

CNN–GNN Fusion for Anatomically-Consistent Multimodal Brain Tumor Classification and Segmentation

by

Alan Mukeyev

Submitted to the Department of Data Science
in partial fulfillment of the requirements for the degree of

Master of Science in Data Science

at the

NAZARBAYEV UNIVERSITY

June 2026

© Nazarbayev University 2026. All rights reserved.

Author
Department of Data Science
April 10, 2026

Certified by
Hari Mohan Rai
Assistant professor
Thesis Supervisor

Accepted by
Yelyzaveta Arkhangelsky
Dean of the School of Engineering and Digital Sciences

CNN–GNN Fusion for Anatomically-Consistent Multimodal Brain Tumor Classification and Segmentation

by

Alan Mukeyev

Submitted to the Department of Data Science
on April 10, 2026, in partial fulfillment of the
requirements for the degree of
Master of Science in Data Science

Abstract

Brain tumor analysis from multimodal magnetic resonance imaging (MRI) remains a challenging task because highlighting tumor subregions properly requires both local feature extraction and understanding of the global brain structure. This thesis proposes a hybrid deep learning framework for multimodal brain tumor segmentation and tumor grade classification that combines a 3D U-Net backbone with a grid-based graph neural network (GNN) refinement module and a DARTS-inspired fusion mechanism. This proposed fusion block tried to learn an optimal combination from 4 parts: CNN-only, GNN-only, averaged, and concatenation-based representations. A combination of these versions allowed the proposed framework to take into account their contributions based on their assigned weights. The dataset used for this paper is the BraTS 2021 dataset. Apart from that, hyperparameters were not selected manually but through Bayesian optimization with Optuna, and the final model was assessed with three-fold nested cross-validation. In the context of experimental results, the model showed competitive segmentation performance. During initial training, composite loss decreased from 0.694 to 0.136, while validation Dice reached 0.689 for enhancing tumor (ET), 0.781 for tumor core (TC), and 0.899 for whole tumor (WT). In nested cross-validation, the final mean Dice scores were 0.758 for ET, 0.819 for TC, and 0.879 for WT, with an overall mean Dice of 0.819. What is interesting to note is that the GNN-refined output was more favored over the raw CNN output. Also, this paper proposed a classification pipeline that is based on 28-dimensional features extracted from predicted tumor regions. This stage achieved a mean accuracy of 0.863 and a mean macro F1-score of 0.740 across folds. Overall, CNN-GNN fusion is a promising direction for brain tumor analysis, and the results in this paper demonstrated this.

Thesis Supervisor: Hari Mohan Rai
Title: Assistant professor

Acknowledgments

I would like to sincerely thank everyone who supported and encouraged me throughout my research journey and the completion of this thesis.

I am especially grateful to my supervisor, Hari Mohan Rai, for his continuous guidance, valuable suggestions, and constant support, which played a significant role in shaping this work.

I would also like to thank the faculty members and my peers for their helpful discussions, insights, and encouragement that enhanced my understanding of the research topic.

Finally, I deeply appreciate all those who, in one way or another, contributed to this work and motivated me throughout the process of writing this Thesis.

Contents

1	Introduction	13
1.1	Motivation	13
1.2	Applications	14
1.3	Challenges	14
1.4	Problem Statement	15
1.5	Contributions	16
1.6	Thesis Organization	18
2	Literature review	19
2.1	Section overview	19
2.2	CNN-Based Architectures for Multimodal Brain Tumor Segmentation	19
2.3	Graph-Based and Hybrid CNN–GNN Models	22
2.4	Multimodal Fusion Strategies	23
2.5	Deep Learning Models for Brain Tumor Classification	24
3	Materials and Methods	27
3.1	Dataset	28
3.2	Data pre-processing	30
3.2.1	Multimodal loading and channel construction	32
3.2.2	Tensor Conversion	32
3.2.3	Orientation standardization	33
3.2.4	Voxel Spacing Normalization	33
3.2.5	Intensity Normalization	34

3.2.6	Label Remapping	34
3.2.7	Foreground Cropping	35
3.2.8	Spatial Padding	36
3.2.9	Patch-Based Sampling	36
3.2.10	Data Augmentation	37
3.3	Composite Loss Function Design	37
3.4	Proposed CNN–GNN Fusion Architecture	41
3.4.1	3D U-Net Backbone	43
3.4.2	Grid-Based GNN-Style Refinement Module	44
3.5	Tumor Grade Classification Pipeline	45
3.6	Evaluation Protocol	47
4	Results and Discussion	49
4.1	Initial Segmentation Training	49
4.2	DARTS Fusion Architecture Analysis	51
4.3	Bayesian Hyperparameter Optimization	54
4.4	Nested Cross-Validation Segmentation Results	55
4.5	Tumor Grade Classification	60
4.6	Discussion	63
5	Conclusion	65
5.1	Key Findings	66
5.2	Future Work	66

List of Figures

3-1	Block diagram of the proposed CNN–GNN fusion methodology pipeline.	29
3-2	Distribution of derived tumor grade categories across the BraTS 2021 dataset. Grade-IV-like cases dominate at 80.2%, while Grade-III-like represents only 5.9%, creating severe class imbalance addressed through SMOTE oversampling in the classification pipeline.	31
3-3	Architecture of the proposed CNN–GNN fusion model.	43
4-1	Initial segmentation training results over 50 epochs: (a) training loss convergence from 0.694 to 0.136, (b) validation Dice scores per tumor region reaching 0.899 (WT), 0.781 (TC), and 0.689 (ET), (c) HD95 improvement from above 50 mm to below 11 mm, (d) voxel-level confusion matrix with BG=1.00, NCR=0.90, ED=0.90, ET=0.82	52
4-2	DARTS fusion weights showing autonomous preference for GNN-only (0.296) over CNN-only (0.217)	53
4-3	Final test Dice scores across three outer folds. All folds demonstrate consistent performance, with WT scores in the 0.866–0.888 range, TC between 0.782–0.846, and ET between 0.731–0.779.	56
4-4	Qualitative segmentation results for representative cases. From left to right: FLAIR modality, ground truth segmentation overlay, model prediction overlay, and mismatch map showing regions of disagreement between prediction and ground truth.	57
4-5	Normalized voxel-level segmentation confusion matrix for Folds 0, 1, 2	58

4-6	DARTS fusion weight evolution during nested cross-validation (Fold 0, 1, 2). The GNN-only operator converged to the highest weight, confirming the pattern observed in the initial training phase.	59
4-7	Tumor grade classification performance across three outer folds. Both overall accuracy (0.861–0.866) and mean F1-score (0.730–0.748) demonstrate consistent cross-fold performance.	62
4-8	Per-class F1-scores across outer folds. Grade-IV achieves consistently high F1 (0.927–0.936), while Grade-II (0.625–0.658) and Grade-III (0.613–0.691) show lower but stable performance reflecting the difficulty of minority class classification.	62

List of Tables

3.1	Summary of the BraTS 2021 dataset used in this study.	31
4.1	Experimental configuration of the proposed CNN–GNN fusion framework	50
4.2	Bayesian hyperparameter optimization results.	54
4.3	Nested cross-validation segmentation results (Dice scores).	55
4.4	Nested cross-validation segmentation results across three outer folds. Dice Similarity Coefficient (DSC) and HD95 are reported per tumor sub-region. HD95 values are mean averages across held-out test sets.	59
4.5	Tumor grade classification results across outer folds.	61
4.6	Tumor grade classification results across outer folds. Per-class F1- scores are shown for each derived grade category.	63
4.7	Overall performance summary of the proposed CNN–GNN fusion frame- work.	64

Chapter 1

Introduction

Brain tumors remain one of the most significant global health concerns today. According to the most recent CBTRUS Statistical Report, the average annual age-adjusted incidence rate of primary brain and other CNS tumors in the United States is roughly 26.05 per 100,000, which means that almost 490,000 cases were diagnosed between 2018 and 2022 [34]. This highlights how important it is to dive deeper into the problem of brain tumors.

1.1 Motivation

Brain tumors are a disease that many people around the world face today. It impacts human's health, their well-being, and early detection of this disease may lead to better impact of disease's treatment. However, automated solutions that can assess brain MRI scans are still not at their final stage and need more improvement to make proper evaluations of people's brain tumor disease. This is what motivated me to work upon a system construction that can both segment brain tumors and also classify them.

Thus, brain tumors are of great importance in medicine and are one of the central problems in medical image computing. They occur from abnormal cell growth within brain tissue regions and lead to high mortality rates by posing great risks to people's health [46]. Among all brain tumors, glioma is considered to be one of the most aggressive and frequent one, and it stands for roughly 80 percents of all malignant

brain tumors. Glioma can be categorized into high-grade and low-grade gliomas (HGG and LGG, respectively), and exact and timely diagnosis is necessary for these disease types[46, 41]. Taken together, there is a high importance of proper brain tumor segmentation, classification, and its related predictive tasks, because the problem is clinically and technically demanding. People receive serious health issues related to brain tumor diagnosis and integration of technical tools may help the medicine.

1.2 Applications

The proposed framework is mainly intended to be used in clinical and research settings. Automated brain tumor segmentation can help radiologists and doctors in revealing the disease and in reducing the effort to delineate the tumor sub-regions and volumetric measurements. Beyond clinical use, the proposed CNN-GNN fusion strategy can be used in other 3D medical image segmentation tasks where local and global contexts matter. Overall, the proposed study is intended to help doctors and clinicians during the work with image scans.

1.3 Challenges

The literature also explain why MRI-based manual tumor assessment may need substitution to automated solution. The reason is that manual segmentation, for example, is time-consuming, subjective, and laborious [51]. Authors from the paper [31] also support the idea that manual assessment can lead human-related errors and relies on clinician experience, which differs among doctors. These observations demonstrate the necessity for the reduction of manual labor in brain tumor field and a gradual shift to automated solutions, which also need to be consistent.

Another major reason why this task is difficult for both manual and automated assessment is that MRI sequences are multimodal, and each one provides a complementary view of brain tumor structure. For example, edema, contrast enhancement, surrounding tissues are presented in different MRI versions. So, it can be said that

information that is obtained from single-image modality is limited; therefore, a multi-modality fusion network is needed [51]. Another paper [47] makes a similar point and highlights the importance of leveraging multi-modal images due to the specific characteristics of each modality. These authors discuss the idea that robust brain tumor analysis should learn from different modalities rather than from a single sequence.

Convolutional neural networks are algorithms that are applied for brain tumor segmentation due to their abilities of feature extraction. Architectures such as FCN and U-Net, as well as advanced variants, Res-U-Net, DenseNet-based models are examples that are used for convolutional neural networks [46]. However, CNN models have a problem related to their solution approach. CNNs struggle with segmentation tasks due to their limited understanding of provided scan data. Their nature is focused on local understanding of images and faces limitations in modeling global relationships between different parts of scan data [17, 25].

These limitations of CNN models showed that some new solutions may need to be integrated for seeking both global and local reasoning. Thus, recent research have also offered another approach where images are represented as graphs that consist of connected brain regions with edges [37]. This allows another approach to be integrated, where CNN captures local features, while Graph Neural Network models are also integrated, and they catch global contextual representation [37, 35, 25]. Authors in these works [44, 17] state the idea that local and global features are both crucial for proper model inference. To summarize, the field has realized that pure local feature extraction is often not enough for an accurate 3d multimodal tumor analysis, and this complementarity between CNN and GNN models is exactly the reason why the CNN-GNN fusion framework is offered in this research paper.

1.4 Problem Statement

This thesis intends to address several important problems related to deep-learning-based brain tumor analysis.

1. CNN-based segmentation models capture local features well but struggle with

modelling global anatomical context, which can limit the accuracy of brain tumor segmentation

2. Some of the existing CNN-GNN hybrid approaches rely on fixed or manually tuned fusion strategies instead of learning the optimal combination based on data.
3. Both segmentation and classification parts of brain tumors are affected by severe class imbalance, which has to be handled carefully.

1.5 Contributions

The classification of brain tumors is also a vital part since early tumor detection can lead to proper treatment and higher survival rate [35]. Some examples that other research authors utilized for brain tumor classification include K-Nearest Neighbors (KNN), K-means, and Support Vector Machine (SVM) [35].

This work has developed an automatic segmentation framework for multimodal brain tumor that is based on a fusion of convolutional and graph-inspired reasoning modules. Therefore, this whole pipeline relies on both a 3D CNN and a grid-based GNN module that helps with contextual consistency. For this research work, it was decided to choose the 3D U-Net as a CNN branch that operates with four-channel MRI input and a GridGNN-based refinement module for catching global contextual information. Apart from that, I applied a DARTS-inspired fusion block that combines CNN, GNN-refined logits, their averaged representation, and a concatenation-based projection as well. This whole design allowed the model to preserve local context and incorporate broader spatial interactions. The BraTS dataset was utilized for the work, and in order to tailor the framework to the characteristics of BraTS tumor subregions, the training pipeline includes multimodal preprocessing, label remapping, ET-biased patch sampling, and a composite objective function.

Beyond segmentation, this work also extends the proposed framework to perform tumor grade classification. The BraTS dataset does not provide explicit labels for

tumor grades; this is why clinically motivated grade categories were taken from the ground-truth segmentation masks that are themselves based on volumetric region ratios: Grade-IV-like (high enhancing tumor proportion), Grade-III-like (moderate tumor core proportion), and Grade-II-like (edema-dominant cases). For every patient case, a trained segmentation model was used to extract a 28-dimensional feature vector from the predicted segmentation output, which captured region volumes, class probabilities, prediction confidence, and spatial entropy characteristics. Also, for addressing the severe class imbalance, SMOTE-inspired oversampling was utilized. After that, a four-layer neural network classifier was trained on these features by using three-fold cross-validation structure. This pipeline demonstrates how the classification part was implemented.

Here are main contributions of this paper:

- **CNN–GNN Fusion with Learnable Architecture Search.**

This work proposes a hybrid segmentation framework that combines a 3D U-Net convolutional backbone with a grid-based Graph Neural Network refinement module for brain tumor segmentation from multimodal MRI data. The main contribution of this part is that unlike other CNN-GNN approaches that rely on fixed fusion strategies or manually chosen combination weights, the proposed framework in this study integrates a DARTS-inspired search mechanism that learns the optimal fusion of four candidate operators: CNN-only, GNN-only, their arithmetic average, and a concatenation-based learned projection.

- **Empirical Validation of GNN Contribution Through Autonomous Weight Discovery.**

Another central finding of this work is that the DARTS fusion mechanism consistently assigned higher weight to the GNN-refined output across all experiments. Specifically, the GNN-only operator converged to approximately 0.296, while the CNN-only operator received approximately 0.217 in the initial training. This pattern was shown without any manual tuning from my side and showed the importance of graph-inspired module incorporation.

- **Segmentation-Derived Tumor Grade Classification Pipeline.** This paper does not only focus on segmentation but also proposes a framework for classification, where grade categories were derived from segmentation outputs. SMOTE-inspired oversampling and class-weighted cross-entropy loss were integrated to handle severe class imbalance.

1.6 Thesis Organization

The remainder of this paper is structured as follows. Firstly, Section 2 provides a review of the literature on multimodal brain tumor segmentation, classification, CNN-based medical image analysis as well as graph-based refinement approaches. Section 3 describes the dataset that was used in the work, the preprocessing pipeline, the model architecture, fusion strategy, loss design, and procedure for hyperparameter optimization. Section 4 is dedicated for results of the experiments and the discussion part. Section 5 is the conclusion of the paper.

Chapter 2

Literature review

2.1 Section overview

This chapter provides a comprehensive review of the existing literature relevant to the proposed CNN-GNN fusion framework for multimodal brain tumor classification and segmentation. Review section is divided into 4 thematic sections that together establish the methodological context of this thesis. Studies explored in this review are selected based on their relevance to multimodal MRI analysis, their use of datasets that are comparable to BraTS, their closeness to the proposed framework. Detailed description of these studies are presented further in this section.

2.2 CNN-Based Architectures for Multimodal Brain Tumor Segmentation

Deep learning has become one of the dominant methodologies for automated brain tumor segmentation, and this is caused by its great ability to capture feature representations from multimodal MRI data, which is used in most of the explored studies. Across the reviewed literature, only studies that report segmentation architecture are considered in this part, along with the metric results. Most segmentation studies are built with the help of a decoder-encoder CNN architecture, such as U-Net and its

other variants. They are strong with dense pixels and voxel-level prediction.

Zhao et al. [51] propose MM-Unet, which is an architecture designed for segmenting brain tumors from multimodal MRI scans. Authors apply multiple encoders for each modalities of MRI scans and a shared decoder that restores pixel-level results of segmentation. This design allows them to take into account properties of each MRI type and outperform single-stream architectures in terms of capturing local and global details. This paper provides results with a mean Dice score of 79.2% and mean Hausdorff95 results of 8.466. Another representative example is SEResU-Net, which is augmented with a U-Net backbone and is explicitly designed for multimodal MRI segmentation [46]. This design’s effectiveness is consistent with high Dice scores across brain tumor subregions. On the BraTS2018 and BraTS2019 datasets SEResU-Net showed Dice coefficients exceeding 0.94 for whole tumor segmentation for the BraTS2018 dataset and outperformed some of the baselines models [46]. In another paper [39], authors utilize a network MM-BiFPN that has 4 separate encoders for each of the MRI modalities and use Bi-FPN layers to connect their features and receive as much information as possible from them. This approach allows them to decide which of the modalities contribute more to the segmentation. Authors also use a method, where information is passed from top to bottom and from bottom to top, which allows them to observe local and global details. This research claims that the MM-BiFPN model has a lower precision value than other models but outperforms previous models in terms of accuracy. Despite strong local feature extraction methods, CNNs are limited in modelling long-range spatial and global anatomical dependencies. Wang et al. [44] proposed a novel TransBTS model that is based on an encoder-decoder structure. Transformer is successful in natural language processing tasks, so the authors decided to apply it for MRI brain Tumor Segmentation. For capturing local context information, 3D CNN is exploited, and volumetric feature maps are collected [44]. These features are then reshaped intentionally to tokens, so that the Transformer can model global dependencies. The decoder part leverages features produced by the Transformer and performs progressive upsampling for building a detailed segmentation map. The TransBTS model achieves a Dice score of 78.69%, 90.98%,

82.85%, respectively, for ET, WT, and TC for the BraTS 2019 dataset, and 78.73%, 90.09%, 81.73% for the BraTS 2020 dataset, which outperforms previous state-of-the-art results for 3D brain tumor segmentation. Another observed strategy included the adoption of attention mechanisms, which emphasized spatial and channel-wise features. AABTS-Net utilizes attention mechanisms to capture long-range dependencies and achieves a competitive Dice score on BraTS2019 validation data [41]. Tumor segmentations are small subregions and attention-augmented architectures are particularly effective for improving the task of brain tumor segmentations. In another research paper, authors proposed another new segmentation approach, which connects a transformer, convolutional neural network, and an encoder-decoder structure for 3D segmentation of brain tumors [17]. The Swin transformer uses a shifted window attention, while the encoder produces features and the decoder rebuilds an output for segmentation goals. Final results on three datasets (BraTS 2019–2021) show that their method provides better results than some of the state-of-the-art methods, including TransBTS. In this paper [42], the authors proposed a Tumnet model, which is a lightweight convolutional neural network. The model was used for classifying and segmenting multimodal MRI and CT scans, which were fused at a pixel level. Datasets were collected from Kaggle and Harvard Med AANLIB. Finally, for classification performance, the model achieved these average results: accuracy at 90%/98%, sensitivity at 96%, and specificity at 95%/99%. Even though the dataset is not large, Tumnet demonstrated that it can provide high values for classification tasks with the right fusion techniques. This paper [31] presents a study where two state-of-the-art CNN architectures were proposed. The first CNN conducts tumor segmentation on multimodal MRI scans with encoder being used for extracting features and reducing dimensionality, and the decoder part rebuilds its output to original size. For this stage, a 3D U-Net-based model was applied. For the second stage EfficientNetB2 was already used for tumor classification tasks (high-grade and low-grade gliomas). The three-class dataset and the BraTS 2019 dataset were used for the models. EfficientNetB2 achieved 99.19% test accuracy, 99.17% precision, 98.94% sensitivity, 99.57% specificity, and 99.06% F1-score, while segmentation outcomes were robust as well:

86.13, 86.75, and 92.41 for et, tc, and wt, respectively, for a Dice score [31]. Thus, the paper presented segmentation and classification parts in a single pipeline with overall strong results. In another study Axial Attention CNN was introduced for segmentation tasks on multimodal MRI images [41]. The attention mechanism processes information along two axes (horizontal and vertical ones separately). This attention part also allows us to focus more on important parts, including brain tumors, and neglect less relevant locations, which provides better results. The BraTS dataset was used for training the network. Even though the full 3D transformer was not utilised, Axial Attention still managed to investigate global and spatial relations well. For BraTS 2021, the model showed great results: DSC values of 0.830, 0.861, and 0.922 and HD values of 17.728, 11.178, and 3.996 for ET, TC, and WT regions, respectively. Transformer-based and attention-based models therefore extend global context modeling beyond plain convolution by observing volumetric dependencies in the entire brain volume.

2.3 Graph-Based and Hybrid CNN–GNN Models

CNN limitations related to modelling long-range spatial and global anatomical dependencies can be overcome with graph-based reasoning modules that are integrated into CNN segmentation pipelines.

Ma et al. [25] used the DGRUnit model for brain tumor segmentation that applies both CNN and GNN. The former one is used to capture local details, while the latter one is used to show relationships between tumors at different regions. The model treats regions as graph nodes and therefore captures relation between tumor areas spatially. The model shows good results for many metrics on the BraTS 2018 dataset. For example, the Nested U-net + DGRUnit demonstrated 0.998, 0.993, 0.997 for ET, WT, and TC regions, respectively, for the specificity metric. Therefore, the paper shows that combining both graphical reasoning and CNN can provide good results that outperform some of the state-of-the-art outcomes. In another paper [37], authors proposed the Joint Graph and Image Convolution Network that connects CNNs with

GNNs, also for the task of tumor segmentation. Authors use this connection because CNN is good at learning spatial information within a region, but can ignore relationships between different regions. Authors used each superpixel as a graph node and connected them according to spatial proximity. The main goal of the project was to show that GNNs can provide good results on their own for tumor segmentation rather than construct highly optimal CNN solutions. Nevertheless, the results of the merged GNN-CNN model provided results better than when they are separate. For the Dice score, models produced 0.925, 0.908, and 0.842 for WT, TC, and ET regions respectively.

2.4 Multimodal Fusion Strategies

In this paper [47], the authors suggested a F2Net network that can work with any number of MRI modalities. F2Net is constructed with encoder-decoder structure and uses 2 Transformer-based feature learning streams as well as cross-modal shared learning network. This allows us to extract separate and joint representations of features. On the BraTS 2019 dataset the model indeed provided high results. For the Dice score, their model had results of 95.01, 94.28, 90.24 for WT, TC, and ET regions, respectively. The model illustrated its effectiveness over many other segmentation methods. Authors in paper [8] offered MVFusFra, which is a multi-view dynamic fusion framework, for combining information from T1, T1ce, T2, and FLAIR modalities. This method effectively integrated multimodal MRI data for segmentation purposes and achieved Dice scores of 0.83682/0.73627/0.71239 for complete/core/enhancing on BraTS2018 [8]. Similarly, MFD-Net introduced a multi-feature decoder for cross-modal interaction. These papers suggest that fusion strategy design is as critical as backbone architecture.

Beyond Dice scores, studies also considered other metrics. For example, boundary-aware evaluation was emphasized by merging Hausdorff distance and recall metrics for the goal of assessing geometric accuracy across the data. MeniSeg reported HD95 and Dice results [50]. Such metrics are crucial for clinical decisions, as boundary

errors can lead to significant mistakes even when the Dice score is relatively strong.

nnU-Net was proposed as a self-configuring segmentation framework that was evaluated on a range of medical images. It showed strong generalization without manual architecture tuning [16]. Even though this paper is not strongly attached to brain tumors explicitly, nnU-Net serves as a strong baseline and can be used for highlighting the features of automated pipeline optimization.

Overall, most of the models used for brain tumor segmentation were trained and tested on BraTS datasets from different years and the results of the papers are shown below in the table.

2.5 Deep Learning Models for Brain Tumor Classification

Deep learning-based brain tumor classification methods that are described in the review studies address important medical questions, such as binary tumor detection, multi-class tumor type classification, and glioma grade discrimination. Classification works that are described in the literature below can be grouped into 4 categories: 1) CNN-based classifiers 2) graph-based (CNN + graph convolution) classification models 3) segmentation-to-classification works 4) multi-stage segmentation-classification pipelines CNN-based techniques stand as primary backbones for classification tasks, and they are combined with preprocessing, fusion, and optimization strategies for better results. One of the examples uses a TumNet classifier, that applies pixel-level fusion with multiple fusion rules. This classifier performs binary tumor classification and achieves an accuracy of 96% by outperforming AlexNet (82%), GoogLeNet (78%), and a conventional CNN (67%) on the same dataset [42]. Another instance of a CNN-based approach includes DeepLabV3+, which is an architecture for classification, and it works by connecting multiple architectural modules, feature fusion, and Bayesian hyperparameter optimization. For this work, the authors managed to achieve the following results: accuracy of up to 95.42%, with sensitivity of 94.20%, precision

of 94.53%, and an area under the ROC curve (AUC) of up to 0.99 [28]. So these mentioned methods incorporate fusion, optimization, or region-focused preprocessing instead of an ordinary baseline CNN architecture. Another method that is used for classification tasks is Graph-based models. In [35] study authors proposed a graph convolutional neural network combined with a CNN classifier that is evaluated on a four-class MRI classification task (no tumor, pituitary tumor, glioma, meningioma) based on the Kaggle brain tumor classification dataset. The best configuration in the paper (Net-2) achieved an accuracy of 95.01%. Beyond the above 2 approaches, this review also considers papers that explicitly use a segmentation-to-classification pipeline, where initially brain tumors are segmented by a model and then classified. One such example is where a 3D U-Net is used for tumor segmentation, and subsequently, a CNN-based classifier model is utilized for classification. This study achieved strong results: sensitivity of 97%, for high-grade glioma (HGG), sensitivity values are up to 99% and 94% for low-grade glioma (LGG), specificity values of 94% for HGG and 99% for LGG [31].

Chapter 3

Materials and Methods

In this section, I provide a detailed overview of materials, pipeline, methodological stages that have been used for this work. The proposed framework focuses on multimodal brain tumor analysis and employs a CNN–GNN fusion strategy for segmentation and classification from MRI dataset. For the current work, the provided approach connects both volumetric convolutional learning and graph-inspired contextual understanding for considering both local information and relations between regions from different parts of brain scans. Here is a workflow that has been applied: dataset was acquired and organized, the pre-processing part was completed, patch-based training samples were generated, a 3D U-Net backbone was used for segmentation, graph-based refinement was completed with the help of a grid-structured neighborhood propagation module, and nested cross-validation was also applied for improving evaluation results. All of these transformations eventually led to a goal of improving metrics higher, so that the model could properly classify and segment brain tumors.

Regarding the core part, then the model architecture is built around a 3D U-Net backbone, which is a CNN model [7, 30]. As an input it is provided with four-channel multimodal MRI data. Basically, the main role of this backbone is to explore hierarchical volumetric features and reconstruct dense segmentation maps with an encoder-decoder structure. After the initial prediction made by the CNN part, a graph-based refinement stage steps in and is applied through a grid-based neighborhood propaga-

tion mechanism. Simply, this refinement part lets operate over the spatial structure of the CNN prediction and exchange information between different regions, allowing the model to receive broader contextual information [43]. This branch is a lightweight GNN-style improvement that works together with local extractions from the CNN backbone [19]. Further, these 2 branches are combined by a DARTS-inspired fusion block that assigns weights to CNN and GNN outputs and learn how much of contribution each of them should make for the final output [23]. This part is central in this work, because it allows not to merely focus just on CNN with post-processing and a graph model, but to concentrate on an architecture that tries to exploit the strengths of both models and apply them properly.

For evaluating the performance of the proposed framework, diverse elements are combined: nested cross-validation for robust assessment, Optuna-based configuration search as well, sliding-window volumetric inference, Dice-based overlap analysis for ET, TC, and WT, mean validation Dice for overall monitoring, and HD95 for boundary-aware assessment. By having this approach, the goal of this study is to achieve a correct detection of brain tumors and make a positive contribution to the field. The next sections will explain techniques and methodologies for brain tumor segmentation and classification in more depth. The overall methodology pipeline is illustrated in Figure 3-1.

3.1 Dataset

The dataset that was used for this work is a publicly available dataset called BraTS 2021 that can be obtained from this study [2]. This is a highly popular dataset for multimodal brain tumor segmentation and classification tasks [3]. Basically, this dataset consists of multi-parametric 3D MRI scans of patients who have glioma.

The dataset consists of 1251 patients’ data for the training part and includes both high-grade gliomas (HGG) and low-grade gliomas (LGG). MRI data is actually publicly available for both training and validation sets, however manual segmentation can be accessed only for the training part. Within the dataset, each subject is rep-

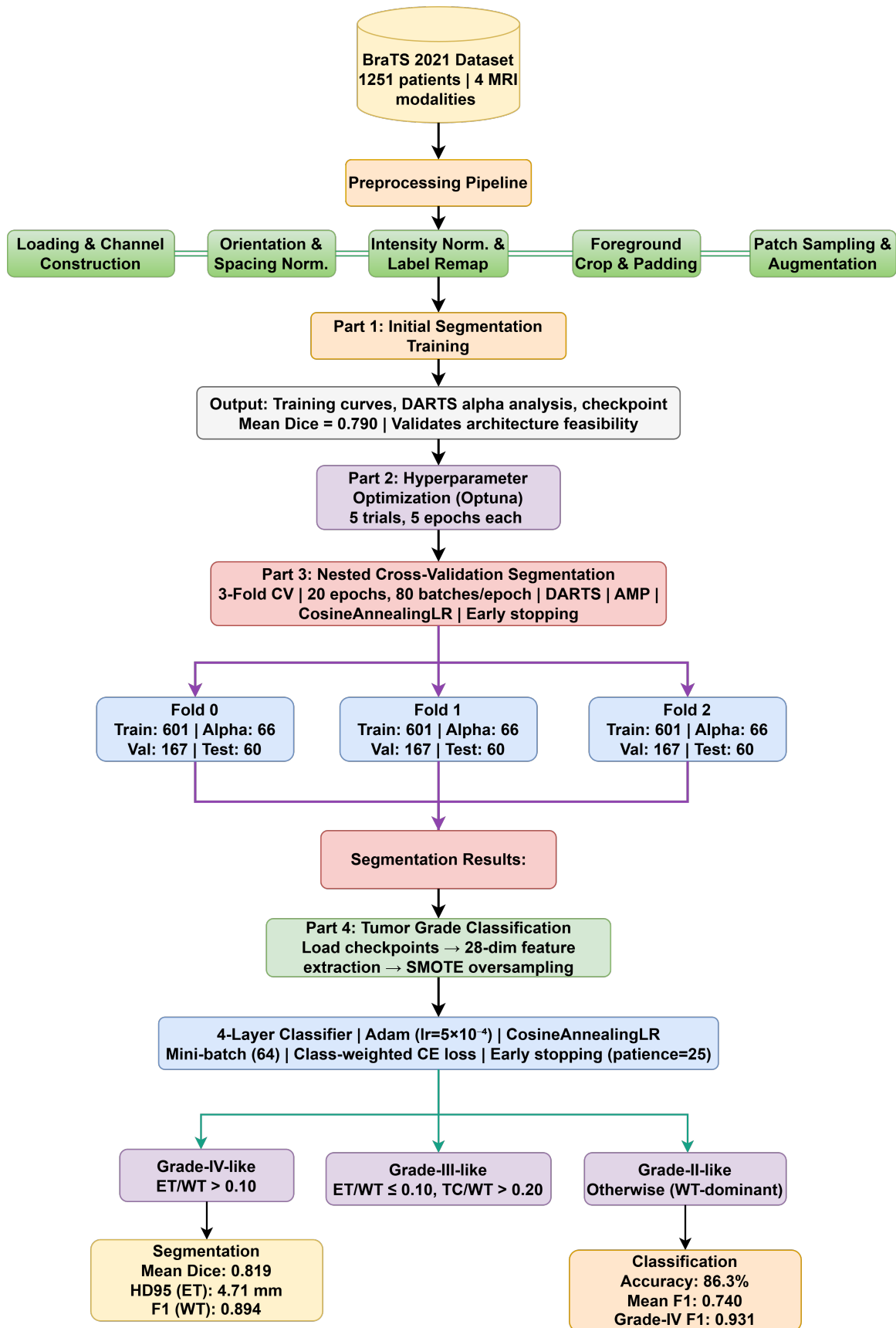


Figure 3-1: Block diagram of the proposed CNN-GNN fusion methodology pipeline.

resented as a 3D volumetric scan in 4 modalities: T1-weighted (T1), T1-weighted contrast-enhanced (T1ce), T2-weighted (T2), and Fluid Attenuated Inversion Recovery (FLAIR). All of these modalities have been preprocessed using a standardized pipeline including skull stripping, resampling to isotropic resolution (1 mm^3), intensity normalization, and a rigid co-registration to a common anatomical template. Each of the contained MRI volume has a spatial size of $240 \times 240 \times 155$ voxels, which enables consistent volumetric processing for all samples.

Apart from these, the scans have the following label encodings: label 0 is used for background information, label 1 is necrotic and non-enhancing tumor core, label 2 is for peritumoral edema, and label 4 is for enhancing tumor. These labels are also grouped into clinically meaningful regions, where enhancing tumor (ET) covers label 4, tumor Core (TC) covers labels 1 and 4, and whole tumor (WT) covers all labels 1, 2, and 4.

Overall, the BraTS dataset’s main focus is pointed at glioma tumors, which is considered to be one of the most aggressive types of brain tumors. If we compare earlier 2D MRI datasets with BraTS dataset, then BraTS dataset has an advantage in that all the data scans are 3D volumetric and they preserve spatial continuity. This is suitable for evaluating advanced architectures that require both local feature extraction and global reasoning of a space, particularly for the proposed in this study CNN-GNN fusion framework.

Figure 3-2 shows the resulting distribution for the dataset: cases with an ET/WT ratio exceeding 0.10 were labeled as Grade-IV-like, cases with ET/WT at most 0.10 but TC/WT exceeding 0.20 were labeled as Grade-III-like, and all remaining cases were labeled as Grade-II-like. Table 3.1 provides detailed information about the dataset used.

3.2 Data pre-processing

For the field of medical image analysis, pre-processing is a highly crucial step as it allows for proper working with high-dimensional multimodal MRI data. particularly,

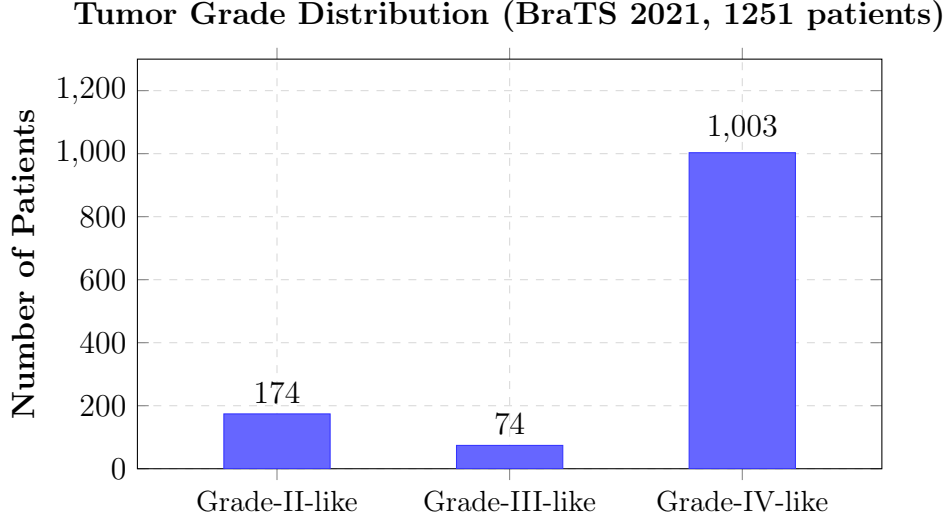


Figure 3-2: Distribution of derived tumor grade categories across the BraTS 2021 dataset. Grade-IV-like cases dominate at 80.2%, while Grade-III-like represents only 5.9%, creating severe class imbalance addressed through SMOTE oversampling in the classification pipeline.

Table 3.1: Summary of the BraTS 2021 dataset used in this study.

Attribute	Details
Dataset	BraTS 2021
Total patients	1,251 (training set with manual annotations)
Tumor type	Glioma (High-Grade and Low-Grade)
MRI modalities	T1, T1ce, T2, FLAIR
Volume dimensions	$240 \times 240 \times 155$ voxels
Voxel resolution	$1 \times 1 \times 1$ mm ³ (isotropic)
Segmentation labels	0 = Background, 1 = NCR, 2 = ED, 4 = ET
Whole Tumor (WT)	Labels {1, 2, 4}
Tumor Core (TC)	Labels {1, 4}
Enhancing Tumor (ET)	Label {4}

this stage allows for converting raw data into a structured and standardized format, which can be used efficiently by deep learning models. In the context of this work, for brain tumor segmentation and classification tasks, pre-processing ensured consistency across samples, numerical stability, and enhanced the model’s ability to explore meaningful anatomical features.

BraTS 2021 dataset, which is the main dataset used for this study, is already partially preprocessed. For example, MRI volumes are already co-registered, skull-

stripped, and resampled. However, in spite of these initial transformations, additional pre-processing work had to be done for this work. So this is why the MONAI framework was applied, and the following pre-processing steps were completed: spatial standardization, intensity normalization, label transformation, foreground extraction, patch-based sampling, and data augmentation [5].

3.2.1 Multimodal loading and channel construction

As was explained before, each data point in the BraTS dataset consists of four co-registered MRI modalities and one file that contains ground-truth segmentation. In the code each patient is converted into a dictionary, where the image field contains paths to t1, t1ce, t2, and flair and the label field points to the segmentation mask. This means that the network in this study does not process a single image but a four-channel 3D tensor, where every channel is related to one MRI sequence. Thus, the main point of loading all modalities for each case jointly is to expose the model to complementary views of the same anatomy.

3.2.2 Tensor Conversion

A fundamental data structure that is used in PyTorch is called a tensor, which is a multi-dimensional array and is specifically adjusted for numerical computations. In the code, the EnsureTyped transformation was utilized for converting MRI data into PyTorch tensors. This step is crucial because many operations that can be used in the training pipeline, such as convolution, backpropagation, and loss computation, are produced in such a way that they can work with tensors only. If data stays in the same original format (for instance, NumPy arrays or other types), then some operations can fail or run incorrectly due to the wrong format usage. This is why tensor conversion is important, as it ensures that the pipeline avoids potential mismatches in type and that operations can be run without errors.

Apart from avoiding these potential problems, Tensor conversion is also needed for computational efficiency. In the current study, PyTorch tensors can be trans-

ferred directly to GPU memory, which lets the model apply hardware acceleration during training process. As a result, this phase leads to computation time reduction, especially for 3D MRI volumes, which are the main data in this research. And most importantly, without tensor conversion, it would not be possible to utilize GPU functionalities to the full.

Thus, this step makes sure that the data is converted into a format that the model can understand, process, and apply to a GPU.

3.2.3 Orientation standardization

After the previous crucial part, the code applies another step that enforces a consistent anatomical orientation convention for all images and labels in the dataset. Actually, the BraTS dataset is already standardized initially and can not be called chaotic, but this step is introduced in order to act as safeguard that guarantees that every sample in the network is in the same axis convention. Practically, this stage is needed for preventing some inconsistencies in orientation as it is important for the network not to train and learn from arbitrary axis locations.

3.2.4 Voxel Spacing Normalization

Another important pre-processing step that is used for this work in voxel spacing normalization, which defines the physical dimensions of each voxel in the MRI volume. The BraTS dataset is generally resampled to a uniform resolution, and explicit spacing normalization in this work is applied for making sure that strict consistency among all the samples is preserved. In terms of this current research, all image scans have been resampled to a size of $1 \times 1 \times 1 \text{ mm}^3$ that guarantees that spatial features have the same scale across all patients, which is a crucial step for convolutional operations, which are applied in this work. Without this step, the model can learn representations inconsistently if they have some variations in voxel size.

Thus, this step is crucial for standardizing spatial resolution and for letting the model interpret anatomical structures properly.

3.2.5 Intensity Normalization

From the paper [2], it is mentioned that the mpMRI scans that are included in the BraTS 2021 challenge dataset were acquired from different scanners and from multiple institutions. Inherently, MRI scans' intensities are not standardized and can vary across diverse scanners and settings. This is why NormalizeIntensity intensity normalization was applied for addressing this problem. For the current work, normalization has been performed for channel-wise and for non-zero voxels only. This was made to make sure that the background regions do not distort the normalization process and do not affect the intensity distribution. Normalization was completed independently for each modality, so that the model can learn modality-specific features effectively.

Overall, the primary goal of this stage is to converge data input to a comparable intensity scale, which enhances numerical stability during the training process and does not allow some intensity ranges to dominate this exact learning phase. Without this transformation, the model would need to learn intensity scaling on its own, which would increase the complexity of the training part and lead to performance reduction.

3.2.6 Label Remapping

In the BraTS 2021 dataset provided segmentation labels are not indexed sequentially, as they contain the following label set: 0,1,2,4, where 0 shows the background part, 1 corresponds to necrotic and non-enhancing tumor core (NCR), 2 to edema (ED), and 4 to enhancing tumor (ET). However, deep learning models, which utilize loss functions such as Dice or cross-entropy, require class indices to be sequential and not chaotic or have gaps between them. Thus, in order to address this requirement, label remapping is applied here in the pre-processing pipeline.

All occurrences of labels with the value 4 are replaced with 3, which means that label set is contiguous now 0,1,2,3. The Lambdad part is seamlessly integrated into the MONAI transformation pipeline and is applied consistently with all samples.

This step is necessary for this research work, because its defined pipeline uses

loss functions such as DiceCELoss, which relies on one-hot encoding, which basically assumes that indices of classes are sequential. Otherwise, non-contiguous labels would lead to incorrect channel mappings and could potentially lead to invalid gradient updates. Importantly, this mapping does not change the meanings of these labels but rather replaces their numerical representation with a convenient one.

3.2.7 Foreground Cropping

MRI volumes that are used in the BraTS 2021 dataset contain large regions of background that do not actually have any meaning for the tumor segmentation or classification task. These regions mainly consist of zero-valued voxels that surround the brain. The problem is that if they are processed incorrectly, then this phase can significantly increase computational cost. That is why foreground cropping is added as a pre-processing step for this work by using the MONAI transformation CropForeground.

This implementation allows to identify smallest possible bounding box that contains all non-zero voxels in a particular scan and crops the image accordingly. An image is used as a source key, and the cropping operation ensures that relevant regions from a scan are preserved.

As explained above, this step affects computational efficiency, and this is the primary motivation for this phase implementation. By removing large empty regions, the input size is reduced, which leads to lower memory consumption and to an increased speed up for both training and inference. This is particularly helpful in the context of this work, because requirements grow rapidly with input size for 3D convolutional networks.

Apart from increased speed, foreground cropping affects the learning process by eliminating unnecessary details from scans and leaving important ones only. This potentially positively affected feature extraction and training behavior, as the model did not focus on unnecessary parts.

3.2.8 Spatial Padding

Following foreground cropping, the resulting volumes can have different dimensions due to differences in anatomical regions for each patient. However, deep learning models, including the one used in this work, require inputs of consistent size, especially during patch-based training strategies. To address this issue, a spatial padding step was applied with the help of the SpatialPadd transformation.

This transformation makes sure that all the cropped volumes have a minimum spatial size of $128 \times 128 \times 128$ voxels. Padding is applied symmetrically where necessary, and original image content is kept in the center within the volume. Without this important stage, cases with lower dimensions could lead to errors or require special handling, which would complicate the training pipeline. Standardization of spatial size through padding excludes this problem.

It also has to be mentioned that padding does not add any new information into the data. Instead, it simply adds neutral voxels around the existing content and preserves the integrity of the original image.

3.2.9 Patch-Based Sampling

For this work, 3D MRI volumes require high memory requirements, and training the model using entire volumes is computationally difficult. This is the reason why a patch-based sampling strategy is utilized in this study, where sub-volumes are extracted from the full image. This approach leads to reduction of memory usage and allows efficient batch processing, stable training on standard GPU hardware, and compatibility with the implemented pipeline.

Important note about this method is that patches are selected based on the distribution of segmentation labels. In the code, the specified ratio [1,2,2,6] assigns higher sampling priority to tumor regions, particularly to the enhancing tumor region. This part is crucial since tumor regions occupy a small portion of the volume, which can lead to class imbalance. And secondly, this phase does not let the model be dominated by background voxels, which can have zero information for the model. Final key note

is that 2 patches are extracted from every volume, which increases data diversity.

3.2.10 Data Augmentation

In the next pre-processing step, data augmentation is applied during training in order to improve the generalization capability of the model and reduce the risk of overfitting [33]. This transformation includes some variability into the dataset and allows the model to learn features by randomly flipping, rotating, and scaling scans along three different spatial axes.

Also, the pipeline uses a probabilistic application that ensures that each training sample is augmented differently. Specifically, this part includes the probabilistic chance that a particular data case will be augmented. This step increases dataset diversity as well, because not all the scans are augmented. Key moment about this data augmentation is that it is applied simultaneously to both images and labels, which preserves spatial alignment.

Thus, data augmentation exposed the model to a wider range of variations, and as a result, this part was applied to positively affect the model’s ability to generalize to unseen data.

3.3 Composite Loss Function Design

Design of the loss function for the segmentation work is carefully built in a way that addresses several important challenges, including class imbalance and the difficulty of segmenting properly small tumor subregions such as Enhancing Tumor (ET). In order to overcome this problem, a composite loss function is offered that integrates a combination of losses applied to different outputs. The proposed CNN-GNN architecture of this research produces three such outputs: fused prediction, output of a CNN branch, and the GNN-refined output. Thus, to effectively take into account all of these components, the total loss is defined as a weighted combination of segmentation losses that are applied to each output. Fourth part is added to this combination, which is an additional class-specific loss for the ET region, as this region provided

low results on initial settings.

Formally, in the code, this loss function is defined as:

$$L_{total} = L_{fused} + 0.2 \cdot L_{cnn} + 0.2 \cdot L_{gnn} + 0.4 \cdot L_{ET} \quad (3.1)$$

where:

- L_{fused} is the primary segmentation loss applied to the final fused output,
- L_{cnn} is the auxiliary loss applied to the CNN output,
- L_{gnn} is the auxiliary loss applied to the GNN-refined output,
- L_{ET} is a focal loss applied specifically to the Enhancing Tumor (ET) class.

Dice + Cross-Entropy Loss (Base Loss) The main segmentation loss that is applied in this work is DiceCELoss from the MONAI framework, which itself is a combination of two losses, namely Dice loss and Cross-Entropy loss. This loss is then applied to three outputs (fused, CNN, and GNN), which ensures that each component of the model is fully supervised.

The Dice loss component measures the spatial overlap between predicted segmentation and a true segmentation scan. It is particularly suitable for this work, because tumors in the data occupy only a small fraction of the whole volume [38]. The Dice coefficient itself follows the classical Dice similarity measure and is used in works such as V-Net [27].

$$Dice = \frac{2 \sum_i p_i t_i}{\sum_i p_i^2 + \sum_i t_i^2} \quad (3.2)$$

where p_i is the predicted probability at voxel i , and t_i is the corresponding ground truth label.

The Dice loss is then defined as [27]:

$$L_{Dice} = 1 - Dice \quad (3.3)$$

What Dice loss do is that it tries to maximize the overlap between the real region and a predicted one by the model, which makes it robust in relation to class imbalance. However, Dice loss alone does not fully penalize voxel-wise classification errors.

Cross-entropy loss is frequently used for classification, but it is sensitive to class imbalance and may underperform for small segmentation part like in this work, while Dice-based loss is more robust to this issue of class imbalance. However, Dice optimization can also produce badly calibrated overconfident predictions [48, 49]. This is the reason why combining both Dice and cross-entropy can be helpful for balancing their complementary strengths and weaknesses. The Cross-Entropy loss for multi-class segmentation is defined as [4]:

$$L_{CE} = - \sum_{c=1}^C t_c \log(p_c) \quad (3.4)$$

where C is the number of classes, t_c is the ground truth label for class c , and p_c is the predicted probability.

In this implementation, the combined loss is defined as:

$$L_{DiceCE} = 0.7 \cdot L_{Dice} + 0.3 \cdot L_{CE} \quad (3.5)$$

So, as it can be seen from the formula, weighting prioritizes region overlap (Dice) and puts less favor to voxel-level accuracy.

Additionally, this work applies softmax activation for modelling outputs and converts ground-truth labels into one-hot representations in order to support multi-class learning.

Deep Supervision via Auxiliary Losses Key aspect about the proposed loss design is that auxiliary supervision has been applied in this research for model’s intermediate outputs [13]. Specifically, CNN and GNN outputs are both supervised with the help of the same DiceCE loss as the final fused prediction. This strategy is being referred to as deep supervision, and it helps to guide feature learning in the early stages of the network and gradually improves gradient flow during training [20].

The auxiliary losses are weighted with a factor of 0.2:

$$L_{aux} = 0.2 \cdot L_{cnn} + 0.2 \cdot L_{gnn} \quad (3.6)$$

This ensures that both CNN and GNN learn meaningful representations even before fusion. Without the above supervision, these intermediate components may produce unstable or weak results, leading to worse final predictions. 0.2 weight is chosen because losses from 2 branches are highly important, but not as important as the final fused model.

Enhancing Tumor Focal Loss In addition to the above general segmentation losses, a specialized loss is introduced in order to improve segmentation part for the Enhancing Tumor (ET) region. This region had the smallest results during training, and an additional emphasis was placed to it. In glioma MRI, these enhancing tumor regions can be small, and their appearance can vary significantly from scan to scan, which makes their segmentation more difficult [21]. As a result, standard loss functions could have underrepresented this region’s importance and led to its weak results during the training process.

Thus, for addressing this issue, focal loss has been implemented in this research work specifically for the ET class only [22]. This loss is defined based on predicted probabilities for class 3 (ET) after softmax:

$$p = \text{softmax}(\text{logits})_{ET} \quad (3.7)$$

Ground truth is converted to a binary mask:

$$t = 1(y = ET) \quad (3.8)$$

Afterwards, focal loss is computed with the following formula:

$$L_{ET} = -(1 - p_t)^\gamma \log(p_t) \quad (3.9)$$

where:

$$p_t = p \cdot t + (1 - p) \cdot (1 - t) \quad (3.10)$$

In the above part, $\gamma = 2$ is used implicitly via the squared term. This part is implemented this way so that easily correctly predicted voxels have less contribution to the loss, while harder or wrong ones provide big contribution.

Finally, ET focal loss is added to the total loss and has a weight of 0.4.

$$L_{ET}^{weighted} = 0.4 \cdot L_{ET} \quad (3.11)$$

So, in the context of this work, it can be seen that huge emphasis is on the ET class only and the final training part gives this region an additional importance.

Final Loss Composition In summary, all loss components are combined together into one final total loss. By integrating Dice, Cross-Entropy, auxiliary supervision, and focal loss for ET, the model is able to learn both local details and global tumor structure.

$$L_{total} = L_{fused} + 0.2 \cdot L_{cnn} + 0.2 \cdot L_{gnn} + 0.4 \cdot L_{ET} \quad (3.12)$$

3.4 Proposed CNN–GNN Fusion Architecture

The proposed framework from this study is designed for brain tumor segmentation from BraTS 2021 MRI scans and is built around an idea that for proper tumor delineation detailed local feature extraction and global understanding are necessary. Multimodal brain MRI dataset, which is used for this study, consists of different modalities providing a diverse view of tumor-related tissue properties. For example, T1 and T1ce are helpful for structural and contrast-enhancing information, T2 is useful for fluid-related abnormalities, FLAIR can be used for edema and infiltrative tumor regions. Thus, the final model is built in a way, so that it can process all modalities jointly and learn volumetric representation from them.

At a high level the architecture combines a convolutional backbone with a graph-based contextual refinement module and also with a learnable fusion mechanism. The main purpose of this design is not to collect different model components together and assign them the same task, but rather to provide them with their own functional roles. For instance, the convolutional branch is responsible for extracting local features from the provided data and producing a dense segmentation map. After this, the graph-inspired branch works in information propagation across neighboring spatial locations to understand contextual situations. Finally, this fused architecture combines complementary outputs into a final prediction. In this sense, it can be stated that the whole pipeline is indeed designed as a hybrid local-contextual framework.

For the convolutional segmentation backbone it has been decided to use a 3D U-Net model, which is used as the primary segmentation hypothesis before contextual refinement. U-Net is able to capture coarse semantic context and fine spatial detail because it has an encoder-decoder structure with skip connections, which makes it a relevant backbone to use for the current task of medical image segmentation, where tumor boundaries as well as lesion regions must be preserved [11]. Even though the CNN branch is strong for local patterns, it can not work properly with distant parts of the volume. For addressing this limitation, the proposed framework applies a grid-based GNN-style refinement module, which does not construct a custom anatomical graph from brain regions. Instead, it works with a smaller 3D grid derived from the CNN output, where each location exchanges information with its neighbors, and as a result, nearby regions support each other. GNN’s role is to introduce a lightweight contextual reasoning stage that acts as an addition to the CNN output without replacing the underlying dense segmentation backbone.

After both branches produce their output, DARTS-inspired fusion block steps into the work. Darts is one of the central ideas proposed in this study. The main reason for its inclusion is that the model does not consider CNN branch to dominate the output, nor does it assume that GNN-refined branch should be more reliable. Instead, the Darts fusion block helps to choose 4 candidate fusion operators and searches which fusion strategy should contribute more and which ones should contribute less. The

first operator preserves the original CNN output (or logits, which is the model’s raw score), second operator keeps GNN-refined logits, third one computes their average and the fourth operator concatenates CNN and GNN logits together along the channel dimension and then applies a $1 \times 1 \times 1$ convolution, so that it can learn a projected mixed representation.

Overall, this mechanism is more flexible than choosing just one branch. For example, the CNN branch may preserve small local details sharp, while for some other regions, the GNN-based branch can provide better contextual consistency. By learning fusion weights, the model can adjust to the contribution of each operator mentioned above to the training data process and combine all of them in different proportions to the final fused output. The overall architecture is illustrated in Figure 3-3.

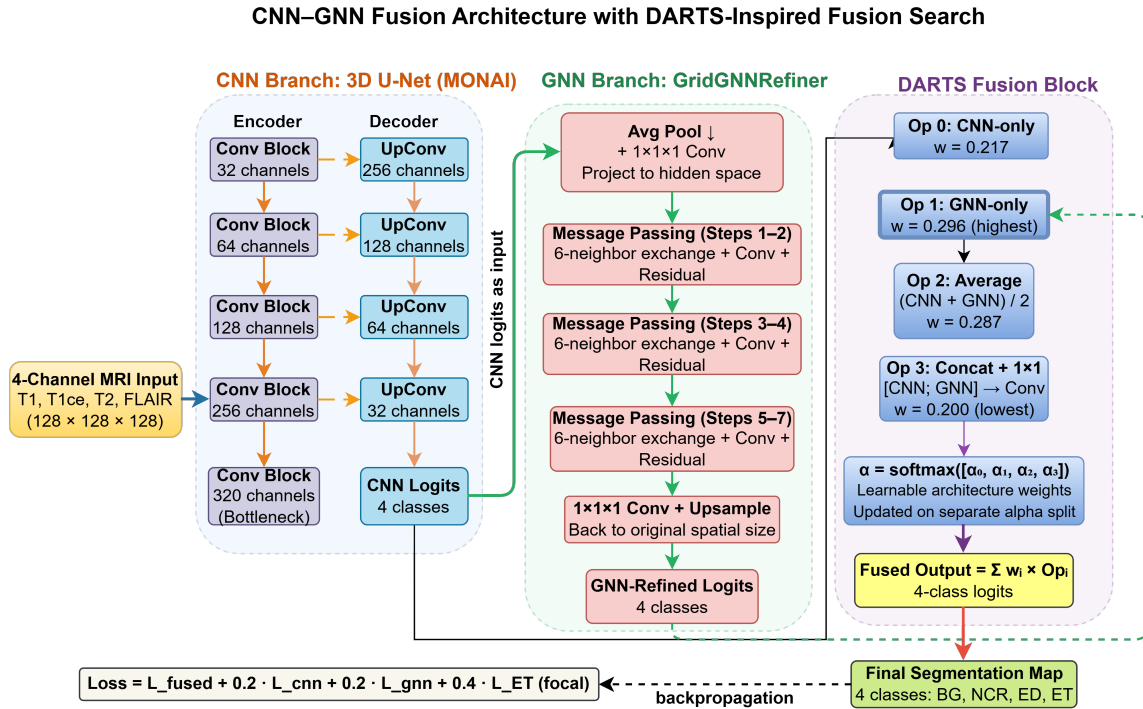


Figure 3-3: Architecture of the proposed CNN–GNN fusion model.

3.4.1 3D U-Net Backbone

This part discusses the CNN branch in detail. This convolutional part is implemented with the help of 3D U-Net architecture from the MONAI framework. U-Net is con-

sidered a widely used encoder-decoder network for medical image segmentation since it connects semantic information from the deep layers with spatial information from shallower layers by utilizing skip connections as well [36].

In the present implementation, this part is configured with:

- `in_channels` = 4 for the four MRI modalities,
- `out_channels` = 4 for the four segmentation classes after label remapping,
- channel widths (32, 64, 128, 256, 320),
- strides (2, 2, 2, 2),
- `num_res_units` = 2,
- dropout of 0.2.

For this part, encoder gradually makes input volumes smaller and learns more complex and more meaningful features from them. Then, the decoder increases this resolution again for producing full segmentation map for the 3D scan. During this reconstruction stage, skip connection passes information from earlier encoder layers to the decoder and helps it make voxel-level predictions more accurate.

3.4.2 Grid-Based GNN-Style Refinement Module

After the initial CNN backbone, the work applies a lightweight graph-inspired refinement module as well. Firstly, the CNN logits (raw outputs of the CNN model) are downsampled with the help of average pooling, which makes representations smaller and reduces costs for computation. After that, a $1 \times 1 \times 1$ convolution is being applied for projecting these pooled features to a hidden feature space.

Then the model performs several important steps. At every step in the 3D grid each location gets information from six nearby directions: left, right, up, down, forward, and backward [10]. Speaking in other words, every nearby position shares information with the others. After each location gets information from its close neighbors,

the model does not directly use this neighbor information. It passes it through convolutional layers, which transforms and improve this information. This new updated information returns back to the current feature representation, so the model preserves old context as well as a new one [12]. This step gets repeated several times, so that this information does not stay only between close neighbors but it spreads farther across the whole 3D scan. In the context of this work, this step is repeated 6 times.

Finally, these refined features are projected back to the class-logit space and get upsampled to the original spatial size. As a result, a prediction is formed, and it complements the original CNN output.

3.5 Tumor Grade Classification Pipeline

Segmentation was described above in high details, while the proposed framework is extended to perform tumor grade classification as well, and classification has a significant role in the provided pipeline. Classification methodology consists of 4 important stages: grade label derivation, feature extraction, class imbalance handling, and classifier training. The BraTS 2021 dataset does not have explicit tumor grade annotations, and grade categories are actually extracted from the ground-truth segmentation masks based on volumetric subregion ratios. For every patient case, volumes of the enhancing tumor (ET), tumor core (TC), and whole tumor (WT) regions are evaluated from the segmentation mask.

In this work, the trained segmentation model was used to generate a prediction via sliding-window inference, which resulted in a 28-dimensional feature vector extraction from the output. This feature vector is organized into 5 groups:

- **Pooled class probabilities (8 dimensions):** Average-pooled and max-pooled softmax probabilities across all four segmentation classes, which capture the overall class distribution in the predicted volume.
- **Region volume ratios (7 dimensions):** Normalized volumes of ET, TC, and WT regions, as well as derived ratios including ET/WT, TC/WT, ET/TC, and

an edema-only ratio $(WT-TC)/WT$.

- **Confidence statistics (6 dimensions):** Mean and maximum predicted probability for each of the three tumors regions (ET, TC, WT) that illustrate the model’s confidence in its predictions.
- **Prediction entropy (4 dimensions):** Mean entropy across the full volume, mean entropy within the tumor region, entropy standard deviation, and maximum entropy, capturing the spatial uncertainty of the segmentation output.
- **Spatial probability standard deviations (3 dimensions):** Standard deviation of predicted probabilities for ET, TC, and WT across spatial locations, reflecting the heterogeneity of the prediction.

Basically, this 28-dimensional vector set provides a compact summary of the tumor extracted from the segmentation output.

Another important issue that was considered in the classification part is handling class imbalance. For this task, SMOTE-inspired oversampling was used to generate synthetic minority-class training samples [6]. For each minority class, samples were generated till the point where the class count reached 50% of the majority class count, resulting in an approximate ratio: 1:1:2 for Grade-II, Grade-III, and Grade-IV classes, respectively. Secondly, class-weighted cross-entropy loss was employed during the training session, where the weight for each class was set inversely proportional to its frequency in the oversampled training data. Thus, SMOTE provides more examples for minority class, while the weighted loss made sure that misclassification of rare classes led to higher penalties.

The training was conducted with the Adam optimizer with a learning rate of 5×10^{-4} and weight decay of 10^{-4} [18]. A CosineAnnealingLR scheduler was applied over a maximum of 100 epochs [24].

3.6 Evaluation Protocol

For assessing the performance of the proposed CNN–GNN fusion framework comprehensively, a set of evaluation metrics was selected for both segmentation and classification tasks.

For the segmentation task, the primary metric used in this work is the Dice Similarity Coefficient (DSC), which quantifies the overlap between the model’s predicted segmentation mask and the ground-truth annotation [40]. While the Dice coefficient captures volumetric overlap, it does not directly show boundary accuracy. For this reason, the Hausdorff Distance at the 95th percentile (HD95) is also employed in this study. HD95 computes the 95th percentile of the set of distances between the surface points of predicted and real regions from scans [14]. Beyond these overlap and boundary metrics, voxel-level precision, recall, and F1-score have also been reported in this work for the segmentation task [29].

For the brain tumor classification task, this work employed overall accuracy and macro-averaged F1-score as the primary evaluation metrics. The former one reflects the proportion of correctly classified cases out of the total number of test samples, while the latter computes the F1-score independently for each class and then takes the unweighted average.

Final results are reported both per-fold and as mean values across all three folds in order to assess cross-fold consistency and overall model performance.

Chapter 4

Results and Discussion

This section provides experimental results that are obtained from the proposed CNN-GNN fusion framework for brain tumor classification and segmentation. There are multiple stages that evaluation was conducted on: initial segmentation training with DARTS-based fusion search, then Bayesian hyperparameter optimization was completed, nested cross-validation for proper segmentation assessment, and tumor grade classification derived from segmentation outputs. The whole experiment was built around the BraTS 2021 dataset, consisting of 1251 patient cases with 4 modalities (T1, T1ce, T2, FLAIR) and was evaluated with the help of standard clinical metrics, including Dice similarity coefficient, Hausdorff distance at the 95th percentile (HD95), precision, recall, and F1-score. Implementation was carried out with PyTorch and the MONAI medical imaging framework that is easy to debug [32]. The training part was performed through the Kaggle platform on GPU P100 and GPU T4. Table 4.1 provides experimental configuration for the proposed CNN-GNN fusion framework.

4.1 Initial Segmentation Training

The first phase of the experiment included fusion model training on 80/20 train-validation split on the full BraTS 2021 dataset. Settings included training on

Table 4.1: Experimental configuration of the proposed CNN–GNN fusion framework

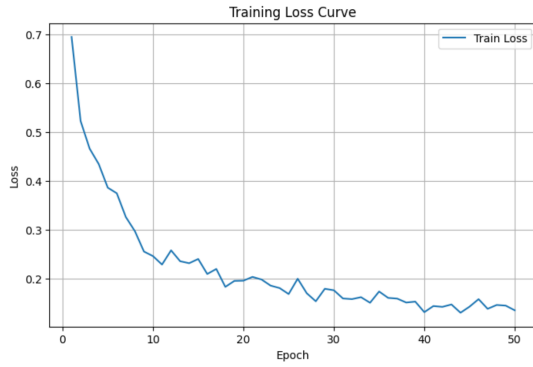
Parameter	Value
<i>Architecture</i>	
CNN backbone	3D U-Net (MONAI), encoder–decoder with skip connections
Input / output channels	4 modalities in, 4 classes out
Channel widths	32, 64, 128, 256, 320 (bottleneck)
Strides	2, 2, 2, 2
Residual units per level	2
GNN hidden dimension	64
GNN normalization	InstanceNorm3d + ReLU
GNN message-passing steps	7 (selected via Bayesian optimisation)
DARTS fusion operators	CNN-only, GNN-only, Average, Concat+1×1×1 Conv
<i>Segmentation Training (Nested CV)</i>	
Weight optimizer	AdamW, lr = 1.27×10^{-4} , weight_decay = 1×10^{-5}
Alpha optimizer	Adam, lr = 1×10^{-3} (DARTS fusion weights only)
LR scheduler	CosineAnnealingLR ($T_{\max} = 20$)
Dropout	0.198 (selected via Bayesian optimisation)
Epochs	20 epochs, 80 batches/epoch
Early stopping	Patience = 10 epochs (validation mean Dice)
Patch size	$128 \times 128 \times 128$ voxels
Patch sampling	RandCropByLabelClassesd, ratios = [1, 2, 2, 6], 2 patches per volume (ET-biased)
Data augmentation	Random flipping along all 3 axes ($p = 0.5$ each)
Mixed precision	AMP enabled
Sliding-window inference	overlap = 0.25, Gaussian weighting
<i>Loss Function</i>	
Primary loss (fused)	DiceCELoss: $0.7 \cdot L_{\text{Dice}} + 0.3 \cdot L_{\text{CE}}$
Auxiliary losses	$0.2 \cdot L_{\text{CNN}} + 0.2 \cdot L_{\text{GNN}}$ (deep supervision)
ET focal loss	$0.4 \cdot L_{\text{ET}}$ (focal loss, $\gamma = 2$, on ET class only)
Total loss	$L_{\text{total}} = L_{\text{fused}} + 0.2L_{\text{CNN}} + 0.2L_{\text{GNN}} + 0.4L_{\text{ET}}$
<i>Classification Training</i>	
Classifier architecture	4-layer MLP (128-128-64-3) with BatchNorm, ReLU, Dropout (0.3/0.3/0.2)
Feature vector	28-dimensional (region volumes, class probs, confidence, entropy, spatial std)
Class imbalance handling	SMOTE-inspired oversampling + class-weighted cross-entropy
Optimizer	Adam, lr = 5×10^{-4} , weight_decay = 1×10^{-4}
LR scheduler	CosineAnnealingLR
Max epochs / early stopping	100 epochs, patience = 25 (val mean F1)
Batch size	64
<i>Evaluation & Hardware</i>	
Evaluation	3-fold nested cross-validation, random_state = 42
Hyperparameter search	Optuna Bayesian optimisation, 5 trials × 5 epochs
Hardware	Kaggle GPU T4
Framework	PyTorch + MONAI

50 epochs with 100 training steps for every epoch by using EMA (Exponential Moving Average) model for validation assessment. Also, for this part, the DARTS strategy was employed, which updated weights for the network on the training split. Architecture fusion parameters (alpha) were updated during this phase as well.

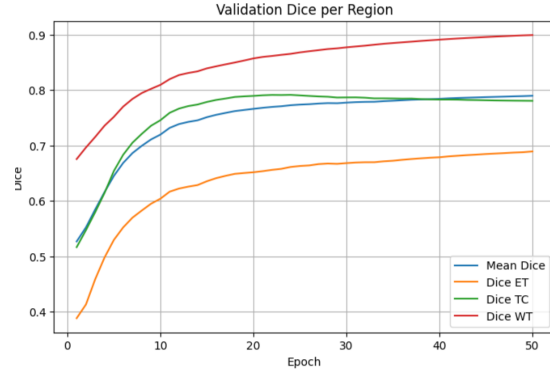
During the training part, all the evaluation metrics were improved. So, composite loss decreased steadily from 0.694 at epoch 1 to 0.136 at epoch 50, demonstrating gradual improvement throughout the full schedule. Mean validation dice improved as well from roughly 0.527 at epoch 1 to 0.790 in the last 50th epoch without signs of overfitting or hitting an early plateau. Validation Dice scores per-region reached the following results by the end of the session: 0.689 for Enhancing Tumor (ET), 0.781 for Tumor Core (TC), and 0.899 for Whole Tumor (WT). The HD95 also demonstrated improvement over training. For instance, at epoch 1, HD95 values were 50.55 mm, 47.23 mm, and 54.38 mm for ET, TC, and WT, respectively, while by epoch 50, these same metrics decreased substantially to 5.77 mm, 5.77 mm, and 10.94 mm, indicating that the model managed to learn to produce geometrically accurate tumor boundaries. The precision values at the final epoch were 0.772, 0.913, and 0.919 for ET, TC, and WT, respectively, while sensitivity values were 0.693, 0.732, and 0.890 for those same regions. This was the initial training step for brain tumor segmentation. As can be seen from the results, ET region showed low results; this is why further improvements were made to increase this result. Figure 4-1 presents the initial training results.

4.2 DARTS Fusion Architecture Analysis

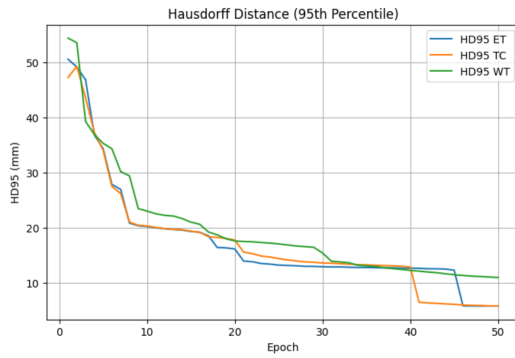
One of the main contributions of this research is the usage of a DARTS-inspired fusion mechanism that learns optimal combinations of CNN and GNN outputs. The model in this study produces four different opinions for every voxel, and



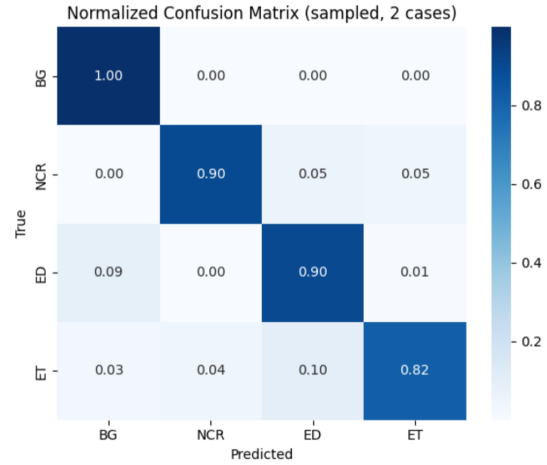
(a) Training loss curve



(b) Validation Dice per region



(c) HD95 per region



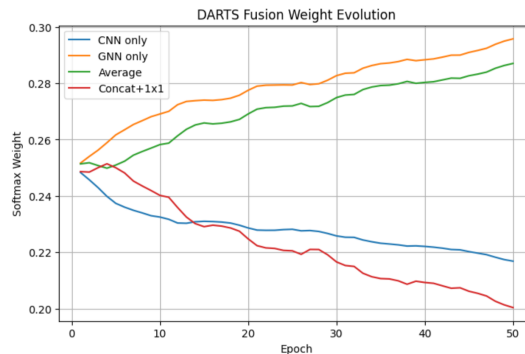
(d) Normalized confusion matrix

Figure 4-1: Initial segmentation training results over 50 epochs: (a) training loss convergence from 0.694 to 0.136, (b) validation Dice scores per tumor region reaching 0.899 (WT), 0.781 (TC), and 0.689 (ET), (c) HD95 improvement from above 50 mm to below 11 mm, (d) voxel-level confusion matrix with BG=1.00, NCR=0.90, ED=0.90, ET=0.82

DARTS does not just choose one of them; instead, it combines all of them with the help of learned weights. Four candidate operations that were selected for this study include: CNN-only output, GNN-only output, their arithmetic average, and a concatenation-based learned projection. The softmax-normalized architecture weights (alpha) evolve during the training part and then show which fusion strategy the model finds most helpful for the segmentation task.

At the beginning, all 4 fusion weights were approximately equal at 0.25 each and did not show any preference for any particular strategy. Differentiation emerged as the training part progressed. By the 50th epoch, weights converged to the following numbers: 0.217 for CNN-only, 0.296 for GNN-only, 0.287 for Average, and 0.200 for Concat+1×1, these results are shown in Figure 4-2.

From these results, several important observations can be drawn. Firstly, the GNN-only operator constantly received highest weights across the whole experiment, which suggests that graph-based contextual refinement apparently provided more beneficial information than the raw CNN output for the final prediction. Another interesting moment to note is that the average fusion operator had the second-highest weight results, which also illustrates the importance of a balanced combination of CNN and GNN outputs. And the final observation is related to the fact that 1×1 convolution did not capture the CNN-GNN complementarity as effectively as other proposed strategies.



(a) DARTS fusion weight evolution

Figure 4-2: DARTS fusion weights showing autonomous preference for GNN-only (0.296) over CNN-only (0.217)

4.3 Bayesian Hyperparameter Optimization

In order to find optimal training hyperparameters for the nested cross-validation phase more emphasis was put on the ET dice score. The Bayesian optimization stage was completed with the help of the Optuna framework [1]. Search space included three most crucial parameters: learning rate(how fast the model learns), dropout rate (how much the model randomly ignores during training to prevent memorization), and the number of GNN message-passing steps (how many times neighbors in the data share information). Overall, a total of 5 trials were conducted, where every trial trained the model on 100 patient cases for 5 epochs and evaluated on 30 cases by using ET Dice as the main optimization objective. Results for the Bayesian search are shown in Table 4.2. As can be seen from it, the best-performing configuration achieved an ET Dice of 0.657 with a learning rate of 1.27×10^{-4} , dropout of 0.198, and 7 GNN steps. This search revealed that moderate learning rates (around 10^{-4}) consistently outperformed both lower values (10^{-5} , which produced ET Dice of 0.605–0.627) and higher values as well. The optimal dropout was around 0.20, and GNN message-passing steps were equal to 7, which meant that deeper graph propagation was better for understanding contextual information.

Table 4.2: Bayesian hyperparameter optimization results.

Trial	Learning Rate	Dropout	GNN Steps	ET Dice
0	1.27×10^{-4}	0.198	7	0.657
1	1.06×10^{-4}	0.188	5	0.655
2	1.10×10^{-5}	0.250	8	0.605
3	1.50×10^{-5}	0.150	7	0.627
4	2.20×10^{-5}	0.211	4	0.644

These best hyperparameters were later used for the full nested cross-validation experiments, which are described in the following sections.

4.4 Nested Cross-Validation Segmentation Results

This section describes in details the results that were obtained during the second segmentation part, where three-fold nested cross-validation was performed using the hyperparameters identified by Optuna ($\text{lr} = 1.27 \times 10^{-4}$, $\text{dropout} = 0.198$, $\text{GNN steps} = 7$). For each outer fold, 1251 patient cases were divided into a training set of 601 cases, an alpha-update set of 66 cases for DARTS fusion weight optimization, a validation set of 167 cases for checkpoint selection, and a held-out test set of 60 cases that the model never encountered during training in any form. Each of those folds were trained for 20 epochs with 80 training batches per epoch, using CosineAnnealingLR scheduling, early stopping with a patience of 10 epochs, mixed-precision training, which speeds up training without losing model accuracy [26]. DARTS optimization was applied as well at every epoch, which led to updates of the fusion architecture weights on the dedicated 66-case alpha split. Eventually, all of these folds were chosen as representative examples, with plots shown below. All three folds in general completed a full 20 epochs without any triggered errors.

Table 4.3 summarizes the final test segmentation results across all three outer folds, and these results are visually confirmed by the outer fold bar chart shown in Figure 4-3 as well.

Table 4.3: Nested cross-validation segmentation results (Dice scores).

Fold	ET Dice	TC Dice	WT Dice	Mean Dice
0	0.765	0.830	0.888	0.828
1	0.731	0.782	0.866	0.793
2	0.779	0.846	0.883	0.836
Mean	0.758	0.819	0.879	0.819

The mean Dice scores across these 3 folds for each of the studied regions showed the following results: 0.758 for ET, 0.819 for TC, and 0.879 for WT, and an overall mean of 0.819. Highest Dice score across all folds was shown by the wt region, which encompasses the largest spatial extent and has the most dis-

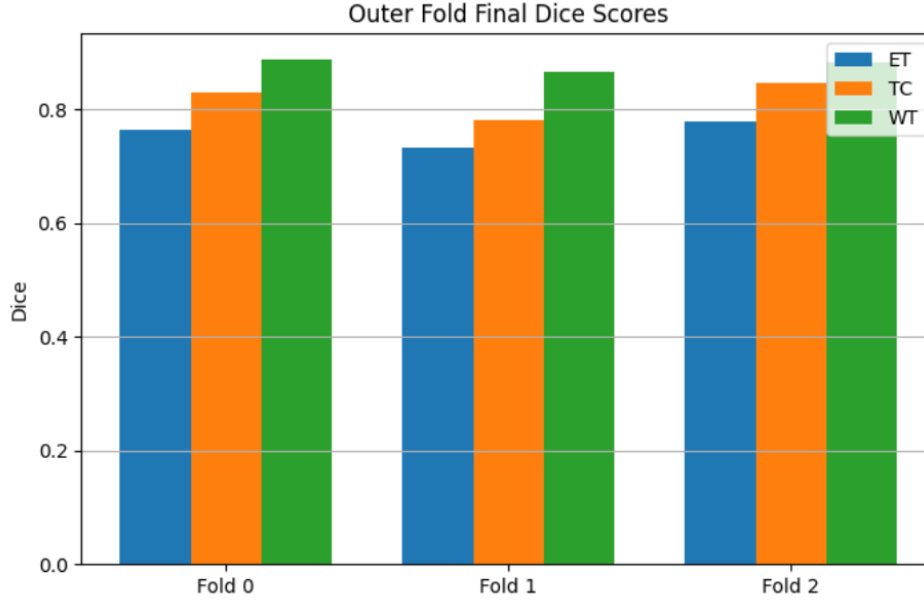


Figure 4-3: Final test Dice scores across three outer folds. All folds demonstrate consistent performance, with WT scores in the 0.866–0.888 range, TC between 0.782–0.846, and ET between 0.731–0.779.

tinctive boundaries among all data in the dataset. The ET region, which had the lowest results after the initial setup and had the smallest volume among all the respective regions, still showed lowest but clinically meaningful Dice scores. The TC region was between these 2 extremes. The HD95 values on the held-out test set were 4.71 mm for ET, 6.16 mm for TC, and 18.79 mm for WT. Precision values for Fold 0, for example, were 0.777 (ET), 0.831 (TC), and 0.881 (WT), while recall values were 0.800 (ET), 0.865 (TC), and 0.909 (WT). The corresponding F1-scores were 0.788, 0.848, and 0.894 for ET, TC, and WT, respectively. These balanced precision and recall values demonstrate that the model did not actually over-segment or under-segment for any region, which is a positive sign, since both false positives and false negatives can lead to undesirable consequences.

To provide a qualitative assessment of the segmentation quality, Figure 4-4 presents representative cases. The visualization displays the FLAIR modality at the axial slice with the tumor area, together with ground truth and predicted segmentation overlays, as well as a mismatch map highlighting a disagreement

region.

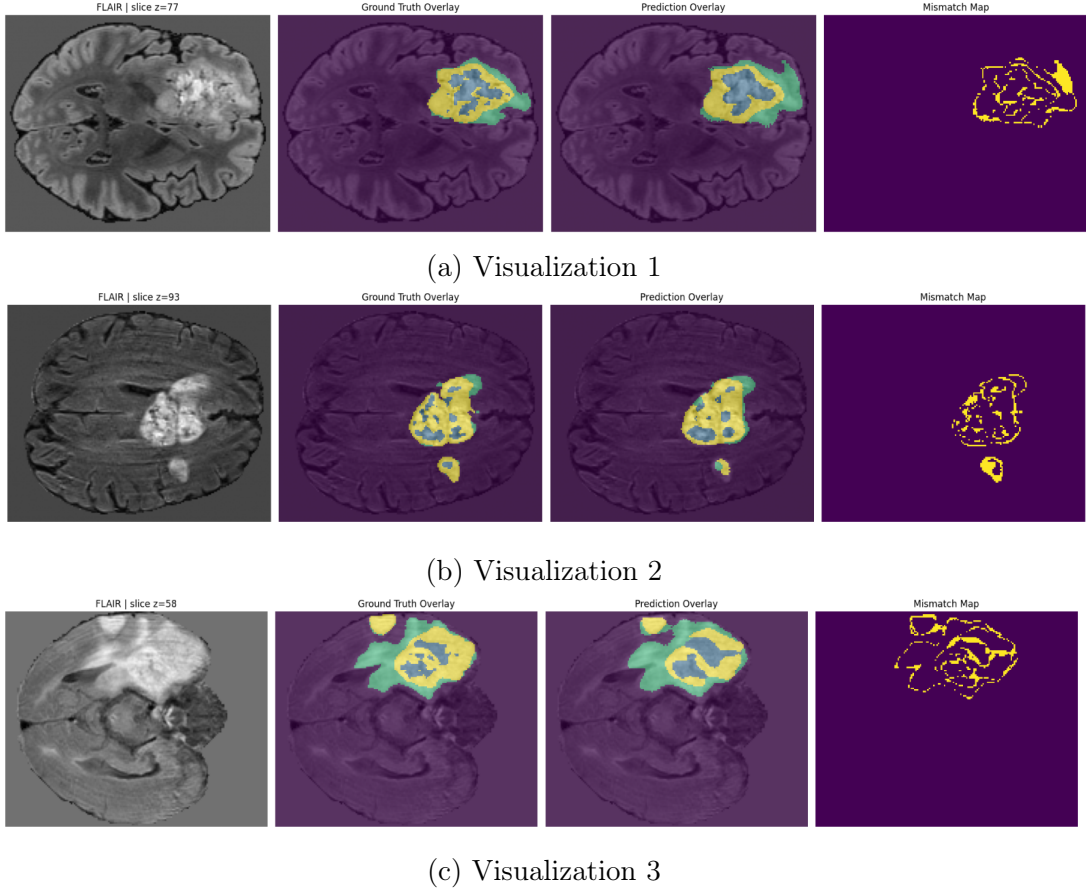


Figure 4-4: Qualitative segmentation results for representative cases. From left to right: FLAIR modality, ground truth segmentation overlay, model prediction overlay, and mismatch map showing regions of disagreement between prediction and ground truth.

The normalized voxel-level confusion matrices for Folds 0, 1, 2 (Figure 4-5) provide deeper insight into per-class segmentation behavior.

The DARTS fusion weight evolution during the stage of nested cross-validation (Figure 4-6) confirmed the findings that were previously described in the paper during the first initial phase. Across all three folds, the GNN-only operator received the highest final weight: 0.297 in Fold 0, 0.295 in Fold 1, and 0.282 in Fold 2. The Average operator received the second-highest weight (0.279, 0.283, and 0.272, respectively), while CNN-only (0.208, 0.212, 0.219) and Concat+1×1 (0.216, 0.210, 0.227) consistently received the lowest weights. These results show

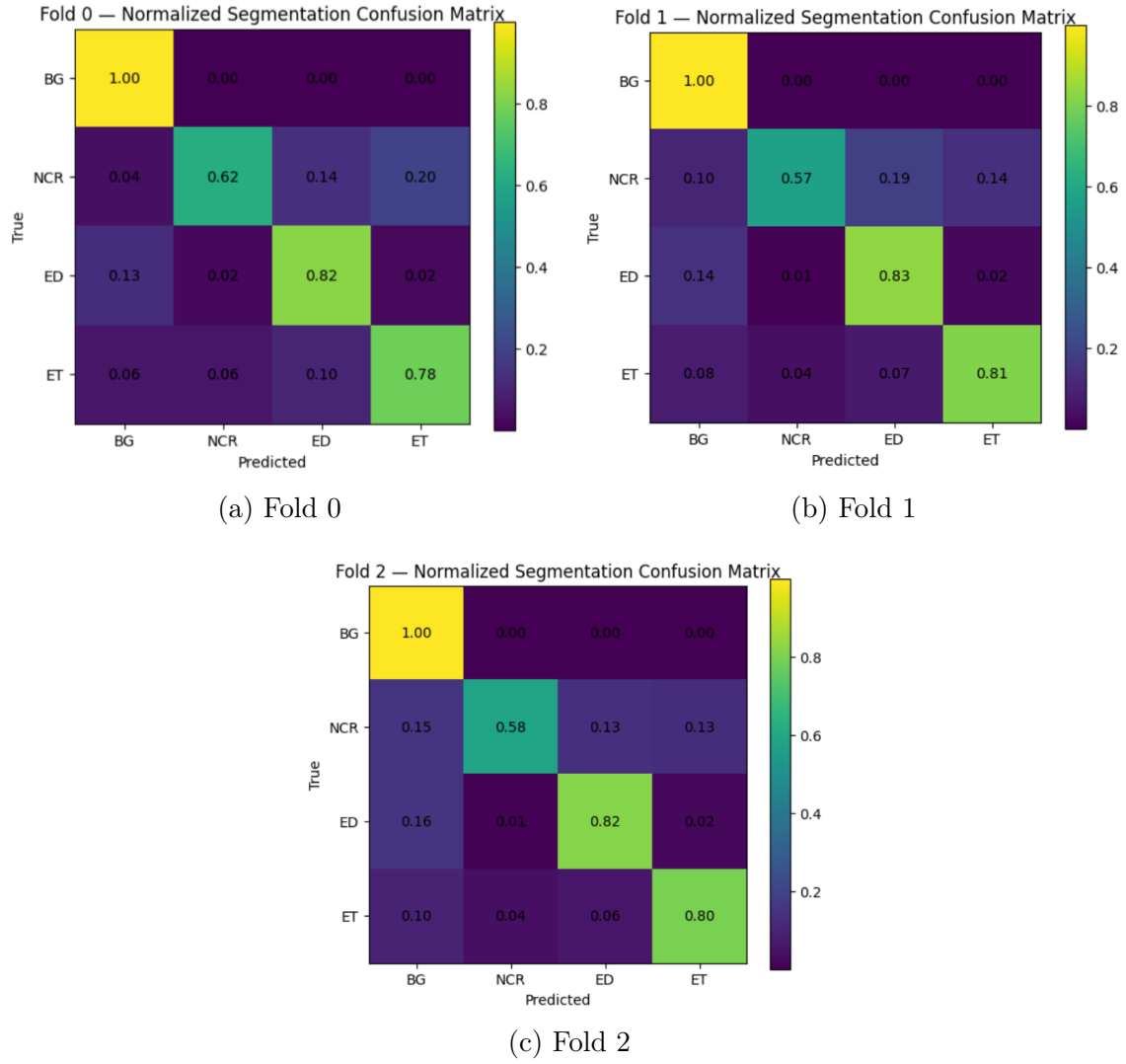


Figure 4-5: Normalized voxel-level segmentation confusion matrix for Folds 0, 1, 2

that the DARTS-discovered fusion strategy did not just randomly work with some particular data partition but reflect genuine architectural benefits of using DARTS in the paper. Table 4.4 provides nested cross-validation segmentation results.

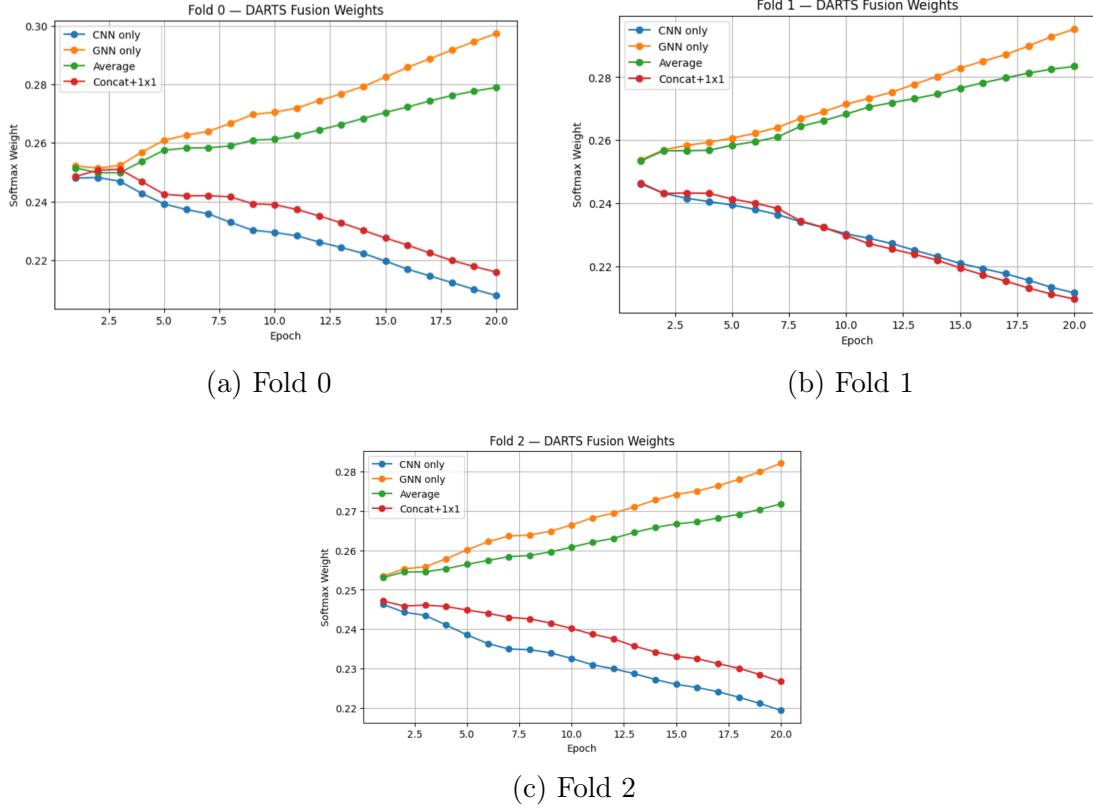


Figure 4-6: DARTS fusion weight evolution during nested cross-validation (Fold 0, 1, 2). The GNN-only operator converged to the highest weight, confirming the pattern observed in the initial training phase.

Table 4.4: Nested cross-validation segmentation results across three outer folds. Dice Similarity Coefficient (DSC) and HD95 are reported per tumor sub-region. HD95 values are mean averages across held-out test sets.

Fold	Dice Score				HD95 (mm)		
	ET	TC	WT	Mean	ET	TC	WT
Fold 0	0.765	0.830	0.888	0.828	—	—	—
Fold 1	0.731	0.782	0.866	0.793	—	—	—
Fold 2	0.779	0.846	0.883	0.836	—	—	—
Mean	0.758	0.819	0.879	0.819	4.71	6.16	18.79

4.5 Tumor Grade Classification

Beyond segmentation, this work also focused on the classification part as a downstream task. The BraTS 2021 dataset does not provide explicit tumor grades; for this reason, clinically meaningful grade categories were derived from the ground-truth segmentation masks, which are based on region ratios depending on volumes. These categories are defined as follows: Grade-IV-like tumors were identified when the ET-to-WT volume ratio exceeded 0.10, Grade-III-like tumors when the ET/WT ratio was at most 0.10 but the TC/WT ratio exceeded 0.20, and Grade-II-like tumors for all remaining cases.

For the feature extraction part, each patient case was processed through the trained segmentation model, where, eventually, a 28-dimensional feature vector was computed from the segmentation output. These features are numerical summaries extracted from the predicted segmentation, which describe the relative size of tumor regions, the model’s confidence in its predictions, and the level of uncertainty across the tumor area. Spoken simply, these features provide a compact description of both the predicted tumor structure as well as how reliable the segmentation part is. The raw training data were imbalanced, where Grade-IV cases constituted approximately 80% of the dataset (562 cases), Grade-II comprised roughly 14% (102 cases), and Grade-III only 6% (44 cases). In order to smooth this issue a little bit, SMOTE-inspired oversampling was utilized in the classification phase. This technique produces synthetic samples for small classes and makes the distribution more balanced [9]. Eventually, class counts were led to the following numbers: 281 Grade-II, 281 Grade-III, and 562 Grade-IV cases (ratio of 1:1:2). Apart from that, this work also applied Class-weighted cross-entropy loss to mitigate the imbalance during gradient computation even further.

The classification model itself consisted of a four-layer neural network with batch normalization, ReLU activations, and dropout regularization at 0.3 [15, 45]. Training was conducted using the Adam optimizer with a learning rate of

5×10^{-4} , CosineAnnealingLR scheduling over 100 maximum epochs, and mini-batch processing with a batch size of 64. Also, this research implemented early stopping with a patience of 25 epochs on the validation mean F1-score. Three-fold cross-validation structure from the segmentation evaluation was maintained, and it used the best segmentation checkpoint from each fold to extract features.

Table 4.5 presents the classification results across all three outer folds, and the outer fold comparison is visualized in Figure 4-7.

Table 4.5: Tumor grade classification results across outer folds.

Fold	Accuracy	Mean F1	Gr-II F1	Gr-III F1	Gr-IV F1
0	0.866	0.730	0.641	0.613	0.936
1	0.861	0.748	0.625	0.691	0.929
2	0.861	0.743	0.658	0.643	0.927
Mean	0.863	0.740	0.641	0.649	0.931

An overall mean accuracy of 86,3% was achieved in the classification part and a mean macro F1-score of 0.740. All the evaluation metrics were pretty similar across the folds, showing high cross-fold consistency. Accuracy values ranged from 0.861 to 0.866 and mean F1 values from 0.730 to 0.748, as shown in Figure 4-7. There is a gap between accuracy (86.3%) and mean F1 (74.0%), which reflects the class imbalance in the test data, where Grade-IV cases dominated at approximately 80%. Per-class analysis, shown in Figure 4-8, reveals a performance hierarchy. For instance, Grade-IV-like tumors were classified with the mean F1-score of 0.931, which is pretty expected since this class has the most distinctive morphological signature, and also has more training sample counts. Precision for Grade-IV was approximately 0.98 across folds, which indicates a small quantity of false positives, and a recall value was around 0.89, suggesting correct classification for the majority of high-grade cases, which indicated that the classifier captured most actual Grade-II cases but sometimes misclassified Grade-IV cases as Grade-II. Grade-III-like tumors had a mean F1-score of 0.649. Table 4.6 provides tumor grade classification results across outer folds.

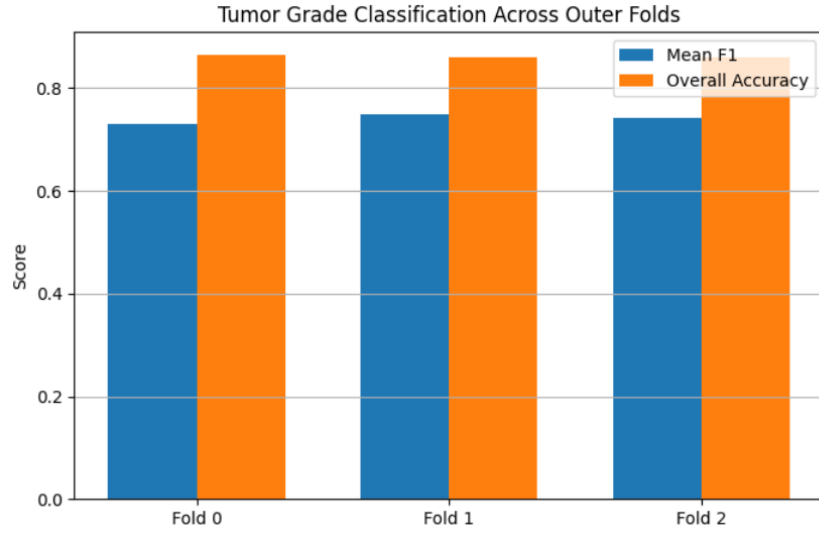


Figure 4-7: Tumor grade classification performance across three outer folds. Both overall accuracy (0.861–0.866) and mean F1-score (0.730–0.748) demonstrate consistent cross-fold performance.

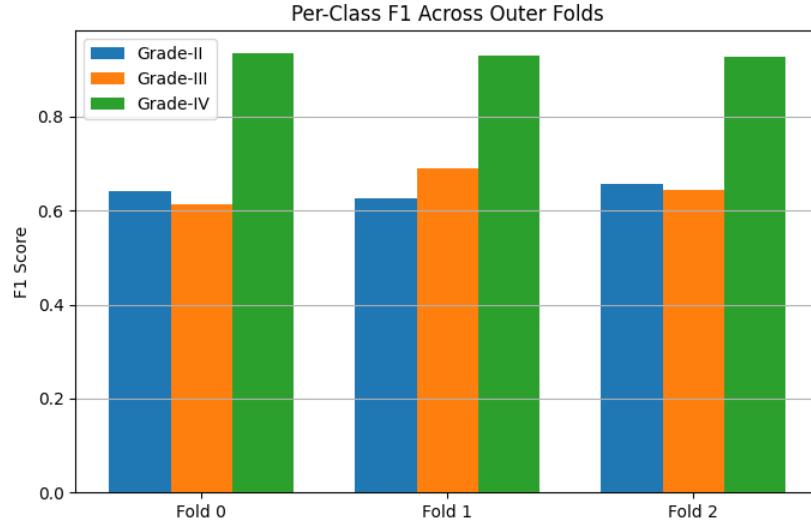


Figure 4-8: Per-class F1-scores across outer folds. Grade-IV achieves consistently high F1 (0.927–0.936), while Grade-II (0.625–0.658) and Grade-III (0.613–0.691) show lower but stable performance reflecting the difficulty of minority class classification.

Table 4.6: Tumor grade classification results across outer folds. Per-class F1-scores are shown for each derived grade category.

Fold	Accuracy	Mean F1	Grade-II F1	Grade-III F1	Grade-IV F1
Fold 0	0.866	0.730	0.641	0.613	0.936
Fold 1	0.861	0.748	0.625	0.691	0.929
Fold 2	0.861	0.743	0.658	0.643	0.927
Mean	0.863	0.740	0.641	0.649	0.931

4.6 Discussion

Experimental results from this work demonstrate that the proposed CNN-GNN fusion framework achieves competitive brain tumor segmentation performance (for instance, mean Dice 0.819) with strong cross-fold results, while also performing well at tumor grade classification (mean accuracy 86.3%, mean F1 0.740) from the same pipeline.

DARTS fusion analysis provided empirical evidence that graph-based contextual reasoning adds more value in brain tumor segmentation, where learned fusion weights favored GNN-refined output over raw CNN output. This finding was consistent at the initial segmentation stage and later with three nested CV folds. The Average fusion operator, which balances CNN and GNN equally, received the second-highest weight by the DARTS optimization.

Another interesting note from this work is the gap between validation and test results on Dice evaluation metrics. In the nested CV experiments, the best validation Dice values (approximately 0.70–0.80) were lower than the final test Dice values (approximately 0.79–0.84), which demonstrates overfitting prevention and reliability of reported test results.

Thirdly, the classification results that were derived in this paper depended on the segmentation model, since BraTS 2021 does not provide clinical tumor grade annotations. Thus, grade labels were derived from volumetric subregion ratios with the help of using clinically motivated thresholds. The strong Grade-IV F1-score (0.931) suggests a strong segmentation model’s capturing ability for

the ET/WT ratio, while the lower Grade-II and Grade-III F1-scores (0.641 and 0.649) also showed that the model had difficulties with subtle volumetric differences.

Overall, the proposed framework has its own advantages. Firstly, the grid-based GNN module operates on a downsampled version of the CNN logits, which adds minimal computational problems, while the DARTS fusion search naturally adapted to data characteristics. However, further research to improve the current situation should definitely be conducted. One of the limitations of this study was related to computational resources, because Kaggle did not allow the use of too many of its resources; otherwise, it would have been possible to train for more epochs and with larger batches, which potentially could have affected the results positively. For classification, a dataset with annotations could have helped as well for training on proper data. Table 4.7 provides an overall performance summary of the proposed CNN-GNN fusion framework.

Table 4.7: Overall performance summary of the proposed CNN-GNN fusion framework.

Task / Metric	ET	TC	WT	Mean
Segmentation				
Dice Score	0.758	0.819	0.879	0.819
HD95 (mm)	4.71	6.16	18.79	—
Dice range	0.731–0.779	0.782–0.846	0.866–0.888	—
Classification				
Accuracy	0.863			
Macro F1	0.740			
DARTS Fusion Weights (mean across folds)				
CNN-only operator	≈ 0.213			
GNN-only operator	≈ 0.291 (highest weight)			
Average operator	≈ 0.278			
Concat+1×1	≈ 0.218			

Chapter 5

Conclusion

This thesis presented a hybrid framework for multimodal brain tumor analysis, which integrated a 3D U-Net, a grid-based GNN refinement module, and a DARTS-inspired fusion mechanism for learning both local and global contextual information. In addition to the segmentation part, this paper also extended the pipeline toward tumor grade classification, where grade categories were derived from segmentation characteristics. This design allowed this research to address both dense voxel-level prediction and case-level tumor grading within a single framework.

Overall, the experimentation results demonstrated that the proposed approach provided effective and robust results. The model achieved mean Dice scores of 0.758 for ET, 0.819 for TC, and 0.879 for WT, with an overall mean Dice of 0.819 across outer folds for the segmentation results. The DARTS fusion weights assigned the highest importance to the GNN-only branch, which meant that graph-based contextual reasoning had the most meaningful contribution. Regarding the classification part, the model achieved a mean accuracy of 86.3% and a mean macro F1-score of 0.740. The strongest results were concluded for Grade-IV-like tumors, while minority classes remained more difficult for classification. CNN-GNN fusion provided consistent results for both the brain tumor segmentation and classification parts.

5.1 Key Findings

Objective 1 – Develop a hybrid CNN–GNN architecture with Learnable Architecture Search.

This objective has been addressed in the paper by combining a 3D U-Net with a grid-based GNN-style refinement module, connected through a DARTS fusion block. There were not signs of overfitting and overall the model showed strong results.

Objective 2 – Empirically determine whether graph-based contextual reasoning meaningfully contributes to brain tumor segmentation.

The DARTS fusion analysis provided clear evidence that the GNN-refined branch is preferred by the model. The GNN-only operator received the highest weight across the whole training process.

Objective 3 – Extend the segmentation framework to perform tumor grade classification within a single pipeline.

Tumor grade categories were derived from segmentation outputs using clinically motivated volumetric ratios and a four-layer classifier was eventually trained on a 28-dimensional feature vector extracted per case from the segmentation part.

5.2 Future Work

Results obtained in this thesis are encouraging, however some limitations remain and few directions need to be explored for the future work:

1. **Computational scaling.** Experiments in this work were conducted with Kaggle GPU resources, which have their own limitations. For detailed training, it is advised to use more powerful resources.
2. **External validation across datasets.** Current study worked with BraTS 2021 dataset only. It would be a good addition to explore the work across

many other different datasets in order to improve current results.

3. **Clinically annotated tumor grades.** Grade categories in this thesis have been derived from segmentation outputs based on volumetric sub-region ratios. Future works would benefit from the use of explicit clinical grade labels in order to validate proposed classification pipeline against ground-truth clinical labels.

Bibliography

- [1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631, 2019.
- [2] Ujjwal Baid, Satyam Ghodasara, Suyash Mohan, Michel Bilello, Evan Calabrese, Errol Colak, Keyvan Farahani, Jayashree Kalpathy-Cramer, Felipe C Kitamura, Sarthak Pati, et al. The rsna-asnr-miccai brats 2021 benchmark on brain tumor segmentation and radiogenomic classification. *arXiv preprint arXiv:2107.02314*, 2021.
- [3] Spyridon Bakas, Hamed Akbari, Aristeidis Sotiras, Michel Bilello, Martin Rozycki, Justin S Kirby, John B Freymann, Keyvan Farahani, and Christos Davatzikos. Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features. *Scientific data*, 4(1):170117, 2017.
- [4] Christopher M Bishop and Hugh Bishop. *Deep learning: Foundations and concepts*. Springer Nature, 2023.
- [5] M Jorge Cardoso, Wenqi Li, Richard Brown, Nic Ma, Eric Kerfoot, Yiheng Wang, Benjamin Murrey, Andriy Myronenko, Can Zhao, Dong Yang, et al. Monai: An open-source framework for deep learning in healthcare. *arXiv preprint arXiv:2211.02701*, 2022.
- [6] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.
- [7] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016.
- [8] Yi Ding, Wei Zheng, Ji Geng, Zhen Qin, Kim-Kwang Raymond Choo, Zhiguang Qin, and Xiaolin Hou. Mvufusfra: A multi-view dynamic fusion framework for multimodal brain tumor segmentation. *IEEE Journal of Biomedical and Health Informatics*, 26(4):1570–1581, 2021.
- [9] Alberto Fernández, Salvador Garcia, Francisco Herrera, and Nitesh V Chawla. Smote for learning from imbalanced data: progress and chal-

- lenges, marking the 15-year anniversary. *Journal of artificial intelligence research*, 61:863–905, 2018.
- [10] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. Pmlr, 2017.
 - [11] Ali Hatamizadeh, Yucheng Tang, Vishwesh Nath, Dong Yang, Andriy Myronenko, Bennett Landman, Holger R Roth, and Daguang Xu. Unetr: Transformers for 3d medical image segmentation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 574–584, 2022.
 - [12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
 - [13] Theophraste Henry, Alexandre Carré, Marvin Lerousseau, Théo Estienne, Charlotte Robert, Nikos Paragios, and Eric Deutsch. Brain tumor segmentation with self-ensembled, deeply-supervised 3d u-net neural networks: a brats 2020 challenge solution. In *International MICCAI Brainlesion Workshop*, pages 327–339. Springer, 2020.
 - [14] Daniel P Huttenlocher, Gregory A. Klanderman, and William J Rucklidge. Comparing images using the hausdorff distance. *IEEE Transactions on pattern analysis and machine intelligence*, 15(9):850–863, 2002.
 - [15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.
 - [16] Fabian Isensee, Paul F Jäger, Simon AA Kohl, Jens Petersen, and Klaus H Maier-Hein. Automated design of deep learning methods for biomedical image segmentation. *arXiv preprint arXiv:1904.08128*, 2019.
 - [17] Yun Jiang, Yuan Zhang, Xin Lin, Jinkun Dong, Tongtong Cheng, and Jing Liang. Swinbts: A method for 3d multimodal brain tumor segmentation using swin transformer. *Brain sciences*, 12(6):797, 2022.
 - [18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
 - [19] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.
 - [20] Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu. Deeply-supervised nets. In *Artificial intelligence and statistics*, pages 562–570. Pmlr, 2015.
 - [21] Hongwei Bran Li, Gian Marco Conte, Qingqiao Hu, Syed Muhammad Anwar, Florian Kofler, Ivan Ezhov, Koen van Leemput, Marie Piraud, Maria Diaz, Byrone Cole, et al. The brain tumor segmentation (brats) challenge 2023: Brain mr image synthesis for tumor segmentation (brasyn). *ArXiv*, pages arXiv–2305, 2024.

- [22] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.
- [23] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018.
- [24] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.
- [25] Qihang Ma, Siyuan Zhou, Chengye Li, Feng Liu, Yan Liu, Mingzheng Hou, and Yi Zhang. Dgrunit: Dual graph reasoning unit for brain tumor segmentation. *Computers in biology and medicine*, 149:106079, 2022.
- [26] Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. Mixed precision training. *arXiv preprint arXiv:1710.03740*, 2017.
- [27] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. Ieee, 2016.
- [28] Shaik Mohammed Huzaiifa et al. Princiresnet brain tumor classification technique for multimodal input-level fusion network. *International Journal of Computing and Digital Systems*, 15(1):1075–1089, 2024.
- [29] Dominik Müller, Iñaki Soto-Rey, and Frank Kramer. Towards a guideline for evaluation metrics in medical image segmentation. *BMC research notes*, 15(1):210, 2022.
- [30] Andriy Myronenko. 3d mri brain tumor segmentation using autoencoder regularization. In *International MICCAI brainlesion workshop*, pages 311–320. Springer, 2018.
- [31] Otung John Peter Odong, Mohammed Abo-Zahhad, and Moataz Abdelwahab. An integrated deep convolutional neural networks framework for the automatic segmentation and grading of glioma tumors using multimodal mri scans. *Discover Applied Sciences*, 7(8):929, 2025.
- [32] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.
- [33] Arjan JM Plompen, O Cabellos, Cyrille de Saint Jean, Michael Fleming, Alejandro Algora, Maurizio Angelone, Pascal Archier, Eric Bauge, Olivier Bersillon, A Blokhin, et al. The joint evaluated fission and fusion nuclear data library, jeff-3.3. *The European Physical Journal A*, 56(7):181, 2020.
- [34] Mackenzie Price, Corey Ballard, Jacob Benedetti, Carol Kruchko, Jill S. Barnholtz-Sloan, and Quinn T. Ostrom. CBTRUS statistical report: Pri-

mary brain and other central nervous system tumors diagnosed in the United States in 2018–2022. *Neuro-Oncology*, 2025.

- [35] M Ravinder, Garima Saluja, Sarah Allabun, Mohammed S Alqahtani, Mohamed Abbas, Manal Othman, and Ben Othman Soufiene. Enhanced brain tumor classification using graph convolutional neural network architecture. *Scientific Reports*, 13(1):14938, 2023.
- [36] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [37] Camillo Saueressig, Adam Berkley, Reshma Munbodh, and Ritambhara Singh. A joint graph and image convolution network for automatic brain tumor segmentation. In *International MICCAI Brainlesion Workshop*, pages 356–365. Springer, 2021.
- [38] Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *International Workshop on Deep Learning in Medical Image Analysis*, pages 240–248. Springer, 2017.
- [39] Nur Suriza Syazwany, Ju-Hyeon Nam, and Sang-Chul Lee. Mm-bifpn: multi-modality fusion network with bi-fpn for mri brain tumor segmentation. *IEEE Access*, 9:160708–160720, 2021.
- [40] Abdel Aziz Taha and Allan Hanbury. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. *BMC medical imaging*, 15(1):29, 2015.
- [41] Weiwei Tian, Dengwang Li, Mengyu Lv, and Pu Huang. Axial attention convolutional neural network for brain tumor segmentation with multi-modality mri scans. *Brain Sciences*, 13(1), 2023.
- [42] M Padma Usha, G Kannan, and M Ramamoorthy. Multimodal brain tumor classification using convolutional tumnet architecture. *Behavioural Neurology*, 2024(1):4678554, 2024.
- [43] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.
- [44] Wenxuan Wang, Chen Chen, Meng Ding, Hong Yu, Sen Zha, and Jiangyun Li. Transbts: Multimodal brain tumor segmentation using transformer. In *International conference on medical image computing and computer-assisted intervention*, pages 109–119. Springer, 2021.
- [45] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.
- [46] Chengdong Yan, Jurong Ding, Hui Zhang, Ke Tong, Bo Hua, and Shaolong Shi. Seresu-net for multimodal brain tumor segmentation. *IEEE Access*, 10:117033–117044, 2022.

- [47] Hengyi Yang, Tao Zhou, Yi Zhou, Yizhe Zhang, and Huazhu Fu. Flexible fusion network for multi-modal brain tumor segmentation. *IEEE Journal of Biomedical and Health Informatics*, 27(7):3349–3359, 2023.
- [48] Michael Yeung, Leonardo Rundo, Yang Nan, Evis Sala, Carola-Bibiane Schönlieb, and Guang Yang. Calibrating the dice loss to handle neural network overconfidence for biomedical image segmentation. *Journal of Digital Imaging*, 36(2):739–752, 2023.
- [49] Michael Yeung, Evis Sala, Carola-Bibiane Schönlieb, and Leonardo Rundo. Unified focal loss: Generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation. *Computerized Medical Imaging and Graphics*, 95:102026, 2022.
- [50] Junting Zhao, Zhaoxu Xing, Zhihao Chen, Liang Wan, Tong Han, Huazhu Fu, and Lei Zhu. Uncertainty-aware multi-dimensional mutual learning for brain and brain tumor segmentation. *IEEE Journal of Biomedical and Health Informatics*, 27(9):4362–4372, 2023.
- [51] Liang Zhao, Jiajun Ma, Yu Shao, Chaoran Jia, Jingyuan Zhao, and Hong Yuan. Mm-unet: A multimodality brain tumor segmentation network in mri images. *Frontiers in Oncology*, Volume 12 - 2022, 2022.