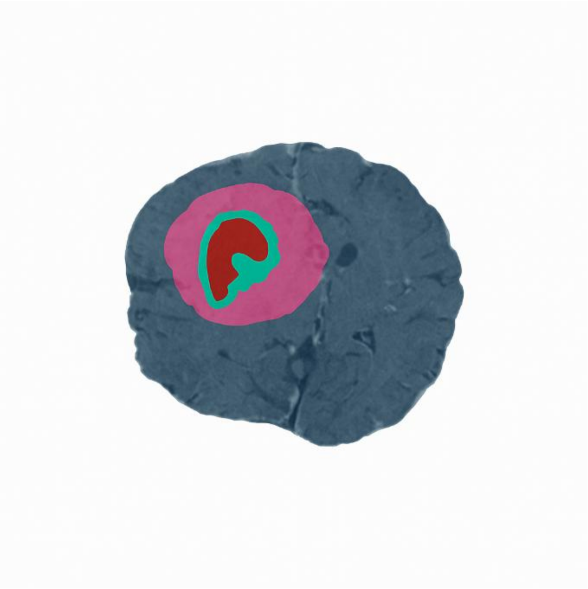


Artificial intelligence applied to brain tumor classification in MRI scans

ANA FILIPA FERREIRA RIBEIRO
(Licenciada em Engenharia Química)

Dissertação para obtenção do grau de Mestre em Engenharia Biomédica

Orientadores:
Doutora Maria Margarida do Carmo Pinto Ribeiro
Doutor Fernando Paulo Neves da Fonseca Cardoso Carreira

Júri:
Presidente: Doutor António Jorge Duarte de Castro Silvestre
Vogais:
Doutor Rafael Neto Henriques
Doutor Fernando Paulo Neves da Fonseca Cardoso Carreira

Setembro 2025

Artificial intelligence applied to brain tumor classification in MRI scans

ANA FILIPA FERREIRA RIBEIRO
(Licenciada em Engenharia Química)

Dissertação para obtenção do grau de Mestre em Engenharia Biomédica

Orientadores:
Doutora Maria Margarida do Carmo Pinto Ribeiro
Doutor Fernando Paulo Neves da Fonseca Cardoso Carreira

Júri:
Presidente: Doutor António Jorge Duarte de Castro Silvestre
Vogais:
Doutor Rafael Neto Henriques
Doutor Fernando Paulo Neves da Fonseca Cardoso Carreira

Setembro 2025

Agradecimentos

Este trabalho não teria sido possível sem o apoio de várias pessoas, às quais deixo aqui o meu reconhecimento.

Agradeço aos meus orientadores, Margarida Ribeiro e Fernando Carreira, pela orientação e disponibilidade. Mas, acima de tudo, pela compreensão e empatia que me permitiram concluir este trabalho após desafios pessoais.

Ao meu tio Fred, com quem vivi por perto a luta ingrata desta doença.

Ao meu querido primo André, que me fez sonhar em salvar o mundo, salvar-lhe a ele, desta doença por tirarem cedo demais aqueles que adoramos.

À minha filha Helena, que no início deste trabalho a sua existência estava longe de ser real, mas sem a qual nunca poderia ter acabado.

À minha mãe, pela incansável ajuda com a neta.

Ao Martim, meu marido e grande pilar, que não me deixa desistir quando quero e acredita em mim quando não consigo.

Ao desenvolver o trabalho de pesquisa e experimentação patentes nesta tese, que se insere num contexto de tecnológico, desde cedo foi para mim evidente que a Inteligência Artificial (IA) poderia e deveria ser utilizada como ferramenta para atingir melhores resultados e aperfeiçoar os caminhos do meu raciocínio e investigação. Também me interessou, de forma empírica, realizar um exame às próprias capacidades interpretativas da IA, pois esse é precisamente um dos temas sobre os quais incide este trabalho, sendo fundamental conhecer os limites e os potenciais da IA enquanto ferramenta analítica. Desta forma, integrei o uso de ferramentas de inteligência artificial generativa disponíveis no mercado, fazendo-o de forma crítica e subordinada aos meus próprios processos de investigação, escrita e pensamento. O uso destas ferramentas traduziu-se em exemplos como o cross-check de raciocínios e procedimentos teóricos por mim implementados, assim como testes à funcionalidade de algoritmos e código que desenvolvi, ou a análise de amostras de dados.

Esta abordagem teve um carácter meramente consultivo, contribuindo para a minha

observação e aprendizagem, por vezes ajudando-me a consolidar e clarificar o meu raciocínio. Ao mesmo tempo, num processo de trabalho solitário, a IA foi por vezes uma forma rápida e eficaz de verificar a solidez de certas soluções ou conclusões teóricas a que cheguei, num caminho que se faz muitas vezes por processos de tentativa e erro. Por considerar que o uso responsável de IA pode contribuir para o trabalho de investigação, desde que regulado de forma crítica e consciente, decidi em plena consciência integrar esta ferramenta, estando convencida de que não só não houve prejuízo na integridade intelectual e no rigor científico do meu trabalho, como também de que o uso da IA propiciou descobertas, correlações e confirmações que, de outra forma, não teriam estado ao meu alcance.

Statement of integrity

I declare that this dissertation / project work / internship report is the result of my personal and independent research. Its content is original, and all sources listed in the bibliographic references were consulted and are duly mentioned in the text. I further declare that all scientific and technical references relevant to the development of the work are duly cited and included in the bibliographic references.

The author

Ana Ribeiro

Lisbon, October 28th

Inteligência Artificial aplicada à classificação de tumores cerebrais em imagens de RM

Resumo

Os gliomas são heterogéneos, difíceis de delinear e consomem tempo clínico. Este trabalho privilegia o pré-processamento em RM e um U-Net 2D, comparando pipelines que melhoram a qualidade das imagens, em vez de otimizar um sistema de segmentação completo sob limitações de recursos computacionais (GPU) e de tempo. Trabalhou-se com dados multimodais do BraTS 2023 (FLAIR, T1Gd e T2), excluindo a sequência T1 nativa por ser considerada redundante na prática. Foram comparados três métodos. O Método 1 aplicou normalização min–max por volume e recorte geométrico. O Método 2 substituiu por *z-score* (apenas nos voxels intracranianos) e centragem guiada pela intensidade do T1Gd. O Método 3 acrescentou correção de bias (N4) e localização sem rótulo baseada no FLAIR (Otsu + componente conexa maior), evitando *label leakage* em teste/validação. Em todos os casos, treinou-se um U-Net 2D (MONAI) com três canais e quatro classes, usando perda *Dice* + *cross-entropy* e *early stopping*. A avaliação seguiu as regiões BraTS (ET, TC, WT) e métricas padrão (*Dice*, *IoU*, sensibilidade, precisão e especificidade).

Os resultados mostram ganhos consistentes do Método 3 face aos restantes. No conjunto de teste, o Método 3 obteve *Dice* médio de 0,626 (ET), 0,682 (TC) e 0,844 (WT), superando o Método 2 em +0,073, +0,153 e +0,072, respetivamente. As melhorias são mais expressivas no núcleo tumoral (TC), a região mais desafiante. Conclui-se que a normalização intracraniana, correção de bias e centragem guiada por FLAIR aumentam a fiabilidade das entradas e a estabilidade do modelo 2D, podendo fornecer uma base sólida para o desenvolvimento de futuras extensões 3D e validação externa.

Palavras-chave: Gliomas; Ressonância Magnética; Pré-processamento.

Artificial intelligence applied to brain tumor classification in MRI scans.

Abstract

Gliomas are heterogeneous, difficult to delineate, and consume significant clinical time. This work prioritizes MRI preprocessing and a simple 2D U-Net, comparing pipelines that improve image quality instead of optimizing a full segmentation system under GPU/time constraints. We used multimodal BraTS 2023 data (FLAIR, T1Gd, and T2), excluding native T1 due to practical redundancy. Three methods were compared. Method 1 applied per-volume min–max normalization and geometric-center cropping. Method 2 replaced this with z-score normalization (computed only on intracranial/non-zero voxels) and centering guided by T1Gd intensity. Method 3 added bias-field correction (N4) and label-free localization based on FLAIR (Otsu + largest connected component), avoiding label leakage in testing/validation. In all cases, a simple 2D U-Net (MONAI) with three input channels and four classes was trained using Dice + cross-entropy loss and early stopping. Evaluation followed the BraTS subregions (ET, TC, WT) and standard metrics (Dice, IoU, sensitivity, precision, and specificity).

Results show consistent gains for Method 3 over the others. On the test set, Method 3 achieved mean Dice of 0.626 (ET), 0.682 (TC), and 0.844 (WT), surpassing Method 2 by +0.073, +0.153, and +0.072, respectively. Improvements are most pronounced in the TC, the most challenging region. In conclusion, the intracranial normalization, bias-field correction, and FLAIR-guided centering increase input reliability and the stability of the 2D model, providing a solid foundation for future 3D extensions and external validation.

Keywords: Gliomas; Magnetic Resonance Imaging; Preprocessing.

List of Symbols and Abbreviations

Abbreviations

CNS	*Central Nervous System*
WHO	*World Health Organization*
IDH	*Isocitrate Dehydrogenase*
MRI	*Magnetic Resonance Imaging*
FLAIR	*Fluid Attenuated Inversion Recovery*
T1GD	*T1-weighted post-contrast*
AI	*Artificial Intelligence*
CNN	*Convolutional Neural Network*
ROI	*Region of Interest*
BraTS	*Brain Tumor Segmentation Challenge*
ET	*Enhancing Tumor*
TC	*Tumor Core*
WT	*Whole Tumor*
ED	*Edema*
NET	*Non-Enhancing Tumor*
NCR	*Necrotic/Cystic Core*
NOS	*Not Otherwise Specified*
CSF	*Cerebrospinal Fluid*
RF	*Radiofrequency*
CT	*Computed Tomography*
FCN	*Fully Convolutional Network*
ReLU	*Rectified Linear Unit*
DSC	*Dice Similarity Coefficient*
FP	*False Positive*
FN	*False Negative*
TP	*True Positive*
TN	*True Negative*
IoU	*Intersection over Union*

Table of Contents

List of Figures

List of Tables

1 Introduction

Gliomas are the most common primary malignant tumors of the central nervous system, exhibiting a broad spectrum of aggressiveness and heterogeneous histological sub-regions including peritumoral edema tissue (ED), necrotic core (NC), and active (AT) or non-enhancing tumor (NET) areas (Bakas et al., 2019). According to global cancer statistics, cancers of the brain and central nervous system (CNS) rank as the 19th most frequently diagnosed malignancy worldwide and the 12th leading cause of cancer-related mortality (Global Cancer Observatory, 2022). Furthermore, projections for Europe indicate that by 2030 both the incidence and mortality of brain/CNS tumors will have increased compared to 2015, with approximately 85,000 new cases and 70,000 deaths expected in 2030 (Ilic & Ilic, 2023).

Gliomas can be classified based on their microscopic appearance into many types: astrocytomas, oligodendrogliomas, and mixed gliomas, each with distinct cellular origins and molecular characteristics. The World Health Organization's (WHO) 2021 Classification of Tumors of the Central Nervous System marks an important advancement in glioma categorization, bringing attention to the integration of molecular markers with traditional histological diagnosis. This latest classification categorizes gliomas into three primary types based on both histological and molecular characteristics: (i) astrocytoma with isocitrate dehydrogenase (IDH) mutations, (ii) oligodendroglioma, which is IDH-mutant and exhibits a co-deletion of chromosomes 1p and 19q, and (iii) glioblastoma, characterized by the absence of IDH mutations (IDH-wildtype) (Louis et al., 2021; Wen & Packer, 2021).

Magnetic Resonance Imaging (MRI) plays an essential role in glioma diagnosis and clinical management, being the most suitable non-invasive technique and enabling detailed visualization of brain anatomy and pathology because of its soft tissue contrast. Multiple MRI sequences provide complementary information about tumor morphology, its extension and characteristics. These sequences include T1-weighted (T1), T2-weighted (T2), fluid-attenuated inversion recovery (FLAIR), and contrast-enhanced T1-weighted (T1Gd) images. For example, T1Gd images highlight tumor borders vividly due to the accumulation of contrast agent in regions where the blood-brain barrier (BBB) is compromised, while T2-weighted images emphasize the ED regions surrounding the tumor. However, the interpretation of these complex, multimodal images can lead to significant challenges like the time-consuming and variability among observers when it comes to manual segmentation, classification of gliomas and the complexity of capturing the heterogeneity of these tumors (Bauer et al., 2013).

Recent research has focused on Artificial Intelligence (AI) techniques as potential tools to support clinicians in diagnosing and predicting outcomes in gliomas. Some models report segmentation accuracies above 80%, suggesting they could improve the precision of tumor delineation (J. Xu et al., 2022). These tools are promising, but they haven't yet been shown to improve survival rates. Even so, as diagnostic accuracy improves, hopefully it will translate into better patient outcomes.

2.1. Motivation and Problem Statement

Integrating AI into neuro-oncology represents a significant development in glioma diagnosis and management. This shift toward AI methods is driven by the inherent complexity and variability of gliomas, which often makes traditional diagnostics challenging. AI techniques, especially deep learning models like convolutional neural networks (CNNs), have already achieved high performance in tasks such as tumor segmentation, classification, and grading (Erickson et al., 2018). These advancements have shown accuracy comparable to expert performance in segmentation tasks, reducing some forms of human error. For instance, certain AI models have already achieved tumor segmentation accuracy above 80-90% (Lotan et al., 2019), revealing its potential to enhance the precision of tumor delineation in medical images.

In routine clinical practice, manually analysing multimodal MRI scans is time-consuming and subjective, often resulting in considerable inter-observer variability. As imaging volumes continue to grow, these challenges are exacerbated, potentially leading to delays in diagnosis and initiation of treatment. AI-based tools could help mitigate these issues by rapidly processing large imaging datasets and detecting subtle patterns that human readers might overlook, thereby providing more objective and consistent interpretations. (Bauer et al., 2013). However, the focus should remain on using these systems as support tools designed to assist and accelerate the diagnostic and treatment process, reducing uncertainty and improving workflow efficiency. It should never be intended to replace the radiologist, whose expertise, clinical judgment, and contextual understanding are irreplaceable

To be clinically useful, the automated output must remain sufficiently close to what radiologists perceive as correct tumor boundaries. When full automation cannot achieve this balance, interactive or semi-automatic methods, in which the user can initialize or refine the segmentation, may provide an effective compromise. In such cases, the variability introduced by limited user interaction tends to be much smaller than that of completely manual procedures. Moreover, usability, ease of integration, and regulatory certification are key factors for clinical acceptance. In practice, broader adoption will likely depend on the availability of validated and user-friendly commercial software integrated with existing medical imaging systems(Bauer et al., 2013).

The primary objective of this project was to develop a reliable AI approach for accurate glioma segmentation on brain MRI using CNN-based techniques. However, during the project, several practical challenges were encountered, including limited access to high-performance GPUs, a tight timeline, and a higher-than-expected level of complexity involved in programming and developing an effective medical image segmentation algorithm. I initially underestimated the level of expertise and effort required to construct a complete segmentation pipeline from scratch. Consequently, the project scope was refined to focus on designing and systematically evaluating an MRI preprocessing pipeline, followed by training the processed images using a 2D U-Net model to assess segmentation performance. This revised focus aimed to better understand how preprocessing techniques influence the accuracy and reliability of glioma segmentation in multimodal brain MRI.

This revised focus emphasizes preprocessing stages that directly affect performance, such as intensity standardization, bias-field correction, and region-of-interest (ROI) localization/cropping strategies suitable for multimodal brain MRI.

1.2. Objectives

In line with the scope adjustment described earlier, shifting from a full segmentation system to a focus on preprocessing MRI under accessible computational constraints, this thesis aims to:

Objective 1 — Consolidate a robust preprocessing pipeline. Establish and justify the core steps most likely to impact the performance in multimodal brain MRI.

Objective 2 — Implement a transparent 2D U-Net baseline (MONAI). Train a compact 2D U-Net with three input channels (FLAIR, T1GD, T2) and remapped labels {0,1,2,3}.

Objective 3 — Evaluate with standard BraTS sub-regions and metrics. Report ET, TC, and WT using Dice, IoU, recall, precision, and specificity.

2 Background

In recent years, segmenting brain tumors in MRI scans has attracted increasing interest from researchers, with a shift from traditional image processing methods to advanced deep learning techniques. This chapter reviews key contributions in this field, discusses current challenges, and outlines future directions.

2.2. Introduction to Gliomas and Subregions

Gliomas are primary brain tumors that arise from glial cells. In adults, diffuse glioma is the most common malignant type, and despite surgery, radiation, and chemotherapy, it tends to recur. This clinical reality is one reason why modern work goes beyond microscope-only diagnosis and looks at both images and tumor genetics (Li et al., 2022). On MRI, gliomas rarely look uniform, they present as a combination of areas that behave differently: while some parts enhance after contrast, some look cystic or necrotic, and others blend into the surrounding brain. To make this complexity easier to study and communicate, the Brain Tumor Segmentation (BraTS) initiative defined image-based subregions that many researchers and clinicians now use when describing pre-operative MRI: enhancing tumor, tumor core and the whole tumor. Because these subregions differ in appearance, they also differ in which MRI sequence sees them best and often behaves differently biologically:

- Enhancing tumor (ET): it presents areas with hyper-intensity on T1-Gd in comparation with the healthy white matter and with the T1 modality. This is usually where active tumor and abnormal vessels break down the blood–brain barrier (Baid et al., 2021).
- Necrotic tumor core (NCR): non-enhancing components of the tumor core together with necrotic/cystic areas, this are typically non-enhancing and T1-Gd hypointense, while solid non-enhancing tumor is often T2/FLAIR-hyperintense (Baid et al., 2021; Louis et al., 2021).

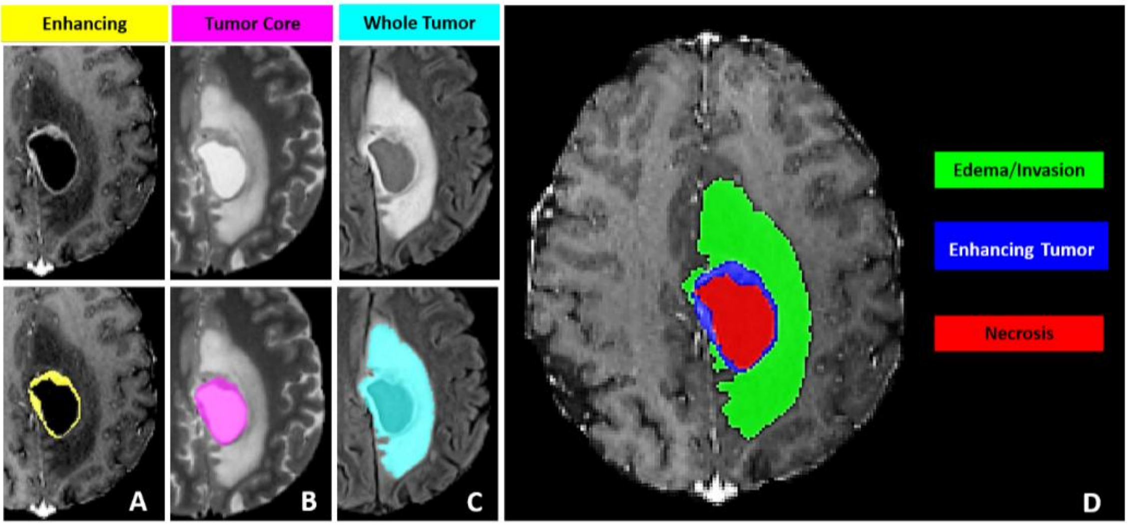


Figure 1: MRI of glioma subregions for the RSNA-ASNR-MICCAI BraTS 2021 challenge.

- Tumor Core (TC): This is the bulk that surgeons aim to resect. It includes two parts: the ET and NCR. The ET is the contrast-avid rim or nodular component and NCR sits centrally and often signals a more aggressive biology.
- Peritumoral Edema (ED): shows the T2-FLAIR-hyperintense abnormality surrounding the core, representing a mixture of vasogenic edema and infiltrative non-enhancing tumor.

Figure 1 shows that the whole tumor (WT) encompasses all abnormal signal, i.e. the entire lesion visible on MRI. On a FLAIR image, the WT (cyan in the figure) includes the tumor core plus the surrounding edema (essentially TC + ED). The TC, shown on a T2 image, includes the enhancing rim together with the non-enhancing core (magenta color, corresponding to ET plus NCR). Meanwhile, the ET is highlighted on the T1-Gd image as the bright enhancing portion (yellow) encircling the dark necrotic center. These different MRI sequences complement each other: T1-Gd is most sensitive to the enhancing tumor, T2 delineates the bulk of the core, and T2-FLAIR best shows the full extent of disease including edema. By segmenting a glioma into these subregions, researchers and clinicians can better describe tumor heterogeneity, measure volumes of each component, and correlate them with outcomes or molecular features. This standardized subregion approach thus facilitates communication and quantitative analysis in the management of diffuse gliomas (Baid et al., 2021a).

2.3. WHO Grading System for Gliomas

Grading indicates how aggressive a diffuse glioma is expected to be. Historically, diffuse gliomas were graded II (diffuse), III (anaplastic), and IV (glioblastoma) based on histology, with features like necrosis and microvascular proliferation pushing a tumor to grade IV. But histology alone left too much uncertainty because patients within the same grade could do very differently. The WHO 2016 grading system update for gliomas made a decisive shift: for adult diffuse gliomas, molecular markers moved to center stage, especially IDH mutation status and 1p/19q codeletion. In practice, a tumor with oligodendroglial morphology and both IDH mutation and 1p/19q codeletion is called oligodendroglioma. Also, an IDH-mutant tumor without 1p/19q codeletion is called astrocytoma and tumors without IDH mutations are labeled IDH-wildtype. When molecular testing cannot be completed, the suffix NOS (“not otherwise specified”) is used.

The 2016 scheme kept the familiar grade II–IV framework but acknowledged important nuances: IDH-mutant astrocytomas with grade-IV histology behave differently from classic IDH-wildtype glioblastoma. Some authors even suggested labeling them “astrocytoma, IDH-mutant, grade IV” to emphasize the biology. Conversely, some diffuse IDH-wildtype tumors that look like grade II–III under the microscope already carry the genetic signature of glioblastoma (e.g., gain of chromosome 7 with loss of 10q, often with TERT-promoter mutation) and tend to behave and be treated like glioblastoma.

2.4. MRI in Glioma Diagnosis and Segmentation

MRI is the primary imaging modality for diagnosing and managing gliomas, due to its unparalleled ability to visualize soft tissues in the brain in a non-invasive manner. In fact, MRI is considered the mainstay of modern neuroimaging, providing superior structural detail and even capturing cellular, vascular, metabolic, and functional properties of brain tumors (Villanueva-Meyer et al., 2017). For aggressive brain tumors like glioblastomas, a contrast-enhanced MRI examination is the standard of care for initial diagnosis and follow-up, preferred over techniques like CT because of MRI’s greater sensitivity to tumor characteristics. MRI scans produce high-contrast images of brain anatomy, which makes them especially valuable for identifying and delineating brain tumors and associated abnormalities. Multiple MRI sequences are typically acquired in glioma patients – T1-weighted, T2-weighted, FLAIR, and post-contrast T1 – with each sequence highlighting different aspects of the tumor and surrounding tissue (Shukla et al., 2017).

2.4.1. MRI Principles and Image Formation

MRI operates on the principles of nuclear magnetic resonance physics. Unlike X-rays or CT scans, MRI uses no ionizing radiation. This means that it explores the fact that the human body is largely composed of water and hydrogen atoms, protons. The main components of an MRI scanner are illustrated in Figure 2, where the patient lies at the center of a powerful superconducting magnet that causes hydrogen protons in tissues to align with the magnetic field. Brief radiofrequency (RF) pulses are then applied to perturb these aligned protons, tipping them out of alignment. When the RF pulse is turned off, the protons relax back toward alignment and in doing so emit faint radio signals. Specialized receiver coils detect these signals, and a computer reconstructs them into cross-sectional images. With computer-controlled magnetic field gradients to encode position, MRI can produce images in any plane and full 3D volumes with high spatial resolution (Coyne & National MagLab, 2015).

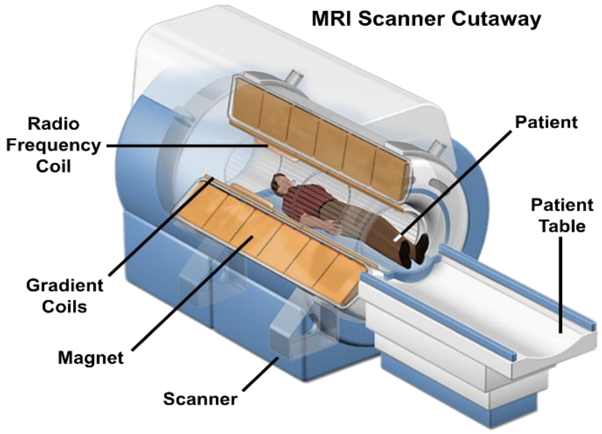


Figure 2: Illustration of an MRI scanner with the main components of the system.

2.4.2. Advantages of MRI Over Other Imaging Techniques

MRI offers several distinct advantages for brain tumor evaluation and is a mainstay across diagnosis, treatment planning, and post-therapy assessment:

- **Non-Ionizing Imaging**: it relies on magnetic fields and radio waves rather than ionizing radiation, which is advantageous for repeated follow-up studies.

- **Superior Soft Tissue Contrast**: provides exceptional soft-tissue contrast and resolution, making it the tool of choice for detailed assessment of brain parenchyma, tumor, and associated changes; this central role in neuro-oncology care is reflected in contemporary clinical practice.

- **Multiplanar & 3D Capability**: it acquires images in any orientation (axial, sagittal, coronal) and as 3D volumes without repositioning the patient, facilitating precise anatomic assessment and planning, as shown in Figure 3 (Coyne & National MagLab, 2015; Villanueva-Meyer et al., 2017).

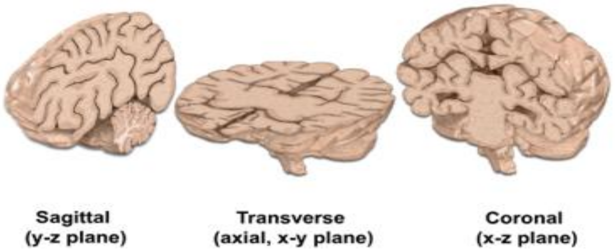


Figure 3: Brain views in three common MRI planes: sagittal, transverse, and coronal.

2.4.3. Standard MRI Modalities in Brain Tumor Imaging

In clinical practice, a set of core sequences is acquired because each contributes complementary information about the tumor and surrounding brain. These structural sequences form the foundation of the brain-tumor MRI exam, as summarized below:

T1-weighted (T1): provide detailed anatomical contrast, with dark CSF and bright white matter, offering a baseline for post-contrast comparison. The critical role of T1 is to be paired with T1 post-contrast to determine enhancement (Gaillard et al., 2015; Villanueva-Meyer et al., 2017). Figure 4 presents examples of T1-weighted slices.

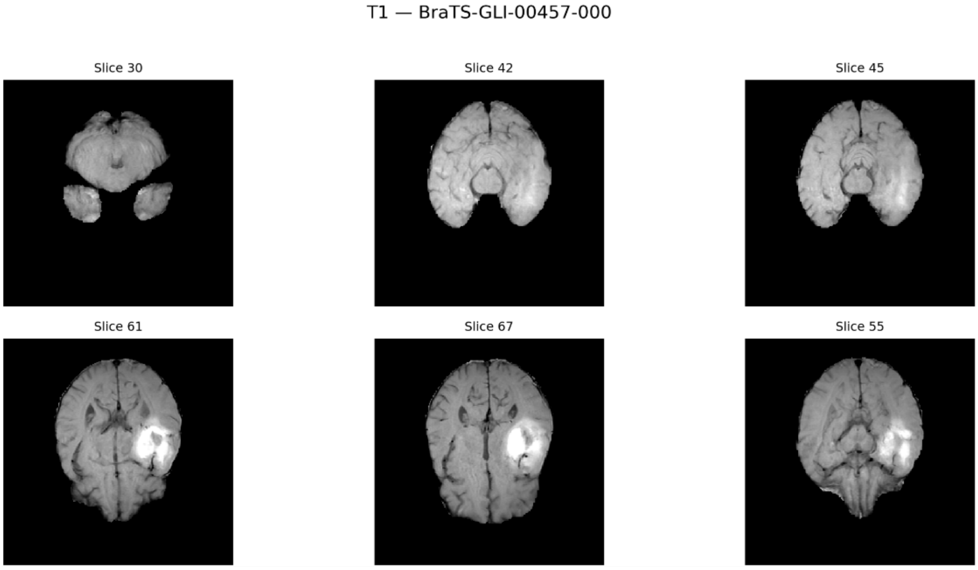


Figure 4: Example of T1-weighted MRI slices from the BraTS 2023 dataset with different cross-sections of the brain

T1-weighted post-contrast (T1Gd): obtained after administering a gadolinium-based contrast agent. After gadolinium, areas of blood–brain barrier breakdown enhance being a hallmark of many high-grade tumors (Figure 5) (Villanueva-Meyer et al., 2017).

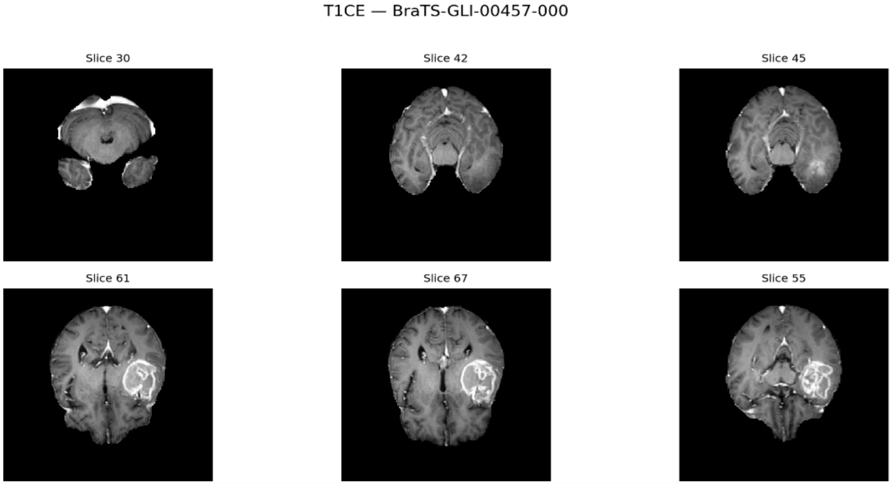


Figure 5: Example of contrast -enhanced T1-weighted MRI slices from the BraTS 2023 dataset, with different cross-sections of the brain.

T2-weighted (T2): highlights differences in tissue water content. As seen in Figure 6, fluid and ED appear as hyperintense areas, which facilitates the detection of lesions, inflammation, demyelination, ischemia, and peritumoral ED (Dhabalia et al., 2024; Jones et al., 2009).

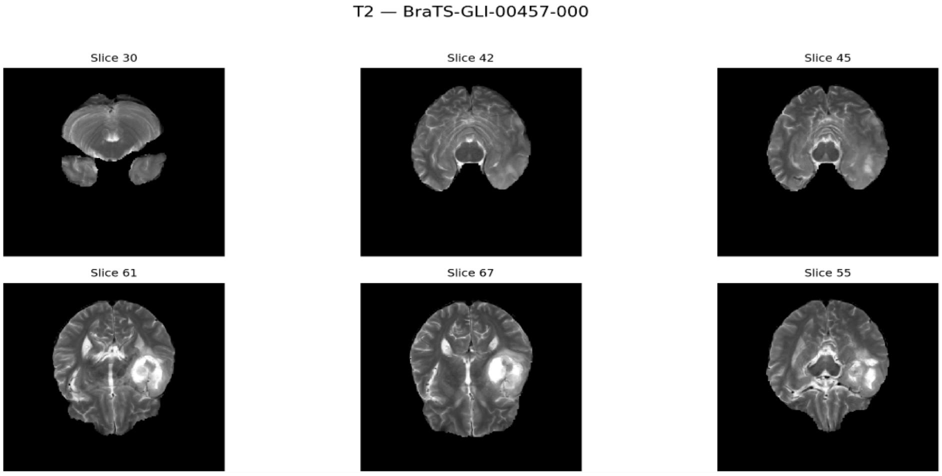


Figure 6: Example of T2-weighted MRI slices from the BraTS 2023 dataset, with different cross-sections of the brain.

Fluid-Attenuated Inversion Recovery (FLAIR): A T2-based inversion-recovery sequence that suppresses free fluid, improving visibility of ED, non-enhancing tumor (NET) and CSF spaces that might be obscured on standard T2. The hyperintense “FLAIR envelope” around gliomas typically represents the combination of ED plus infiltrating tumor cells. Most of the glioblastoma recurrences occur within this T2/FLAIR envelope, as seen in Figure 7, highlighting FLAIR’s utility for delineating disease beyond the enhancing core (Gaillard et al., 2015; Niknejad et al., 2013; Shukla et al., 2017).

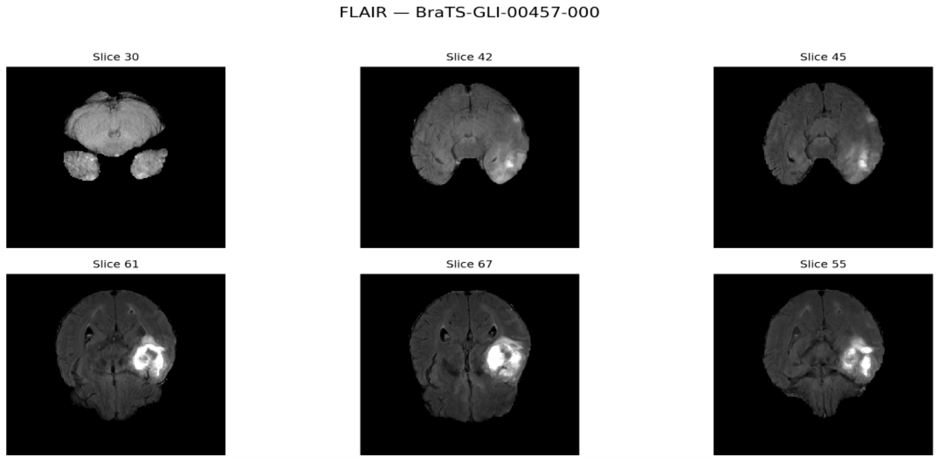


Figure 7: Example of FLAIR MRI slices from the BraTS 2023 dataset, with different cross-sections of the brain.

Each of these MRI modalities contributes uniquely to diagnosis. T1 and T1Gd delineate the tumor's solid components and relationship to blood vessels or the blood–brain barrier, whereas T2 and FLAIR depict the full extent of brain tissue affected by the tumor. By examining all sequences together, clinicians can distinguish tumor tissue from ED, assess mass effect, and plan biopsies or surgeries more safely (Gaillard et al., 2008; Villanueva-Meyer et al., 2017).

3. Related Work

3.1. Brain Tumor Segmentation: From Manual Delineation to AI

Imaging alone is not sufficient, the information in MRI scans needs to be quantified and interpreted for clinical decision-making. Segmentation of brain tumors on MRI, like delineating the tumor's boundaries and sub-regions on the images, is a crucial step in diagnosis, treatment planning, and monitoring of gliomas. By identifying the exact region of the tumor on each scan, clinicians can calculate tumor volume, plan surgical resection or radiation targeting and track how the tumor changes over time (Wadhwa et al., 2019). In fact, accurate MRI-based tumor segmentation has been shown to aid in surgery planning, guide postoperative assessments, and even correlate with improved patient outcomes. For example, knowing the precise extent of a glioma helps neurosurgeons remove as much tumor as safely possible, and consistent segmentation across serial MRIs allows detection of tumor progression or response to therapy at the earliest time point, which can impact survival (Rios Piedra et al., 2016).

Traditionally, tumor segmentation on MRI is performed manually by expert radiologists or neurosurgeons. In routine clinical practice, a specialist will carefully outline the tumor on each MRI slice to create a 3D representation of the lesion. This manual process, while considered the reference standard, is highly labor-intensive and time-consuming, as a single brain MRI study can contain hundreds of images that need examination. Moreover, manual delineation is inherently subjective because it depends on the expert's experience and judgment. Different observers may produce slightly different segmentations on the same scan, especially for diffuse tumor margins, leading to significant inter-observer variability. Even the same person may not outline the tumor exactly identically each time. This variability and potential bias in manual annotations can affect downstream decisions or evaluations of treatment response. In short, while manual MRI segmentation provides meaningful results, its variable and not scalable to the growing volume of imaging data (Liu et al., 2023; Rios Piedra et al., 2016) plus the predictions of this brain tumor in the future as mentioned earlier. These challenges have created a strong motivation to develop automated methods for brain tumor segmentation.

Segmenting a medical image means cutting it into meaningful pieces, marking the edges of a tumor, organ or lesion. Clear outlines are vital because they help doctors measure disease, plan treatments and monitor progress (Dorfner et al., 2025; Y. Xu et al., 2024). In oncology, neurology and cardiology, precise segmentation of structures like brain tumors or scar tissue lets clinicians measure volumes, guide radiotherapy or surgery and track therapy response, reducing differences between observers (Dorfner et al., 2025).

Manual segmentation is still considered the gold standard where radiologists or technologists draw boundaries on each image slice. This approach is accurate but slow, requires training and is impractical for large studies. Semi-automatic methods ask a user to place a seed point and then grow the region, saving time yet still relying on manual inputs and leading to inter-observer variability. Fully automatic methods rely on algorithms to find regions of interest without much human intervention and enable the creation of large, consistent datasets, which are essential for radiomics and artificial intelligence (Ma et al., 2024).

For brain tumors, segmentation is particularly challenging because gliomas are intrinsically heterogeneous: their sub-regions exhibit distinct intensity profiles across different MRI sequences. This heterogeneity in appearance and shape means that segmentation of brain tumors in multimodal MRI scans is among the most difficult tasks in medical image analysis. The segmentation algorithm must therefore integrate information from multiple modalities and account for variable tumor appearance (Bakas et al., 2019), as mentioned in chapter 2.1.

Before deep learning, several traditional algorithms were used for medical image segmentation. These techniques are simple and fast but rely on strong assumptions about object intensity or shape and deteriorate in the presence of noise, heterogeneous tissues or low contrast (Y. Xu et al., 2024):

1. **Thresholding** – these methods choose one or more intensity values to separate the ROI from the background. Global thresholding uses a single threshold for the entire image, while local approaches adapt to regional variations. These methods are quick but assume a bimodal histogram and only work well when there is clear contrast, so intensity variations or

similar-coloured tissues may lead to incorrect segmentation (Rios Piedra et al., 2016; Y. Xu et al., 2024).

2. **Region-based methods** – algorithms like region growing and watershed expand areas from seed points based on intensity similarity and local connectivity. They can detect continuous structures and handle some noise, but they depend heavily on correct seed selection and region homogeneity. In heterogeneous tissues, such as tumors with necrotic areas, they tend to over or under-segment (Rios Piedra et al., 2016; Y. Xu et al., 2024).

3. **Edge-based detection** – classical operators such as Sobel, Prewitt, Laplacian or Canny detect intensity discontinuities to outline boundaries. These methods work for structures with clear edges, like vessel walls or bones, but are highly sensitive to noise and may fail when boundaries are weak or fuzzy (Rios Piedra et al., 2016; Y. Xu et al., 2024).

4. **Clustering and fuzzy methods** – algorithms like k-means or fuzzy C-means group pixels with similar characteristics. Fuzzy C-means allows a pixel to belong partly to multiple clusters, adapting better to intensity gradients. However, these methods assume the number of clusters is known and are sensitive to noise and often require post-processing to remove artefacts (Y. Xu et al., 2024).

5. **Graph-based approaches** – methods like graph cuts represent the image as a graph and seek a minimum-energy partition. It produces smooth segmentation and coherent boundaries for complex topologies but are computationally demanding and rely heavily on the cost function and graph quality (Y. Xu et al., 2024).

These classical techniques share common limitations: they capture only simple intensity or texture features, ignore broader spatial context and require manual parameter tuning. These methods, although efficient, extract features from a single image and are sensitive to noise, resulting in low robustness. Since the 2010s, CNNs have transformed medical image segmentation. Fully convolutional networks (FCNs) and particularly the U-Net, combine down-sampling and up-sampling pathways with skip connections that preserve edge details and recover global context. This architecture allows accurate segmentation even when the ROI is small or has a low signal-to-noise ratio, and U-Net models achieve high Dice scores (around 0.92) and substantially outperform traditional methods (Wan et al., 2023). Despite significant advances, CNNs remain limited by their dependence on large, annotated datasets to mitigate overfitting, and models trained on one image domain or population often fail to generalise across different modalities or institutions (Ma et al., 2024).

3.2. The U-net model

Since its introduction by Ronneberger in 2015, the U-Net has become a benchmark architecture for biomedical image segmentation. Its design follows a distinctive U-shaped layout, as illustrated in Figure 8, with two symmetrical components: a contracting path nominated as encoder and an expanding path nominated as decoder. At the center lies a bottleneck layer that compresses the information into a compact representation, typically implemented with two convolutional layers activated by Rectified Linear Unit (ReLU), a simple non-linear function (Equation 3.1) that introduces non-linearity and helps networks learn complex patterns while mitigating gradient-vanishing effects.

$$f(x) = max(0, x) \tag{3.1}$$

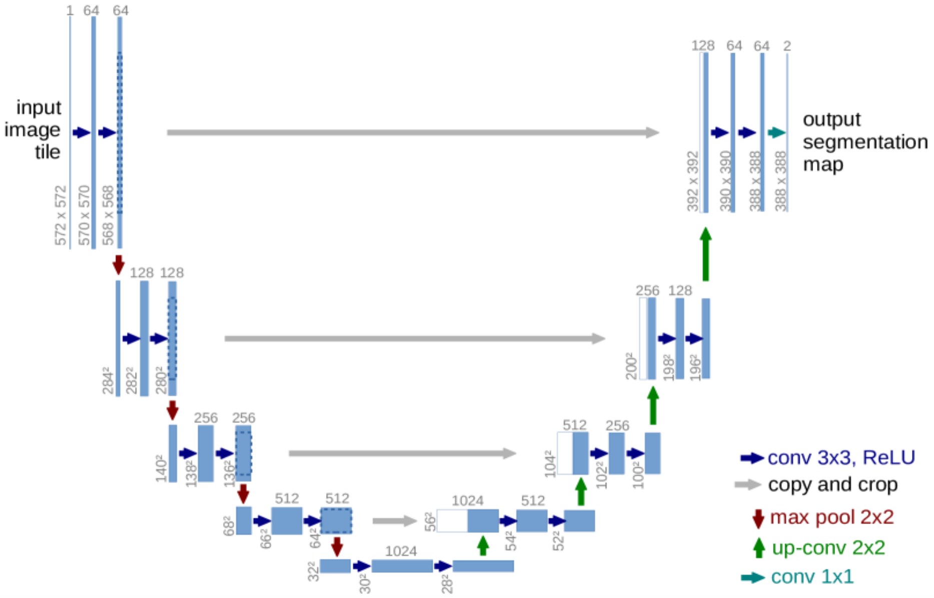


Figure 8: U-Net architecture for image segmentation.

In the encoder, the network repeatedly applies two 3×3 convolutions without padding, each followed by ReLU, and then a 2×2 max-pooling operation with a stride of 2 to reduce spatial resolution. With each downsampling step, the number of feature channels doubles. The decoder works in the opposite direction: it upsamples the feature maps, applies a 2×2 “up-convolution” to halve the number of channels, concatenates the result with a cropped feature map from the corresponding encoder stage, and then applies two

3×3 convolutions with ReLU. Cropping is required because the unpadded convolutions reduce feature map size. Finally, a 1×1 convolution maps each 64-channel vector to the target classes. This setup produces a fully convolutional network with 23 layers (Azad et al., 2022; Ronneberger et al., 2015).

One challenge in the downsampling process is that pooling layers inevitably lose fine spatial detail. To counter this, U-Net employs skip connections that link each encoder stage directly to its decoder counterpart. By merging high-resolution features from the encoder with the upsampled decoder features, the model can combine fine-grained detail with broader contextual information, improving localization accuracy. These skip connections are widely considered essential, as they help preserve local patterns and improve the model's ability to generalize. Over the years, researchers have proposed variations such as U-Net 3+ (H. Huang et al., 2020) , which add more connection points, and BiO-Net (Xiang et al., 2020), which introduces bidirectional links, all retaining the core idea of fusing features from different scales.

Because U-Net uses valid convolutions, no padding, its output is slightly smaller than the input. To handle large images without losing coverage, the original paper proposed the overlap-tile strategy: the image is split into overlapping tiles, each tile is processed independently, and the results are stitched back together. Tiles are cropped so that pooling layers always operate on even-sized feature maps. To maintain context near the image borders, the method mirrors the pixels at the edges. In the original architecture, the bottleneck was kept simple, with just two convolutional layers and ReLU activations encoding the compressed representation. More recent work, however, shows that attention mechanisms can enhance performance. For example, the positional attention block in MA-Net uses self-attention to model spatial dependencies in the bottleneck. Other variants add lightweight spatial attention to highlight important features or introduce frequency-based recalibration mechanisms. These developments illustrate how the original U-Net has served as a flexible foundation for architectures that capture richer context and longer-range relationships (Azad et al., 2022; Ronneberger et al., 2015).

3.3. U-Net Variants for Glioma Segmentation in MRI

The 3D U-Net variant was proposed to handle volumetric segmentation in medical images by extending the original 2D U-Net to three dimensions. The main motivation was to capture spatial continuity in brain MRI data, since segmenting separate 2D slices limits tumor context to the slice plane. Unlike the conventional U-Net, 3D U-Net replaces all 2D operations, like convolutions or pooling, with 3D operations, enabling learning from full volumes and even from sparsely annotated slices in some cases. This makes it possible to obtain dense three-dimensional segmentations of gliomas, leveraging inter-slice information that the original U-Net would not capture (Çiçek et al., 2016). Evidence in glioma segmentation shows the advantages of 3D U-Net. For example, a 3D U-Net–based model was applied to the BraTS 2018, with training adaptations and an ensemble of models, achieving competitive results. This method achieved 9th place in the highly competitive BraTS 2018 challenge, automatically segmenting glioma subregions (ED, NCR, ET). Six 3D U-Nets were trained for this challenge, reaching mean Dice values of the validation set 0.7946 / 0.9114 / 0.8304 and on the test set (191 cases) 0.754 / 0.878 / 0.799 for ET/WT/TC respectively. This indicates that incorporating 3D context improves accuracy over the original 2D U-Net in segmenting heterogeneous gliomas on MRI (Feng et al., 2020).

The U-Net++ was developed to reduce the semantic gap between encoder and decoder layers in the traditional U-Net. In the original U-Net, skip-connected features may be at very different abstraction levels (shallow encoder vs. deep decoder), making precise reconstruction harder. U-Net++ addresses this by introducing nested, dense skip connections between encoder and decoder, forming intermediate sub-U-Nets. In other words, it inserts extra convolutions along the skip paths, creating multi-level connection routes that progressively refine features before they reach the decoder. The architecture also employs deep supervision on intermediate maps to aid training. As a distinctive trait, U-Net++ links encoder and decoder through multiple intermediate layers, instead of single direct connections, making the representations more semantically similar and easing optimization. The authors reported performance gains over the classic U-Net with an average increase of about 3.9 IoU points across several biomedical segmentation datasets (Zhou et al., 2018). Recent brain-tumor studies show that dense-skip variants like a multi-plane U-Net++ applied to BraTS2021 data reported strong performance: Dice of ~0.79 for ET, ~0.84 for TC, and 0.906 for WT (Roth et al., 2022). These original reports confirm that U-Net++ yields more accurate glioma segmentations, thanks to its ability to fuse multi-scale features and reduce encoder–decoder discrepancies during mask reconstruction.

Unlike the previous variants, nnU-Net (no-new-Net) is not a fixed new architecture, but a self-configuring framework designed to automate the design and tuning of U-Net models for a target dataset. It analyses the training data and automatically configures preprocessing, network architecture, training, and post-processing using a dataset fingerprint and rule-based dependencies to infer pipeline choices**.** By default, it trains a 2D U-Net, a full-resolution 3D U-Net, and a 3D cascade, then selects the best configuration and decides whether post-processing is beneficial**.** Rather than introducing new building blocks, nnU-Net adds an automation layer, so the algorithm decides how to configure a U-Net-like template to achieve the best performance on the given dataset, without specialized manual intervention (Isensee et al., 2020). nnU-Net served as a competitive baseline for BraTS and, with some task-specific optimizations, won 1st place in BraTS 2020. The method achieved Dice scores of 88.95%, 85.06%, and 82.03% for WT, TC and ET, respectively, on the BraTS 2020 test set (Isensee et al., 2021).

Although U-Net and its variants, such as nnU-Net, have achieved strong performance in BraTS, they remain inherently limited by the local receptive fields of convolutional layers. This means that long-range dependencies and global contextual information, which are crucial for capturing the heterogeneous structure of gliomas, are not fully exploited. To address this gap, researchers have recently explored the use of Transformers, which employ global self-attention to model distant relationships within the image. Building on this idea, the TransUNet model (Chen et al., 2021) emerged as a hybrid approach that combines convolutional feature extraction with Transformer-based global encoding. In TransUNet, the CNN first extracts feature maps from the image, which are then split into tokens and processed by a Transformer encoder (ViT) to capture global context. The decoder then upsamples these representations and merges them with high resolution CNN feature maps through skip connections, preserving local detail needed for accurate lesion localization. This design leverages both the fine detail provided by CNNs and the long-range relationships modeled by Transformers, leading to more accurate segmentations. This hybrid CNN Transformer methods achieve state-of-the-art results on the BraTS benchmark for glioma segmentation, reporting Dice scores of about 0.93 for WT, 0.93 for TC, and 0.89 for ET. These results highlight the effectiveness of Transformers in complementing CNNs for precise tumor delineation (Wang et al., 2023). Table 1 summarizes four influential U-Net variants used for glioma segmentation.

Table 1: Summary of four influential U-Net variants used for glioma segmentation on MRI, with motivation idea, architectural differences and quantitative BraTS results.

Variant	Motivation	Architectural Differences	Evidence on BraTS
3D U-Net (Çiçek et al., 2016)	Exploit volumetric context for coherent masks across slices.	All 2D ops → 3D (conv/pool/upsample); train on 3D patches; sparse labels possible.	BraTS 2018 — Ensemble of six 3D U-Nets: Validation: 0.911 / 0.830 / 0.795 Test (191 cases): 0.878 / 0.799 / 0.754, 9th Rank (Feng et al., 2020)
UNet++ (Zhou et al., 2018b)	Reduce encoder–decoder semantic gap; stronger multi-scale fusion.	Nested, dense skip pathways + deep supervision; progressive refinement.	BraTS 2021 — Multi-plane UNet++ ensemble: 0.906 / 0.835 / 0.792 (Roth et al., 2022)
nnU-Net (Isensee et al., 2020)	Self-configuring pipeline (2D/3D/cascade) — strong, comparable baseline.	Auto-configures preprocessing, architecture, training, post-processing; picks best model/ensemble.	BraTS 2020 — Test (1st place): 0.8895 / 0.8506 / 0.8203 (Isensee et al., 2021))
TransUNet (hybrid CNN–Transformer) (Chen et al., 2021)	Add global self-attention to capture long-range relations while preserving local detail.	CNN encoder → tokens → ViT encoder (global context) → U-Net-style decoder with skips.	Hybrid examples: • BiTr-UNet (BraTS 2021): 0.934 / 0.930 / 0.889 • TransBTS (BraTS 2019/2020): 0.901 / 0.817 / 0.787 (Wang et al., 2023)

Clinical use depends on trust and ease of use. Many research tools are still hard for clinicians to handle, and they are accepted only when results look close to what a radiologist would draw. When full automation is not good enough, simple interactive tools, so clinicians can initialize or quickly correct a mask, help reduce variability while keeping objectivity in follow-up. These methods should support, not replace, clinical judgment, and adoption requires user-friendly software and proper certification (Bauer et al., 2013).

4. Dataset Description and Preprocessing Steps

4.1. BraTS Dataset

The BraTS challenge has been fundamental in driving significant advances in biomedical imaging by providing standardized datasets and evaluation frameworks. The challenge uses multi-parametric MRI scans, including T1, T2, T1Gd and T2-FLAIR volumes. This multi-modal approach allows algorithms to leverage complementary information from different MRI sequences, enhancing segmentation accuracy.

The evolution of the BraTS challenge demonstrates the field's progress, from an initial 35 training cases in 2012 to 285 cases in 2018, the challenge has expanded its scope to include overall survival prediction, focusing on the clinical relevance of segmentation tasks. The 2023 edition expands significantly, encompassing approximately 4,500 cases to address diverse populations, tumor types, clinical challenges, and algorithmic considerations. The challenge consists of multiple tasks, including segmentation of gliomas, meningiomas, brain metastases, and pediatric tumors, as well as algorithmic generalizability, handling missing data, and data augmentation techniques (Bakas et al., 2019).

The training and validation data are available through the Synapse platform, with ground truth annotations meticulously curated and approved by expert neuroradiologists. These annotations serve as the reference for evaluating segmentation algorithms submitted to the challenge.

For Task 1, Tumor Sub-Region Segmentation, initial automated segmentations were generated using a fusion of top-performing BraTS segmentation algorithms, including nnU-Net, DeepScan, and DeepMedic. These preliminary segmentations were subsequently manually refined by neuroradiology experts following a standardized

annotation protocol. The final segmentations were approved by board-certified neuroradiologists with extensive experience in glioma imaging, ensuring high-quality annotations for training and evaluation.
The dataset consists of four segmentation labels, which define the different tumor sub-regions:

- Label 0 (Backgroun); non-tumor brain regions and surrounding tissue.
- Label 1 NCR: Non-enhancing necrotic regions of the tumor, typically appearing hypointense on T1Gd MRI.
- Label 2 ED: The hyperintense region surrounding the tumor, seen on T2-FLAIR, which includes infiltrative tumor and vasogenic ED.
- Label 3 ET: Contrast-enhancing portions of the tumor, appearing hyperintense on T1Gd MRI.

To report the BraTS subregions, the following mapping was adopted:

- ET = Label 3;
- TC = Label 1 + 3;
- WT = Label 1 + 2 + 3.

These segmentations follow established VASARI imaging features, ensuring consistency with radiological standards and aiding in the development of automated segmentation models capable of distinguishing between different tumor sub-regions (Baid et al., 2021).

4.2. Data Preprocessing

The data preprocessing pipeline and the subsequent segmentation models were implemented using the Python 3 programming language. The development relied on established libraries for medical imaging and array manipulation, specifically Nibabel and SimpleITK for NIfTI file handling, NumPy for tensor operations, and scikit-learn for data splitting and normalization. The specific implementation steps and code structure are detailed in the following sections. Diagram illustrated in Figure 9 shows the sequential processing operations applied in this baseline method, highlighting the per-volume min–max normalization and the reliance on the geometric center for cropping when masks are unavailable.

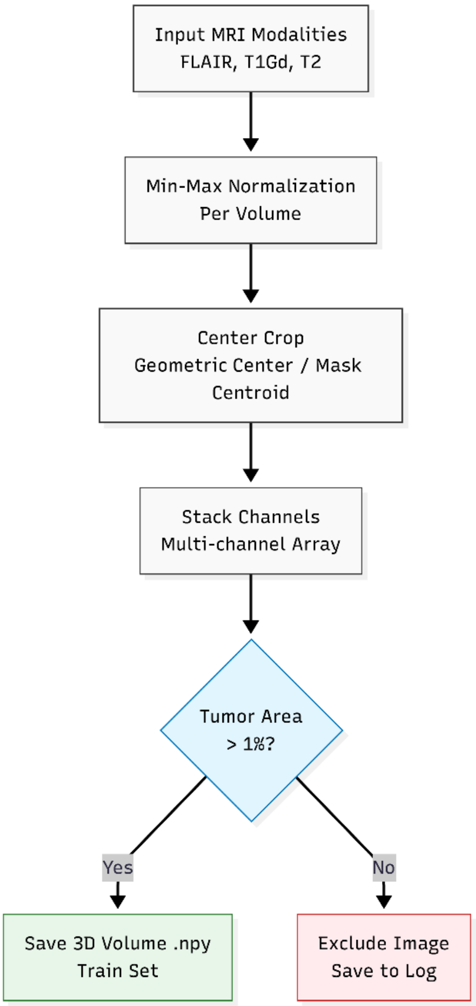


Figure 9 Method 1 preprocessing diagram

4.2.1. Exclusion on T1 modality

Recent studies have found substantial redundancy among the MRI sequences for glioma segmentation tasks. In particular, the native T1 often provides little unique benefit when T1Gd, T2, and FLAIR is available. One study observed that plain T1 was the weakest single-modality input – models using only T2 or FLAIR outperformed T1-only models, and T1Gd alone yielded the highest overlap scores among single sequences (Farhan et al., 2025). This suggests that much of the anatomical detail in T1 is already captured or superseded by the other sequences, especially by T1Gd for TC and by FLAIR/T2 for ED. These findings align with the intuition that T1 and T1Gd largely overlap in highlighting the TC, while FLAIR and T2 overlap in showing the tumor's edematous extent. In short, there is significant duplicative information across the four MRI channels, raising the question of whether all are needed for accurate segmentation (De Sutter et al., 2024).

Another study evaluated U-Net models on BraTS data with different modality subsets, found that a model using only T1Gd + FLAIR matched or exceeded the performance of a model using all four modalities. In their results, the two-sequence combination of T1Gd and FLAIR achieved slightly higher Dice scores for TC and ET than the full four-sequence model, both on cross-validation and an external test set. They conclude that its possible to achieve high accuracy using only two T1Gd + FLAIR, with no need for the native T1 or T2 to get near-optimal results. This reduced-modality approach not only maintained accuracy but was suggested to improve model generalizability and practicality by lowering data requirements (J. Huang et al., 2025).

Based on this evidence and the hardware/time constraints, T1 was excluded and used only T1Gd, FLAIR, and T2 modalities.

4.2.2. N4 Bias Field Correction

MRI images often suffer from intensity inhomogeneity called bias field, manifested as slow variations in brightness across the image. This low-frequency artifact makes the same tissue appear with different intensities in different locations, hindering quantitative analysis and potentially inducing errors in glioma segmentation algorithms. Bias correction aims to eliminate these spurious variations, making tissue intensities more homogeneous and comparable, thereby improving image quality and the reliability of subsequent segmentation. Let $v(x)$ be the observed image intensity at voxel x, modelled as in Equation 4.1.

$$v(x) = u(x)f(x) + n(x) \tag{4.1}$$

Where $u(x)$ is the true tissue intensity, $f(x)$ low-frequency multiplicative bias field, and $n(x)$ the additive noise. Applying the natural logarithm (assuming $n(x) \approx 0$) it results in the Equation 4.2 (Tustison et al., 2010).

$$log\, v(x) = log\, u(x) + log\, f(x) \tag{4.2}$$

The N4ITK (N4 Bias Field Correction) algorithm was proposed as an improvement over the N3 method, introducing a more robust B-spline fitting routine and a multi-resolution scheme to iteratively refine the correction. The algorithm assumes that the low-frequency distortion can be viewed as the convolution of the image intensity histogram with a Gaussian. Thus, the iterative process alternates between: (i) deconvolving the histogram to remove the broadening caused by bias), (ii) remapping image intensities based on the deconvolution, and (iii) fitting and smothing the estimated bias field using a low-frequency B-spline surface (ANTsX, 2021).

The preprocessing pipeline is designed to prepare the BraTS 2023 dataset for efficient and effective training for the deep learning model. The pipeline includes the following steps:

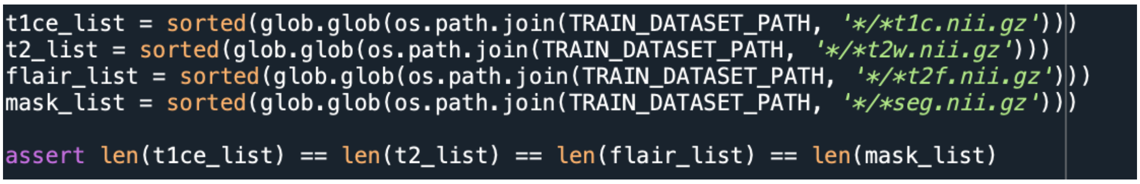
4.2.3. Data Loading and Organization

It started by locating where the raw training and validation data are and organizing into structured directories for training, testing, and validation. The training set includes the three MRI modalities (FLAIR, T1GD, T2) and segmentation masks, while validation set contains only the modalities, with no ground-truth masks.
To structure the output, specific directories are created for processed training images, masks, and validation images as shown in Listing 1.

```
output_dir = 'processed_data_v2/'
output_dir_images = os.path.join(output_dir, 'images')
output_dir_masks = os.path.join(output_dir, 'masks')
os.makedirs(output_dir_images, exist_ok=True)
os.makedirs(output_dir_masks, exist_ok=True)

output_dir_validation = 'processed_validation_data_v2/'
output_dir_images_validation = os.path.join(output_dir_validation, 'images')
os.makedirs(output_dir_images_validation, exist_ok=True)
```

Listing 1: Directory setup for processed data (train/test/validation).

Once the directories are prepared, the script compiles file lists for each modality and the corresponding segmentation masks using the glob library as seen in Listing 2. Each list is sorted to ensure a consistent and reproducible ordering of files across modalities. Since BraTS training data typically includes a complete set of modalities and a segmentation mask for each patient, the script includes an assertion to confirm that all four lists are the same length. This validation step ensures that each patient's data is complete and correctly aligned.



```
t1ce_list = sorted(glob.glob(os.path.join(TRAIN_DATASET_PATH, '*/*t1c.nii.gz')))
t2_list = sorted(glob.glob(os.path.join(TRAIN_DATASET_PATH, '*/*t2w.nii.gz')))
flair_list = sorted(glob.glob(os.path.join(TRAIN_DATASET_PATH, '*/*t2f.nii.gz')))
mask_list = sorted(glob.glob(os.path.join(TRAIN_DATASET_PATH, '*/*seg.nii.gz')))

assert len(t1ce_list) == len(t2_list) == len(flair_list) == len(mask_list)
```

Listing 2: Use of the glob library to compile file lists for each modality and masks.

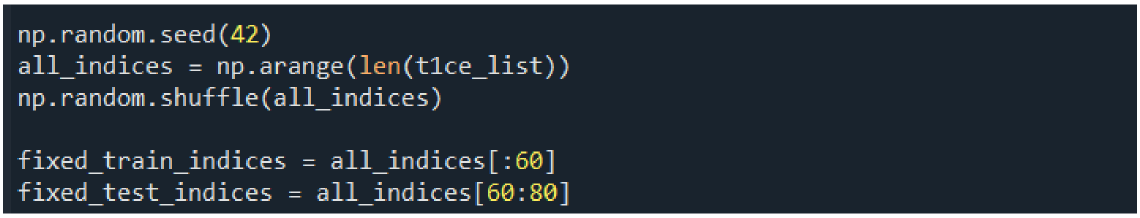
4.2.4. Data Splitting and Validation Handling

The section of the code shown in Listing 2 handles the partitioning of the dataset into training, test, and validation subsets. Originally, the intention was to divide the BraTS 2023 data into three subsets following a 90/10/10 ratio for training, testing, and validation, respectively. However, since a dedicated test set was not available, it had to be extracted from the official training set. The dataset was initially split using scikit-learn's train_test_split() method, which offered a standard and reproducible way to create a training/testing division.

```
train_indices, test_indices = train_test_split(range(len(t1ce_list)), test_size=0.1, random_state=42)
```

Listing 3: Initial random split with scikit-learn's train_test_split().

Despite this initial strategy, it became evident that the provided BraTS training and validation datasets were extremely large, each containing hundreds of high-resolution, multi-modal 3D volumes. Processing the full dataset required significant memory and time, which posed substantial computational challenges during development. To overcome this limitation and ensure that experiments could be conducted efficiently on standard hardware, a simplified and lighter setup was adopted for the final implementation: a fixed and reduced subset of 84 patient cases was selected using NumPy's shuffling mechanism, replacing the previous dynamic splitting logic. This subset included 60 scans for training, 20 for internal testing, and 4 from the validation cohort for internal validation, as shown in Listing 4.

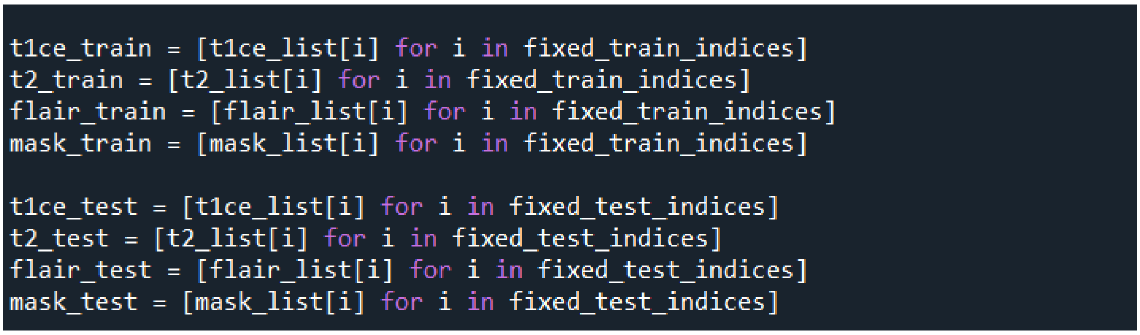


```
np.random.seed(42)
all_indices = np.arange(len(t1ce_list))
np.random.shuffle(all_indices)

fixed_train_indices = all_indices[:60]
fixed_test_indices = all_indices[60:80]
```

Listing 4: Fixed-seed subset selection with NumPy for reproducible training/testing indices.

The use of `np.random.seed(42)` ensures that the same subset of cases is consistently selected across executions, allowing direct comparison of different preprocessing methods on an identical set of images and facilitating fair, reproducible evaluation. A posteriori, it became clear that the random subset may not guarantee strict comparability between cases, like the presence and size of ET, which weakens the assurance that all samples are on equal footing for evaluation. Due to time and computational constraints, it was not feasible to re-implement a stratified sampling strategy. Since the BraTS dataset is carefully curated and standardized, large systematic differences are unlikely, though some imbalance may still exist.

After generating the randomized indices, the selected file paths for each MRI modality and their corresponding segmentation masks are extracted based on these indices. The variables `fixed_train_indices` and `fixed_test_indices` contain the shuffled list positions used to define the training and testing samples, respectively. Using list comprehensions, each modality (T1GD, T2, FLAIR) and the mask is sliced to form the final subsets as shown in Listing 5.



```
t1ce_train = [t1ce_list[i] for i in fixed_train_indices]
t2_train = [t2_list[i] for i in fixed_train_indices]
flair_train = [flair_list[i] for i in fixed_train_indices]
mask_train = [mask_list[i] for i in fixed_train_indices]

t1ce_test = [t1ce_list[i] for i in fixed_test_indices]
t2_test = [t2_list[i] for i in fixed_test_indices]
flair_test = [flair_list[i] for i in fixed_test_indices]
mask_test = [mask_list[i] for i in fixed_test_indices]
```

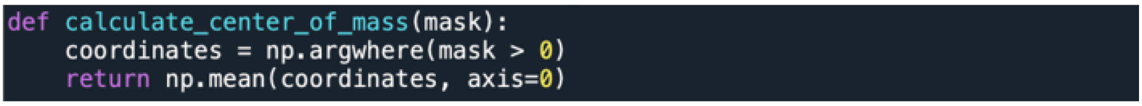
Listing 5: Building train/test subsets by indexing the sorted modality and mask lists.

This indexing approach ensures that all modalities and masks belonging to the same patient are selected consistently across splits. By referencing the same indices across all four lists, it guarantees that each training or testing sample consists of the correct and complete set of multimodal inputs along with its corresponding segmentation mask. This logic is essential for maintaining patient-level coherence during the preprocessing and training stages.

The same procedure is applied to the validation set, with the only difference being that segmentation masks are not available for these cases.

4.2.5. Center of Mass Calculation

Since most tumors occupy only a small portion of the total brain volume, processing the entire 3D image is computationally inefficient and often includes irrelevant regions. To avoid this, the preprocessing pipeline extracts a ROI around the tumor. As shown in Listing 6, the center of this ROI is determined by computing the spatial centroid, the center of mass, of all positive voxels in the segmentation mask. In this work, all 3D volumes are handled as NumPy arrays with axis order (Z, Y, X). Here, Z corresponds to the axial dimension, Y to the vertical direction within each slice, and X to the horizontal direction.



```
def calculate_center_of_mass(mask):
    coordinates = np.argwhere(mask > 0)
    return np.mean(coordinates, axis=0)
```

Listing 6: Center of mass function computes the centroid of all positive voxels in the binary tumur mask.

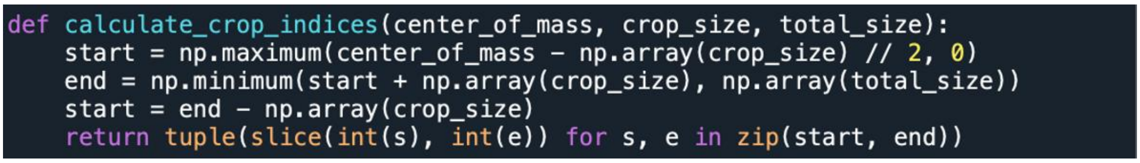
However, this initial version operated under the assumption that tumor masks would always contain non-zero values, calculating the mean position of all positive voxels. While effective in most cases, this approach breaks when the mask is empty for instance, due to images without tumors. Later, when these invalid values are used to crop the image, the code fails and stops the processing.

```python
def calculate_center_of_mass(mask):
    coordinates = np.argwhere(mask > 0)
    return np.mean(coordinates, axis=0) if len(coordinates) > 0 else np.array(mask.shape) // 2
```

Listing 7: Edited function of the center of mass.

To address this, it was implemented a conditional statement that checks if any non-zero coordinates exist. When positive voxels are present, it proceeds with the standard center of mass calculation and when faced with an empty result, no positive voxels, it falls back to using the geometric center of the image volume. When segmentation masks are not available, like in the validation set, it is not possible to localize the tumor directly. In these cases, the pipeline uses the geometric center of the image as a fallback. Although this does not guarantee the tumor is centered in the crop, it ensures that the image can still be processed without failure. A more robust strategy, used in a later version (Method 2) estimates the tumor location based on image intensity, which improves the likelihood of including the tumor in the cropped region.

Later, it was identified that the logic for handling empty masks and defaulting to the image center was being duplicated across different parts of the pipeline. To improve maintainability and reduce redundancy, the function was modified to incorporate all fallback logic within a single, centralized implementation as shown in Listing 8.



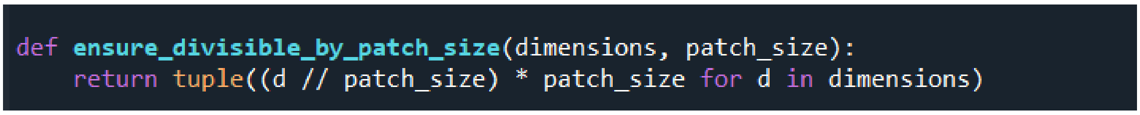
```python
def calculate_crop_indices(center_of_mass, crop_size, total_size):
    start = np.maximum(center_of_mass - np.array(crop_size) // 2, 0)
    end = np.minimum(start + np.array(crop_size), np.array(total_size))
    start = end - np.array(crop_size)
    return tuple(slice(int(s), int(e)) for s, e in zip(start, end))
```

Listing 8: Crop-index calculator

A crop window is centered on the computed center-of-mass and the start index is set as the center minus half the crop size. To ensure that the crop remains entirely within the image boundaries, the indices are adjusted accordingly. Because center_of_mass and subsequent calculations might produce floating-point values, int(s) and int(e) simply cast those values to integers so they can be used as valid slice boundaries.

Since reliable segmentation masks were available, it seemed reasonable that the center-of-mass could be accurately computed based solely on these masks. This approach offered a significant advantage in its straightforward implementation and computational efficiency. However, while this approach functioned effectively for the train and test sets where ground truth segmentation masks were available, it revealed a significant limitation when applied to the validation set, which did not include masks.

Without annotation, the original method could no longer determine the tumor's location, highlighting the need for a more general and robust solution capable of functioning in fully unsupervised scenarios.

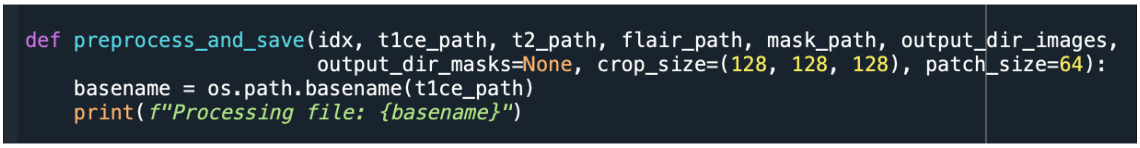


```
def ensure_divisible_by_patch_size(dimensions, patch_size):
    return tuple((d // patch_size) * patch_size for d in dimensions)
```

Listing 9: Patch-size adjustment

4.2.6. Preprocess function

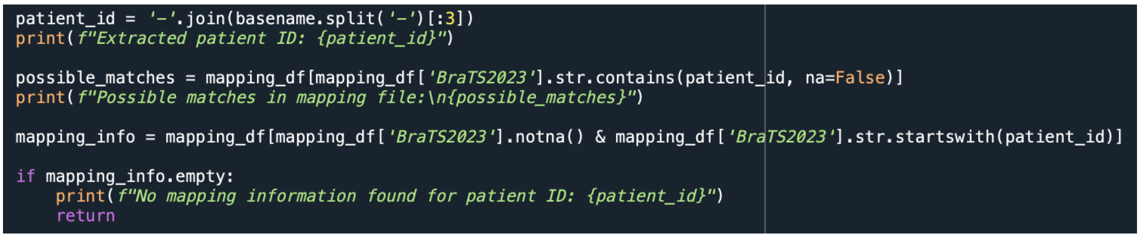
The preprocess_and_save function, shown in Listing 10, is responsible for reading raw multimodal MRI volumes, applying intensity normalization, cropping the image around the tumor region, or approximate center, and saving the processed outputs to disk.



```
def preprocess_and_save(idx, t1ce_path, t2_path, flair_path, mask_path, output_dir_images,
                        output_dir_masks=None, crop_size=(128, 128, 128), patch_size=64):
    basename = os.path.basename(t1ce_path)
    print(f"Processing file: {basename}")
```

Listing 10: Function header and defaults parameters.

For each patient, the function receives file paths for the modalities, an optional segmentation mask, output directories, and default parameters for crop size (128×128×128) and patch size (64).

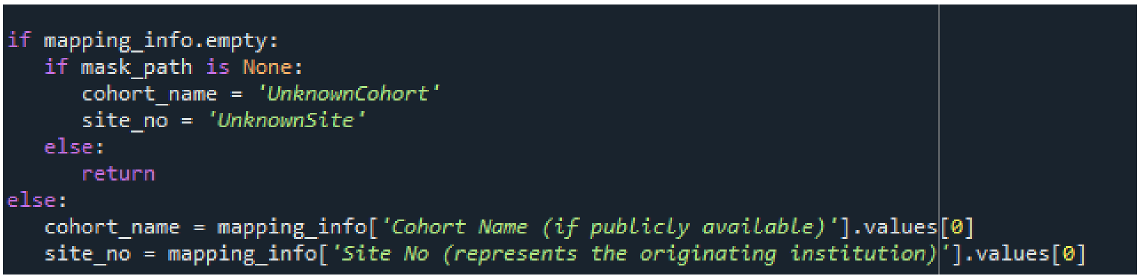


```
patient_id = '-'.join(basename.split('-')[:3])
print(f"Extracted patient ID: {patient_id}")

possible_matches = mapping_df[mapping_df['BraTS2023'].str.contains(patient_id, na=False)]
print(f"Possible matches in mapping file:\n{possible_matches}")

mapping_info = mapping_df[mapping_df['BraTS2023'].notna() & mapping_df['BraTS2023'].str.startswith(patient_id)]

if mapping_info.empty:
    print(f"No mapping information found for patient ID: {patient_id}")
    return
```

Listing 11: Mapping lookup and file naming.

As illustrated in Listing 11, the patient ID is taken directly from the files name by splitting the name into parts and joining the first three, for example BraTS-GLI-00458. This ID is then used to look for the patient in the mapping file, which contains the cohort and site information. In the above code, if no match was found in the mapping file the function stopped and did not save anything, like the validation images that has no mapping information. The output files were also named using the loop index, such as image_{idx}_{cohort}_{site}.npy, which made it hard in the end to compare the same patient across different methods.

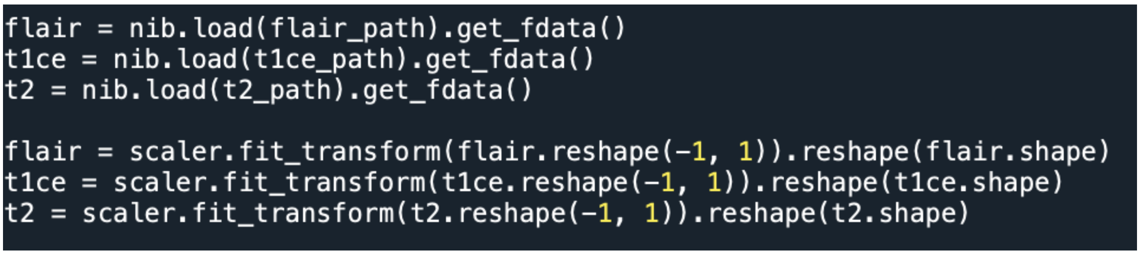
In the revised code, in the validation set, if there is no match in the mapping file the function now keeps going, uses “UnknownCohort” and “UnknownSite” as defaults, and saves the file using only the patient ID as the name as shown in Listing 12. This makes sure that the same validation case will have the same name in all methods, which makes visual and numerical comparisons easy. For the training and test sets, the behaviour stays the same if there is no mapping: the case is skipped, and a short message is shown. If the mapping information is found, the file is saved with a name that includes the patient ID plus the cohort and site (ex.: BraTS-GLI-00458_CPTAC-GBM_20.npy), and the mask file gets the same name with `_mask` added.



```
if mapping_info.empty:
   if mask_path is None:
      cohort_name = 'UnknownCohort'
      site_no = 'UnknownSite'
   else:
      return
else:
   cohort_name = mapping_info['Cohort Name (if publicly available)'].values[0]
   site_no = mapping_info['Site No (represents the originating institution)'].values[0]
```

Listing 12: Handling missing mapping entries.

As seen in Listing 13, each MRI modality is loaded from its respective NIfTI file via Nibabel, producing NumPy arrays. These volumes often vary in intensity range, so the script reshapes each into a two-dimensional array (-1, 1) and applies a MinMaxScaler to rescale intensities to [0, 1]. Afterwards, each array is reshaped back to its original 3D form. This approach normalizes each modality per image independently, which can lead to intensity inconsistencies across the dataset.



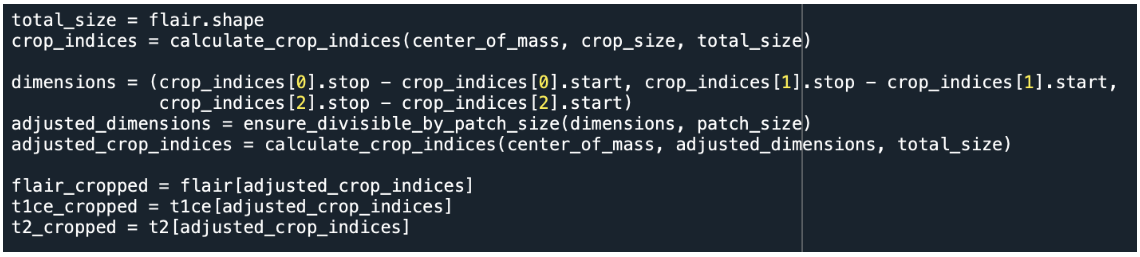
```
flair = nib.load(flair_path).get_fdata()
t1ce = nib.load(t1ce_path).get_fdata()
t2 = nib.load(t2_path).get_fdata()

flair = scaler.fit_transform(flair.reshape(-1, 1)).reshape(flair.shape)
t1ce = scaler.fit_transform(t1ce.reshape(-1, 1)).reshape(t1ce.shape)
t2 = scaler.fit_transform(t2.reshape(-1, 1)).reshape(t2.shape)
```

Listing 13: Modality loading and per-volume min–max normalisation (FLAIR, T1Gd, T2).

```
if mask_path is not None:
    mask = nib.load(mask_path).get_fdata().astype(np.uint8)
    center_of_mass = calculate_center_of_mass(mask)
else:
    mask = None
    center_of_mass = np.array(flair.shape) // 2
```

Listing 14: Mask loading and crop center selection: centroid of the mask when available, geometric center otherwise.

If a segmentation mask is provided, it is loaded and cast to an 8-bit integer array. The function calculate_center_of_mass is then used to determine the tumor's center based on all non-zero voxels. If no mask is provided, or if the mask contains no tumor voxels, the function falls back to using the geometric center of the volume as shown in Listing 14.

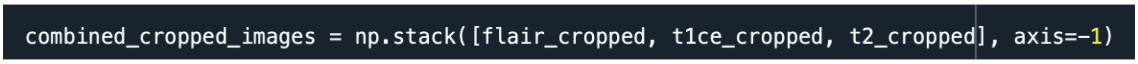


```
total_size = flair.shape
crop_indices = calculate_crop_indices(center_of_mass, crop_size, total_size)

dimensions = (crop_indices[0].stop - crop_indices[0].start, crop_indices[1].stop - crop_indices[1].start,
              crop_indices[2].stop - crop_indices[2].start)
adjusted_dimensions = ensure_divisible_by_patch_size(dimensions, patch_size)
adjusted_crop_indices = calculate_crop_indices(center_of_mass, adjusted_dimensions, total_size)

flair_cropped = flair[adjusted_crop_indices]
t1ce_cropped = t1ce[adjusted_crop_indices]
t2_cropped = t2[adjusted_crop_indices]
```

Listing 15: Calculation of crop limits and patch-size adjustment.

Once the center of mass is established, the code determines the maximum allowable region to extract. It does so by calling calculate_crop_indices, which returns the start and end positions of the crop in each dimension, ensuring that the selected region stays within the image boundaries. The dimensions of this initial crop are then inspected, and a helper function (ensure_divisible_by_patch_size) adjusts them so that the final cropped volume is a multiple of the chosen patch size. A second call to calculate_crop_indices re-computes the slicing bounds based on these adjusted dimensions as shown in Listing 15.

Using these refined indices, the script slices each modality (FLAIR, T1GD, and T2) around the tumor region or its approximate center. This results in three uniform 3D sub-volumes, which are then stacked along the last axis, the channel dimension, producing a single array that holds all three modalities together as demonstrated in Listing 16.

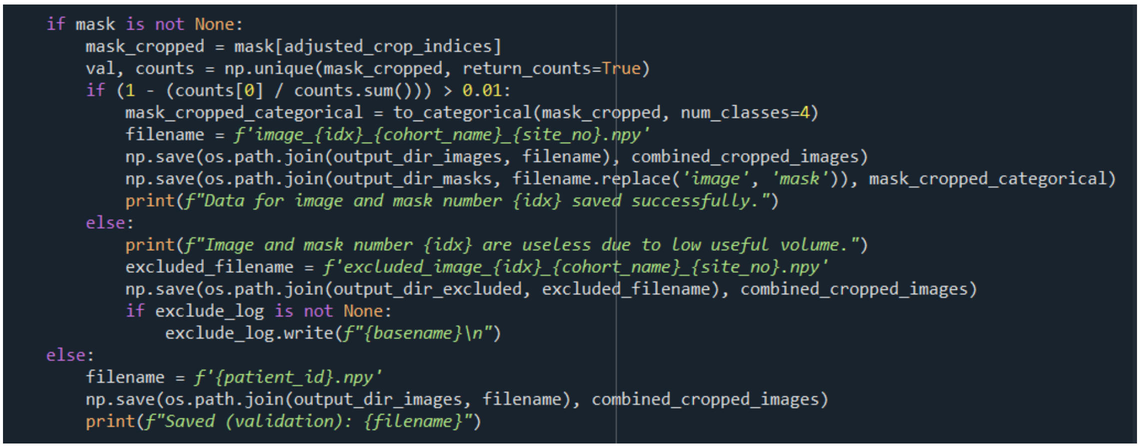


```
combined_cropped_images = np.stack([flair_cropped, t1ce_cropped, t2_cropped], axis=-1)
```

Listing 16: Stacking the cropped FLAIR, T1Gd and T2 into a single array.

For images with corresponding masks, the mask was first cropped to align with the dimensions of the processed image. Class balance was then assessed by calculating the proportion of non-background voxels within the mask—i.e., voxels indicating tumor presence. This evaluation step is particularly relevant in medical image segmentation, where tumor regions often comprise only a small portion of the entire volume.

As illustrated in Listing 17, to maintain dataset quality, a threshold of 1% tumor presence was employed. If the proportion of tumor voxels exceeded this threshold, the sample was deemed informative and retained for further processing. The segmentation mask was then converted to a categorical format using one-hot encoding to facilitate multi-class segmentation. Both the processed image and the categorical mask were saved as NumPy arrays, with filenames including the sample index, cohort name, and site number to ensure traceability and reproducibility. Samples falling below the 1% tumor threshold were saved in a separate directory as “excluded” cases. These crops were stored without their corresponding masks and were named only with a sequential index plus cohort/site (e.g., excluded_image_7_Private Collection_1.npy), rather than with the full BraTS identifier. This design broke traceability and later caused mis-matched overlays.



```
if mask is not None:
    mask_cropped = mask[adjusted_crop_indices]
    val, counts = np.unique(mask_cropped, return_counts=True)
    if (1 - (counts[0] / counts.sum())) > 0.01:
        mask_cropped_categorical = to_categorical(mask_cropped, num_classes=4)
        filename = f'image_{idx}_{cohort_name}_{site_no}.npy'
        np.save(os.path.join(output_dir_images, filename), combined_cropped_images)
        np.save(os.path.join(output_dir_masks, filename.replace('image', 'mask')), mask_cropped_categorical)
        print(f"Data for image and mask number {idx} saved successfully.")
    else:
        print(f"Image and mask number {idx} are useless due to low useful volume.")
        excluded_filename = f'excluded_image_{idx}_{cohort_name}_{site_no}.npy'
        np.save(os.path.join(output_dir_excluded, excluded_filename), combined_cropped_images)
        if exclude_log is not None:
            exclude_log.write(f"{basename}\n")
else:
    filename = f'{patient_id}.npy'
    np.save(os.path.join(output_dir_images, filename), combined_cropped_images)
    print(f"Saved (validation): {filename}")
```

Listing 17: Crop of the mask and initial 1% tumor check for saving or exclusion.

The threshold itself was intended to reduce the dataset size by discarding volumes considered “uninformative,” keeping only crops with a clear tumor signal to make training more efficient. In practice, this strategy failed: many excluded crops contained visible tumor tissue, removed only because the tumor occupied less than 1% of the crop. Excluding such cases also eliminated valuable variability. In clinical settings, small tumors, post-surgical cavities, or subtle ED are common, and robust models must be exposed to them. Figures show that excluded crops still contained pathology, and when mismatched masks were overlaid, the error became obvious.

For these reasons, the threshold rule was abandoned. Retaining all cases makes the dataset richer and more realistic: it contains both strong tumor examples and cases with little or no tumor. For neural network training, this broader spectrum improves sensitivity to true pathology, reduces false positives, and supports better generalization across patients and sites. In short, the dataset is stronger when it reflects the full clinical range, rather than being filtered to only the “easy” tumor cases.

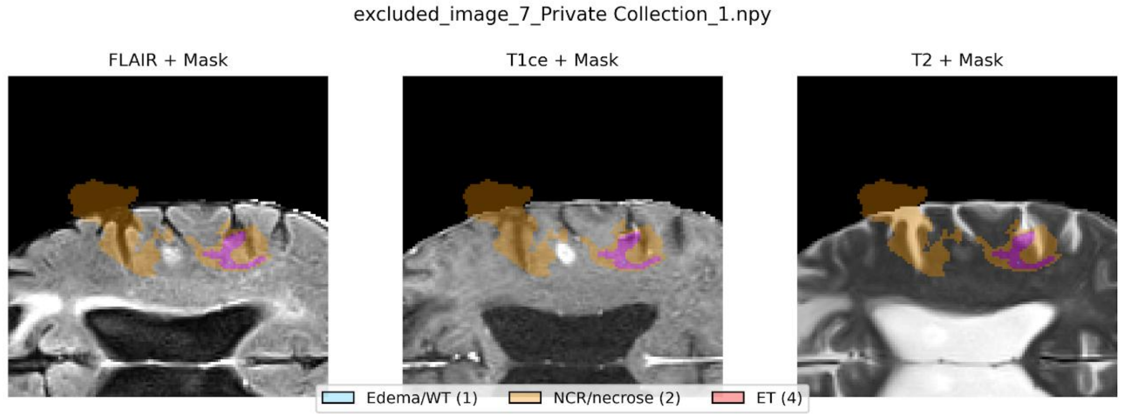


Figure 10: Example of an excluded crop overlaid with a wrong mask, due to the missing patient ID and missing cropped mask.

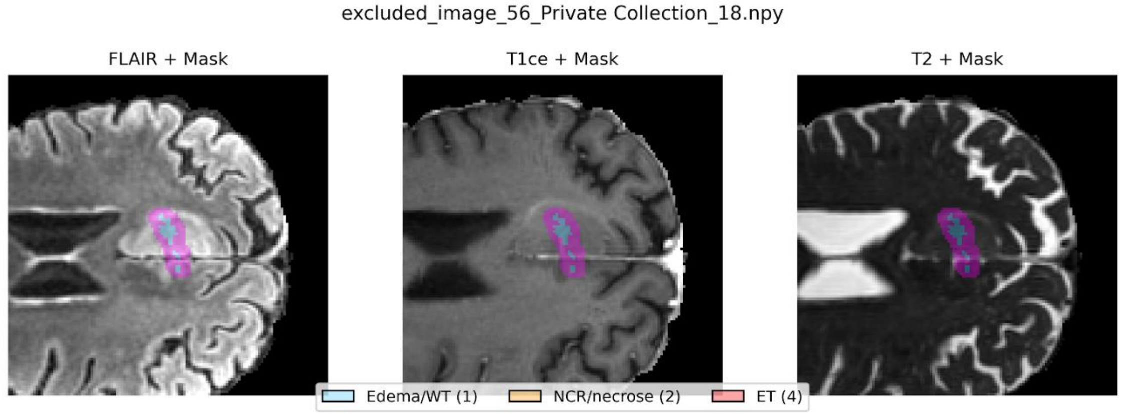
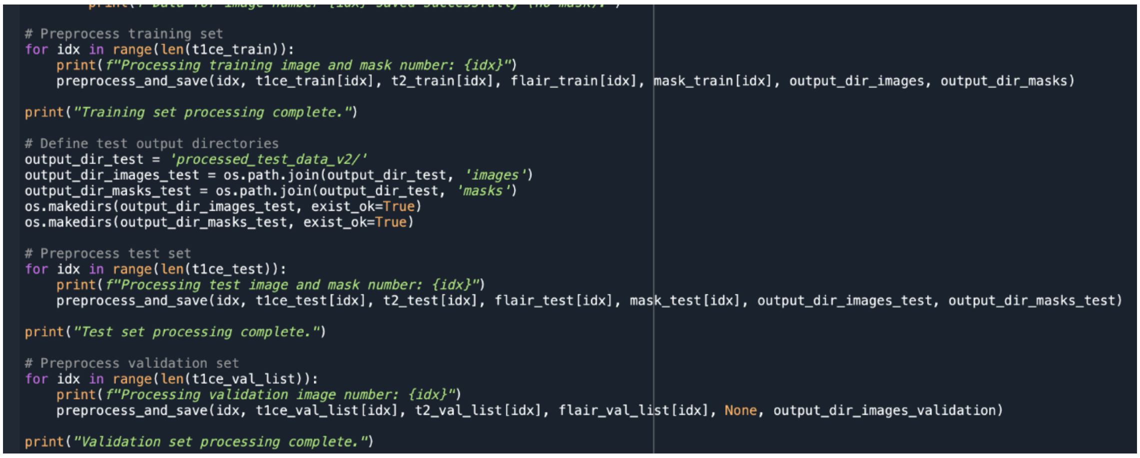


Figure 11: Example of an excluded crop with an overlay from the wrong subject.

Method 2 corrected these limitations by always cropping the mask together with the images, keeping all cases (no 1% exclusion), and ensuring consistent filenames even when cohort/site metadata was missing, as shown in Figure 10-11. Adopting the full BraTS-GLI-XXXX-YYYY identifier in filenames/logs would eliminate the residual ambiguity across both methods.

4.2.7. Implementation of Dataset Processing

The dataset processing is implemented through a modular structure that handles the training, testing, and validation subsets using separate loops. Each subset undergoes the same preprocessing steps, ensuring consistency across partitions while preserving strict separation to avoid data leakage during model evaluation.



```
# Preprocess training set
for idx in range(len(t1ce_train)):
    print(f"Processing training image and mask number: {idx}")
    preprocess_and_save(idx, t1ce_train[idx], t2_train[idx], flair_train[idx], mask_train[idx], output_dir_images, output_dir_masks)

print("Training set processing complete.")

# Define test output directories
output_dir_test = 'processed_test_data_v2/'
output_dir_images_test = os.path.join(output_dir_test, 'images')
output_dir_masks_test = os.path.join(output_dir_test, 'masks')
os.makedirs(output_dir_images_test, exist_ok=True)
os.makedirs(output_dir_masks_test, exist_ok=True)

# Preprocess test set
for idx in range(len(t1ce_test)):
    print(f"Processing test image and mask number: {idx}")
    preprocess_and_save(idx, t1ce_test[idx], t2_test[idx], flair_test[idx], mask_test[idx], output_dir_images_test, output_dir_masks_test)

print("Test set processing complete.")

# Preprocess validation set
for idx in range(len(t1ce_val_list)):
    print(f"Processing validation image number: {idx}")
    preprocess_and_save(idx, t1ce_val_list[idx], t2_val_list[idx], flair_val_list[idx], None, output_dir_images_validation)

print("Validation set processing complete.")
```

Listing 18: Training-set loop calling preprocess_and_save with modality and mask paths.

This section of the code processes the training, test, and validation subsets using structured loops and a unified preprocessing function as demonstrated in Listing 18.

For the training set, the script iterates over the selected training indices and passes the corresponding T1GD, T2, FLAIR, and segmentation mask files to the preprocess_and_save function. The function receives the output directories for processed images and masks, along with a log file used to record excluded samples.

Before processing the test set, output directories for test images and masks are defined and created using os.makedirs(). The test loop follows the same structure as the training loop. Each case is processed with all four input volumes (including the mask), and excluded samples are recorded in the same log file. This ensures consistent processing across both sets. The validation set is processed similarly, with one key difference: no segmentation masks are available. To reflect this, the mask argument is explicitly set to None when calling the function. This allows the function to bypass mask-related operations while applying the same preprocessing steps (e.g., normalization, cropping, and stacking) to the image data. Only the processed images are saved.

Each loop ends with a confirmation message to indicate the completion of processing for that dataset split.

After executing the preprocessing script, an unexpected issue was observed: the validation data directory remained empty despite successful preprocessing of training and testing datasets. This anomaly prompted further investigation into the underlying cause.

Upon detailed inspection of the preprocessing script, it was determined that the function preprocess_and_save relied on metadata from the mapping file (BraTS2023_2017_GLI_Mapping.xlsx) to validate patient IDs and retrieve necessary metadata (cohort names and site numbers). While this worked correctly for training and testing datasets, the validation dataset lacked corresponding entries in the mapping file and segmentation masks, causing the script to skip validation cases entirely.

Initially, the use of segmentation masks to compute the center of mass seemed appropriate and sufficient, especially in the context of the well-annotated BraTS training data. At that point, the assumption that masks would always be available was reasonable given the limited scope and experience. However, as the project progressed and shifted toward real-world validation scenarios—where annotations were absent—this design choice revealed itself as a limitation. Recognizing this constraint led to the development of a more flexible and robust approach, which reflects a natural and important part of the learning and research process.

Also, the normalization method initially implemented was reconsidered. The approach used was normalizing each modality per image independently, potentially leading to intensity inconsistencies across the dataset: it rescales each scan to [0,1] using that image's own minimum and maximum, which makes it sensitive to outliers and can yield inconsistent intensity distributions across scans (Scikit, 2025). Therefore, MinMaxScaler was replaced with the Z-score normalization method, addressing these concerns and improving the overall robustness of the preprocessing pipeline.

The identified issues were thoroughly analyzed, and the corrections, including the normalization method update, were integrated into the second preprocessing method (method 2), thus ensuring proper handling and processing of validation cases and improved dataset consistency.

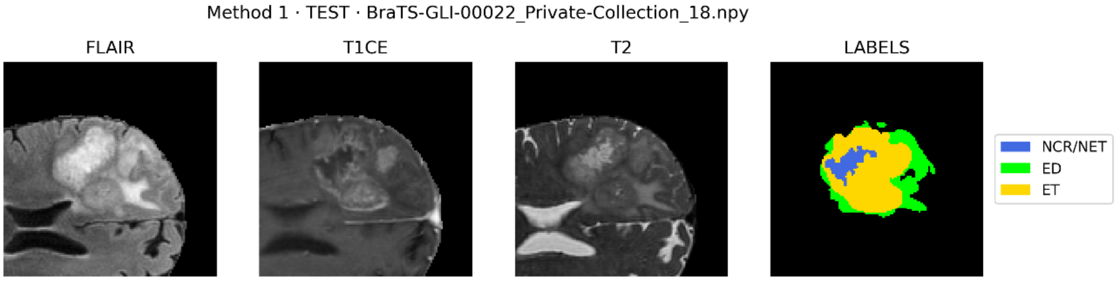


Figure 12: Example of FLAIR, T1GD, and T2 slices from the test set, with the corresponding segmentation labels showing necrotic core, ED, and enhancing tumor.

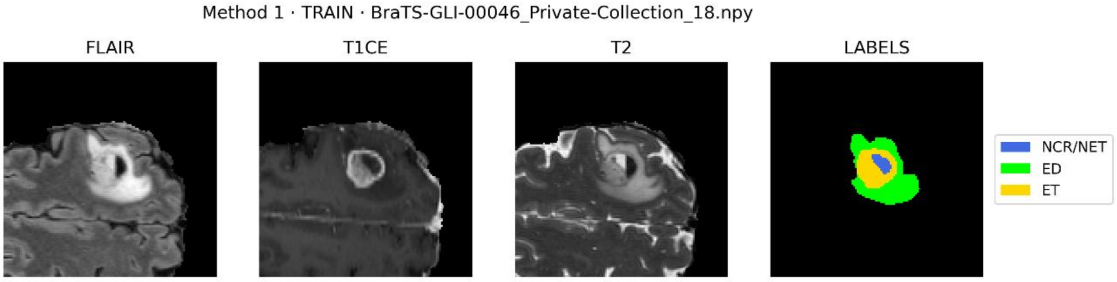


Figure 13: Example of FLAIR, T1GD, and T2 slices from the train set, with the corresponding segmentation labels showing necrotic core, ED, and enhancing tumor.

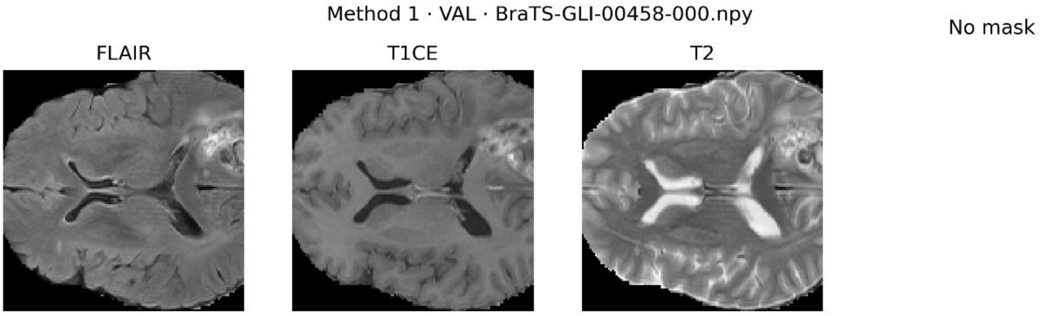


Figure 14: Example of FLAIR, T1GD, and T2 slices from the validation set. No segmentation mask is provided.

4.3. Method 2

Building upon Method 1, the second approach introduces structural changes to address the identified limitations regarding intensity scaling and crop positioning. The updated pipeline for Method 2, which incorporates Z-score normalization and T1Gd-guided centering, is schematically represented in Figure 15.

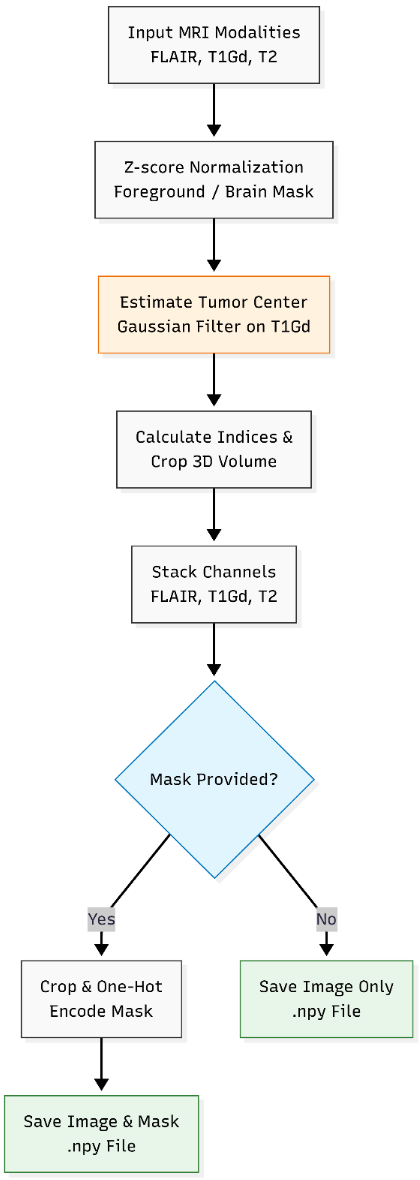


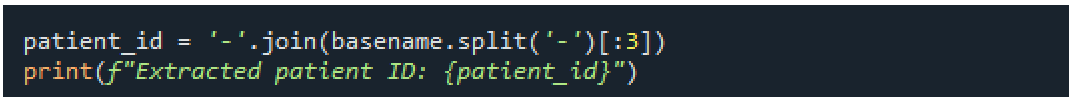
Figure 15 Method 2 preprocessing diagram

4.3.1. Transition from Method 1 to Method 2

Given the constraints posed by the BraTS 2023 validation set—which lacks segmentation masks—an updated preprocessing strategy was developed to accommodate these cases while maintaining comparability across data splits. Method 2 was built directly from the codebase of Method 1 but adapted to introduce four core improvements:

4.3.2. Fixing the patient identifier

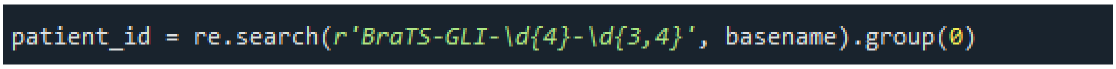
The evolution of the patient-ID extraction is summarised in Listings 19 to 21. Method 1 used a partial identifier, Method 2 first attempted a pattern-matching search, and the final version extracts the ID directly from the case-folder name.



```
patient_id = '-'.join(basename.split('-')[:3])
print(f"Extracted patient ID: {patient_id}")
```

Listing 19: Original patient-ID extraction using the case-folder name

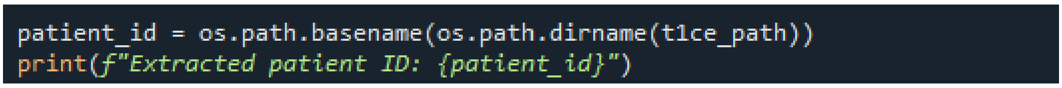
This gave IDs like BraTS-GLI-00166 and cut off the last part. Because of this, different patients could end up with the same ID. For excluded cases, the filenames had only a running number (e.g., excluded_image_7_Private Collection_1.npy) and no patient ID at all. This made it very hard to know which subject the file belonged to and caused mistakes when trying to match images with masks later. In Method 2, the first attempt was to use a regular expression (regex) to capture the full BraTS ID, for example BraTS-GLI-00166-000.



```
patient_id = re.search(r'BraTS-GLI-\d{4}-\d{3,4}', basename).group(0)
```

Listing 20: Pattern-matching attempt to capture the full BraTS identifier (Method 2, first fix).

Some file paths did not exactly match the pattern, so the regex returned nothing, and the program crashed with an error. The final and simplest solution was to use the folder name as the ID. In the BraTS dataset, each patient is stored in a folder already named with the full identifier (e.g., BraTS-GLI-00166-000).

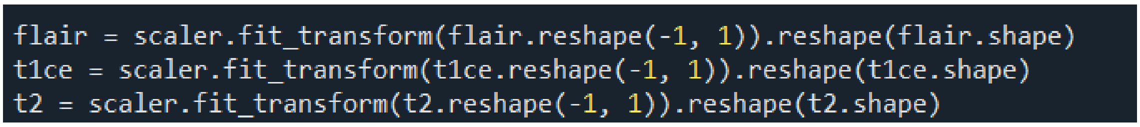


```
patient_id = os.path.basename(os.path.dirname(t1ce_path))
print(f"Extracted patient ID: {patient_id}")
```

Listing 21: Final solution in Method 2: extract patient ID from the case-folder name

4.3.3. Substituting MinMaxScaler with Z-score normalization

The normalization procedure in Method 1, illustrated in Listing 22, scales the voxel intensities of the three modalities (FLAIR, T1Gd, T2) to the range [0, 1] using the MinMaxScaler from scikit-learn.

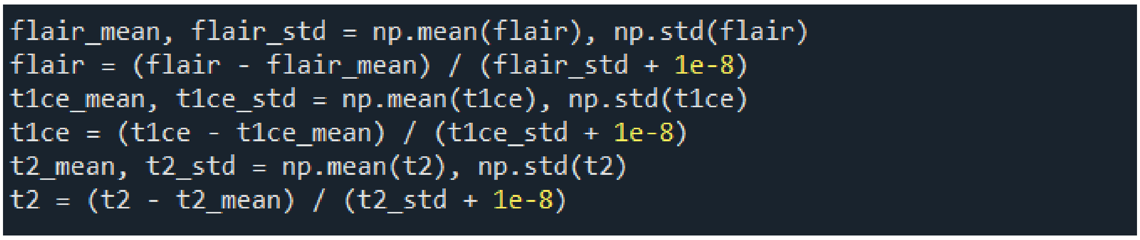


```
flair = scaler.fit_transform(flair.reshape(-1, 1)).reshape(flair.shape)
t1ce = scaler.fit_transform(t1ce.reshape(-1, 1)).reshape(t1ce.shape)
t2 = scaler.fit_transform(t2.reshape(-1, 1)).reshape(t2.shape)
```

Listing 22: Original Min–Max scaling per volume (Method 1).

The per-image Min–Max step maps each scan to [0, 1] based on its own minimum and maximum. Because the mapping depends on these image-specific extrema, the method is sensitive to outliers and its per-image range compression can yield inconsistent intensity distributions across scans (Scikit, 2025).

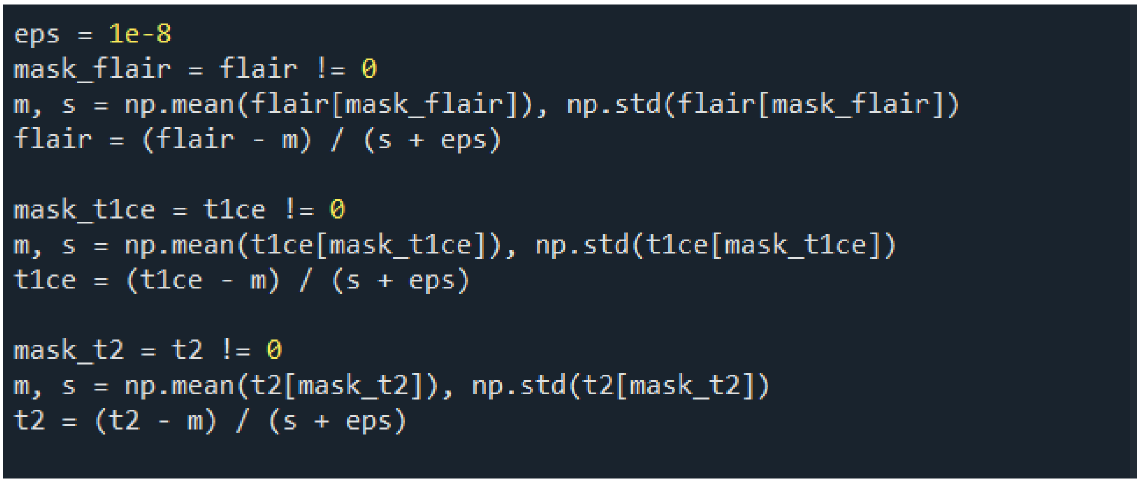
In method 2, Min–Max was replaced with Z-score normalization is applied per volume and per modality, where μ is the mean, σ the standard deviation (Cheng, 2025). This change follows common MRI preprocessing practice and helps make intensities more comparable across cases. First, μ and σ were computed over the entire volume. The normalization was applied before the lesion-center crop. This gives approximately zero mean and unit variance at the full volume.
Listing 23 shows the implementation of the Z-score normalization.



```
flair_mean, flair_std = np.mean(flair), np.std(flair)
flair = (flair - flair_mean) / (flair_std + 1e-8)
t1ce_mean, t1ce_std = np.mean(t1ce), np.std(t1ce)
t1ce = (t1ce - t1ce_mean) / (t1ce_std + 1e-8)
t2_mean, t2_std = np.mean(t2), np.std(t2)
t2 = (t2 - t2_mean) / (t2_std + 1e-8)
```

Listing 23: Z-score normalization per modality (Method 2).

After reviewing the literature (Ranjbarzadeh et al., 2021), the code was updated to compute μ and σ only on foreground voxels for each modality and then apply the transform to the whole volume before cropping. This avoids large background areas (zeros) from skewing the statistics and matches the idea of doing Z-score inside the brain region rather than in background. Listing 24 presents this masked Z-score step.



```
eps = 1e-8
mask_flair = flair != 0
m, s = np.mean(flair[mask_flair]), np.std(flair[mask_flair])
flair = (flair - m) / (s + eps)

mask_t1ce = t1ce != 0
m, s = np.mean(t1ce[mask_t1ce]), np.std(t1ce[mask_t1ce])
t1ce = (t1ce - m) / (s + eps)

mask_t2 = t2 != 0
m, s = np.mean(t2[mask_t2]), np.std(t2[mask_t2])
t2 = (t2 - m) / (s + eps)
```

Listing 24: Masked Z-score: compute μ and σ within a brain-only mask; set background to zero.

To evaluate the impact of normalization, two Z-score approaches were compared:

- **Whole volume**: statistics calculated using all voxels, including background;
- **Foreground only**: statistics calculated using only voxels inside the brain (non-zero values).

In the whole-volume approach, the large number of zero-intensity background voxels biased the results. This caused brain intensities to shift upward (mean around 2.0–2.5 instead of 0) and produced smaller, inconsistent standard deviations across subjects. With the foreground-only approach, brain intensities stayed centered (mean ≈ 0, std ≈ 1) and consistent across different patients and modalities, as illustrated in the histograms (Figure 16). For this reason, the pipeline adopts foreground-only Z-score normalization. In practice, the mean and standard deviation are computed only from brain voxels for each modality, and the transformation is applied to the entire volume before cropping around the lesion.

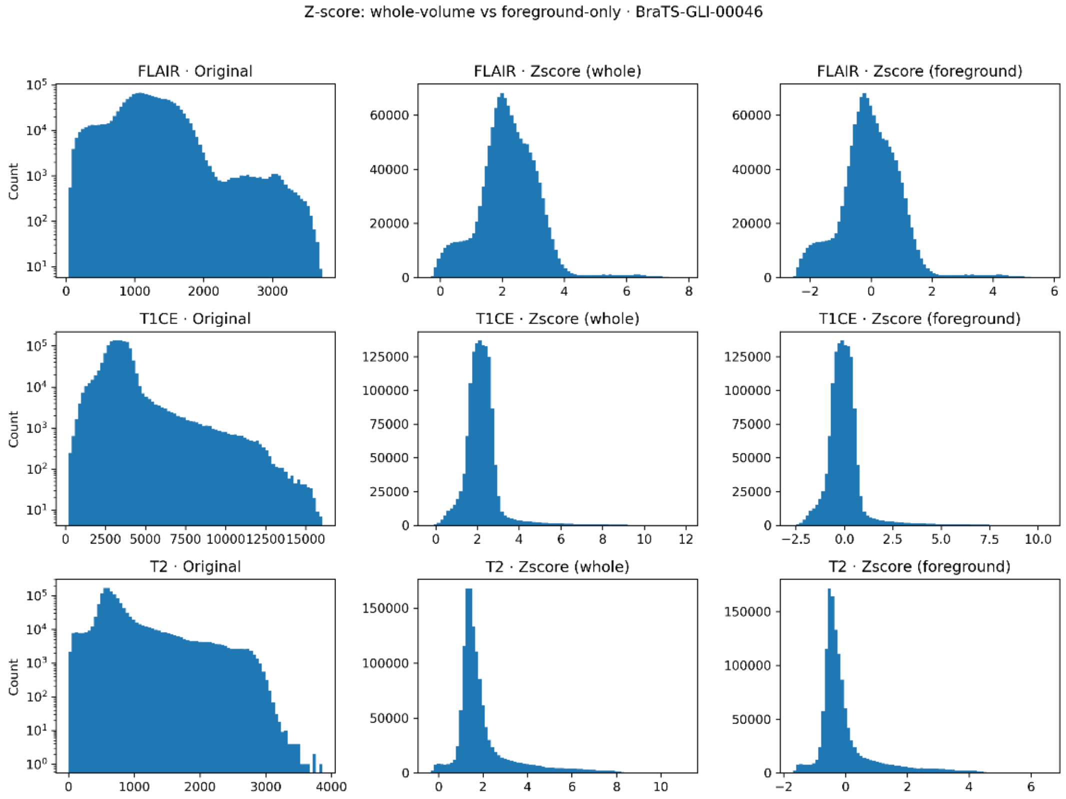
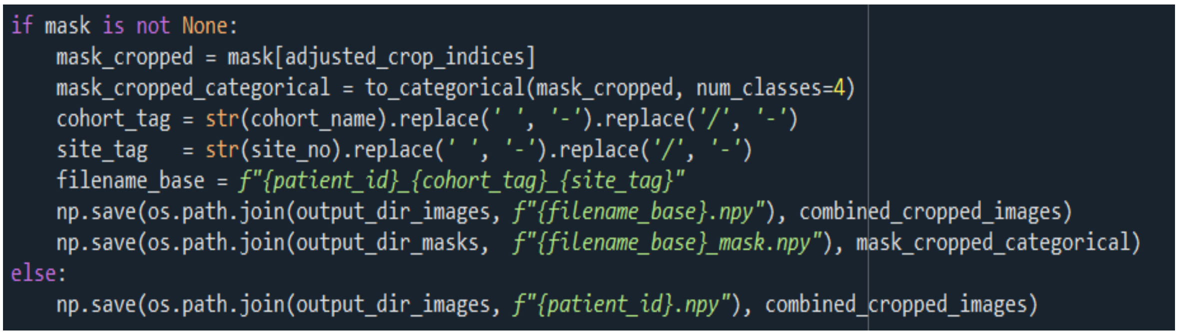


Figure 16: Histograms of FLAIR, T1GD, and T2 intensities before and after Z-score normalization in the whole brain and on the foreground.

4.3.4. Retention of all images regardless of tumor size

In Method 1, a logic block discarded images if the tumor area comprised less than 1% of the crop. This exclusion logic helped focus training on tumor-positive examples but created a critical flaw for processing the validation dataset, which contains no segmentation masks at all. As a result, those cases were skipped entirely. In Method 2, this logic was removed, and all images were saved regardless of tumor volume. This simplified implementation is shown in Listing 25 and ensures that cases with no annotation, such as those in the validation set, were still preprocessed and included in the pipeline output.

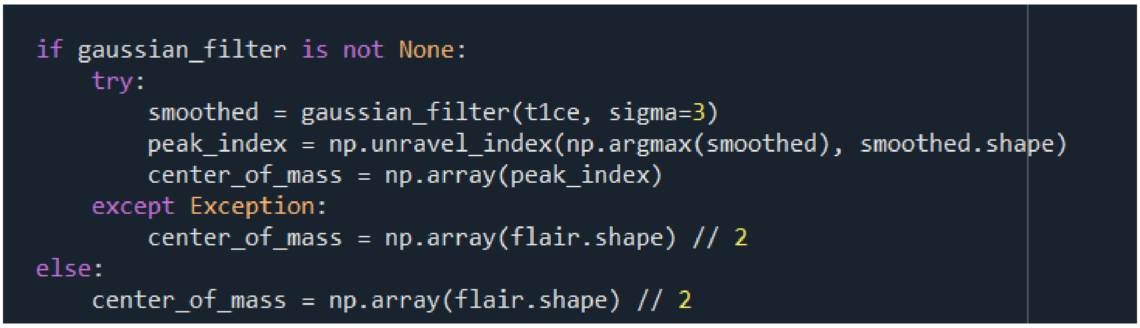


```
if mask is not None:
    mask_cropped = mask[adjusted_crop_indices]
    mask_cropped_categorical = to_categorical(mask_cropped, num_classes=4)
    cohort_tag = str(cohort_name).replace(' ', '-').replace('/', '-')
    site_tag   = str(site_no).replace(' ', '-').replace('/', '-')
    filename_base = f"{patient_id}_{cohort_tag}_{site_tag}"
    np.save(os.path.join(output_dir_images, f"{filename_base}.npy"), combined_cropped_images)
    np.save(os.path.join(output_dir_masks,  f"{filename_base}_mask.npy"), mask_cropped_categorical)
else:
    np.save(os.path.join(output_dir_images, f"{patient_id}.npy"), combined_cropped_images)
```

Listing 25: Simplified save block in Method 2: all images saved regardless of tumor size.

4.3.5. Estimating tumor location in the absence of segmentation masks

Method 1 relies entirely on the segmentation mask to calculate the center of mass for cropping. However, in the validation dataset, where no mask exists, Method 1 defaulted to the center of the volume, a weak heuristic that can lead to crops missing the tumor entirely. In Method 2, another strategy was implemented using the T1GD image, as shown in Listing 26.



```
if gaussian_filter is not None:
    try:
        smoothed = gaussian_filter(t1ce, sigma=3)
        peak_index = np.unravel_index(np.argmax(smoothed), smoothed.shape)
        center_of_mass = np.array(peak_index)
    except Exception:
        center_of_mass = np.array(flair.shape) // 2
else:
    center_of_mass = np.array(flair.shape) // 2
```

Listing 26: Mask-less center estimation via Gaussian-smoothed T1Gd peak, fallback to image center on failure.

This technique uses a Gaussian filter to blur the T1GD image, followed by locating the brightest voxel in the smoothed volume, under the assumption that tumors enhance with contrast. This estimate increases the likelihood of cropping around relevant regions, even in the absence of masks. Cropping is a common preprocessing step for volumetric brain MRI because it reduces the amount of data that must be stored and processed. The challenge is to ensure that the cropped region still contains the tumor; otherwise, the model may never learn the tumor features.

4.4. Method 3

The final preprocessing pipeline, illustrated in Figure 17, integrates a set of robust techniques designed to maximize data consistency and segmentation accuracy.

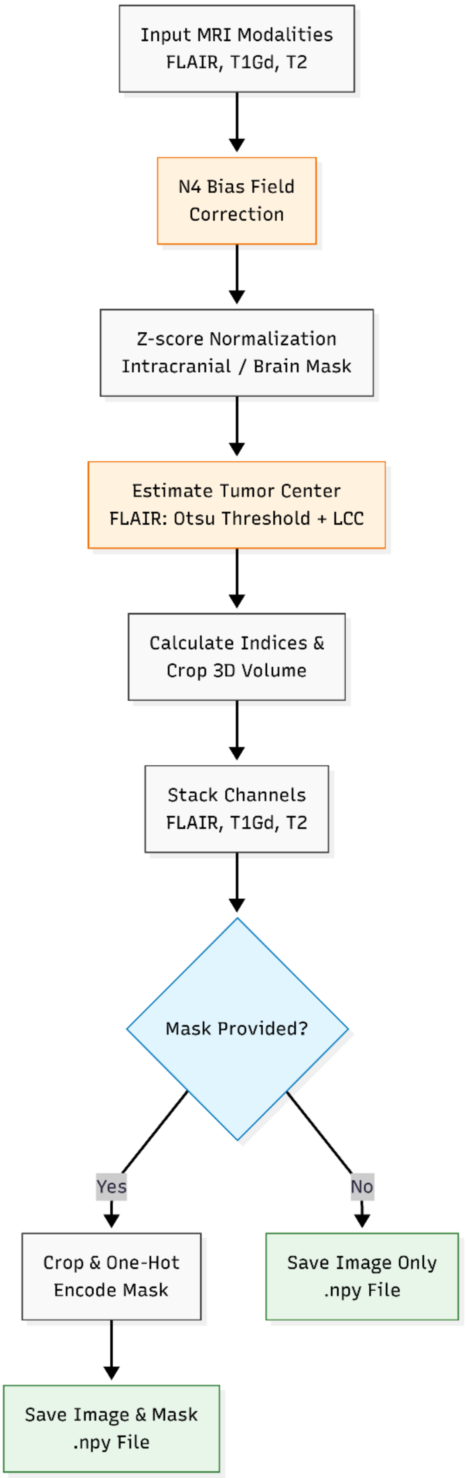
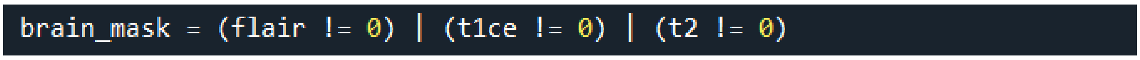


Figure 17 Method 3 preprocess diagram.

Compared with Method 2, which centred the crop at the brightest voxel of a Gaussian-smoothed post-contrast T1, it was observed that this method was vulnerable to non-tumor enhancement and local artefacts, occasionally mis-centring the ROI. To address this, Method 3 switches the primary localization cue to FLAIR, which reliably highlights peritumoral ED and NET as hyperintense regions, providing a cleaner lesion–background contrast. Then, it was applied Otsu's global threshold on the FLAIR volume to obtain a first, label-free lesion mask and refine it with light morphological cleaning. The largest connected component defines the ROI and its centroid sets the crop center, avoiding label leakage (Nyo et al., 2022).

In case the FLAIR doesn't present no plausible region, the use of the T1Gd is retained as a fallback. Finally, Z-score is computed only over intracranial (non-zero) voxels rather than the whole volume, to avoid background bias and improve cross-case consistency (Ranjbarzadeh et al., 2021).

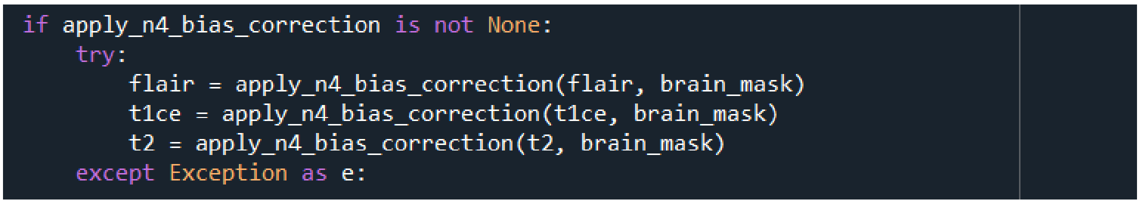
The main code steps implemented in Method 3 are summarised in Listings 27–30, which include the construction of the brain mask, optional N4 bias-field correction, masked Z-score normalisation, and the final tumor-center estimation logic.



```
brain_mask = (flair != 0) | (t1ce != 0) | (t2 != 0)
```

Listing 27: Construction of the brain mask from non-zero voxels across modalities.

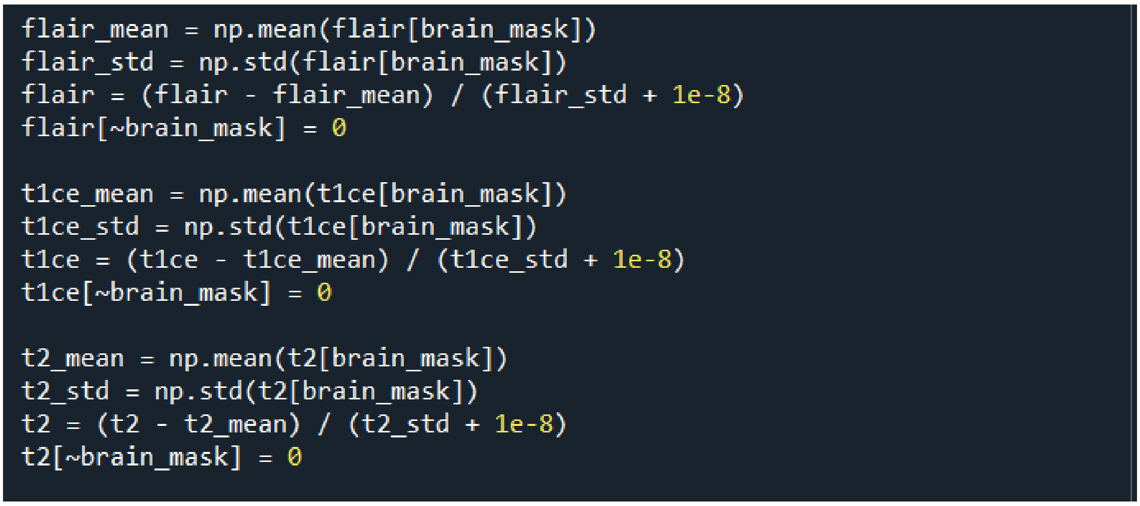
Before normalising each sequence, construct a brain mask from the union of non-zero voxels across the three modalities. Since the BraTS data are skull-stripped, background voxels are typically zeros and should not influence the computed mean and standard deviation. Masking out non-brain voxels is a common practice when performing z-score normalisation on MRI volumes.



```
if apply_n4_bias_correction is not None:
    try:
        flair = apply_n4_bias_correction(flair, brain_mask)
        t1ce = apply_n4_bias_correction(t1ce, brain_mask)
        t2 = apply_n4_bias_correction(t2, brain_mask)
    except Exception as e:
```

Listing 28: N4 bias-field correction using SimpleITK and the ANTsX implementation.

In this work, N4 bias field correction was performed using the SimpleITK implementation of the ANTsX N4 algorithm, applied to FLAIR, T1Gd and T2 within the intracranial mask (Lowekamp et al., 2013).

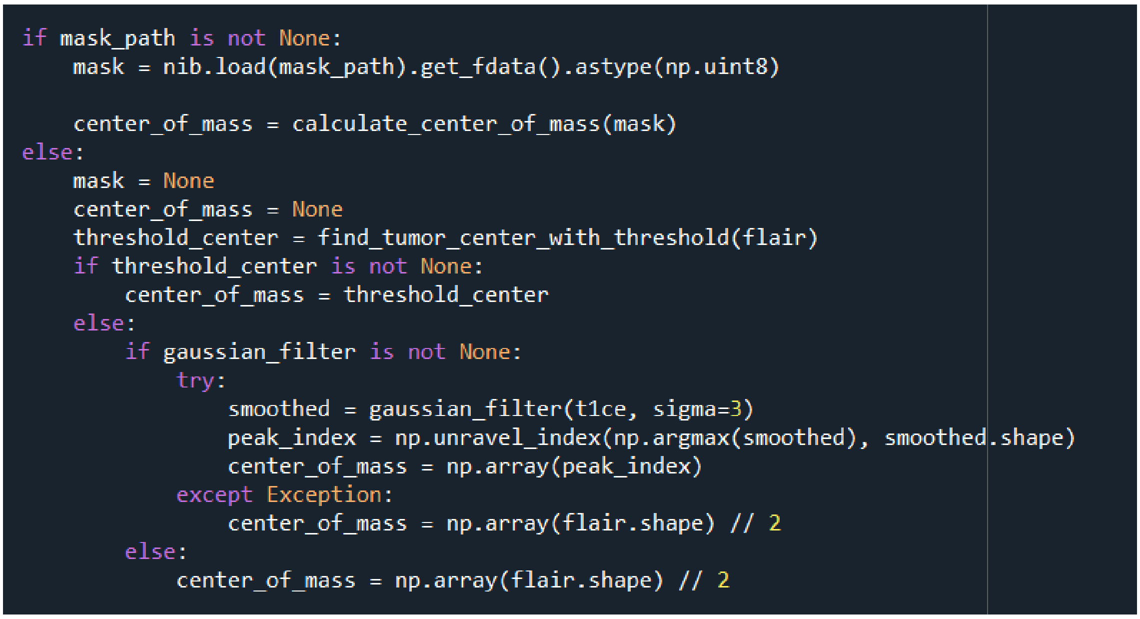


```
flair_mean = np.mean(flair[brain_mask])
flair_std = np.std(flair[brain_mask])
flair = (flair - flair_mean) / (flair_std + 1e-8)
flair[~brain_mask] = 0

t1ce_mean = np.mean(t1ce[brain_mask])
t1ce_std = np.std(t1ce[brain_mask])
t1ce = (t1ce - t1ce_mean) / (t1ce_std + 1e-8)
t1ce[~brain_mask] = 0

t2_mean = np.mean(t2[brain_mask])
t2_std = np.std(t2[brain_mask])
t2 = (t2 - t2_mean) / (t2_std + 1e-8)
t2[~brain_mask] = 0
```

Listing 29: Z-score normalization within the brain mask, background set to zero.

Compute mean and standard deviation only within the brain mask for z-score normalisation. Adding a small epsilon avoids division by zero for uniform volumes. After normalisation, set voxels outside the brain mask to zero so that background does not contain arbitrary values.

Use the provided segmentation mask to compute the center of mass directly. Without a segmentation mask, attempt to estimate the tumor center using a simple thresholding approach. Otsu’s method is used to detect hyperintense voxels in the FLAIR volume, and the largest connected component’s mean coordinate is returned. This provides a coarse localization of the tumor region. If thresholding fails or the required libraries are unavailable, fall back to the Gaussian-smoothed peak method implemented previously.



```
if mask_path is not None:
    mask = nib.load(mask_path).get_fdata().astype(np.uint8)

    center_of_mass = calculate_center_of_mass(mask)
else:
    mask = None
    center_of_mass = None
    threshold_center = find_tumor_center_with_threshold(flair)
    if threshold_center is not None:
        center_of_mass = threshold_center
    else:
        if gaussian_filter is not None:
            try:
                smoothed = gaussian_filter(t1ce, sigma=3)
                peak_index = np.unravel_index(np.argmax(smoothed), smoothed.shape)
                center_of_mass = np.array(peak_index)
            except Exception:
                center_of_mass = np.array(flair.shape) // 2
        else:
            center_of_mass = np.array(flair.shape) // 2
```

Listing 30: Computation of the tumor center: mask-based centroid, threshold-based estimation, and Gaussian-smoothed fallback.

Although the BraTS images are skull-stripped, the cropped volumes still include a large background region with intensity zero. When Otsu’s method is applied to the whole volume, the zero background dominates the histogram and the threshold is placed between background and brain tissue. By restricting Otsu to brain voxels only, the histogram becomes bimodal with one peak for normal tissue and another for hyperintense voxels (Figure 18-19). The resulting threshold is closer to the true separation between normal and tumor tissue.

.

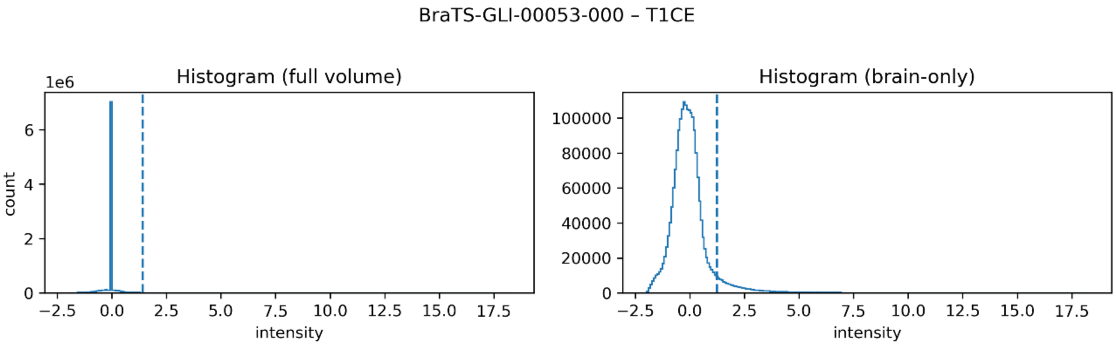


Figure 18: Intensity histograms from a T1Gd image: full volume (left) vs brain-only (right). The zero background dominates the full-volume histogram and biases thresholding, masking removes this effect.

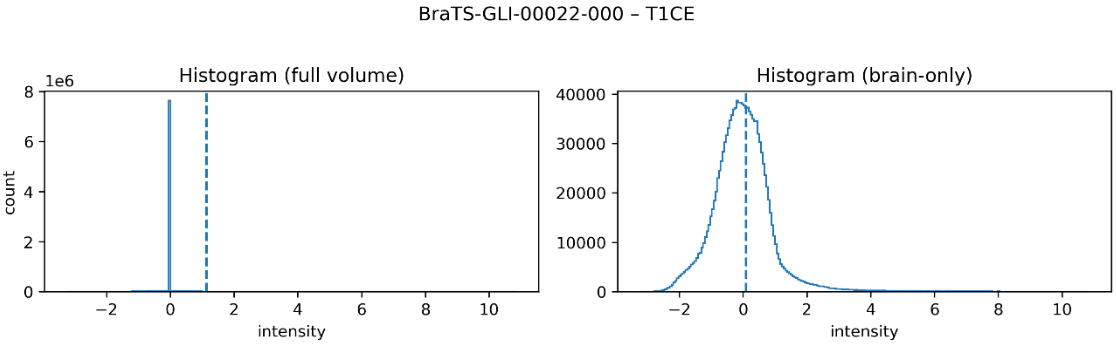


Figure 19: Same comparison for a different subject, illustrating the consistency of the brain-only distribution.

4.5. Examples of Preprocessed Images

A total of 1,470 available images underwent the complete preprocessing pipeline described previously, ensuring consistent normalization and formatting for the subsequent training phase. To illustrate the output quality and the effect of these transformations on the MRI data, representative examples are displayed in Figures 20, 21, and 22.

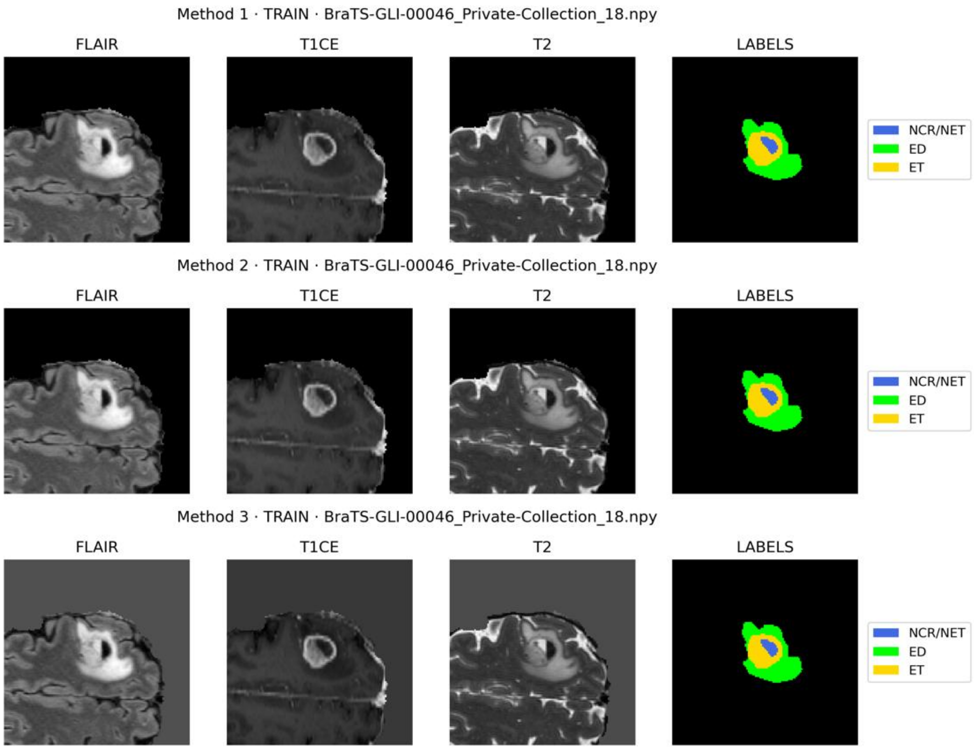


Figure 20: Images from the training set after preprocessing with Methods 1, 2, and 3. Each row shows FLAIR, T1GD, and T2 with the corresponding segmentation labels.

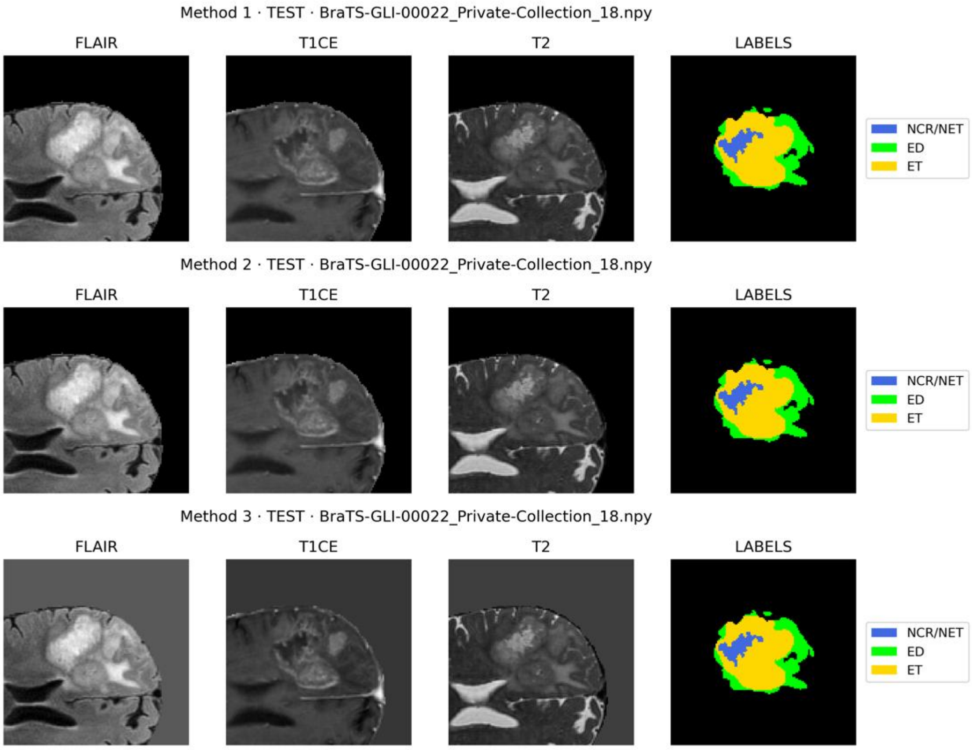


Figure 21: Images from the test set after preprocessing with Methods 1, 2, and 3. Each row shows FLAIR, T1GD, and T2 with the corresponding segmentation labels.

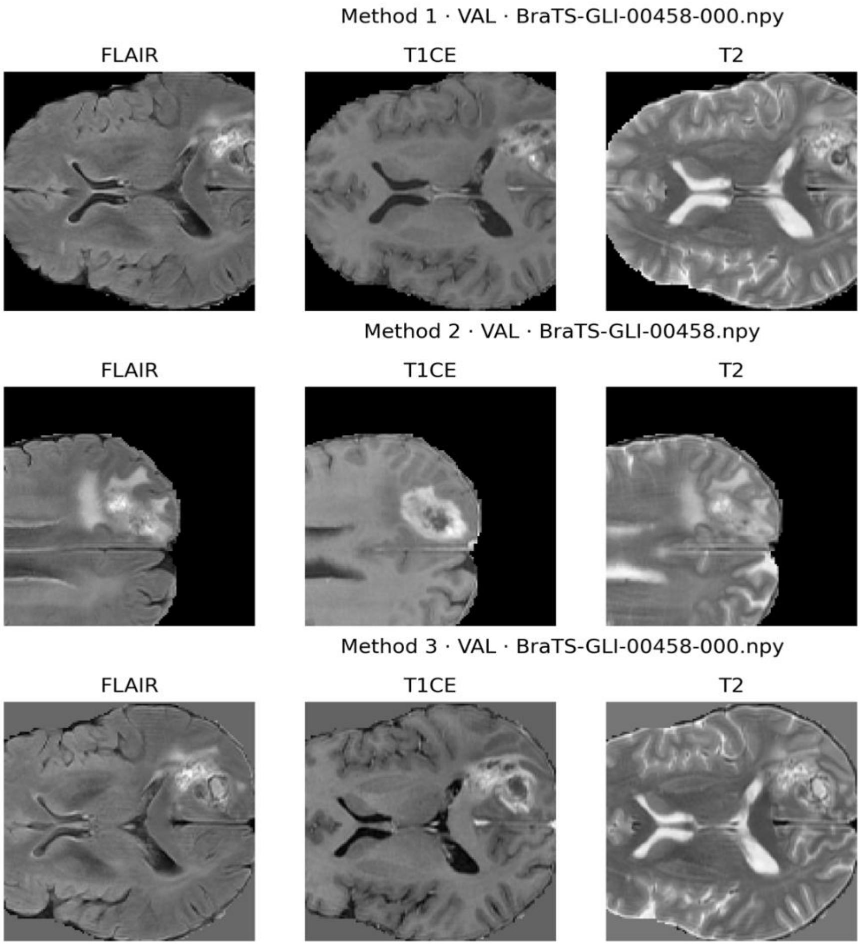


Figure 22: Images from the validation set after preprocessing with Methods 1, 2, and 3. Each row shows FLAIR, T1GD, and T2 with the corresponding segmentation labels.

Late in the project it became clear that the preprocessing did not allow a fair comparison between methods. Test crops in earlier pipelines were centerd using the ground-truth mask (mask-guided ROI). That creates an oracle ROI: the crop already points to the tumor, which is not realistic and inflates test performance. At the same time, different methods prepared inputs differently (mask-guided in some runs, image-only in others), so test metrics were not comparable. Additional traceability issues also played a role earlier (e.g., truncated or missing patient IDs and excluded files without masks), which complicated auditing and made inconsistencies harder to spot. Together, these factors meant that a model could be trained, but any numbers reported across methods would not be a fair.

To fix this, the test (and validation) preprocessing was made image-only across all methods, matching the real inference scenario:

- Train: the mask may still guide the crop (only to place the ROI efficiently).
- Test & Validation: the crop center is chosen from the images only using a simple, robust rule—Otsu's threshold computed on brain voxels, followed

by a light morphological closing, and then the center of the largest connected component. The ground-truth mask, if available, is cropped afterwards with the same indices and used only for evaluation. This avoids label leakage and makes test metrics realistic and comparable. In Method 3 this logic is already implemented via find_tumor_center_with_threshold with Otsu-on-brain and binary_closing(…, iterations=1) before selecting the largest component (Figure 23-25).

- ID handling: patient IDs are taken from the subject folder name (e.g., BraTS-GLI-00166-000) both in file names and in the reproducibility list. This keeps all outputs traceable and aligned across methods.

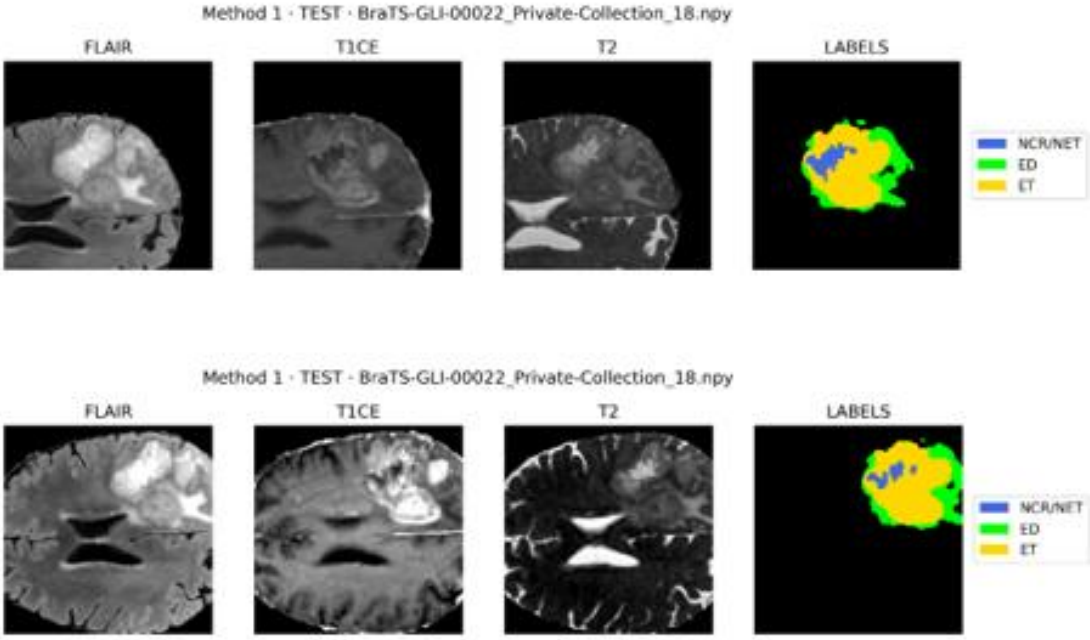


Figure 23: Method 1 Before: crops used the mask. After: crops use images only, making tests more realistic for the model.

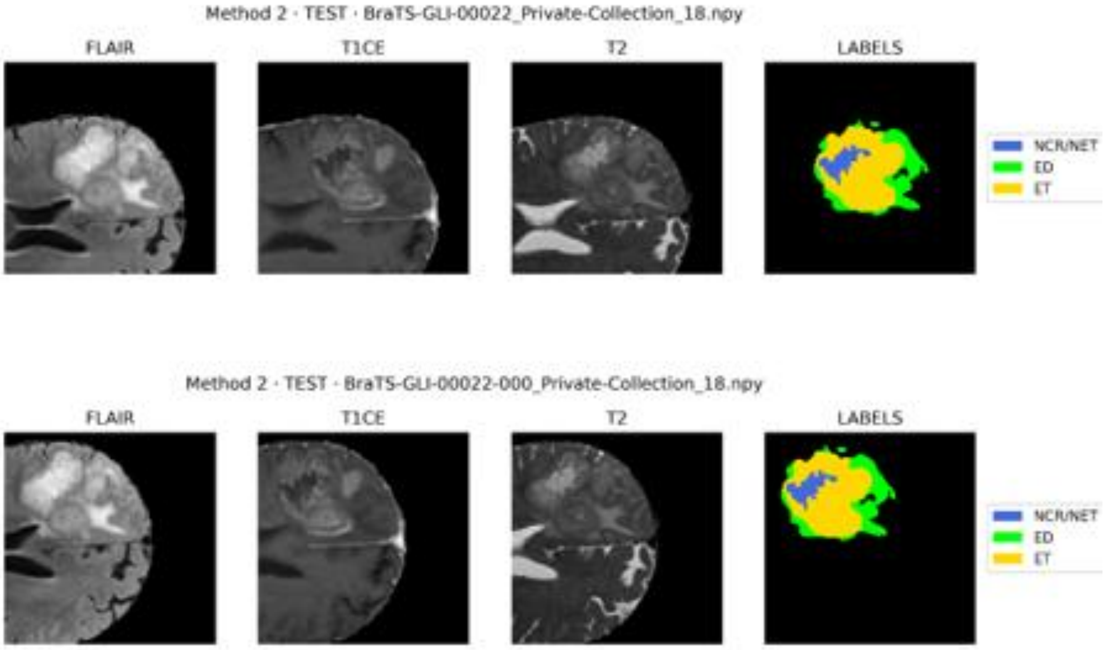


Figure 24: Method 2 Before: crops used the mask. After: crops use images only, making tests more realistic for the model.

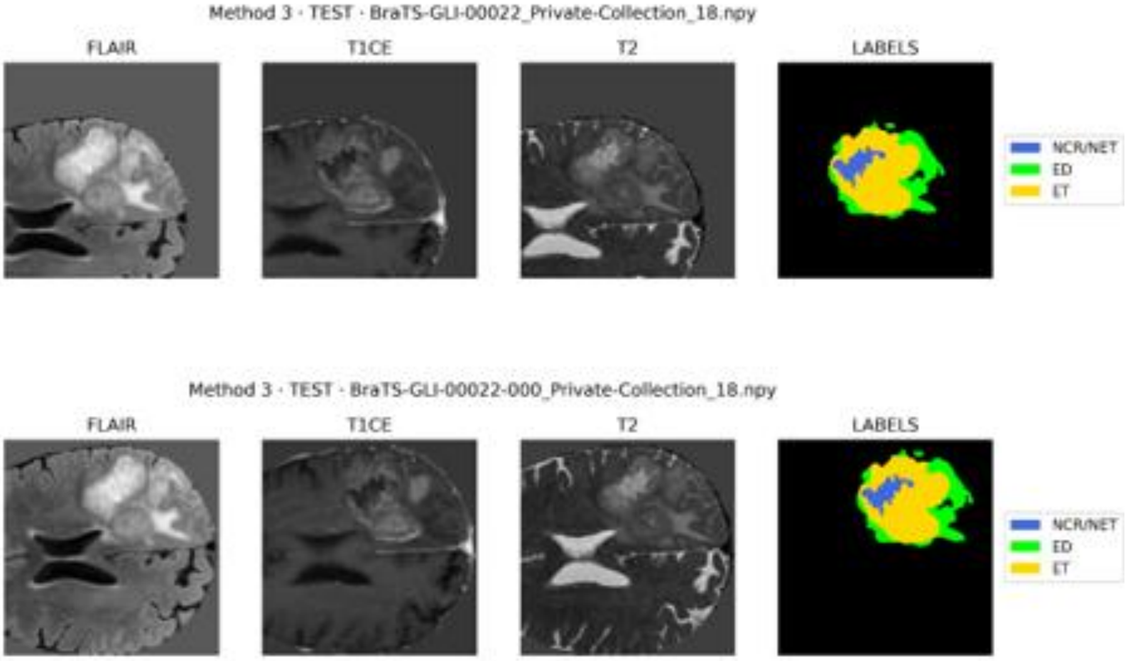


Figure 25: Method 3 Before: crops used the mask. After: crops use images only, making tests more realistic for the model.

5. 2D U-Net Model implementation with MONAI

5.1. Transforms and tensor preparation

For this project, a 2D U-Net model was built using MONAI, based on the official BraTS tutorial, adopting its general workflow with minor adaptations to suit the available data and limited hardware, specifically regarding 2D slice extraction. The resulting end-to-end pipeline, ranging from data ingestion and preprocessing to model training and evaluation, is schematically illustrated in Figure 26.

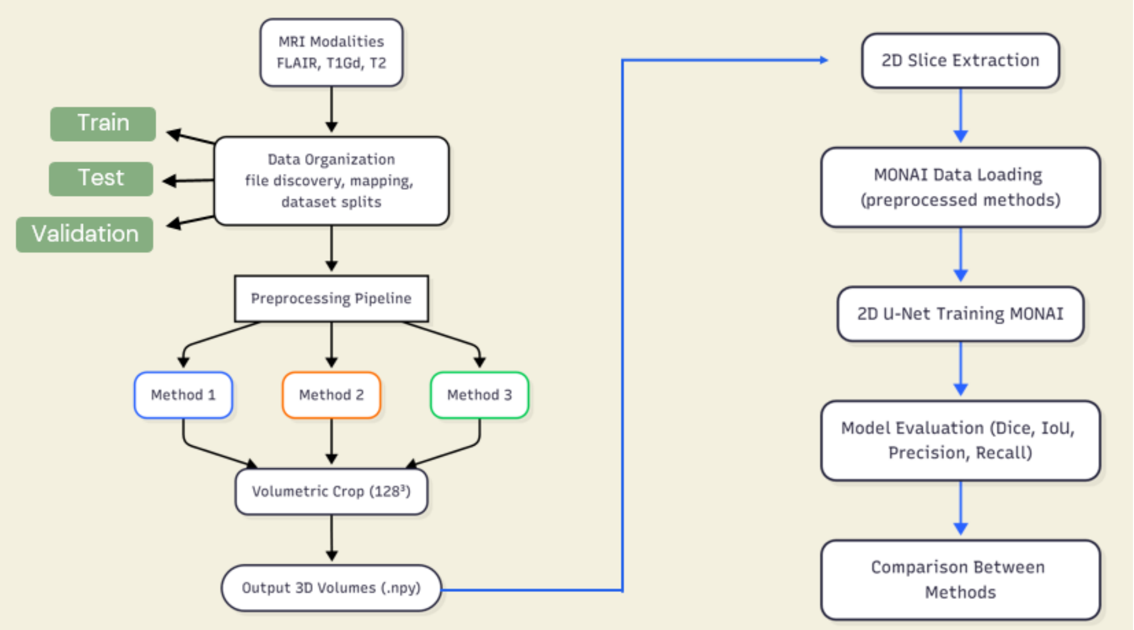
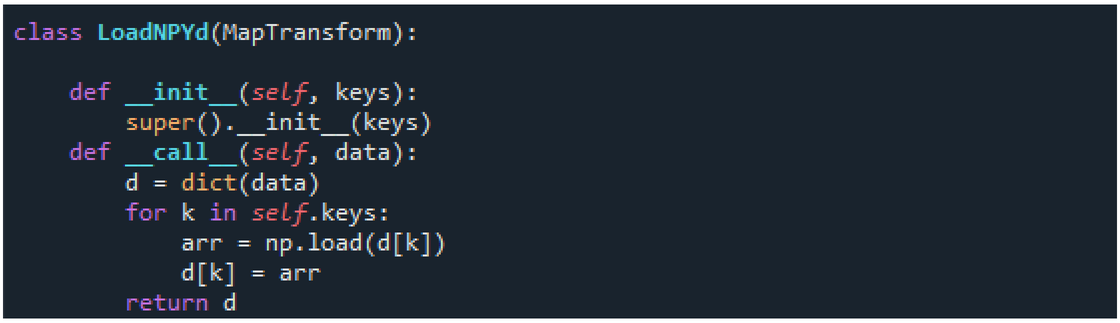


Figure 26 Schematic overview of the end-to-end segmentation framework.

5.1.1. On data preparation:

- The input data were saved as NumPy files (.npy) instead of the usual NIfTI format (see Listing 31).
- For each patient, the three MRI modalities were stacked together to form a 3-channel 2D image, and the segmentation mask was stored separately.
- Since the GPU available could not handle a 3D U-Net, only 2D slices were used. From each MRI volume, the slice with the largest tumor area was chosen. If no tumor was present, the central slice was selected.



```python
class LoadNPYd(MapTransform):

    def __init__(self, keys):
        super().__init__(keys)
    def __call__(self, data):
        d = dict(data)
        for k in self.keys:
            arr = np.load(d[k])
            d[k] = arr
        return d
```

Listing 31: Custom MONAI map transform to read .npy arrays instead of NIfTI.

5.1.2. In the preprocessing step:

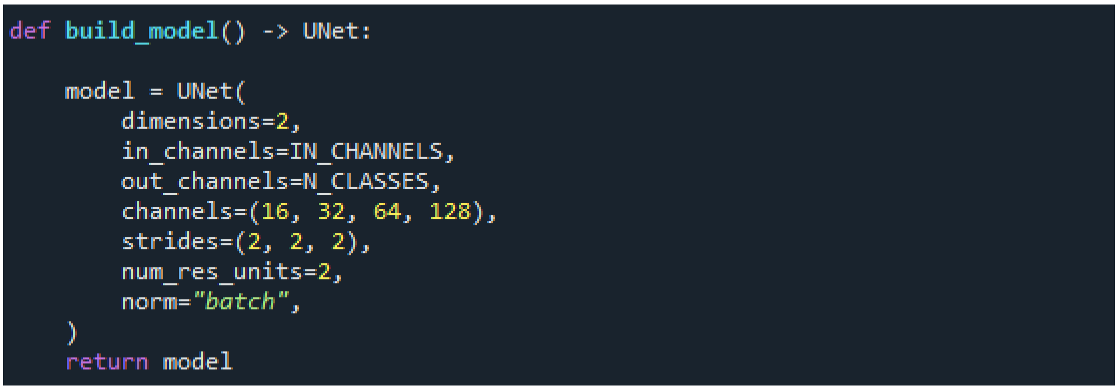
- Labels from the BraTS dataset were remapped from {0, 1, 2, 4} to {0, 1, 2, 3} to keep the order sequential.
- The MONAI transform pipeline was used to load data, arrange channels, and convert them into tensors for training
- Intensity normalization, Z-score, was already done before, so it was not repeated inside MONAI.

```
def build_transforms() -> Tuple[Compose, Compose]:

    def _remap_brats(lbl):
        lbl = lbl.copy()
        lbl[lbl == 4] = 3
        return lbl

    train_t = Compose([
        LoadNPYd(keys=["image", "label"]),
        Lambdad(keys="label", func=_remap_brats),
        EnsureChannelFirstd(keys="image", channel_dim=-1),
        AddChanneld(keys="label"),
        #NormalizeIntensityd(keys="image", channel_wise=True),
        ToTensord(keys=["image", "label"]),
        EnsureTyped(keys=["image", "label"]),])
    val_t = Compose([
        LoadNPYd(keys=["image", "label"]),
        Lambdad(keys="label", func=_remap_brats),
        EnsureChannelFirstd(keys="image", channel_dim=-1),
        AddChanneld(keys="label"),
        #NormalizeIntensityd(keys="image", channel_wise=True),
        ToTensord(keys=["image", "label"]),
        EnsureTyped(keys=["image", "label"]),])
    return train_t, val_t
```

Listing 32: Transformations used before training: loads .npy files, remaps label 4 to 3, arranges channels.

5.1.3. Model architecture and training:

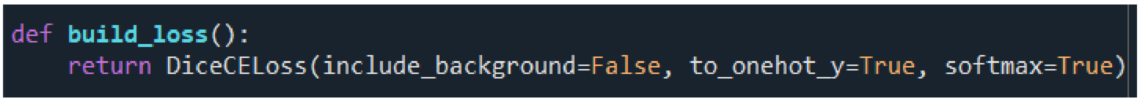
- The network used was a standard 2D U-Net with 128×128 input size, three channels, and four output classes.



```
def build_model() -> UNet:

    model = UNet(
        dimensions=2,
        in_channels=IN_CHANNELS,
        out_channels=N_CLASSES,
        channels=(16, 32, 64, 128),
        strides=(2, 2, 2),
        num_res_units=2,
        norm="batch",
    )
    return model
```

Listing 33: 2D U-Net with 3 input channels and 4 output classes, designed for 128 × 128 slices.

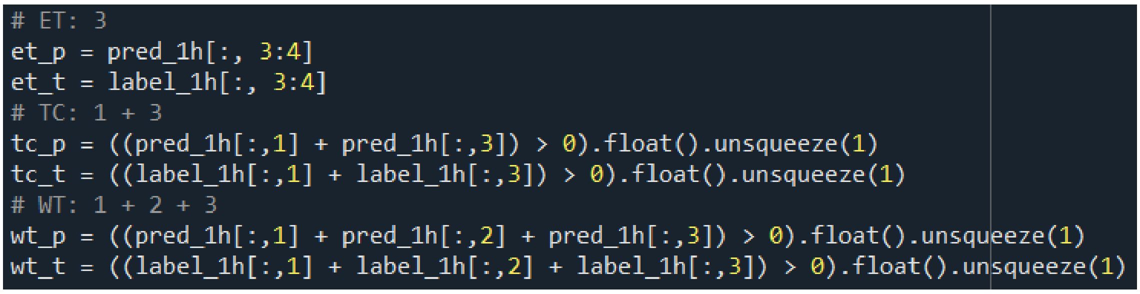
- It has three downsampling and three upsampling levels, connected by skip connections to keep both local and global details.
- The loss function combined Dice + Cross-Entropy, ignoring the background to focus on tumor areas.



```
def build_loss():
    return DiceCELoss(include_background=False, to_onehot_y=True, softmax=True)
```

Listing 34: Dice + Cross-Entropy loss ignoring background to focus on tumor regions.

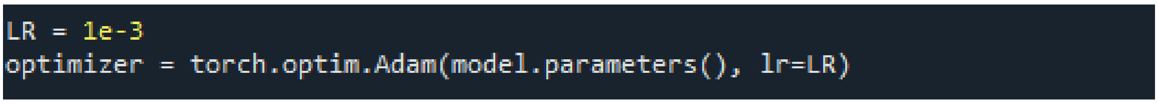
- Combination of the four output classes, into the three BraTS subregions used for evaluation.



```
# ET: 3
et_p = pred_1h[:, 3:4]
et_t = label_1h[:, 3:4]
# TC: 1 + 3
tc_p = ((pred_1h[:,1] + pred_1h[:,3]) > 0).float().unsqueeze(1)
tc_t = ((label_1h[:,1] + label_1h[:,3]) > 0).float().unsqueeze(1)
# WT: 1 + 2 + 3
wt_p = ((pred_1h[:,1] + pred_1h[:,2] + pred_1h[:,3]) > 0).float().unsqueeze(1)
wt_t = ((label_1h[:,1] + label_1h[:,2] + label_1h[:,3]) > 0).float().unsqueeze(1)
```

Listing 35: Combination of predicted segmentation classes into BraTS subregions for further metric computation.

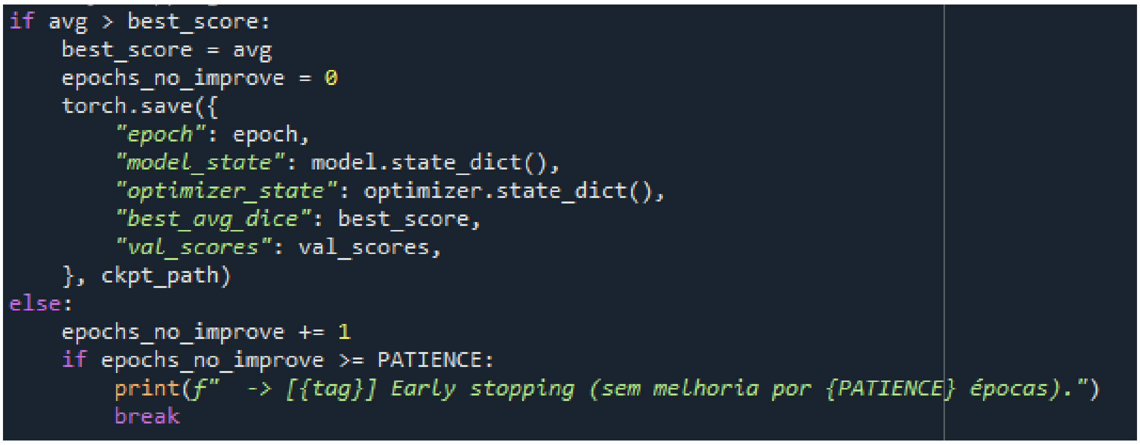
- Training used the Adam optimizer with learning rate 0.001, up to 25 epochs, and batch size 2.



```
LR = 1e-3
optimizer = torch.optim.Adam(model.parameters(), lr=LR)
```

Listing 36: Training with Adam optimizer.

- The early stopping criterion monitored the average validation Dice across ET, TC and WT, training stops after 5 epochs without improvement, saving the best checkpoint automatically.



```
if avg > best_score:
    best_score = avg
    epochs_no_improve = 0
    torch.save({
        "epoch": epoch,
        "model_state": model.state_dict(),
        "optimizer_state": optimizer.state_dict(),
        "best_avg_dice": best_score,
        "val_scores": val_scores,
    }, ckpt_path)
else:
    epochs_no_improve += 1
    if epochs_no_improve >= PATIENCE:
        print(f"  -> [{tag}] Early stopping (sem melhoria por {PATIENCE} épocas).")
        break
```

Listing 37: Early stopping and model saving.

During training and validation, MONAI only computed the Dice score as the main segmentation metric. After training, the best model weights from each preprocessing method were saved as a checkpoint file. These checkpoints were then used for the test stage, where predicted segmentation masks were compared with the corresponding ground-truth masks. For each BraTS region, pixel-level True Positives (TP), False Positives (FP), False Negatives (FN), and True Negatives (TN) were counted. These counts were later used to calculate Precision, Recall, Specificity, IoU, and Dice per patient during evaluation.

At last, the pipeline was repeated separately for each preprocessing method - Method 1, 2, 3. For each method, a model was trained from scratch and validated.

5.2. Evaluation Metrics

The following metrics were used to quantitatively evaluate the segmentation performance of each trained model.

Dice Coefficient (DSC): An overlap metric between the predicted and ground truth segmentation, calculated as Equation 4.1.

$$DSC = \frac{2TP}{2TP+FP+FN} \tag{4.1}$$

It is equivalent to the F1-score (harmonic mean between precision and recall) and is widely used in medical tumor segmentation, as it provides a robust indicator of the correspondence between the model's segmented volume and the annotated reference volume (Müller et al., 2022).

Intersection over Union (IoU): Also known as the Jaccard index, it measures the fraction of intersection over the union of the predicted and ground truth regions (Equation 4.2).

$$IoU = \frac{TP}{2TP+FP+FN} \tag{4.2}$$

It is an overlap metric like Dice, but it tends to penalize discrepancies (false positive or false negative regions) more severely. IoU values range from 0 to 1, indicating from no overlap to perfect agreement between segmentations (Müller et al., 2022).

Sensitivity/recall: Corresponds to the model's ability to recover true positives, defined as the proportion of correctly identified true positives among all existing positives (Equation 4.3).

$$Sensitivity = \frac{TP}{TP+FN} \tag{4.3}$$

In the context of glioma segmentation, sensitivity indicates the fraction of the tumor volume truly detected by the model (also called the true positive rate). A high recall means that few tumor areas were left unsegmented (low FN), but it does not account for false alarms (Müller et al., 2022).

Precision: Reflects the accuracy of the model's positive predictions, i.e., the proportion of pixels classified as tumor that indeed correspond to tumor in the ground truth (Equation 4.4).

$$Precision = \frac{TP}{TP+FP} \tag{4.4}$$

This metric quantifies how free of false positives the proposed segmentation is—high precision implies that the model rarely segments healthy regions as tumor (Cray, 2025).

Specificity: The proportion of true negatives correctly identified, given by Equation 4.5.

$$Specificity = \frac{TN}{TN+FP} \tag{4.5}$$

It indicates how well the model recognizes the background (non-tumoral region), avoiding confusing normal structures with lesions. Values close to 1 are common due to the large background area in medical images, therefore, specificity is useful to confirm that the algorithm rarely segments normal tissue as tumor. Nevertheless, it is not a good indicator of tumor segmentation quality since if a model labeled everything as background it would achieve high specificity.

These metrics provides a complementary insight into segmentation model performance. Dice and IoU evaluate global overlaps, while sensitivity and precision reveal the incidence of false-negative and false-positive errors, respectively (Cray, 2025). The combination of these metrics in the analysis enables a comprehensive evaluation of segmentation quality, aligned with recommended practices in challenges such as BraTS for quantifying glioma segmentation algorithm performance (Müller et al., 2022)

6. Results

6.1. Methods preprocessing images

6.1.1. Method 1

Min–max and mask-centerd crop: the lesion is fully included and well positioned. Min–max strongly boosts intensities (especially on T1GD), which gives high contrast but can compress the grey-scale range differently in each modality (Figure 27).

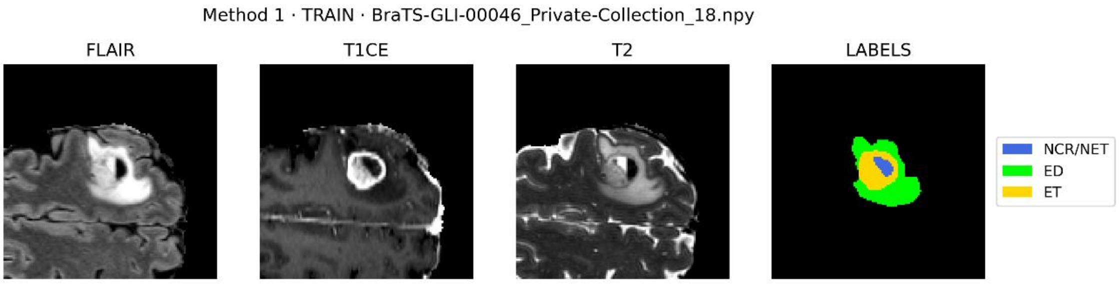


Figure 27: Method 1 preprocessed images from the train set.

Min–max and geometric center: the ROI is tight and, in this example, it slightly clips the edge of the lesion, so some context is lost. The modalities stay spatially aligned, but per-image min–max makes brightness less consistent across FLAIR/T1GD/T2 (Figure 28).

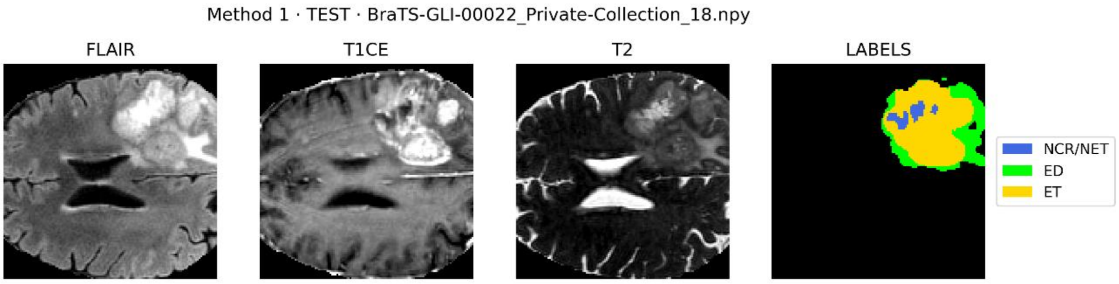


Figure 28: Method 1 preprocessed images from the test set.

Min–max and geometric center; the tumor can be off-center relative to the crop. The field of view is small, so there is a risk of losing information at the borders. The background is black and all three modalities are visible, but their intensity balance differs (Figure 29).

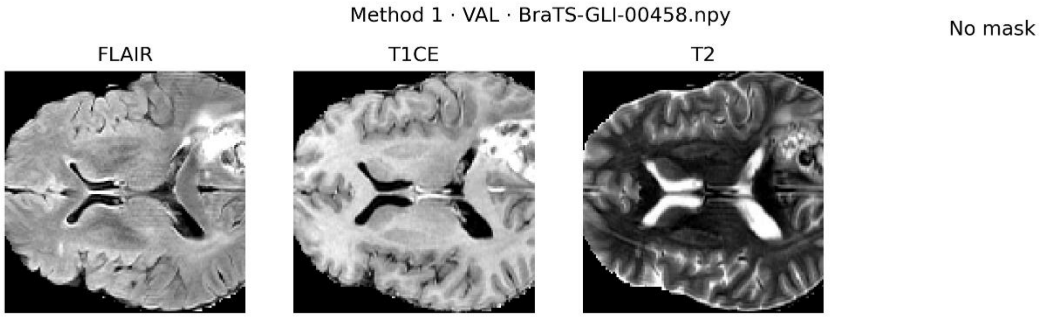
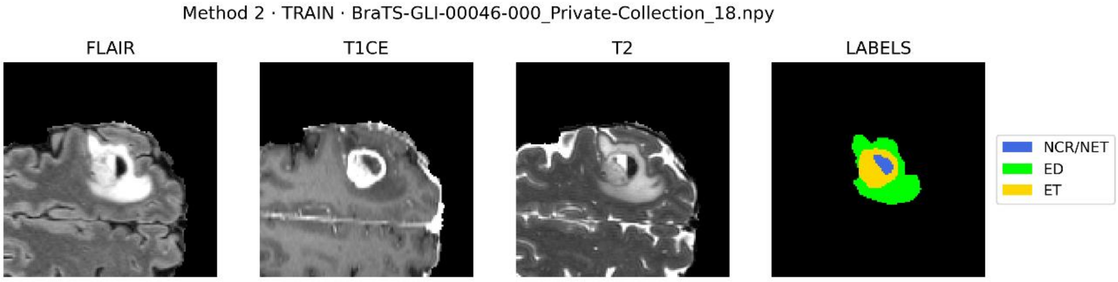


Figure 29: Method 1 preprocessed images from the validation set.

6.1.2. Method 2

Z-score (within the brain mask) and mask-centerd crop: the lesion is fully included and the contrast across FLAIR/T1GD/T2 is balanced, there is less saturation than with min–max (Figure 30).



Figur9)e 30: Method 2 preprocessed images from the train set.

Z-score + image-based centring (brightest voxel in T1GD): the tumor stays fully inside the ROI and is reasonably centerd; there is no clipping at the edges. The modalities keep a more stable intensity scale than in Method 1.

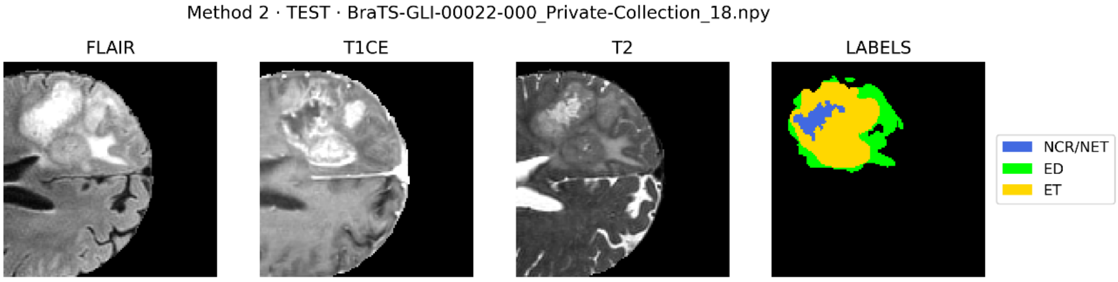


Figure 31: Method 2 preprocessed images from the test set.

Z-score and same rule as TEST (brightest voxel in T1GD): the lesion remains entirely within the crop, and the field of view keeps some context. The background is black and the cross-modality balance is consistent as seem in Figure 32.

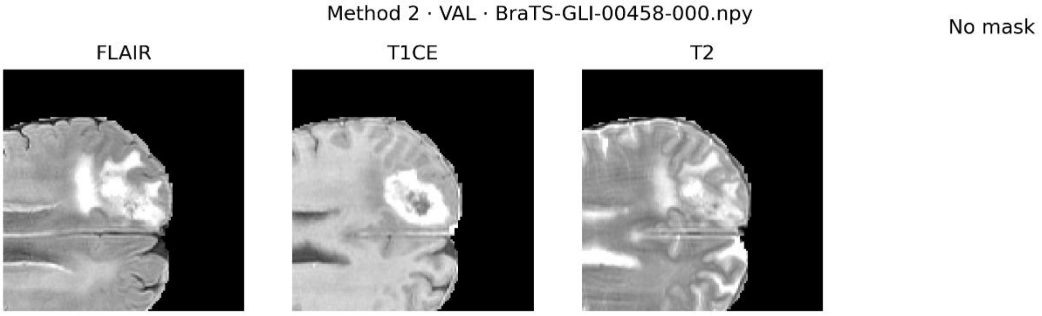


Figure 32: Method 2 preprocessed images from the validation set.

6.1.3. Method 3

With crop centerd by the tumor mask (center of mass), in figure 33, the lesion is fully included and the three modalities (FLAIR, T1GD, T2) keep stable contrast and anatomical alignment. No truncation at the borders.

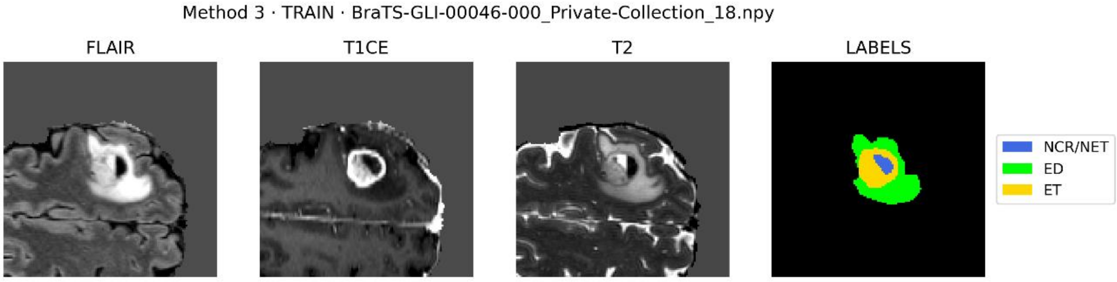


Figure 33: Method 3 preprocessed images from the train set.

N4 + Z-score within the brain mask (Figure 34), image-only centring (Otsu on brain-masked FLAIR; T1GD fallback not needed here). The lesion stays completely within the ROI and is reasonably centerd; no information is lost at the borders. The three modalities keep a stable, unsaturated intensity scale.

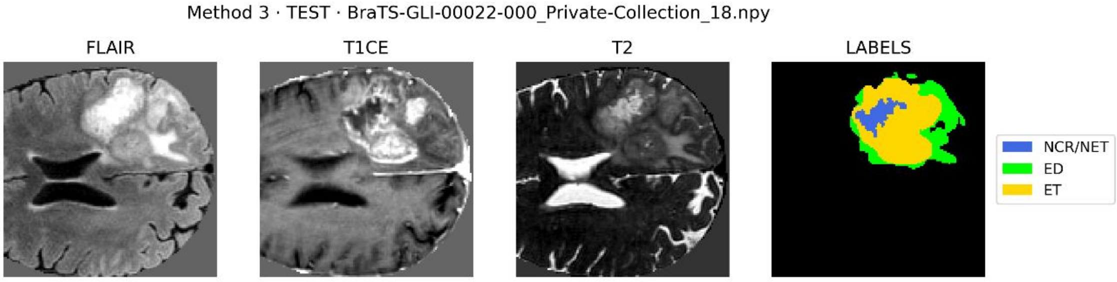


Figure 34: Method 3 preprocessed images from the test set.

N4 + Z-score within the brain mask, same rule as TEST. The crop keeps the lesion fully included and preserves enough surrounding context. The tumor is not cut as seen in Figure 35.

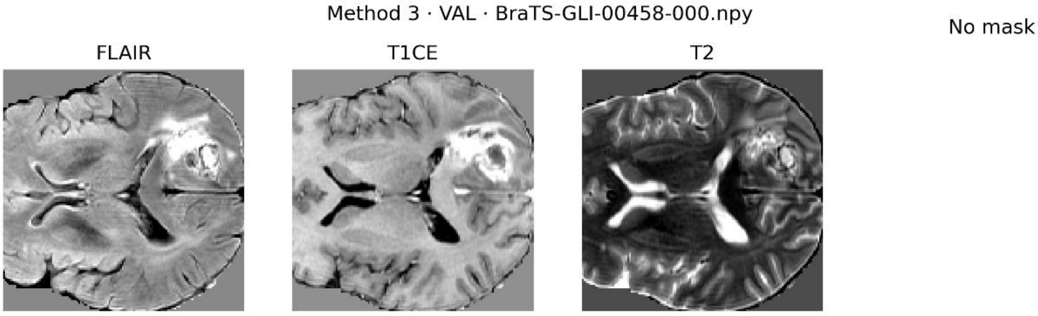


Figure 35: Method 3 preprocessed images from the validation set.

Method 1 uses per-image min–max and a geometric-center crop when no mask is available (VAL/TEST). This produces strong contrast but less consistent brightness between modalities and a tight field of view, which can truncate the lesion or reduce context. In TRAIN the mask prevents cropping errors; in TEST/VAL the geometric-center rule is simple but less reliable for keeping the whole lesion inside the ROI. In short, per-image min–max compresses the grey-scale range differently per modality, making FLAIR/T1GD/T2 harder to compare directly. Method 2 replaces min–max with Z-score inside the brain mask, which stabilises the intensity scale and reduces saturation. In TEST/VAL, image-based centring on T1GD (brightest voxel after light smoothing) includes the lesion better than the geometric-center rule in Method 1, lowering the risk of cropping it. However, very bright non-tumor structures can occasionally shift the center slightly—an issue Method 3 addresses with a FLAIR-based rule. Finaly, method 3 produces cleaner and more consistent inputs: image-only centring in TEST/VAL keeps the lesion fully inside the field of view without labels, and Z-score within the brain mask stabilises the scale across FLAIR/T1GD/T2, avoiding the over/under-saturation seen with per-image min–max.

6.2. Internal validation results

The model was trained for each method with the best epoch selected, using the internal validation split. Table 2 and Figure 36 show a clear ranking across methods: Method 3 achieves the highest Dice in all BraTS regions—ET = 0.729, TC = 0.718, WT = 0.888, with a macro-average of 0.778. Method 2 follows with ET = 0.670, TC = 0.691, WT = 0.879, macro=0.747, and Method 1 is lower with ET = 0.663, TC = 0.619, WT = 0.867, macro=0.716. Moving from Method 2 to Method 3 yields consistent gains: ET +0.059 (+8%), TC +0.027 (+4%), WT +0.009 (+1%). It's important to note that Method 1 was validated on 10 cases, while Methods 2–3 used 12, the smaller n for Method 1 reflects images that were excluded by its preprocessing, this makes direct comparison not reliable.

Table 2: Validation Dice by BraTS region (n = 12).

	ET	TC	WT	n_val	MacroAvg
Method 1	0,663	0,619	0,867	10	0,716
Method 2	0,670	0,691	0,879	12	0,747
Method 3	0,729	0,718	0,888	12	0,778

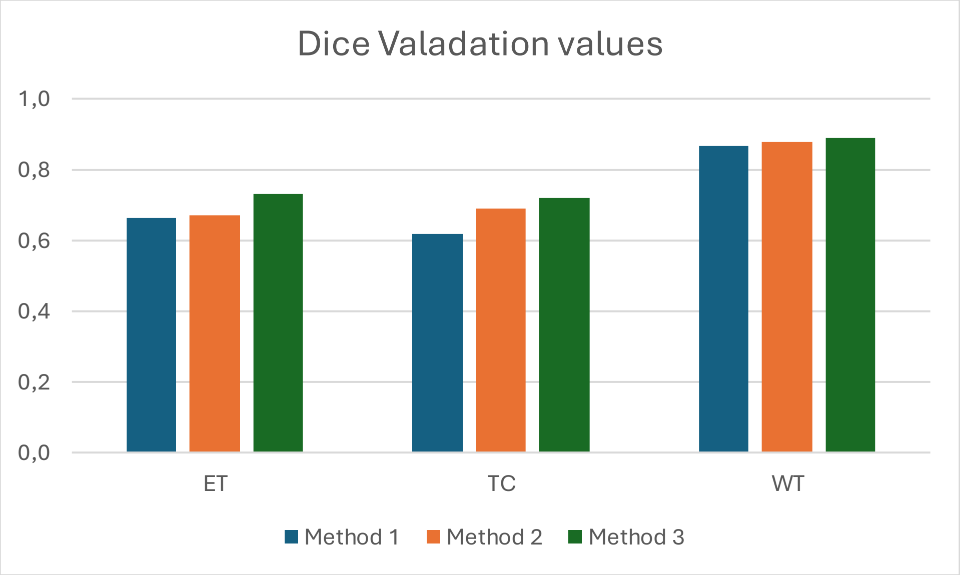


Figure 36: Dice validation values for ET, TC, and WT.

6.3. Internal Test results

Table 2 and Figure 33 show the average Dice values for the three BraTS regions on the test set, comparing the three preprocessing methods. Method 1 had the lowest Dice values; Method 2 improved over Method 1 and Method 3 achieved the best overall results as shown.

The improvements from Method 3 compared to Method 2 are +0.073 for ET, +0.153 for TC, and +0.072 for WT. In relative terms, these gains are 13% for ET, 29% for TC, and 9% for WT. The largest improvement is in TC region, which is usually the most difficult to segment.

Table 3: Test Dice by BraTS region.

	ET	TC	WT	n_test	MacroAvg
Method 1	0,440	0,418	0,777	18	0,545
Method 2	0,553	0,529	0,771	20	0,618
Method 3	0,626	0,682	0,844	20	0,717

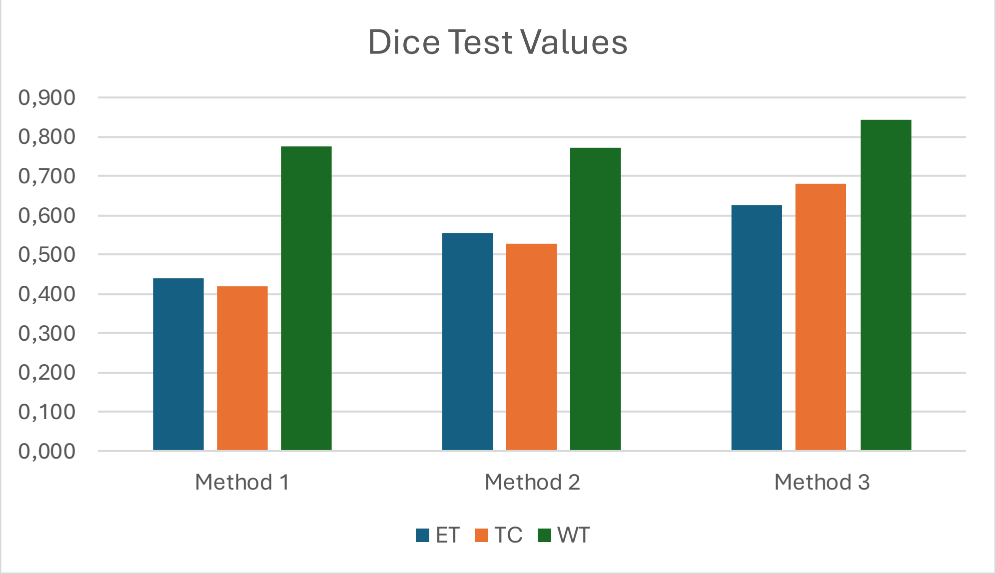


Figure 37: Test set Dice scores for the three preprocessing methods.

These outcomes reflect the impact of the different preprocessing steps. Method 1 applied Min–Max normalization separately to each image and cropped around the geometric center; this can yield inconsistent intensity scaling across modalities and occasionally clip parts of the tumor, which reduces ET and TC performance. Method 2 used Z-score normalization within the brain and centered the crop using the T1GD channel, stabilising intensities and improving TC while giving modest gains for ET and WT. Method 3 went further by normalising intensities within the brain and using a FLAIR-based cropping strategy. This approach ensures the lesion stays fully inside the crop and maintains a consistent intensity scale across FLAIR, T1GD and T2, yielding the highest ET and TC Dice values and the best overall macro averages.

6.4. Comparison of Methods

There’s a clear ordering in Figure 38, with M1 < M2 < M3. Method 1 shows the lowest central tendency and the widest spread while Method 2 shifts the distribution upward but retains a long lower tail and Method 3 with a higher median and lower dispersion, so results are more consistent across patients. On the common 18 case subset used for the plots, the medians/IQRs are 0.393 / 0.472 (M1), 0.715 / 0.485 (M2), and 0.786 / 0.391 (M3).

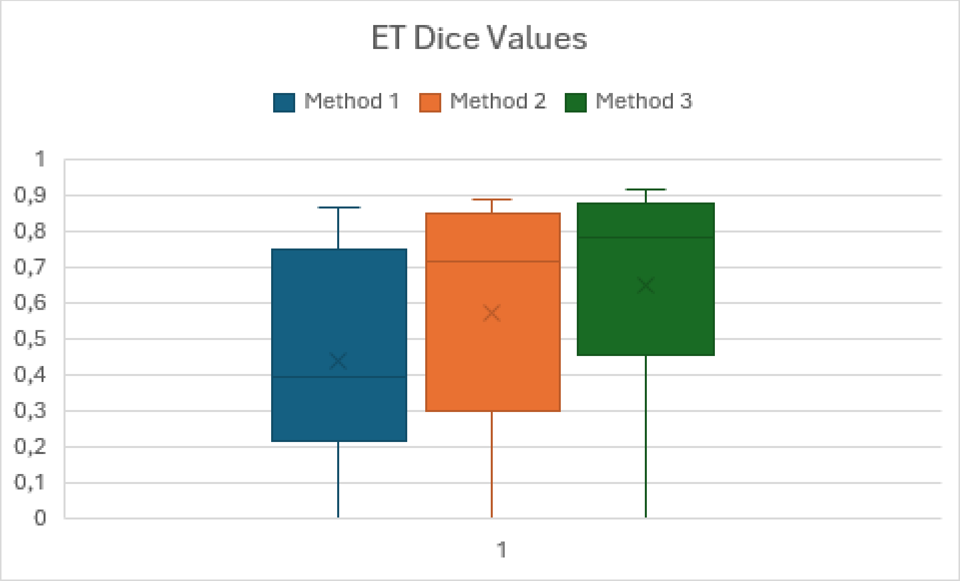


Figure 38: Boxplot of ET Test set Dice values.

For TC the same pattern is seen, with Method 1 being the weakest and demonstrating wide variability, Method 2 raises the distribution but with substantial interquartile width (Figure 39). Method 3 gives the highest median and a much tighter box, showing better robustness in this harder region; however, the mean is lower than the median, showing that a subset of low scores still pulls the average down. For the 18 cases common to all methods, medians/IQRs are 0.311 / 0.403 (M1), 0.624 / 0.459 (M2), and 0.793 / 0.207 (M3).

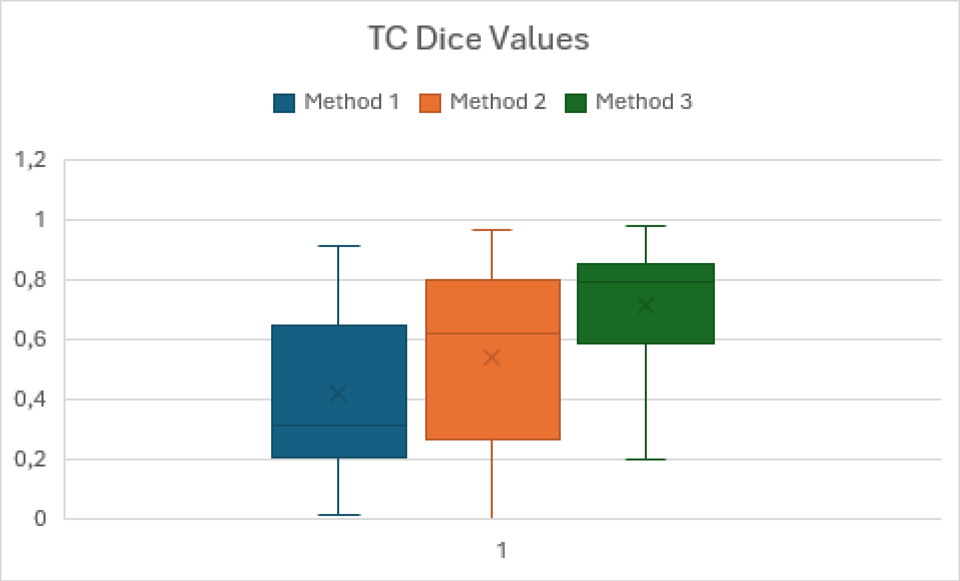


Figure 39: Boxplot of TC Test set Dice values.

Regarding the Whole Tumor region, Figure 40 shows that all three methods achieve high Dice scores, although they differ significantly in consistency. Method 1 shows broader dispersion; Method 2 contains a single extreme low point (Dice ≈ 0.021) which the boxplot marks as an outlier, extending the lower tail; Method 3 exhibits the most compact distribution (smallest IQR) while maintaining a high median. On the common 18-case subset, medians/IQRs are 0.859 / 0.219 (M1), 0.906 / 0.175 (M2), and 0.921 / 0.068 (M3).

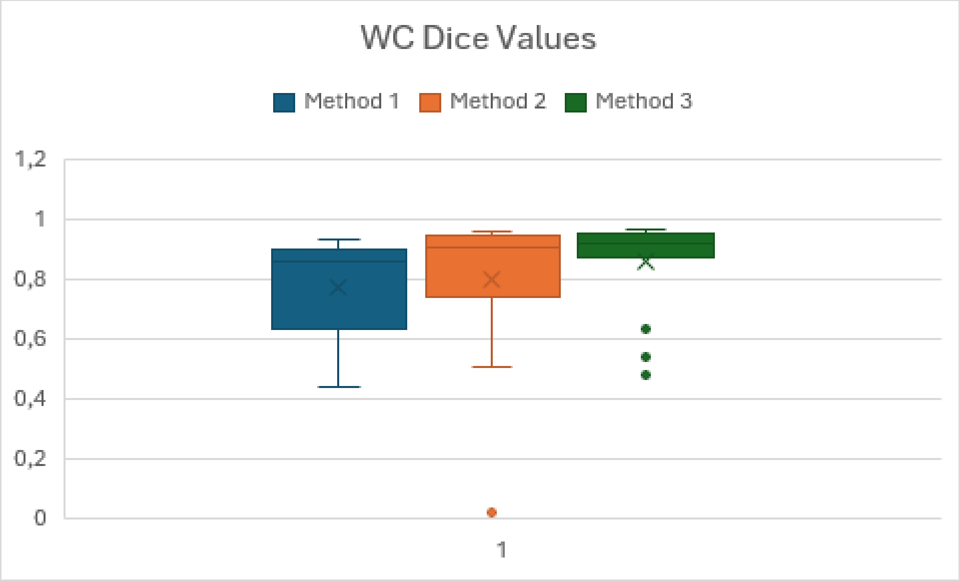


Figure 40: Boxplot of WT Test set Dice values.

6.5. Metrics Summary

The test results for the three preprocessing methods were evaluated using standard segmentation metrics for each tumor region, as summarized in Tables 4, 5, and 6. Methods 2 and 3 were each tested on 20 patients, whereas Method 1 used a slightly smaller test set of 18 patients. Therefore, comparisons between Method 2 and Method 3 are more directly reliable since they share the same test sample size, while Method 1's results may be less directly comparable due to its smaller n.

Overall, Method 3 achieved the best segmentation performance, with higher values in all metrics across all regions. For example, Method 3 presents consistently higher Dice scores and IoU across ET, TC, and WT regions compared to Method 2 (Table 5). This improvement was especially pronounced in the TC region: Method 3's average TC Dice was substantially higher (around 0.68, versus approximately 0.53 with Method 2), and its TC recall was also much improved (around 0.69 vs 0.46), indicating that Method 3 captured the core tumor regions more completely.

Futhermore, Table 6 shows that Method 3 showed higher precision in TC (around 0.80 vs 0.70 for Method 2), meaning fewer false positive predictions within the core, while maintaining a high TC specificity (on the order of 0.86) – a slight trade-off compared to Method 2's TC specificity (~0.89) but still indicating that most non-core regions were correctly identified as tumor-free. In the ET region, Method 3 also outperformed Method 2, with higher ET Dice (~0.63 vs 0.55) and recall (~0.63 vs 0.57), and notably improved precision (~0.74 vs 0.61), reflecting fewer false alarms in non-enhancing areas. For the WT region, Method 3 again achieved better results (WT Dice ~0.84 vs ~0.77 in Method 2), along with higher recall (~0.92 vs ~0.81), indicating more accurate and comprehensive segmentation of the entire tumor extent.

In contrast, the results for Method 1, detailed in Table 4, reveal the lowest performed overall, with considerably lower Dice and recall values in all regions (for instance, ET Dice was only about 0.44 and TC Dice ~0.42), reflecting a tendency to miss a large portion of the tumor with many false negatives. Method 1 did exhibit high precision and specificity in the ET and TC categories (precision ~0.78–0.80 and specificity above 0.92 for those regions), meaning it produced relatively few false positives; however, this came at the cost of much lower recall (TC recall ~0.35), implying that Method 1 often failed to detect significant parts of the tumor.

Table 4: Method 1 segmentation metrics summary.

region	n	dice	recall	precision	specificity	iou
ET	18	0,440 ± 0,277	0,400 ± 0,295	0,796 ± 0,325	0,962 ± 0,090	0,323 ± 0,249

TC	18	0,418 ± 0,267	0,354 ± 0,292	0,782 ± 0,300	0,929 ± 0,119	0,302 ± 0,242
WT	18	0,777 ± 0,172	0,848 ± 0,160	0,795 ± 0,247	0,000 ± 0,000	0,662 ± 0,209

Table 5: Method 2 segmentation metrics summary.

region	n	dice	recall	precision	specificity	iou
ET	20	0,553 ± 0,339	0,565 ± 0,354	0,609 ± 0,370	0,908 ± 0,091	0,449 ± 0,301
TC	20	0,529 ± 0,322	0,457 ± 0,312	0,703 ± 0,369	0,888 ± 0,130	0,420 ± 0,295
WT	20	0,771 ± 0,251	0,813 ± 0,233	0,761 ± 0,271	0,000 ± 0,000	0,679 ± 0,269

Table 6: Method 3 segmentation metrics summary.

region	n	dice	recall	precision	specificity	iou
ET	20	0,626 ± 0,311	0,630 ± 0,302	0,744 ± 0,308	0,929 ± 0,108	0,519 ± 0,297
TC	20	0,682 ± 0,255	0,688 ± 0,277	0,800 ± 0,231	0,861 ± 0,191	0,563 ± 0,254
WT	20	0,844 ± 0,155	0,916 ± 0,105	0,801 ± 0,201	0,000 ± 0,000	0,755 ± 0,202

7. Discussion

The results indicate that Method 3 outperformed all other preprocessing methods across all tumor sub-regions. This method achieved the highest mean Dice scores, with the most notable improvement in the TC region (+29%), as well as gains of +13% in ET and +9% in WT, compared to Method 2. These improvements confirm that the additional steps introduced in Method 3, N4 bias field correction, intracranial z-score normalization, and FLAIR-based ROI centering, substantially enhanced segmentation performance.

The TC region, generally the most challenging subregion to segment, benefited the most from the improved pipeline. The U-Net was able to better distinguish internal tumor structures when inputs were carefully and consistently preprocessed. This demonstrates that, even with datasets such as BraTS, which are already partially pre-processed, like skull stripping, preprocessing remains an important and valuable step. It provides a relatively simple yet effective strategy to achieve better results, enabling more precise diagnosis and localization of gliomas in a clinical context.

The N4 bias field correction algorithm is typically used to correct intensity inhomogeneities in MR images, which often exhibit low-frequency artifacts that make the same tissue appear with different intensities across the field of view. This became especially important in this dataset because the images originated from multiple institutions and scanners, even under similar acquisition protocols, data are susceptible to such artifacts. N4 estimates and removes a smooth, multiplicative bias field, producing more homogeneous tissue intensities. By normalizing illumination, N4 enables the model to focus on true tissue patterns like contrast enhancement, rather than acquisition artifacts (ANTsX, 2021; Tustison et al., 2010) . This likely improved the model's overall reliability and helped drive the better segmentation results seen with Method 3.

Another improvement in Method 3 was the use of intracranial z-score normalization in place of the simple min–max scaling used in Method 1. Method 1's per-volume min–max normalization is highly sensitive to intensity outliers and can be inconsistent across scanners (Scikit, 2025). In contrast, Methods 2 and 3 applied *z*-score normalizations computed only on the non-zero (intracranial) voxels of each MRI volume. This approach avoids being skewed by the zero-valued background and ensures that each imaging modality has a mean of approximately 0 and a standard deviation of 1 within the brain region. Such normalization is essential because MRI intensity units are arbitrary and can vary widely between different scanners and protocols. By placing all scans on a comparable intensity scale, *z*-score normalization reduced intensity variability across patients and institutions, contributing to a more stable and robust training process. This standardization likely aided Method 3, and Method 2 to a lesser extent, in achieving higher accuracy than Method 1.

Method 1 used a fixed crop at the image center, often missing off-center lesions. Method 2 improved on this by centering on the brightest voxel in T1Gd, which works for strongly enhancing tumors but fails for low-grade gliomas, non-enhancing regions, or non-tumoral bright spots, like vessels. Method 3 replaced this with an image-only strategy using FLAIR: Otsu thresholding followed by largest connected component selection created a brain/tumor mask without labels. This guaranteed that the tumor and surrounding ED were included in the field of view. FLAIR is particularly suited for this, since both ED and NET are hyperintense on FLAIR/T2. As a result, the U-Net tend to receive inputs where the lesion was fully visible and spatially consistent, which likely explains the strong Dice improvements in TC.

7.1.1. Study Limitations

Despite the clear performance gains achieved, several limitations of this study must be acknowledged. First, the approach used a 2D U-Net model operating on axial slices, which means that each slice was segmented independently, without leveraging any 3D spatial context (Yang et al., 2025). Gliomas, however, are three-dimensional lesions, and continuity across slices can provide important cues for segmentation. Using only a single representative slice per patient for training and validation, to avoid slice redundancy from the same tumor, further reduced the effective dataset size, which increases the risk of overfitting and limits the model’s exposure to the full variability of tumor appearance. Indeed, the training cohort was relatively small, and the test set included only 20 cases for Methods 2 and 3 (and 18 for Method 1), which restricts statistical confidence in the reported results. With such a limited sample, performance metrics can vary widely, making it difficult to draw very strong or generalizable conclusions about real-world performance. Additionally, the model was not evaluated on any independent external dataset, so its generalizability to new data from other sources remains unproven. All of these factors, working with 2D slices instead of volumetric data, a small sample size, and lack of external validation, constrain the broader applicability of the findings.

Additionally, excluding the native T1-weighted MRI modality – while reducing data redundancy and computational load – may have sacrificed valuable information. The ET is defined by areas that appear hyper-intense on T1Gd when compared to the baseline T1 image, whereas NCR components are typically hypo-intense on T1Gd relative to T1 (Baid et al., 2021). Leaving out T1 means the model could not use these intensity differences, which may have made it harder to clearly separate ET and NCR within the TC.

The small test set also restricts statistical confidence in the results. With only a limited number of cases for evaluation, performance metrics can vary widely, making it difficult to draw strong conclusions or generalize the model's accuracy. Another practical constraint was limited time for optimization and advanced experimentation, although Python courses were completed and considerable skills were gained, the level of expertise and time was still not sufficient to pursue more advanced experiments. Finally, a bug in computing TN for the WT class invalidated reported specificity values for that region, although overlap metrics like Dice, were not affected.

7.1.2. Future Directions

Planned improvements include correcting the TN bug and applying lightweight post-processing to predictions, such as removing very small, isolated components or filling holes. These simple steps are common in frameworks like nnU-Net and MONAI and have proven effective in reducing false positives. A hybrid ROI centering strategy combining FLAIR and T1Gd, combining method 2 and method 3 approaches, could be interest to increase robustness by ensuring both diffuse ED and focal enhancement are included. More importantly, future work should move beyond 2D toward 2.5D or full 3D U-Nets, which better capture volumetric context. Also, including the T1 sequence in future work could provide complementary information and improve segmentation performance for these substructures. Expanding training data is also critical: if more annotated cases are not available, data augmentation and semi-supervised learning can exploit unlabelled volumes to improve generalization.

A persistent barrier to reproducible tumor segmentation is the lack of fully standardized MR acquisition across centers. As already emphasized, heterogeneity in sequence parameters, contrast timing, and resolution can undermine model robustness and limit portability beyond the site of development. Standardized, multi-site protocols would reduce the need for site-specific fine-tuning and are therefore a prerequisite for reliable clinical deployment of automatic methods. Another recurrent issue is the gap between performance obtained in well-controlled research settings and what is achievable in day-to-day clinical routine.

Results reported on clean datasets or feasibility studies may not carry over when confronted with protocol variability, motion, treatment-related effects, or multifocal disease. Clinical workflows are also constrained by runtime. While true real-time segmentation is difficult, computation times beyond a few minutes are unacceptable in routine practice (Bauer et al., 2013). In this setting, full preprocessing of the initial cohort routinely required an overnight run, and single-case inference frequently exceeded one hour, clearly incompatible with real-world expectations. These constraints motivated the downsized training cohort and focus on lightweight, preprocessing-centric pipelines rather than heavier 3D models.

8. Conclusion

This work achieved the proposed objective of comparing different preprocessing pipelines for glioma segmentation in MRI scans while holding a simple segmentation model U-Net 2D. The results show that the combination of intracranial normalization, bias-field correction, and FLAIR-guided centering, Method 3, yielded the best segmentation performance among the approaches tested. This more comprehensive pipeline outperformed the simpler methods, Methods 1 and 2, across all evaluated tumor subregions (ET, TC, and WT), with particularly notable gains in TC, with an increase of approximately 0.15 in the Dice coefficient (from about 0.53 to 0.68) relative to Method 2. These findings confirm that careful and consistent image preprocessing can significantly improve input quality and, consequently, the stability of the 2D model, showing that even a relatively basic architecture can achieve more accurate segmentations when provided with higher-quality data.

Despite these results, there are limitations to consider. Segmentation was performed in 2D, processing each axial slice independently; although this approach simplified training and highlighted differences between pipelines, it sacrifices spatial continuity across adjacent slices, which may limit the precision with which the full tumor volume is delineated. Additionally, the methodology was developed and evaluated exclusively with data from the BraTS 2023 challenge; although this dataset is extensive and demanding, it does not fully reproduce the variability of images obtained in real clinical settings with different scanners and acquisition protocols. It therefore remains unclear to what extent the observed gains generalize to clinical scenarios outside the controlled environment of that challenge.

Based on these observations, two main directions emerge for future work. The first is the natural extension to a three-dimensional segmentation model, such as a 3D U-Net, that leverages full volumetric information and ensures continuity across slices, potentially improving the detection of tumor structures throughout the volume. The same preprocessing steps developed in this project (intracranial normalization, bias-field correction, FLAIR-guided centering) are expected to be adaptable to a 3D pipeline, although it will be necessary to investigate the impact of increased dimensionality on computational cost and performance. The second direction is the external validation of the approach on real clinical data, obtained from different institutions and MRI scanners, to assess the robustness and generalization capability of the method outside the BraTS context. Such evaluation in a clinical setting will verify whether the segmentation improvements persist and translate into practical benefits, such as more reliable identification of tumor components, streamlining diagnosis and treatment planning.

Finally, reflecting on this work, I conclude that investing in preprocessing quality can be truly beneficial to AI models for brain tumor segmentation.

References

ANTsX. (2021, January 12). *N4BiasFieldCorrection*. GitHub Wiki. https://github.com/ANTsX/ANTs/wiki/N4BiasFieldCorrection

Azad, R., Aghdam, E. K., Rauland, A., Jia, Y., Avval, A. H., Bozorgpour, A., Karimijafarbigloo, S., Cohen, J. P., Adeli, E., & Merhof, D. (2022). *Medical Image Segmentation Review: The success of U-Net*. http://arxiv.org/abs/2211.14830

Baid, U., Ghodasara, S., Mohan, S., Bilello, M., Calabrese, E., Colak, E., Farahani, K., Kalpathy-Cramer, J., Kitamura, F. C., Pati, S., Prevedello, L. M., Rudie, J. D., Sako, C., Shinohara, R. T., Bergquist, T., Chai, R., Eddy, J., Elliott, J., Reade, W., … Bakas, S. (2021a). *The RSNA-ASNR-MICCAI BraTS 2021 Benchmark on Brain Tumor Segmentation and Radiogenomic Classification*. http://arxiv.org/abs/2107.02314

Baid, U., Ghodasara, S., Mohan, S., Bilello, M., Calabrese, E., Colak, E., Farahani, K., Kalpathy-Cramer, J., Kitamura, F. C., Pati, S., Prevedello, L. M., Rudie, J. D., Sako, C., Shinohara, R. T., Bergquist, T., Chai, R., Eddy, J., Elliott, J., Reade, W., … Bakas, S. (2021b). *The RSNA-ASNR-MICCAI BraTS 2021 Benchmark on Brain Tumor Segmentation and Radiogenomic Classification*. http://arxiv.org/abs/2107.02314

Bakas, S., Reyes, M., Jakab, A., & Bauer Helbling, S. (2019). *Identifying the Best Machine Learning Algorithms for Brain Tumor Segmentation, Progression Assessment, and Overall Survival Prediction in the BRATS Challenge*. https://www.researchgate.net/publication/331576745

Bauer, S., Wiest, R., Nolte, L. P., & Reyes, M. (2013). A survey of MRI-based medical image analysis for brain tumor studies. In *Physics in Medicine and Biology* (Vol. 58, Issue 13). https://doi.org/10.1088/0031-9155/58/13/R97

Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A. L., & Zhou, Y. (2021). *TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation*. http://arxiv.org/abs/2102.04306

Cheng, W. L. (2025). *Performing Z-Score Normalization in Python*. https://gist.github.com/weiloon-datature/3b3949529d2131946893f64a13b2a396#file-zscore_normalization-py

Coyne, K., & National MagLab. (2015). *MRI: A Guided Tour*. https://nationalmaglab.org/magnet-academy/read-science-stories/science-simplified/mri-a-guided-tour/

Cray, D. (2025, July 23). *What are different evaluation metrics used to evaluate image segmentation models?* https://www.geeksforgeeks.org/computer-vision/what-are-different-evaluation-metrics-used-to-evaluate-image-segmentation-models/

De Sutter, S., Wuts, J., Geens, W., Vanbinst, A. M., Duerinck, J., & Vandemeulebroucke, J. (2024). Modality redundancy for MRI-based glioblastoma segmentation. *International Journal of Computer Assisted Radiology and Surgery*, *19*(10), 2101–2109. https://doi.org/10.1007/s11548-024-03238-4

Dhabalia, R., Kashikar, S. V, Parihar, P. S., & Mishra, G. V. (2024). Unveiling the Intricacies: A Comprehensive Review of Magnetic Resonance Imaging (MRI) Assessment of T2-Weighted Hyperintensities in the Neuroimaging Landscape. *Cureus*. https://doi.org/10.7759/cureus.54808

Dorfner, F. J., Patel, J. B., Kalpathy-Cramer, J., Gerstner, E. R., & Bridge, C. P. (2025). A review of deep learning for brain tumor analysis in MRI. In *npj Precision Oncology* (Vol. 9, Issue 1). Nature Research. https://doi.org/10.1038/s41698-024-00789-2

Erickson, B. J., Korfiatis, P., Kline, T. L., Akkus, Z., Philbrick, K., & Weston, A. D. (2018). Deep Learning in Radiology: Does One Size Fit All? *Journal of the American College of Radiology*, *15*(3), 521–526. https://doi.org/10.1016/j.jacr.2017.12.027

Farhan, A. S., Khalid, M., & Manzoor, U. (2025). XAI-MRI: an ensemble dual-modality approach for 3D brain tumor segmentation using magnetic resonance imaging. *Frontiers in Artificial Intelligence*, *8*. https://doi.org/10.3389/frai.2025.1525240

Feng, X., Tustison, N. J., Patel, S. H., & Meyer, C. H. (2020). Brain Tumor Segmentation Using an Ensemble of 3D U-Nets and Overall Survival Prediction Using Radiomic Features. *Frontiers in Computational Neuroscience*, *14*. https://doi.org/10.3389/fncom.2020.00025

Gaillard, F., Agazzi, G. M., & Ryan, P. (2008). Glioblastoma, IDH-wildtype. In *Radiopaedia.org*. Radiopaedia.org. https://doi.org/10.53347/rID-4910

Gaillard, F., Baba, Y., & Bell, D. (2015). MRI sequences (overview). In *Radiopaedia.org*. Radiopaedia.org. https://doi.org/10.53347/rID-37346

Global Cancer Observatory. (2022, February 8). *Brain Central Nervous System Fact Sheet*. Globocan 2022.

Huang, H., Lin, L., Tong, R., Hu, H., Zhang, Q., Iwamoto, Y., Han, X., Chen, Y.-W., & Wu, J. (2020). *UNET 3+: A FULL-SCALE CONNECTED UNET FOR MEDICAL IMAGE SEGMENTATION*.

Huang, J., Yagmurlu, B., Molleti, P., Lee, R., VanderPloeg, A., Noor, H., Bareja, R., Li, Y., Iv, M., & Itakura, H. (2025). Brain tumor segmentation using deep learning: high performance with minimized MRI data. *Frontiers in Radiology*, *5*. https://doi.org/10.3389/fradi.2025.1616293

Ilic, I., & Ilic, M. (2023). International patterns and trends in the brain cancer incidence and mortality: An observational study based on the global burden of disease. *Heliyon*, *9*(7). https://doi.org/10.1016/j.heliyon.2023.e18222

Isensee, F., Jaeger, P. F., Full, P. M., Vollmuth, P., & Maier-Hein, K. H. (2020). *nnU-Net for Brain Tumor Segmentation*. http://arxiv.org/abs/2011.00848

Isensee, F., Jaeger, P. F., Kohl, S. A. A., Petersen, J., & Maier-Hein, K. H. (2021). nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods*, *18*(2), 203–211. https://doi.org/10.1038/s41592-020-01008-z

Jones, J., Moore, C., & Haouimi, A. (2009). T2-weighted image. In *Radiopaedia.org*. Radiopaedia.org. https://doi.org/10.53347/rID-6345

Li, Z.-C., Yan, J., Zhang, S., Liang, C., Lv, X., Zou, Y., Zhang, H., Liang, D., Zhang, Z., Chen, Y., & Cn, C. O. (2022). *Glioma survival prediction from whole-brain MRI without tumor segmentation using deep attention network: a multicenter study*. https://doi.org/10.1007/s00330-022-08640-7/Published

Liu, Z., Tong, L., Chen, L., Jiang, Z., Zhou, F., Zhang, Q., Zhang, X., Jin, Y., & Zhou, H. (2023). Deep learning based brain tumor segmentation: a survey. *Complex and Intelligent Systems*, *9*(1), 1001–1026. https://doi.org/10.1007/s40747-022-00815-5

Lotan, E., Jain, R., Razavian, N., Fatterpekar, G. M., & Lui, Y. W. (2019). State of the art: Machine learning applications in Glioma Imaging. In *American Journal of Roentgenology* (Vol. 212, Issue 1, pp. 26–37). American Roentgen Ray Society. https://doi.org/10.2214/AJR.18.20218

Louis, D. N., Perry, A., Wesseling, P., Brat, D. J., Cree, I. A., Figarella-Branger, D., Hawkins, C., Ng, H. K., Pfister, S. M., Reifenberger, G., Soffietti, R., Von Deimling, A., & Ellison, D. W. (2021). The 2021 WHO classification of tumors of the central nervous system: A summary. *Neuro-Oncology*, *23*(8), 1231–1251. https://doi.org/10.1093/neuonc/noab106

Lowekamp, B. C., Chen, D. T., Ibáñez, L., & Blezek, D. (2013). The design of simpleITK. *Frontiers in Neuroinformatics*, *7*(DEC). https://doi.org/10.3389/fninf.2013.00045

Ma, J., He, Y., Li, F., Han, L., You, C., & Wang, B. (2024). Segment anything in medical images. *Nature Communications*, *15*(1). https://doi.org/10.1038/s41467-024-44824-z

Müller, D., Soto-Rey, I., & Kramer, F. (2022). *TOWARDS A GUIDELINE FOR EVALUATION METRICS IN MEDICAL IMAGE SEGMENTATION*.

Niknejad, M. T., Chieng, R., & Baba, Y. (2013). Fluid attenuated inversion recovery. In *Radiopaedia.org*. Radiopaedia.org. https://doi.org/10.53347/rID-21760

Ranjbarzadeh, R., Bagherian Kasgari, A., Jafarzadeh Ghoushchi, S., Anari, S., Naseri, M., & Bendechache, M. (2021). Brain tumor segmentation based on deep learning and an attention mechanism using MRI multi-modalities brain images. *Scientific Reports*, *11*(1). https://doi.org/10.1038/s41598-021-90428-8

Rios Piedra, E. A., Taira, R. K., El-Saden, S., Ellingson, B. M., Bui, A. A. T., & Hsu, W. (2016). *Assessing Variability in Brain Tumor Segmentation to Improve Volumetric Accuracy and Characterization of Change * HHS Public Access*. 380–383. https://doi.org/10.1109/BHI

Ronneberger, O., Fischer, P., & Brox, T. (2015). *U-Net: Convolutional Networks for Biomedical Image Segmentation*. http://arxiv.org/abs/1505.04597

Roth, J., Keller, J., Franke, S., Neumuth, T., & Schneider, D. (2022). *Multi-plane UNet++ Ensemble for Glioblastoma Segmentation* (pp. 285–294). https://doi.org/10.1007/978-3-031-08999-2_23

Scikit. (2025). *Compare the effect of different scalers on data with outliers*. https://scikit-learn.org/stable/auto_examples/preprocessing/plot_all_scaling.html

Shukla, G., Alexander, G. S., Bakas, S., Nikam, R., Talekar, K., Palmer, J. D., & Shi, W. (2017). Advanced magnetic resonance imaging in glioblastoma: A review. In *Chinese Clinical Oncology* (Vol. 6, Issue 4). AME Publishing Company. https://doi.org/10.21037/cco.2017.06.28

Tustison, N. J., Avants, B. B., Cook, P. A., Zheng, Y., Egan, A., Yushkevich, P. A., & Gee, J. C. (2010). N4ITK: Improved N3 bias correction. *IEEE Transactions on Medical Imaging*, *29*(6), 1310–1320. https://doi.org/10.1109/TMI.2010.2046908

Villanueva-Meyer, J. E., Mabray, M. C., & Cha, S. (2017). Current clinical brain tumor imaging. *Clinical Neurosurgery*, *81*(3), 397–415. https://doi.org/10.1093/neuros/nyx103

Wadhwa, A., Bhardwaj, A., & Singh Verma, V. (2019). A review on brain tumor segmentation of MRI images. In *Magnetic Resonance Imaging* (Vol. 61, pp. 247–259). Elsevier Inc. https://doi.org/10.1016/j.mri.2019.05.043

Wan, B., Hu, B., Zhao, M., Li, K., & Ye, X. (2023). Deep learning-based magnetic resonance image segmentation technique for application to glioma. *Frontiers in Medicine*, *10*. https://doi.org/10.3389/fmed.2023.1172767

Wang, P., Yang, Q., He, Z., & Yuan, Y. (2023). Vision transformers in multi-modal brain tumor MRI segmentation: A review. *Meta-Radiology*, *1*(1), 100004. https://doi.org/10.1016/j.metrad.2023.100004

Wen, P. Y., & Packer, R. J. (2021). The 2021 WHO Classification of Tumors of the Central Nervous System: Clinical implications. In *Neuro-Oncology* (Vol. 23, Issue 8, pp. 1215–1217). Oxford University Press. https://doi.org/10.1093/neuonc/noab120

Xiang, T., Zhang, C., Liu, D., Song, Y., Huang, H., & Cai, W. (2020). *BiO-Net: Learning Recurrent Bi-directional Connections for Encoder-Decoder Architecture*. http://arxiv.org/abs/2007.00243

Xu, J., Meng, Y., Qiu, K., Topatana, W., Li, S., Wei, C., Chen, T., Chen, M., Ding, Z., & Niu, G. (2022). Applications of Artificial Intelligence Based on Medical Imaging in Glioma: Current State and Future Challenges. In *Frontiers in Oncology* (Vol. 12). Frontiers Media S.A. https://doi.org/10.3389/fonc.2022.892056

Xu, Y., Quan, R., Xu, W., Huang, Y., Chen, X., & Liu, F. (2024). Advances in Medical Image Segmentation: A Comprehensive Review of Traditional, Deep Learning and Hybrid Approaches. In *Bioengineering* (Vol. 11, Issue 10). Multidisciplinary Digital Publishing Institute (MDPI). https://doi.org/10.3390/bioengineering11101034

Yang, S., Jeong, J. S., Song, D., Han, J. Y., Lim, S. H., Kim, S., Yoo, J. Y., Kim, J. M., Kim, J. E., Huh, K. H., Lee, S. S., Heo, M. S., & Yi, W. J. (2025). Comparison of 2D, 2.5D, and 3D segmentation networks for mandibular canals in CBCT images: a study on public and external datasets. *BMC Oral Health*, *25*(1). https://doi.org/10.1186/s12903-025-06483-4

Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2018). *UNet++: A Nested U-Net Architecture for Medical Image Segmentation*. http://arxiv.org/abs/1807.10165

Appendices

Table 7: Method-3 vs Method-2 gains: Absolute and percentage improvements for ET, TC, WT (validation set).

Δ	ET	TC	WT
M3-M2	0,059	0,027	0,009
%M3-M2	8%	4%	1%

Table 8: Method-3 vs Method-2 gains: Absolute and percentage improvements for ET, TC, WT (test set).

Δ	ET	TC	WT
M3-M2	0,073	0,153	0,072
%M3-M2	13%	29%	9%

Boxplot Data

Table 9: WT Dice Values Test per patient per method

WT Dice Values Test	Method 1	Method 2	Method 3
BraTS-GLI-00022_Private-Collection_18	0,934	0,906	0,891
BraTS-GLI-00053_Private-Collection_18	0,814	0,504	0,882
BraTS-GLI-00233_Private-Collection_1	0,913	0,956	0,956
BraTS-GLI-00426_Private-Collection_1	0,892	0,958	0,959
BraTS-GLI-00468_Private-Collection_18	0,438	0,751	0,919
BraTS-GLI-00481_Private-Collection_18	0,929	0,021	0,873
BraTS-GLI-00533_Private-Collection_18	0,850	0,943	0,921
BraTS-GLI-00569_IvyGAP_16	0,490	0,710	0,540
BraTS-GLI-00590_IvyGAP_16	0,712	0,931	0,928
BraTS-GLI-00608_Private-Collection_18	0,893	0,906	0,967
BraTS-GLI-00676_Private-Collection_18	0,544	0,827	0,960
BraTS-GLI-00688_Private-Collection_18	0,845	0,546	0,483
BraTS-GLI-01135_Private-Collection_1	0,871	0,850	0,930
BraTS-GLI-01167_Private-Collection_4	0,661	0,841	0,636
BraTS-GLI-01231_Private-Collection_1	0,928	0,918	0,935
BraTS-GLI-01386_Private-Collection_1	0,889	0,954	0,951
BraTS-GLI-01442_TCGA-GBM_6	0,868	0,957	0,921
BraTS-GLI-01524_TCGA-LGG_14	0,506	0,945	0,879

Table 10: Boxplot statistics: Q1, median, Q3, and IQR for WT.

	Method 1	Method 2	Method 3
Q1	0,674	0,770	0,880
Mediana	0,859	0,906	0,921
Q3	0,893	0,945	0,947
IQR	0,219	0,175	0,068

Table 11: ET Dice Values Test per patient per method.

ET Dice Values Test	Method 1	Method 2	Method 3
BraTS-GLI-00022_Private-Collection_18	0,348	0,789	0,787
BraTS-GLI-00053_Private-Collection_18	0,223	0,000	0,767
BraTS-GLI-00233_Private-Collection_1	0,866	0,850	0,906
BraTS-GLI-00426_Private-Collection_1	0,854	0,882	0,919
BraTS-GLI-00468_Private-Collection_18	0,187	0,569	0,491
BraTS-GLI-00481_Private-Collection_18	0,602	0,000	0,785
BraTS-GLI-00533_Private-Collection_18	0,747	0,774	0,789
BraTS-GLI-00569_IvyGAP_16	0,435	0,466	0,308
BraTS-GLI-00590_IvyGAP_16	0,328	0,890	0,847
BraTS-GLI-00608_Private-Collection_18	0,027	0,325	0,374
BraTS-GLI-00676_Private-Collection_18	0,435	0,831	0,877
BraTS-GLI-00688_Private-Collection_18	0,535	0,661	0,482
BraTS-GLI-01135_Private-Collection_1	0,351	0,769	0,869
BraTS-GLI-01167_Private-Collection_4	0,153	0,219	0,129
BraTS-GLI-01231_Private-Collection_1	0,286	0,556	0,579
BraTS-GLI-01386_Private-Collection_1	0,768	0,850	0,888
BraTS-GLI-01442_TCGA-GBM_6	0,767	0,864	0,918
BraTS-GLI-01524_TCGA-LGG_14	0,000	0,000	0,000

Table 12: Boxplot statistics: Q1, median, Q3, and IQR for ET.

Q1	0,239	0,360	0,485
Mediana	0,393	0,715	0,786
Q3	0,711	0,845	0,875
IQR	0,472	0,485	0,391

Table 13: TC Dice Values Test per patient per method

TC Dice Values Test	Method 1	Method 2	Method 3
BraTS-GLI-00022_Private-Collection_18	0,322	0,729	0,876
BraTS-GLI-00053_Private-Collection_18	0,685	0,000	0,699
BraTS-GLI-00233_Private-Collection_1	0,828	0,852	0,889
BraTS-GLI-00426_Private-Collection_1	0,714	0,912	0,681
BraTS-GLI-00468_Private-Collection_18	0,121	0,493	0,478
BraTS-GLI-00481_Private-Collection_18	0,488	0,000	0,846
BraTS-GLI-00533_Private-Collection_18	0,557	0,692	0,871
BraTS-GLI-00569_IvyGAP_16	0,300	0,287	0,371
BraTS-GLI-00590_IvyGAP_16	0,259	0,813	0,849

BraTS-GLI-00608_Private-Collection_18	0,010	0,148	0,198
BraTS-GLI-00676_Private-Collection_18	0,247	0,570	0,811
BraTS-GLI-00688_Private-Collection_18	0,294	0,677	0,732
BraTS-GLI-01135_Private-Collection_1	0,170	0,487	0,775
BraTS-GLI-01167_Private-Collection_4	0,138	0,203	0,461
BraTS-GLI-01231_Private-Collection_1	0,219	0,421	0,628
BraTS-GLI-01386_Private-Collection_1	0,618	0,729	0,836
BraTS-GLI-01442_TCGA-GBM_6	0,918	0,971	0,979
BraTS-GLI-01524_TCGA-LGG_14	0,632	0,796	0,840
	0,418	0,543	0,712

Table 14: Boxplot statistics: Q1, median, Q3, and IQR for TC.

Q1	0,226	0,321	0,641
Mediana	0,311	0,624	0,793
Q3	0,629	0,779	0,848
IQR	0,403	0,459	0,207

Table 15: Method 1 per-patient evaluation metrics and data: TP, FP, FN, TN per case and region.

patient	region	tp	fp	fn	tn	dice	recall	precision	specificity	iou
BraTS-GLI-00022_Private-Collection_18	ET	367	0	1376	1045	0,348	0,211	1,000	1,000	0,211
BraTS-GLI-00022_Private-Collection_18	TC	369	0	1552	867	0,322	0,192	1,000	1,000	0,192
BraTS-GLI-00022_Private-Collection_18	WT	2441	109	238	0	0,934	0,911	0,957	0,000	0,876
BraTS-GLI-00053_Private-Collection_18	ET	51	2	353	2295	0,223	0,126	0,962	0,999	0,126
BraTS-GLI-00053_Private-Collection_18	TC	584	374	163	1580	0,685	0,782	0,610	0,809	0,521
BraTS-GLI-00053_Private-Collection_18	WT	1855	64	782	0	0,814	0,704	0,967	0,000	0,687
BraTS-GLI-00233_Private-Collection_1	ET	275	47	38	1953	0,866	0,879	0,854	0,977	0,764
BraTS-GLI-00233_Private-Collection_1	TC	347	67	77	1822	0,828	0,818	0,838	0,965	0,707
BraTS-GLI-00233_Private-Collection_1	WT	1942	19	352	0	0,913	0,847	0,990	0,000	0,840
BraTS-GLI-00426_Private-Collection_1	ET	710	73	170	3536	0,854	0,807	0,907	0,980	0,745
BraTS-GLI-00426_Private-Collection_1	TC	951	621	142	2775	0,714	0,870	0,605	0,817	0,555
BraTS-GLI-00426_Private-Collection_1	WT	3617	94	778	0	0,892	0,823	0,975	0,000	0,806
BraTS-GLI-00468_Private-Collection_18	ET	149	1	1296	2670	0,187	0,103	0,993	1,000	0,103
BraTS-GLI-00468_Private-Collection_18	TC	149	1	2156	1810	0,121	0,065	0,993	0,999	0,065
BraTS-GLI-00468_Private-Collection_18	WT	1153	0	2963	0	0,438	0,280	1,000	0,000	0,280
BraTS-GLI-00481_Private-Collection_18	ET	340	8	441	2035	0,602	0,435	0,977	0,996	0,431
BraTS-GLI-00481_Private-Collection_18	TC	348	0	731	1745	0,488	0,323	1,000	1,000	0,323
BraTS-GLI-00481_Private-Collection_18	WT	2451	103	270	0	0,929	0,901	0,960	0,000	0,868
BraTS-GLI-00533_Private-Collection_18	ET	367	97	152	1030	0,747	0,707	0,791	0,914	0,596

BraTS-GLI-00533_Private-Collection_18	TC	426	43	635	542	0,557	0,402	0,908	0,927	0,386
BraTS-GLI-00533_Private-Collection_18	WT	1217	160	269	0	0,850	0,819	0,884	0,000	0,739
BraTS-GLI-00569_IvyGAP_16	ET	614	1505	92	9256	0,435	0,870	0,290	0,860	0,278
BraTS-GLI-00569_IvyGAP_16	TC	643	1476	1531	7817	0,300	0,296	0,303	0,841	0,176
BraTS-GLI-00569_IvyGAP_16	WT	3725	7513	229	0	0,490	0,942	0,332	0,000	0,325
BraTS-GLI-00590_IvyGAP_16	ET	105	0	430	3630	0,328	0,196	1,000	1,000	0,196
BraTS-GLI-00590_IvyGAP_16	TC	105	1	600	3459	0,259	0,149	0,991	1,000	0,149
BraTS-GLI-00590_IvyGAP_16	WT	2304	1650	211	0	0,712	0,916	0,583	0,000	0,553
BraTS-GLI-00608_Private-Collection_18	ET	14	0	1014	2260	0,027	0,014	1,000	1,000	0,014
BraTS-GLI-00608_Private-Collection_18	TC	14	52	2750	472	0,010	0,005	0,212	0,901	0,005
BraTS-GLI-00608_Private-Collection_18	WT	2654	170	464	0	0,893	0,851	0,940	0,000	0,807
BraTS-GLI-00676_Private-Collection_18	ET	145	2	375	3763	0,435	0,279	0,986	1,000	0,278
BraTS-GLI-00676_Private-Collection_18	TC	147	0	894	3244	0,248	0,141	1,000	1,000	0,141
BraTS-GLI-00676_Private-Collection_18	WT	1599	2686	0	0	0,544	1,000	0,373	0,000	0,373
BraTS-GLI-00688_Private-Collection_18	ET	72	4	121	1117	0,535	0,373	0,947	0,996	0,366
BraTS-GLI-00688_Private-Collection_18	TC	78	1	374	861	0,294	0,173	0,987	0,999	0,172
BraTS-GLI-00688_Private-Collection_18	WT	961	306	47	0	0,845	0,953	0,759	0,000	0,731
BraTS-GLI-01135_Private-Collection_1	ET	190	10	692	2804	0,351	0,215	0,950	0,996	0,213
BraTS-GLI-01135_Private-Collection_1	TC	225	11	2187	1273	0,170	0,093	0,953	0,991	0,093
BraTS-GLI-01135_Private-Collection_1	WT	2851	91	754	0	0,871	0,791	0,969	0,000	0,771
BraTS-GLI-01167_Private-Collection_4	ET	164	1663	147	2846	0,153	0,527	0,090	0,631	0,083
BraTS-GLI-01167_Private-Collection_4	TC	220	1620	1117	1863	0,139	0,165	0,120	0,535	0,074
BraTS-GLI-01167_Private-Collection_4	WT	2378	2281	161	0	0,661	0,937	0,510	0,000	0,493
BraTS-GLI-01231_Private-Collection_1	ET	60	34	265	1705	0,286	0,185	0,638	0,980	0,167
BraTS-GLI-01231_Private-Collection_1	TC	61	40	396	1567	0,219	0,134	0,604	0,975	0,123
BraTS-GLI-01231_Private-Collection_1	WT	1788	166	110	0	0,928	0,942	0,915	0,000	0,866
BraTS-GLI-01386_Private-Collection_1	ET	330	9	190	2174	0,768	0,635	0,974	0,996	0,624
BraTS-GLI-01386_Private-Collection_1	TC	375	2	462	1864	0,618	0,448	0,995	0,999	0,447
BraTS-GLI-01386_Private-Collection_1	WT	2165	28	510	0	0,890	0,809	0,987	0,000	0,801
BraTS-GLI-01442_TCGA-GBM_6	ET	519	21	295	2363	0,767	0,638	0,961	0,991	0,622
BraTS-GLI-01442_TCGA-GBM_6	TC	2051	16	351	780	0,918	0,854	0,992	0,980	0,848
BraTS-GLI-01442_TCGA-GBM_6	WT	2452	384	362	0	0,868	0,871	0,865	0,000	0,767
BraTS-GLI-01524_TCGA-LGG_14	ET	0	55	0	8709	0,000	0,000	0,000	0,994	0,000
BraTS-GLI-01524_TCGA-LGG_14	TC	1232	55	1379	6098	0,632	0,472	0,957	0,991	0,462

BraTS-GLI-01524_TCGA-LGG_14	WT	2967	5673	124	0	0,506	0,960	0,343	0,000	0,339

Table 16: Method 2 per-patient evaluation metrics and data: TP, FP, FN, TN per case and region.

patient	region	tp	fp	fn	tn	dice	recall	precision	specificity	iou
BraTS-GLI-00022-000_Private-Collection_18	ET	1157	112	507	1413	0,789	0,695	0,912	0,927	0,652
BraTS-GLI-00022-000_Private-Collection_18	TC	1169	100	771	1149	0,729	0,603	0,921	0,920	0,573
BraTS-GLI-00022-000_Private-Collection_18	WT	2639	462	88	0	0,906	0,968	0,851	0,000	0,828
BraTS-GLI-00053-000_Private-Collection_18	ET	0	241	0	919	0,000	0,000	0,000	0,792	0,000
BraTS-GLI-00053-000_Private-Collection_18	TC	0	256	0	904	0,000	0,000	0,000	0,779	0,000
BraTS-GLI-00053-000_Private-Collection_18	WT	391	684	85	0	0,504	0,821	0,364	0,000	0,337
BraTS-GLI-00233-000_Private-Collection_1	ET	278	63	35	2045	0,850	0,888	0,815	0,970	0,739
BraTS-GLI-00233-000_Private-Collection_1	TC	327	17	97	1980	0,852	0,771	0,951	0,992	0,742
BraTS-GLI-00233-000_Private-Collection_1	WT	2216	127	78	0	0,956	0,966	0,946	0,000	0,915
BraTS-GLI-00426-000_Private-Collection_1	ET	851	198	29	2874	0,882	0,967	0,811	0,936	0,789
BraTS-GLI-00426-000_Private-Collection_1	TC	1016	118	77	2741	0,912	0,930	0,896	0,959	0,839
BraTS-GLI-00426-000_Private-Collection_1	WT	3637	204	111	0	0,959	0,970	0,947	0,000	0,920
BraTS-GLI-00468-000_Private-Collection_18	ET	543	45	776	1932	0,570	0,412	0,924	0,977	0,398
BraTS-GLI-00468-000_Private-Collection_18	TC	581	7	1186	1522	0,493	0,329	0,988	0,995	0,328
BraTS-GLI-00468-000_Private-Collection_18	WT	1981	12	1303	0	0,751	0,603	0,994	0,000	0,601
BraTS-GLI-00481-000_Private-Collection_18	ET	0	36	0	154	0,000	0,000	0,000	0,811	0,000
BraTS-GLI-00481-000_Private-Collection_18	TC	0	36	0	154	0,000	0,000	0,000	0,811	0,000
BraTS-GLI-00481-000_Private-Collection_18	WT	2	38	150	0	0,021	0,013	0,050	0,000	0,011
BraTS-GLI-00530-000_Private-Collection_18	ET	0	1	92	771	0,000	0,000	0,000	0,999	0,000
BraTS-GLI-00530-000_Private-Collection_18	TC	0	1	232	631	0,000	0,000	0,000	0,998	0,000
BraTS-GLI-00530-000_Private-Collection_18	WT	167	603	94	0	0,324	0,640	0,217	0,000	0,193
BraTS-GLI-00533-000_Private-Collection_18	ET	597	279	70	1016	0,774	0,895	0,682	0,785	0,631
BraTS-GLI-00533-000_Private-Collection_18	TC	767	123	559	513	0,692	0,578	0,862	0,807	0,529
BraTS-GLI-00533-000_Private-Collection_18	WT	1751	115	96	0	0,943	0,948	0,938	0,000	0,893
BraTS-GLI-00569-000_IvyGAP_16	ET	440	916	94	3593	0,466	0,824	0,325	0,797	0,303
BraTS-GLI-00569-000_IvyGAP_16	TC	463	893	1406	2281	0,287	0,248	0,341	0,719	0,168
BraTS-GLI-00569-000_IvyGAP_16	WT	2777	1618	648	0	0,710	0,811	0,632	0,000	0,551
BraTS-GLI-00590-000_IvyGAP_16	ET	157	7	32	2231	0,890	0,831	0,957	0,997	0,801
BraTS-GLI-00590-000_IvyGAP_16	TC	157	40	32	2198	0,814	0,831	0,797	0,982	0,686
BraTS-GLI-00590-000_IvyGAP_16	WT	2113	178	136	0	0,931	0,940	0,922	0,000	0,871

BraTS-GLI-00608-000_Private-Collection_18	ET	147	9	603	1535	0,325	0,196	0,942	0,994	0,194
BraTS-GLI-00608-000_Private-Collection_18	TC	156	0	1801	337	0,148	0,080	1,000	1,000	0,080
BraTS-GLI-00608-000_Private-Collection_18	WT	1899	26	369	0	0,906	0,837	0,987	0,000	0,828
BraTS-GLI-00676-001_Private-Collection_18	ET	392	32	128	1367	0,831	0,754	0,925	0,977	0,710
BraTS-GLI-00676-001_Private-Collection_18	TC	418	7	623	871	0,570	0,402	0,984	0,992	0,399
BraTS-GLI-00676-001_Private-Collection_18	WT	1353	391	175	0	0,827	0,886	0,776	0,000	0,705
BraTS-GLI-00688-000_Private-Collection_18	ET	78	20	60	853	0,661	0,565	0,796	0,977	0,494
BraTS-GLI-00688-000_Private-Collection_18	TC	195	9	177	630	0,677	0,524	0,956	0,986	0,512
BraTS-GLI-00688-000_Private-Collection_18	WT	380	230	401	0	0,546	0,487	0,623	0,000	0,376
BraTS-GLI-01135-000_Private-Collection_1	ET	351	66	145	994	0,769	0,708	0,842	0,938	0,625
BraTS-GLI-01135-000_Private-Collection_1	TC	356	87	664	449	0,487	0,349	0,804	0,838	0,322
BraTS-GLI-01135-000_Private-Collection_1	WT	1150	111	295	0	0,850	0,796	0,912	0,000	0,739
BraTS-GLI-01167-000_Private-Collection_4	ET	148	895	163	1863	0,219	0,476	0,142	0,676	0,123
BraTS-GLI-01167-000_Private-Collection_4	TC	247	850	1090	882	0,203	0,185	0,225	0,509	0,113
BraTS-GLI-01167-000_Private-Collection_4	WT	2226	517	326	0	0,841	0,872	0,812	0,000	0,725
BraTS-GLI-01231-000_Private-Collection_1	ET	156	61	188	1692	0,556	0,454	0,719	0,965	0,385
BraTS-GLI-01231-000_Private-Collection_1	TC	159	118	319	1501	0,421	0,333	0,574	0,927	0,267
BraTS-GLI-01231-000_Private-Collection_1	WT	1779	234	84	0	0,918	0,955	0,884	0,000	0,848
BraTS-GLI-01341-000_Private-Collection_1	ET	158	31	62	205	0,773	0,718	0,836	0,869	0,630
BraTS-GLI-01341-000_Private-Collection_1	TC	188	1	94	173	0,798	0,667	0,995	0,994	0,664
BraTS-GLI-01341-000_Private-Collection_1	WT	233	174	49	0	0,676	0,826	0,573	0,000	0,511
BraTS-GLI-01386-000_Private-Collection_1	ET	593	168	41	2853	0,850	0,935	0,779	0,944	0,739
BraTS-GLI-01386-000_Private-Collection_1	TC	675	87	414	2479	0,729	0,620	0,886	0,966	0,574
BraTS-GLI-01386-000_Private-Collection_1	WT	3336	204	115	0	0,954	0,967	0,942	0,000	0,913
BraTS-GLI-01442-000_TCGA-GBM_6	ET	808	245	9	1959	0,864	0,989	0,767	0,889	0,761
BraTS-GLI-01442-000_TCGA-GBM_6	TC	2381	120	24	496	0,971	0,990	0,952	0,805	0,943
BraTS-GLI-01442-000_TCGA-GBM_6	WT	2773	204	44	0	0,957	0,984	0,932	0,000	0,918
BraTS-GLI-01524-000_TCGA-LGG_14	ET	0	158	0	3117	0,000	0,000	0,000	0,952	0,000
BraTS-GLI-01524-000_TCGA-LGG_14	TC	1811	157	771	536	0,796	0,701	0,920	0,773	0,661
BraTS-GLI-01524-000_TCGA-LGG_14	WT	2936	247	92	0	0,945	0,970	0,922	0,000	0,897

Table 17: Method 3 per-patient evaluation metrics and data: TP, FP, FN, TN per case and region.

patient	region	tp	fp	fn	tn	dice	recall	precision	specificity	iou

BraTS-GLI-00022-000_Private-Collection_18	ET	1111	49	553	1663	0,787	0,668	0,958	0,971	0,649
BraTS-GLI-00022-000_Private-Collection_18	TC	1548	47	392	1389	0,876	0,798	0,971	0,967	0,779
BraTS-GLI-00022-000_Private-Collection_18	WT	2712	649	15	0	0,891	0,995	0,807	0,000	0,803
BraTS-GLI-00053-000_Private-Collection_18	ET	266	24	138	2358	0,767	0,658	0,917	0,990	0,622
BraTS-GLI-00053-000_Private-Collection_18	TC	408	13	339	2026	0,699	0,546	0,969	0,994	0,537
BraTS-GLI-00053-000_Private-Collection_18	WT	2200	208	378	0	0,883	0,853	0,914	0,000	0,790
BraTS-GLI-00233-000_Private-Collection_1	ET	274	18	39	2129	0,906	0,875	0,938	0,992	0,828
BraTS-GLI-00233-000_Private-Collection_1	TC	401	77	23	1959	0,889	0,946	0,839	0,962	0,800
BraTS-GLI-00233-000_Private-Collection_1	WT	2253	166	41	0	0,956	0,982	0,931	0,000	0,916
BraTS-GLI-00426-000_Private-Collection_1	ET	837	105	43	3712	0,919	0,951	0,889	0,973	0,850
BraTS-GLI-00426-000_Private-Collection_1	TC	1074	986	19	2618	0,681	0,983	0,521	0,726	0,517
BraTS-GLI-00426-000_Private-Collection_1	WT	4331	277	89	0	0,960	0,980	0,940	0,000	0,922
BraTS-GLI-00468-000_Private-Collection_18	ET	557	267	888	2435	0,491	0,386	0,676	0,901	0,325
BraTS-GLI-00468-000_Private-Collection_18	TC	852	405	1453	1437	0,478	0,370	0,678	0,780	0,314
BraTS-GLI-00468-000_Private-Collection_18	WT	3526	31	590	0	0,919	0,857	0,991	0,000	0,850
BraTS-GLI-00481-000_Private-Collection_18	ET	642	213	139	2430	0,785	0,822	0,751	0,919	0,646
BraTS-GLI-00481-000_Private-Collection_18	TC	961	231	118	2114	0,846	0,891	0,806	0,902	0,734
BraTS-GLI-00481-000_Private-Collection_18	WT	2651	703	70	0	0,873	0,974	0,790	0,000	0,774
BraTS-GLI-00530-000_Private-Collection_18	ET	1	1	154	381	0,013	0,007	0,500	0,997	0,006
BraTS-GLI-00530-000_Private-Collection_18	TC	1	1	352	183	0,006	0,003	0,500	0,995	0,003
BraTS-GLI-00530-000_Private-Collection_18	WT	334	137	66	0	0,767	0,835	0,709	0,000	0,622
BraTS-GLI-00533-000_Private-Collection_18	ET	532	218	66	1031	0,789	0,890	0,709	0,826	0,652
BraTS-GLI-00533-000_Private-Collection_18	TC	1032	126	179	510	0,871	0,852	0,891	0,802	0,772
BraTS-GLI-00533-000_Private-Collection_18	WT	1578	179	90	0	0,922	0,946	0,898	0,000	0,854
BraTS-GLI-00569-000_IvyGAP_16	ET	570	2427	136	6445	0,308	0,807	0,190	0,726	0,182
BraTS-GLI-00569-000_IvyGAP_16	TC	1467	4277	707	3127	0,371	0,675	0,255	0,422	0,227
BraTS-GLI-00569-000_IvyGAP_16	WT	3542	5606	430	0	0,540	0,892	0,387	0,000	0,370
BraTS-GLI-00590-000_IvyGAP_16	ET	416	31	119	2166	0,847	0,778	0,931	0,986	0,735
BraTS-GLI-00590-000_IvyGAP_16	TC	560	54	145	1973	0,849	0,794	0,912	0,973	0,738
BraTS-GLI-00590-000_IvyGAP_16	WT	2363	217	152	0	0,928	0,940	0,916	0,000	0,865
BraTS-GLI-00608-000_Private-Collection_18	ET	240	14	788	2122	0,374	0,234	0,945	0,993	0,230
BraTS-GLI-00608-000_Private-Collection_18	TC	304	0	2460	400	0,198	0,110	1,000	1,000	0,110
BraTS-GLI-00608-000_Private-Collection_18	WT	2961	46	157	0	0,967	0,950	0,985	0,000	0,936
BraTS-GLI-00676-001_Private-Collection_18	ET	426	25	94	1133	0,877	0,819	0,945	0,978	0,782

BraTS-GLI-00676-001_Private-Collection_18	TC	718	11	323	626	0,811	0,690	0,985	0,983	0,683
BraTS-GLI-00676-001_Private-Collection_18	WT	1548	79	51	0	0,960	0,968	0,951	0,000	0,923
BraTS-GLI-00688-000_Private-Collection_18	ET	62	2	131	1503	0,483	0,321	0,969	0,999	0,318
BraTS-GLI-00688-000_Private-Collection_18	TC	262	2	190	1244	0,732	0,580	0,992	0,998	0,577
BraTS-GLI-00688-000_Private-Collection_18	WT	540	690	468	0	0,483	0,536	0,439	0,000	0,318
BraTS-GLI-01135-000_Private-Collection_1	ET	734	74	148	2756	0,869	0,832	0,908	0,974	0,768
BraTS-GLI-01135-000_Private-Collection_1	TC	1570	69	842	1231	0,775	0,651	0,958	0,947	0,633
BraTS-GLI-01135-000_Private-Collection_1	WT	3228	107	377	0	0,930	0,895	0,968	0,000	0,870
BraTS-GLI-01167-000_Private-Collection_4	ET	170	2151	141	2839	0,129	0,547	0,073	0,569	0,069
BraTS-GLI-01167-000_Private-Collection_4	TC	1191	2636	146	1328	0,461	0,891	0,311	0,335	0,300
BraTS-GLI-01167-000_Private-Collection_4	WT	2472	2749	80	0	0,636	0,969	0,474	0,000	0,466
BraTS-GLI-01231-000_Private-Collection_1	ET	135	6	190	1775	0,579	0,415	0,957	0,997	0,408
BraTS-GLI-01231-000_Private-Collection_1	TC	226	37	231	1612	0,628	0,495	0,859	0,978	0,458
BraTS-GLI-01231-000_Private-Collection_1	WT	1848	208	50	0	0,935	0,974	0,899	0,000	0,878
BraTS-GLI-01341-000_Private-Collection_1	ET	181	32	61	350	0,796	0,748	0,850	0,916	0,661
BraTS-GLI-01341-000_Private-Collection_1	TC	211	2	93	318	0,816	0,694	0,991	0,994	0,690
BraTS-GLI-01341-000_Private-Collection_1	WT	251	320	53	0	0,574	0,826	0,440	0,000	0,402
BraTS-GLI-01386-000_Private-Collection_1	ET	580	93	54	3022	0,888	0,915	0,862	0,970	0,798
BraTS-GLI-01386-000_Private-Collection_1	TC	1070	402	19	2258	0,836	0,983	0,727	0,849	0,718
BraTS-GLI-01386-000_Private-Collection_1	WT	3402	298	49	0	0,952	0,986	0,920	0,000	0,907
BraTS-GLI-01442-000_TCGA-GBM_6	ET	759	78	58	2354	0,918	0,929	0,907	0,968	0,848
BraTS-GLI-01442-000_TCGA-GBM_6	TC	2371	70	34	774	0,979	0,986	0,971	0,917	0,958
BraTS-GLI-01442-000_TCGA-GBM_6	WT	2776	432	41	0	0,922	0,985	0,865	0,000	0,854
BraTS-GLI-01524-000_TCGA-LGG_14	ET	0	235	0	3653	0,000	0,000	0,000	0,940	0,000
BraTS-GLI-01524-000_TCGA-LGG_14	TC	2203	374	466	845	0,840	0,825	0,855	0,693	0,724
BraTS-GLI-01524-000_TCGA-LGG_14	WT	3048	774	66	0	0,879	0,979	0,798	0,000	0,784