhao, Y. X., Zhang, Y. M., & Liu, C. L. (2019, October). Bag of Tricks for 3D MRI Brain Tumor Segmentation. In International MICCAI Brainlesion Workshop (pp. 210-220). Springer, Cham.

[68] Zhou, C., Chen, S., Ding, C., & Tao, D. (2018, September). Learning contextual and attentive information for brain tumor segmentation. In International MICCAI Brainlesion Workshop (pp. 497-507). Springer, Cham.

[69] Zhou, C., Ding, C., Lu, Z., Wang, X., & Tao, D. (2018, September). One-pass multi-task convolutional neural networks for efficient brain tumor segmentation. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 637-645). Springer, Cham.

[70] Zhang, J., Shen, X., Zhuo, T., & Zhou, H. (2017). Brain tumor segmentation based on refined fully convolutional neural networks with a hierarchical dice loss. arXiv preprint arXiv:1712.09093.

[71] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2018). Unet++: A nested u-net architecture for medical image segmentation. In Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support (pp. 3-11). Springer, Cham.

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

A Appendix

A.1 Ethical Approval Letter

University of St Andrews | FOUNDED 1413

University Teaching and Research Ethics Committee

19 June 2020

Dear Daria, Carina and Neha,

Thank you for submitting your ethical application, which was considered by the School of Computer Science Ethics Committee, where the following documents were reviewed:

1. Ethical Application Form

The School of Computer Science Ethics Committee has been delegated to act on behalf of the University Teaching and Research Ethics Committee (UTREC) and has granted this application ethical approval. The particulars relating to the approved project are as follows -

Approval Code:	CS14954	**Approved on:**	19.06.20	**Approval Expiry:**	19.06.25
Project Title:	Brain Tumour Segmentation (in MRI) using Deep Learning				
Researcher(s):	Daria Sergeevna Tkachova Carina Norregaard Neha Ghatia				
Supervisor(s):	Dr David Harris-Birtill and Dr Lewis McMillan				

Approval is awarded for five years. Projects which have not commenced within two years of approval must be re-submitted for review by your School Ethics Committee. If you are unable to complete your research within the five year approval period, you are required to write to your School Ethics Committee Convener to request a discretionary extension of no greater than 6 months or to re-apply if directed to do so, and you should inform your School Ethics Committee when your project reaches completion.

If you make any changes to the project outlined in your approved ethical application form, you should inform your supervisor and seek advice on the ethical implications of those changes from the School Ethics Convener who may advise you to complete and submit an ethical amendment form for review.

Any adverse incident which occurs during the course of conducting your research must be reported immediately to the School Ethics Committee who will advise you on the appropriate action to be taken.

Approval is given on the understanding that you conduct your research as outlined in your application and in compliance with UTREC Guidelines and Policies (http://www.st-andrews.ac.uk/utrec/guidelinespolicies/). You are also advised to ensure that you procure and handle your research data within the provisions of the Data Provision Act 1998 and in accordance with any conditions of funding incumbent upon you.

Yours sincerely

Wendy Boyter

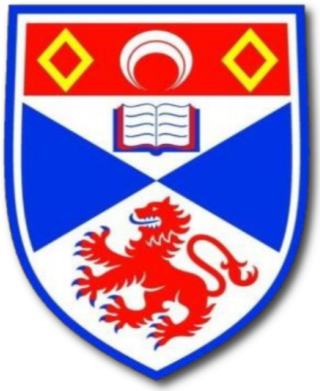
ethics-cs@st-andrews.ac.uk

The University of St Andrews is a charity registered in Scotland: No SC013532

School Ethics Committee Administrator

A.2 User Manual

User Manual



University of St Andrews

School of Computer Science

Brain Tumour Segmentation (in MRI) using Deep Learning Project

Daria Sergeevna Tkachova

Matriculation Number: 190024486

14 August 2020

Steps to execute

In order to run the code for this project please execute the following steps:

1. Read in/load the data from the "Data" file into the program. Please note that initially patients images from the BraTS 2018 [5],[1],[2][3],[4] dataset of high-grade gliomas (HGG) [6] and low-grade gliomas (LGG) [6] patients images need to be placed in the "Data" file (this is not done by the code and needs to be done manually separately in order to run the program). The reading in of the data is then done by running these python files:

- dataLoad_HGG.py
- dataLoad_LGG.py
- dataLoad_HGG_3D.py
- dataLoad_LGG_3D.py

2. Complete the tumour segmentation by running and training the different models and evaluating them on the validation data by running the following python files:

- ModelUnet_Group_2D_HGG.py
- ModelUnet_Group_2D_LGG.py
- 2DModel_HGG_Ronneberger_U-Net.py
- 2DModel_LGG_Ronneberger_U-Net.py
- 3DModel_HGG_Ronneberger_U-Net.py
- 3DModel_LGG_Ronneberger_U-Net.py

3. Evaluate the performance of each model on the test data by loading each of the models by running the following python files, for confusion matrix and image generation respectively:

- 2D_HGG_test_evaluate.py
- 2D_LGG_test_evaluate.py
- 2D_HGG_test_evaluate_images.py
- 2D_LGG_test_evaluate_images.py
- 3D_HGG_test_evaluate.py
- 3D_LGG_test_evaluate.py
- 3D_HGG_test_evaluate_images.py
- 3D_LGG_test_evaluate_images.py

4. Additionally evaluate the performance of each model on the training set by running the following python files:

- 2D_HGG_train_evaluate.py
- 2D_LGG_train_evaluate.py
- 3D_HGG_train_evaluate.py
- 3D_LGG_test_evaluate.py

References

[1] 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. https://doi.org/10.1038/sdata.2017.117

[2] Bakas, S., Reyes, M., Jakab, A., Bauer, S., Rempfler, M., Crimi, A., ... Prastawa, M. (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.

[3] Bakas, S., Akbari, H., Sotiras, A. (2017). Segmentation labels for the pre-operative scans of the TCGA-GBM collection. The Cancer Imaging Archive.

[4] Bakas, S., Akbari, H., Sotiras, A., Bilello, M., Rozycki, M., Kirby, J., ... Davatzikos, C. (2017). Segmentation labels and radiomic features for the pre-operative scans of the TCGA-LGG collection. The cancer imaging archive, 286.

[5] Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., Lanczi, L., Gerstner, E., Weber, M. A., Arbel, T., Avants, B. B., Ayache, N., Buendia, P., Collins, D. L., Cordier, N., Corso, J. J., ... Van Leemput, K. (2015). The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). IEEE transactions on medical imaging, 34(10), 1993–2024. https://doi.org/10.1109/TMI.2014.2377694

[6] Stefani, A., Rahmat, R., Harris-Birtill, D. (2020, July). Autofocus Net: Autofocused 3D CNN for Brain Tumour Segmentation. In Annual Conference on Medical Image Understanding and Analysis (pp. 43-55). Springer, Cham.