Università degli Studi di Modena e Reggio Emilia

Dipartimento di Ingegneria Enzo Ferrari

Master's Degree in Artificial Intelligence Engineering

Fast and Sliceous: Synthesis of Missing Brain MRI Modalities for Improved Tumor Segmentation

---

Author
Omar Carpentiero

Supervisor
Prof. Federico Bolelli

Co-supervisor
Dott. Kevin Marchesini

Academic Year 2024-2025

Abstract

Fast and Sliceous: Synthesis of Missing Brain MRI Modalities for Improved Tumor Segmentation

The synthesis of missing MRI modalities has emerged as a critical strategy to address incomplete multi-parametric imaging in brain tumor diagnosis and treatment planning. Recent advances in generative models, particularly GANs and diffusion-based approaches, have shown promising results in cross-modality MRI generation, although challenges persist in preserving anatomical fidelity and minimizing synthesis artifacts. Building on the Hybrid Fusion GAN (HF-GAN) framework, several enhancements are introduced to improve synthesis quality and generalization across tumor types. These include the application of z-score normalization, optimization of network components for faster and more stable training, and the extension of the pipeline to support multi-view generation across diverse brain tumor categories such as gliomas, metastases, and meningiomas. The approach emphasizes refinement of 2D slice-based generation to ensure intra-slice coherence and reduce intensity inconsistencies, ultimately facilitating more accurate and robust tumor segmentation in scenarios with missing imaging modalities.

Keywords: Image Synthesis, MRI, Multimodal, BraTS, Brain Tumor Imaging, GANs, Medical Imaging.

QR codes linking to the public source code and to the repository containing all experiments:

To my father,

who first sparked my passion
for learning and discovery.

Acknowledgements

I thank **Prof. Costantino Grana**, **Prof. Federico Bolelli** and **Dr. Kevin Marchesini** for guiding and supporting me throughout my internship and the preparation of this thesis. Their support has been fundamental at every step, and I am immensely grateful to them for everything they have done for me.

I thank my family for always being there throughout this journey, especially my grandmothers ”Nonna Pina” and ”Ma” and my mother ”Emily”.

I thank Katerina for always being by my side, patiently listening to my **endless monologues** in which I detailed **very specific aspects** of the training and the architecture, **expecting her to follow along and understand everything I was saying.** She tolerated not only the technical digressions, but also my **repeated attempts** to re-explain the **same concept** from multiple angles, as though she were preparing to submit her own paper on the subject. Unfortunately, despite the **very in-depth knowledge** she acquired over time, *no such paper has yet appeared*, probably because the challenge itself sounded far too intimidating to approach directly.

I thank my uni-best-friend Mr. Ago D. (Ago is canonically his name) for being, without a doubt, the most *unbearably insufferable* person in the entire world, constantly lamenting every little thing and always being the ”mountain guy”, as if the weight of the peaks personally rested on his shoulders.

I also thank Tobia for jokingly telling me what not to do with my career, advice that I openly ignored, doing **exactly** the opposite of what he suggested.

I thank all the Mind the Brain team for making this possible and for reigniting my passion for research. Their support has been crucial in shaping me into the person I am today.

I thank all my colleagues at AImageLab for letting me work these past months in such a healthy and supportive community.

I thank my fellow swimmers, simply because they swim—a lot.

Finally, I thank my friends for their support and company.

Contents

Abstract in English iii

Dedication iv

Acknowledgements v

Contents vi

List of Figures ix

List of Tables xi

1 Introduction 1
- 1.1 Motivations . . . 1
- 1.2 Objectives . . . 2
- 1.3 Contents Outline . . . 2

2 Context Overview 5
- 2.1 MRI . . . 5
- 2.2 Brain Tumors and Their Imaging Characteristics . . . 6
- 2.3 MRI Image Representation . . . 9
- 2.4 The NIFTI format . . . 10
- 2.5 U-Net . . . 14
- 2.6 Automatic Brain Tumor Segmentation . . . 14
- 2.7 Evaluation Metrics . . . 17
- 2.8 Additional Loss Metrics . . . 18
- 2.9 Multimodal Learning in Artificial Intelligence . . . 19
- 2.10 Generative Adversarial Networks (GANs) . . . 20

3 BraTS 23
- 3.1 Challenge Overview . . . 23
- 3.2 MICCAI and the BraTS Platform . . . 24
- 3.3 Evolution of BraTS Tasks . . . 24

3.4 BraTS 2025 Tasks . . . 25
3.5 The BraTS Python Package . . . 25
3.6 BraTS 2025 Key Dates . . . 26
3.7 Submission Protocol . . . 27
3.8 Task 8: Brasyn . . . 28
3.9 Last Year's Winners: HF-GAN . . . 28

4 Method **35**
4.1 Summary . . . 35
4.2 Preprocessing . . . 35
4.3 The Models . . . 38
4.3.1 The Generator . . . 39
4.3.2 The Discriminator . . . 45
4.3.3 The Segmenter . . . 46
4.4 The Loss Function . . . 47
4.5 The Training . . . 49
4.6 The Stacking Pipeline . . . 52

5 Experiments and Results **55**
5.1 Runs . . . 56
5.2 Discussion . . . 60

6 Conclusions and Future Work **63**
6.1 Conclusions . . . 63
6.2 Future Work . . . 64

A The Flaws of SSIM on Background Dominated Images **65**
A.1 Introduction . . . 65
A.2 SSIM in medical imaging . . . 66
A.3 Consequences . . . 67
A.4 A Simple Solution . . . 67

B Challenge Paper Publication **69**

Bibliography **83**

List of Figures

2.1 Representative axial slices from four MRI modalities 7
2.2 3D renderings of a glioma, a metastasis and meningioma 8
2.3 Axial 2D scans of a glioma, a metastasis, and a meningioma 8
2.4 3D render of a T1c volume from the glioma training set 10
2.5 Default U-Net architecture 15
2.6 Simple visual representation of the GAN framework in the context of natural images 21

3.1 A visual representation of the original first stage implementation. 30
3.2 Visual representation of the Refiner network 31
3.3 Dataset structure 33

4.1 The four histograms representing data distribution 36
4.2 Axial slice under different clamping strategies 37
4.3 Architecture of the generator 40
4.4 Architecture of a downsampling stage. 41
4.5 Architecture of one of the five encoders 42
4.6 Architecture of the modality infuser module 44
4.7 Architecture of an upsampling stage. 45
4.8 Architecture of the decoder 45
4.9 Architecture of one of the five encoders 46
4.10 Simple representation of the segmenter. 46
4.11 The full 3d generation pipeline 53

5.1 Comparisons of real and reconstructed images 61

List of Tables

2.1 NIfTI header structure (first 348 bytes). 13

4.1 Max values in relation with different percentiles 38
4.2 MRI intensity values of the training set 38
4.3 2D dataset sample counts 39

5.1 Experimental results obtained combining different losses 56
5.2 Characterization of settings employed in different experiments 57

Chapter 1

Introduction

This thesis will describe an automatic approach for generating a missing modality from the three available ones in the context of brain MRI. The approach will include the development of a model, its training, evaluation, and participation in Task 8 of the BraTS [1] challenge, an international benchmark organized annually in conjunction with MICCAI to evaluate algorithms for brain tumor analysis, followed by the final results. Moreover, this work led to the submission of a paper to the BraTS MICCAI challenge, and we will be cited as co-authors in this year's official BraTS publication. Starting from the available dataset, we developed and submitted a complete solution to the challenge, consolidating the entire workflow from data preprocessing to model deployment.

1.1 Motivations

In the field of medical imaging, automatic brain tumor segmentation from brain MRI has long been a critical research area, attracting significant attention due to its potential to enhance clinical workflows and improve patient outcomes. Automatic segmentation can drastically increase both the accuracy and speed of tumor diagnosis, reducing the dependency on time-consuming manual annotations by radiologists and minimizing interobserver variability.

However, the most effective segmentation systems typically require all four MRI modalities—T1, T1c, T2, and FLAIR—to achieve optimal performance. In practice, these modalities are not always available. Missing modalities can arise from acquisition errors, incomplete

scans, or the infeasibility of performing certain sequences, such as T1c, which requires administering a contrast agent to the patient and scheduling a second MRI session close in time to the initial scans. These limitations make it impossible to directly apply the best state-of-the-art algorithms, which are designed to work with all four modalities.

To address this, the scientific community [2–5] has explored models capable of segmenting tumors using an arbitrary subset of available modalities. While effective, these methods often require designing and training specialized models.

An alternative approach, which is the focus of this thesis, involves leveraging generative models to synthesize the missing modalities. By generating the absent data, it becomes possible to directly apply high-performing, state-of-the-art [6] segmentation models that expect all four modalities, without the need for retraining or modifying them. This strategy not only facilitates the reuse of established models but also has the potential to recover performance levels comparable to those achieved with complete data. Despite being less frequently explored in the literature, generative approaches promise a robust solution to the missing-modality problem and open new avenues for improving clinical applicability in scenarios with incomplete MRI datasets.

1.2 Objectives

The objective of this thesis is to provide a detailed description of all the steps undertaken, the experiments conducted, the considerations made, and the results obtained that led to the development of a generative model capable of synthesizing missing brain MRI scans. The thesis will provide all the necessary background and tools for a proper understanding of the problem, as well as implementation details that illustrate how the model works and the optimizations that made it significantly more efficient in its final versions.

1.3 Contents Outline

A summary of what will be addressed in the following Chapters is now presented.

- Chapter 2 introduces the medical and artificial intelligence background required to contextualize the scope of this thesis. It discusses the fundamentals of magnetic resonance imaging (MRI), including its clinical significance, acquisition protocols, and image formats. Furthermore, it provides an overview of automatic tumor segmentation, with particular attention to the U-Net architecture.
- Chapter 3 presents an overview of the BraTS challenge, outlining its objectives, the specific task addressed in this work, and the approaches adopted by last year's winning teams. It further details the dataset employed and the evaluation metrics used.
- Chapter 4 describes the proposed methodology, including the training pipeline, the network architectures, and the adopted loss functions.
- Chapter 5 reports the main experiments, presenting the results obtained.
- Chapter 6 presents the conclusions and possible directions for future work.

Chapter 2

Context Overview

This Chapter provides an overview of the scope of this work, explaining the medical and technical topics related to the problem addressed.

2.1 MRI

Magnetic Resonance Imaging (MRI) is a non-invasive imaging technique that exploits the magnetic properties of hydrogen nuclei in the body to produce high-resolution images of internal structures. During an MRI scan, the patient is placed in a strong magnetic field, which aligns the nuclear spins of hydrogen protons. Radiofrequency pulses are then applied to perturb this alignment, and the resulting signals emitted by the relaxing protons are measured by the scanner. These signals are processed to generate detailed images that reflect differences in tissue properties, such as proton density, relaxation times, and molecular environment. By adjusting the sequence parameters, MRI can produce a variety of contrasts that highlight different anatomical and pathological features. In brain tumor imaging, the most commonly used sequences include T1, T1 with contrast enhancement (T1c), T2, and FLAIR. Each modality provides complementary information: T1 images offer good anatomical detail, T1c highlights areas with blood-brain barrier disruption, T2 emphasizes fluid-rich regions such as edema, and FLAIR suppresses cerebrospinal fluid signals to enhance the visibility of lesions. The combination of these sequences is critical for accurate tumor visualization and subsequent segmentation. Among the MRI modalities, T2 images are often considered

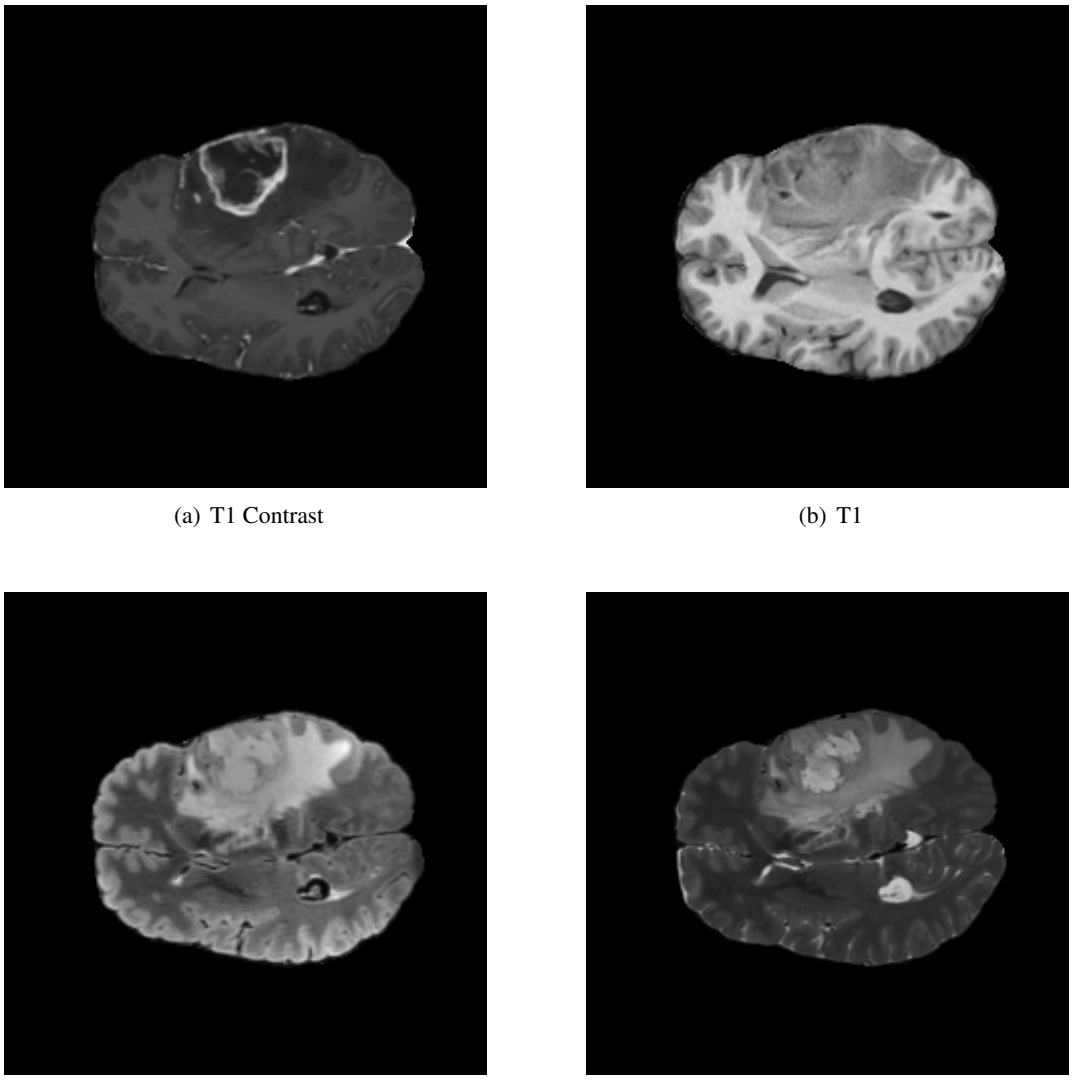
the least informative for tumor segmentation. They are sometimes acquired using fast, low-resolution scans and subsequently upsampled via interpolation. While it effectively highlights fluid-rich areas, it provides limited contrast between tumor tissue and surrounding structures, as many regions—including edema and normal anatomy—appear similarly bright. A complete brain MRI acquisition protocol typically produces four 3D volumes per subject, corresponding to the T1, T1c, T2, and FLAIR sequences. These four modalities together provide complementary anatomical and pathological information, forming the foundation for accurate tumor visualization. A visual example of a complete acquisition protocol can be seen in Figure 4.1

2.2 Brain Tumors and Their Imaging Characteristics

Brain tumors are abnormal collections of cells within the brain, and may be categorized as primary (originating in the brain) or secondary (metastatic, spreading from another organ). Both types are harmful and potentially deadly. Metastatic tumors are often characterized by the presence of multiple distinct tumor bodies, whereas primary tumors, such as gliomas and meningiomas, usually arise as a single mass, as shown in Figure 2.2 as a 3D rendering and in Figure 2.3 to highlight their structure in the context of a complete 2D scan.

Gliomas are primary tumors arising from glial cells within the brain parenchyma, the functional tissue of the brain. Being intra-axial, they grow inside the brain tissue and often infiltrate the surrounding areas. On imaging, gliomas typically exhibit irregular shapes, heterogeneous signal intensities, and variable contrast enhancement, reflecting regions of necrosis, edema, or hemorrhage. High-grade gliomas, such as glioblastomas, are particularly aggressive and associated with poor clinical outcomes.

Metastatic brain tumors arise from cancers outside the central nervous system and spread hematogenously to the brain. These lesions are typically intra-axial, lodging within the brain parenchyma without the diffuse infiltration seen in gliomas. They frequently present as multiple, well-defined lesions at the gray–white matter junction and are surrounded by disproportionate vasogenic edema. Imaging features often correlate with the advanced stage of the primary systemic cancer, and prognosis depends primarily on the characteristics of the primary tumor.

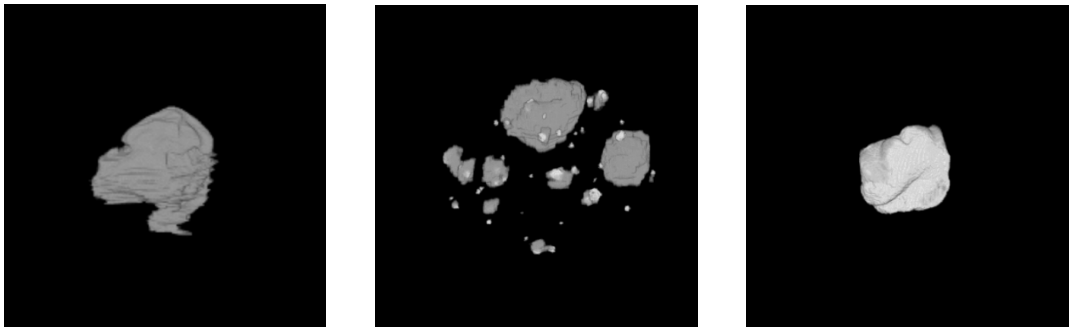


(a) T1 Contrast (b) T1

(c) FLAIR (d) T2

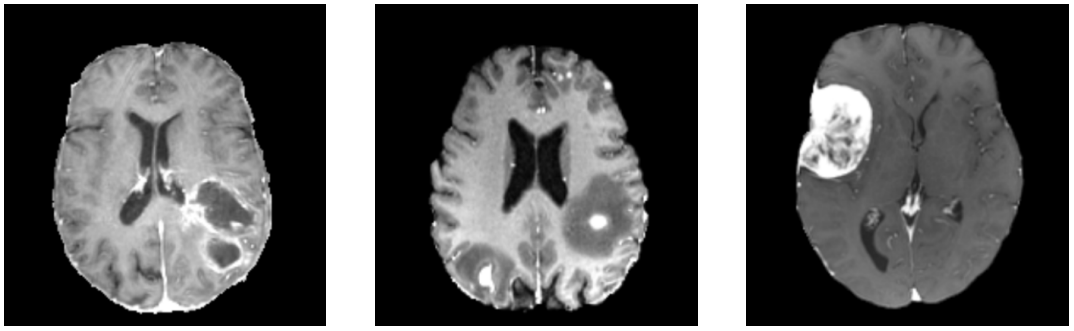
FIGURE 2.1: Representative axial slices from four MRI modalities: T1 with contrast (T1 Contrast), standard T1 (T1), FLAIR, and T2 images. Contrast differences result from projecting the unbounded tensor values to the 0–255 range. The slices are taken from subject GLI-00492 from the BraTS Dataset

Meningiomas are extra-axial, developing outside the brain parenchyma and attached to the dura, the tough outer membrane that covers the brain. They generally appear as well-circumscribed, homogeneous masses. While most meningiomas are benign and slow-growing, they can still produce significant clinical effects through mass effect or recurrence. Unlike gliomas, they compress the brain rather than infiltrating it.



(a) Glioma tumor (b) Metastatic tumor (c) Meningioma tumor

FIGURE 2.2: 3D renderings of a glioma, a metastasis and meningioma, segmentation mask from their respective training sets. The images highlight the distinct appearances of the three tumor types, with gliomas typically forming a single infiltrative mass, metastases appearing as multiple separate lesions, and meningiomas presenting well-defined boundaries.



(a) Glioma tumor (b) Metastatic tumor (c) Meningioma tumor

FIGURE 2.3: Axial 2D scans of a glioma, a metastasis, and a meningioma from their respective training sets. The images highlight the distinct appearances of the three tumor types, with gliomas showing an irregular infiltrative mass that blends with surrounding tissue, metastases appearing as multiple well-circumscribed intra-axial lesions often surrounded by edema, and meningiomas presenting as extra-axial tumors with sharp, well-defined margins compressing the adjacent brain.

On MRI, brain tumors present differently depending on the sequence and the pathological subregion. The tumor core often shows contrast enhancement on T1 with contrast (T1c), indicating breakdown of the blood–brain barrier. Surrounding edema and infiltrative zones are typically hyperintense on T2 or FLAIR. Necrotic or cystic regions may appear hypointense on T1 (and hyperintense on T2). Because of this heterogeneity, segmenting tumors is challenging: different subregions exhibit varying signal intensities across modalities, and boundaries with healthy tissue are sometimes ambiguous. An example can be seen in Figure 4.1.

In this study, a conventional three-class decomposition of tumor tissue is employed:

- **Necrotic / non-enhancing tumor core.** The dead or non-enhancing portion (often hypointense in T1c).
- **Enhancing tumor.** The actively growing, contrast-enhancing portion seen in T1c;
- **Peritumoral edema.** The swollen or infiltrated region around the core (visible especially in FLAIR/T2).

These three classes are also aggregated into composite labels:

- **Enhancing tumor (ET)**. The enhancing tumor class alone.
- **Tumor core (TC)**. The union of enhancing + necrotic/non-enhancing core.
- **Whole tumor (WT)**. The union of all three classes (enhancing + necrotic core + edema).

2.3 MRI Image Representation

An MRI scan can be considered a discrete representation of a continuous spatial distribution of tissue properties. Given a three-dimensional position vector x, the function f(x) represents the MRI signal intensity at that position. It is not possible to store or process MRI data as a continuous function, so it must be discretized into a three-dimensional matrix of $w \times d \times h$ elements called voxels.

Discretization involves two main steps. The first is sampling, in which a value is assigned to each voxel based on the underlying continuous signal. The second is quantization, which restricts these values to a finite set of discrete levels determined by the scanner's bit depth. For example, a voxel with b bits can represent 2^b discrete intensity levels, often referred to as grey-levels.

MRI volumes are typically acquired as a stack of slices along one or more axes. The acquired slices are then automatically stacked, as shown in Figure 2.4 The slice distance is defined as the spacing between adjacent slices. If the in-plane voxel size (pixel spacing) is equal to the slice distance, the volume is said to be isotropic, meaning voxels have a cubic shape. Otherwise, the

volume is anisotropic, which can affect image analysis and processing, including segmentation and registration tasks. This argument can be extended to three-dimensional images, that are considered as three-dimensional matrices of $w \times d \times h$ elements called voxels. A volume can be sliced in all three dimensions in order to get images. The distance between two adjacent images is called *slice distance*. If the pixel distance is equal to the slice distance, then the volume is *isotropic*, and this means that the voxels have a perfectly cubic shape. Otherwise, the volume is *anisotropic*.

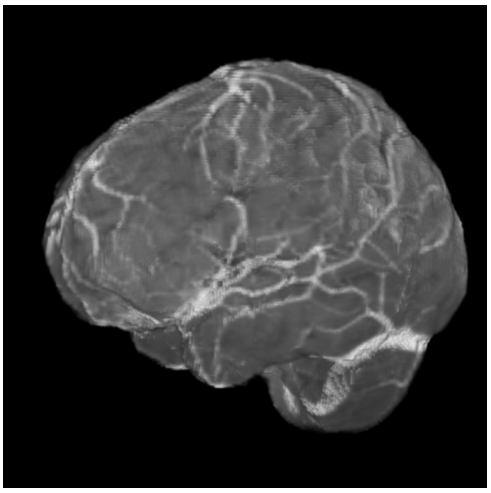


FIGURE 2.4: 3D render of a T1c volume from the glioma training set

2.4 The NIFTI format

The establishment of dedicated standards in medical imaging is essential to ensure consistency, reliability, and interoperability across devices, institutions, and studies. General-purpose image formats such as PNG or JPEG are unsuitable, as they cannot accommodate the multidimensional data, metadata, and quantitative information required in clinical and research settings. Dedicated standards enable accurate comparison of results, support the development and validation of computational tools, and ultimately improve both practice and research outcomes.

Medical Image Formats and Standards

MRI data are typically stored in standardized digital formats to ensure interoperability and ease of processing. One of the most widely used formats in neuroimaging research is NIfTI (Neuroimaging Informatics Technology Initiative). NIfTI files store volumetric MRI data as three-dimensional matrices of voxels, along with essential metadata describing the image acquisition and spatial orientation. This format has become a standard for computational pipelines, particularly in tasks such as brain tumor segmentation. Although the more general medical imaging format DICOM (Digital Imaging and Communications in Medicine) is commonly used in clinical settings, it is less convenient for computational analysis due to its complex hierarchical structure. Notably, the BraTS challenge does not provide DICOM data, relying exclusively on NIfTI-formatted volumes.

File Types: .nii and .nii.gz

NIfTI files can exist as uncompressed `.nii` files or compressed `.nii.gz` files, which reduce storage requirements without losing information. Each file contains both the image data and a header with metadata, making it self-contained and easily portable across software tools. The compression is lossless and fully compatible with most neuroimaging software.

Voxel Data and Discretization

A NIfTI image stores voxel intensities as discrete values representing the underlying MRI signal. Each voxel corresponds to a small volume element of the scanned tissue and has defined dimensions along each axis, known as the voxel dimensions. These dimensions are critical for volume measurements, resampling, and accurate spatial analysis. While voxel values can be stored as 16-bit or 32-bit integers or floating-point numbers in general, in the BraTS challenge the NIfTI images use `float64` internally, but the final voxel values are integers within the range of `uint32`.

Header information

The NIfTI header stores crucial details about the image, including:

- **Voxel dimensions.** The physical spacing between voxels along each axis, which ensures correct scaling in 3D visualization and analysis.
- **Data type and bit depth.** Determines how intensity values are represented, affecting precision and quantization.
- **Image orientation.** Specifies the spatial orientation of the volume, allowing consistent alignment across subjects and modalities.
- **Additional metadata.** Optional fields may include details about the scanning protocol, patient anonymized information, and acquisition parameters.

A complete representation of the header information is shown in Table 2.1.
Preserving header information is crucial when data are loaded, processed, and subsequently modified, either partially or fully. Any loss or corruption of the header can lead to incorrect voxel dimensions, orientation errors, or misalignment between modalities. This is particularly important in tasks where the output must remain aligned with the input, such as tumor segmentation or image generation, where spatial fidelity is essential for accurate analysis and model performance.

Advantages and Usage

The NIfTI format is widely supported by major neuroimaging libraries, including `Nibabel`, `ANTs`, and `FSL`. Compared to other formats, NIfTI provides a compact, self-contained representation of the entire 3D volume, simplifying loading, preprocessing, and visualization in computational pipelines. Its standardized structure facilitates reproducibility, dataset sharing, and integration with machine learning models for segmentation or analysis. The precise voxel size and orientation information ensures accurate registration, resampling, and analysis of multi-modal MRI datasets.

Type	Name	Offset	Size	Description
int	sizeof_hdr	0B	4B	Header size (must be 348)
char	data_type[10]	4B	10B	Not used; Analyze compatibility
char	db_name[18]	14B	18B	Not used; Analyze compatibility
int	extents	32B	4B	Not used; Analyze compatibility
short	session_error	36B	2B	Not used; Analyze compatibility
char	regular	38B	1B	Not used; Analyze compatibility
char	dim_info	39B	1B	Encoding directions (phase, frequency, slice)
short	dim[8]	40B	16B	Data array dimensions
float	intent_p1	56B	4B	1st intent parameter
float	intent_p2	60B	4B	2nd intent parameter
float	intent_p3	64B	4B	3rd intent parameter
short	intent_code	68B	2B	NIfTI intent
short	datatype	70B	2B	Data type
short	bitpix	72B	2B	Bits per voxel
short	slice_start	74B	2B	First slice index
float	pixdim[8]	76B	32B	Grid spacings (unit per dimension)
float	vox_offset	108B	4B	Offset into `.nii` file
float	scl_slope	112B	4B	Data scaling slope
float	scl_inter	116B	4B	Data scaling offset
short	slice_end	120B	2B	Last slice index
char	slice_code	122B	1B	Slice timing order
char	xyzt_units	123B	1B	Units of pixdim[1..4]
float	cal_max	124B	4B	Maximum display intensity
float	cal_min	128B	4B	Minimum display intensity
float	slice_duration	132B	4B	Time for one slice
float	toffset	136B	4B	Time axis shift
int	glmax	140B	4B	Not used; Analyze compatibility
int	glmin	144B	4B	Not used; Analyze compatibility
char	descrip[80]	148B	80B	Any text
char	aux_file[24]	228B	24B	Auxiliary filename
short	qform_code	252B	2B	Use quaternion fields
short	sform_code	254B	2B	Use affine fields
float	quatern_b	256B	4B	Quaternion b parameter
float	quatern_c	260B	4B	Quaternion c parameter
float	quatern_d	264B	4B	Quaternion d parameter
float	qoffset_x	268B	4B	Quaternion x shift
float	qoffset_y	272B	4B	Quaternion y shift
float	qoffset_z	276B	4B	Quaternion z shift
float	srow_x[4]	280B	16B	1st row affine transform
float	srow_y[4]	296B	16B	2nd row affine transform
float	srow_z[4]	312B	16B	3rd row affine transform
char	intent_name[16]	328B	16B	Name or meaning of the data
char	magic[4]	344B	4B	Magic string

TABLE 2.1: NIfTI header structure (first 348 bytes).

2.5 U-Net

The U-Net [7] is a convolutional neural network originally introduced for biomedical image segmentation, and it has since become a standard baseline for many medical imaging tasks, including BraTS. Its architecture is characterized by a symmetric encoder–decoder structure with skip connections, forming a characteristic "U" shape, as shown in Figure X. U-Net is particularly suitable in contexts where the output is a *pixel- or voxel-wise map of tissue or structure labels*, such as segmentation masks of tumor subregions in MRI volumes.

- **Encoder (contracting path).** The encoder progressively reduces the spatial resolution while increasing the number of feature channels. This is achieved through repeated applications of convolutional layers, nonlinear activations, and downsampling (e.g., max pooling). The encoder captures increasingly abstract and contextual features of the input MRI slices.
- **Bottleneck.** At the bottom of the U, the network learns highly compressed, semantic representations of the data, integrating both local and global context.
- **Decoder (expanding path).** The decoder gradually upsamples the feature maps to restore the original spatial resolution. Transposed convolutions (or upsampling followed by convolution) are used, and the decoder refines the segmentation prediction at increasingly finer scales.
- **Skip connections.** To recover spatial detail lost during downsampling, feature maps from the encoder are concatenated with corresponding layers in the decoder. This allows the network to combine semantic context from deep layers with fine-grained localization from shallow layers, which is particularly important in medical imaging where small structures (e.g., tumor subregions) must be precisely delineated.

2.6 Automatic Brain Tumor Segmentation

Automatic brain tumor segmentation is a critical task in medical image analysis, facilitating accurate diagnosis, treatment planning, and monitoring of tumor progression. Over the years,

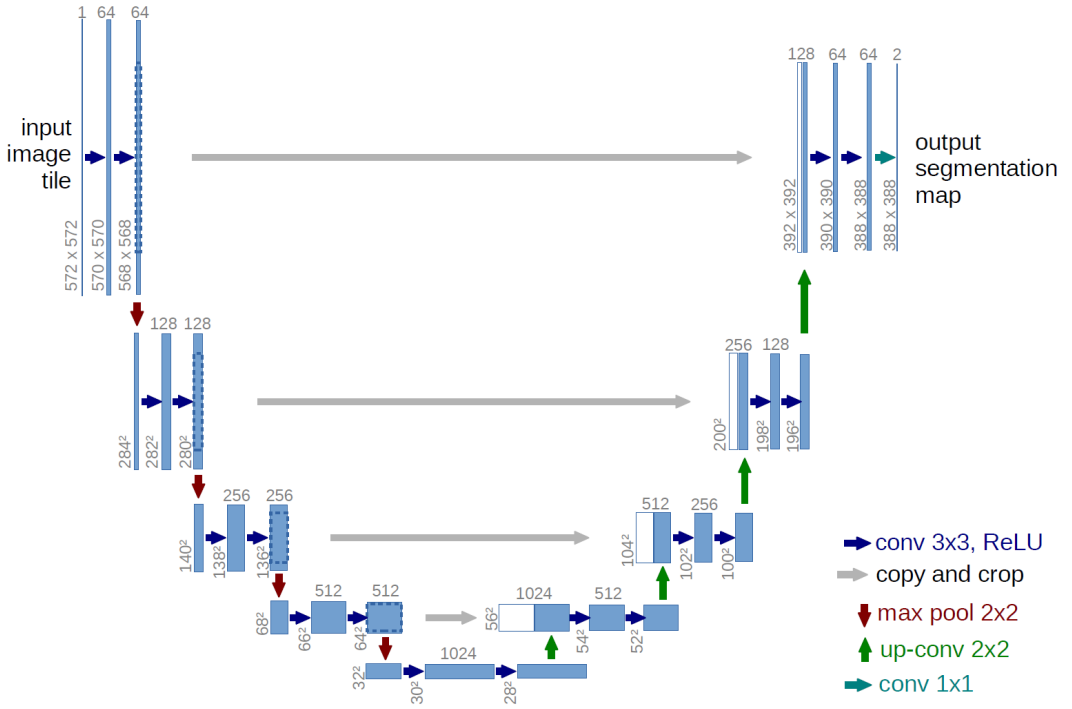


FIGURE 2.5: Default U-Net architecture. The figure shows the U-shaped encoder–decoder network with skip connections, which combine contextual information from downsampling with precise localization from upsampling.

segmentation techniques have evolved from traditional image processing methods to advanced deep learning approaches, significantly enhancing segmentation accuracy and efficiency.

Classical Segmentation Methods

Early approaches to brain tumor segmentation primarily relied on traditional image processing techniques, which include:

- **Thresholding.** This method involves segmenting the image based on intensity thresholds, distinguishing tumor regions from healthy tissue. However, it is sensitive to noise and variations in image intensity. Thresholding generally performs best after skull-stripping (deskulling), as hyperintense structures such as mucus in the nasal cavities can otherwise be incorrectly included in the segmentation of T1 images.

- **Region Growing.** Starting from seed points, this technique grows regions by adding neighboring pixels that have similar properties, such as intensity. It requires careful seed selection and is susceptible to over- or under-segmentation [8].
- **Clustering.** Algorithms like K-means and Gaussian Mixture Models classify pixels into clusters based on intensity or texture features. While they can handle variability in tissue types, they may struggle with complex tumor structures. [9]
- **Atlas-based Segmentation.** This method involves registering a pre-labeled anatomical atlas to the target image, transferring the segmentation labels. It is effective for standard anatomical structures but less reliable for heterogeneous tumors due to the significant variability in tumor shape, size, and intensity [10].

Despite their utility, these methods often fall short in accurately delineating complex and heterogeneous tumor regions, paving the way for more advanced techniques.

Deep Learning-Based Segmentation

The advent of deep learning has revolutionized brain tumor segmentation. Convolutional Neural Networks (CNNs), particularly U-Net and its variants, have demonstrated exceptional performance in segmenting brain tumors from multi-modal MRI images. These models automatically learn hierarchical features, capturing both local and global context, which is crucial for accurate segmentation. Recent advancements include 3D U-Net [11] architectures, which extend U-Net to three dimensions to process volumetric data and capture spatial dependencies across slices, with variants like nnU-Net introducing adaptive architectures tailored to specific datasets [12–15]. In addition, CNN-Transformer hybrid models, such as BiTr-Unet [16], combine CNNs for local feature extraction with Transformer blocks to capture long-range dependencies, improving contextual modeling and delineation of complex tumor boundaries. Residual and dense connection strategies, used in architectures like 3D Res-U-Net and Dense-U-Net, further enhance feature reuse and gradient flow, improving segmentation accuracy for small or heterogeneous tumor subregions.

2.7 Evaluation Metrics

A wide range of quantitative scores exists to evaluate medical image synthesis and segmentation, each emphasizing different aspects of image quality and task performance. However, for the purposes of this work, only a subset of metrics is required. In particular, reconstruction fidelity will be assessed with the Structural Similarity Index Measure (SSIM), while the evaluation of segmentation performance will rely on three complementary metrics: the Dice Similarity Coefficient (DICE), the Hausdorff Distance (HD), and the Normalized Surface Distance (NSD). The following subsections describe these metrics in detail and outline their relevance to the present study.

- **Structural Similarity Index Measure (SSIM).** Evaluates the perceptual similarity between the reconstructed image and the reference image, taking into account luminance, contrast, and structural information. SSIM values range from -1 to 1, with 1 indicating perfect similarity.

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

  where μ_x and μ_y are the mean intensities, σ_x^2 and σ_y^2 the variances, and σ_{xy} the covariance of images x and y. C_1 and C_2 are small constants to stabilize the division.

- **Dice Similarity Coefficient (DSC).** Measures the overlap between predicted and ground truth segmentations. A DSC of 1 indicates perfect overlap, while 0 indicates no overlap.

$$\text{DICE} = \frac{2\,|P \cap GT|}{|P| + |GT|}, \quad \text{where } P \text{ is the predicted segmentation and } GT \text{ is the ground truth.}$$

- **Hausdorff Distance (HD).** Quantifies the maximum distance between the predicted and ground truth boundaries. The 95th percentile of HD (HD95) is often used to mitigate the influence of outliers.

$$H(P, GT) = \max\left\{ \sup_{p \in P} \inf_{g \in GT} d(p, g),\ \sup_{g \in GT} \inf_{p \in P} d(g, p) \right\}$$

where A and B are sets of points (e.g., surfaces of the segmentations), and $d(a, b)$ is a distance metric (usually the Euclidean distance).

- **Normalized Surface Dice (NSD).** Evvaluates the similarity between predicted and ground truth surfaces within a predefined tolerance distance. Unlike the volumetric Dice coefficient, NSD emphasizes boundary accuracy, making it particularly relevant for small or irregular tumor structures. It is especially important for the enhancing tumor, the actively growing portion of the tumor that surrounds the necrotic core.

$$\text{NSD}(P, GT) = \frac{\left|\{p \in \partial P \mid d(p, \partial GT) \leq \tau\}\right| + \left|\{g \in \partial GT \mid d(g, \partial P) \leq \tau\}\right|}{|\partial P| + |\partial GT|}$$

where ∂P and ∂GT are the surfaces of the predicted and ground truth segmentations, $d(\cdot, \cdot)$ is a distance metric (usually the Euclidean distance), and τ is a tolerance threshold that allows small deviations between the surfaces to be considered acceptable, accounting for minor segmentation inaccuracies and discretization effects.

These metrics offer a comprehensive evaluation of segmentation performance, guiding model development and comparison.

2.8 Additional Loss Metrics

In addition to the evaluation metrics described in the previous section, certain loss functions are used during training to guide the model toward accurate reconstruction, realistic synthesis, and correct segmentation. These functions can also be interpreted as quantitative measures of discrepancy between predictions and targets, and are particularly relevant for model optimization. In this work, we focus on the Mean Absolute Error (MAE) for reconstruction, the Binary Cross-Entropy (BCE) loss for discriminator and classification tasks, and the Focal Tversky loss for segmentation.

- **Mean Absolute Error (MAE).** Quantifies the average absolute difference between the predicted image and the ground truth. MAE is widely used as a reconstruction loss because it directly penalizes pixel-wise deviations, encouraging the generator to

produce outputs that are numerically close to the target. Values closer to 0 indicate better reconstruction.

$$\text{MAE}(x, y) = \frac{1}{N} \sum_{i=1}^{N} |x_i - y_i|$$

where x and y are the predicted and reference images, respectively, and N is the total number of pixels.

- **Binary Cross-Entropy (BCE).** Measures the discrepancy between predicted probabilities and binary labels, and is used for both adversarial and classification tasks. BCE effectively evaluates how well the discriminator can distinguish real from generated samples, or how accurately a network predicts binary labels. Lower BCE values indicate better performance.

$$\text{BCE}(p, t) = -\frac{1}{N} \sum_{i=1}^{N} \left[t_i \log(p_i) + (1 - t_i) \log(1 - p_i) \right]$$

where p_i is the predicted probability for sample i, $t_i \in \{0, 1\}$ is the target label, and N is the number of samples in the batch.

- **Focal Tversky Loss.** Designed for highly imbalanced segmentation tasks, the Focal Tversky loss modifies the standard Tversky index by introducing a focusing parameter γ to emphasize hard-to-classify regions, such as small tumors. This loss helps the network prioritize difficult voxels while maintaining overall segmentation accuracy.

$$\text{FTL} = \sum_{i=0}^{C-1} (1 - \text{TI}_i)^{\gamma}, \quad \text{TI}_i = \frac{\text{TP}_i}{\text{TP}_i + \alpha \text{FP}_i + \beta \text{FN}_i}$$

where TI_i is the Tversky index for class i, $\text{TP}_i, \text{FP}_i, \text{FN}_i$ are true positives, false positives, and false negatives, α and β control the penalty for FP and FN, γ is the focusing parameter, and C is the number of segmentation classes.

2.9 Multimodal Learning in Artificial Intelligence

Multimodal learning in artificial intelligence (AI) focuses on integrating information from multiple heterogeneous sources in order to build richer and more robust representations.

In many real-world scenarios, data is inherently multimodal: for instance, vision can be complemented by language, or audio signals may provide context to visual scenes. By learning to combine modalities, AI systems are able to capture complementary cues, disambiguate noisy or incomplete signals, and improve overall performance in complex tasks. This integration is typically achieved through fusion strategies, which may occur at different levels of the model architecture, such as early fusion (input-level concatenation), late fusion (decision-level aggregation), or hybrid approaches involving attention mechanisms and cross-modal interactions.

The importance of multimodality is particularly evident in the medical imaging domain, where different acquisition techniques or contrast settings provide complementary perspectives on the same anatomical structures. Instead of relying on a single source of information, multimodal systems exploit the unique characteristics of each modality, enabling more accurate diagnoses and better-informed clinical decisions. Moreover, multimodal learning is well-suited to scenarios involving missing or corrupted data, as the system can infer the missing information from the available modalities.

In the context of this work, the problem is inherently multimodal, as a complete brain MRI acquisition consists of four distinct sequences (T1, T1ce, T2, and FLAIR), each contributing complementary information to the synthesis task.

2.10 Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs), first introduced by Goodfellow et al. [17], are a class of deep learning models designed to generate realistic synthetic data by learning the underlying distribution of a given dataset. GANs have gained tremendous attention due to their remarkable ability to produce high-quality images, videos, and other data modalities, often indistinguishable from real samples. This explanation is particularly important in the context of our work, since the paradigm of our model follows the same adversarial framework.

The fundamental architecture of a GAN consists of two neural networks—the generator G and the discriminator D—that are trained simultaneously in a minimax game. The generator takes as input a random noise vector z sampled from a known distribution (typically Gaussian or uniform) and produces a synthetic sample $G(z)$ intended to resemble real data. The

discriminator, on the other hand, receives either a real sample x or a generated sample $G(z)$ and predicts the probability that the input comes from the real data distribution rather than the generator. The networks are optimized using the following objective:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))].$$

Here, the generator seeks to minimize the probability that the discriminator correctly identifies fake samples, while the discriminator aims to maximize its ability to distinguish real from generated data (see Figure 2.6). Through this adversarial process, the generator progressively improves, producing samples increasingly similar to the real data distribution.

A key strength of GANs is their ability to learn complex, high-dimensional data distributions without requiring explicit density estimation. Unlike traditional generative models such as Variational Autoencoders (VAEs), which often produce blurred outputs due to probabilistic approximations, GANs are capable of generating sharp and realistic images by leveraging the adversarial training framework. Despite their success, GANs are notoriously difficult to train.

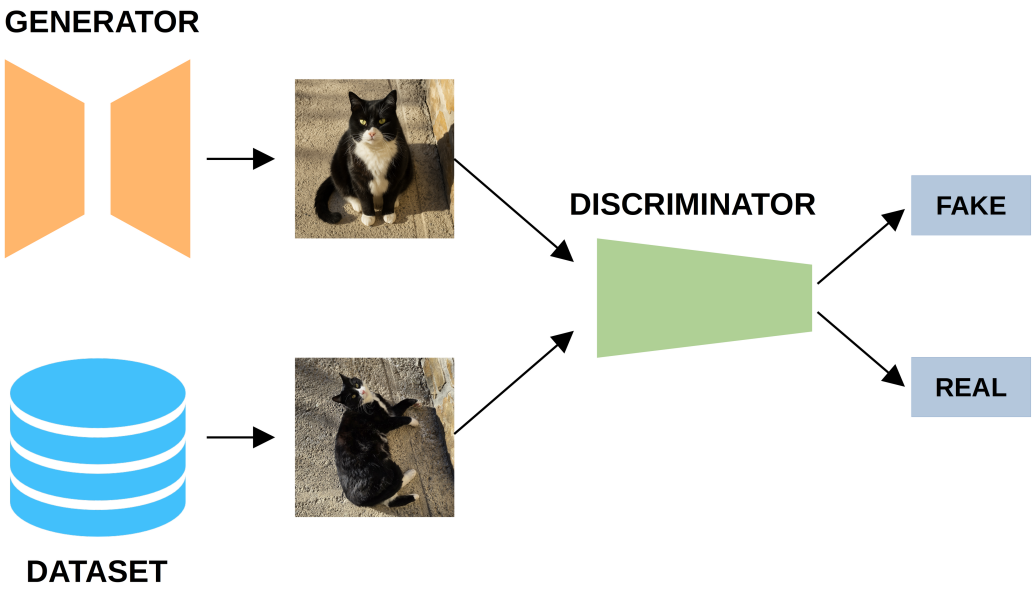


FIGURE 2.6: Simple visual representation of the GAN framework in the context of natural images[18]

Common challenges include mode collapse, where the generator produces limited variations

of outputs, and training instability, which arises from the delicate balance required between the generator and discriminator.

In recent years, GANs have been partially outperformed by diffusion models [19], which offer more stable training and often superior image quality. However, the scope of this thesis focuses on GANs, as there exist well-established GAN architectures and pre-trained models that provide a strong starting point for our work, enabling rapid development and experimentation.

In summary, GANs represent a powerful and flexible framework for generative modeling, offering unparalleled capabilities in creating realistic synthetic data. Their adversarial nature encourages continuous improvement of generated samples, making them particularly suitable for domains where high-fidelity data is crucial. Importantly, understanding this framework is essential for our work, as our model adopts the same adversarial paradigm for learning and generating realistic outputs.

Chapter 3

BraTS

This chapter provides an overview of the Brain Tumor Segmentation (BraTS) Challenge, outlining its objectives and available tasks, with particular attention to tumor segmentation, missing modality generation, the 2025 Lighthouse Challenge, and the new Python ecosystem for reproducible AI development.

3.1 Challenge Overview

Since its inception at MICCAI 2012, the Brain Tumor Segmentation (BraTS) challenge has advanced brain tumor image analysis by benchmarking algorithmic advances, providing high-quality annotated datasets, and tasking participants with developing innovative solutions to clinically relevant problems along the disease course and across different brain tumor entities.

In collaboration with leading clinical organizations such as AI-RANO, RSNA, ASNR, NIH, FDA, ASFNR, and CBTN, the BraTS 2025 Lighthouse Challenge expands this effort by addressing additional clinical needs, including longitudinal assessment of tumor response, generalizability of segmentation methods across different tumor entities, and inclusion of tumor types for which annotated data are currently limited.

As a novelty in 2025, algorithmic performance will be tested against human expert inter-rater variability, furthering insights into the clinical applicability of BraTS solutions.

3.2 MICCAI and the BraTS Platform

The BraTS challenge is hosted annually as part of the Medical Image Computing and Computer Assisted Intervention (MICCAI) conference[20], a leading venue in medical imaging and computational methods in healthcare. MICCAI provides an international forum for the dissemination of state-of-the-art research, and BraTS offers a benchmarking platform where participants can compare algorithms on standardized datasets with well-defined evaluation metrics such as Dice score and Hausdorff distance.

Participants who perform exceptionally in BraTS are often invited to present their methods in oral sessions at MICCAI workshops, increasing visibility and facilitating knowledge transfer between the computational and clinical communities. The challenge encourages the development of algorithms that are not only technically accurate but also clinically meaningful, reflecting real-world variability in multi-center MRI datasets.

3.3 Evolution of BraTS Tasks

From its initial focus on pre-operative glioma sub-region segmentation, BraTS has consistently introduced new tasks and diversified its objectives:

- Segmentation of glioma subregions based on multimodal MRI.
- Expansion to include additional tumor entities—such as meningioma, brain metastases, and pediatric tumors—as seen in editions like BraTS-PEDs and BraTS-METS.
- Introduction of synthesis tasks, including global missing-modality generation (BraSyn) and local inpainting, reflecting an emphasis on image completion and data augmentation strategies .
- Broadening clinical impact by including longitudinal studies, generalizability assessments, and inter-rater variability benchmarking.
- Alignment with MICCAI standards for reproducibility, fair benchmarking, and clinical relevance.

3.4 BraTS 2025 Tasks

In 2025, the BraTS Cluster of Challenges encompasses eleven distinct tasks, each addressing a critical topic under one of three computational categories: Sub-region Segmentation (SEG), Image Synthesis (SYN), or Classification (CLASS). The tasks are:

1. Pre- and Post-Treatment Adult Glioma (SEG)
2. Pre-Treatment Intracranial Meningioma (SEG)
3. Pre-Radiotherapy Intracranial Meningioma (SEG)
4. Pre- and Post-Treatment Brain Metastases (SEG)
5. Brain Glioma in the underserved sub-Saharan African patient population (SEG)
6. Pre-Treatment Pediatric Tumor Patients in partnership with multiple related societies (SEG)
7. Generalizability of Segmentation Methods Across Tumors (SEG)
8. MRI Global Synthesis (SYN)
9. MRI Local Inpainting (SYN)
10. Assessing the Heterogeneous Histologic Landscape of Glioma (CLASS)
11. Predicting the Tumor Response During Therapy (CLASS)

3.5 The BraTS Python Package

To foster reproducible research and streamline participation, the BraTS organizers have developed the `brats-toolkit`[6], a dedicated Python package that significantly lowers the barrier to entry. This toolkit is designed to standardize the entire experimental workflow, from data acquisition to evaluation and submission. It is publicly available on PyPI for easy installation and its source code is hosted on GitHub,[1] promoting transparency and community contributions.

[1] https://github.com/BrainLesion/BraTS

Key features of the `brats-toolkit` include:

- **Programmatic Data Access**. Provides utilities to easily download and manage the extensive BraTS datasets directly within a Python environment, ensuring participants use the correct, versioned data.
- **Standardized Preprocessing**. Includes functions for essential preprocessing steps such as NIfTI file handling, image co-registration, and normalization, which are crucial for consistent model performance.
- **Dockerized Baseline Models**. Offers access to containerized, pre-trained models that represent state-of-the-art architectures from previous challenges. These serve as powerful baselines and enable researchers to replicate leaderboard results with minimal setup.
- **Integration with Deep Learning Libraries**. Designed for seamless compatibility with major frameworks like PyTorch (often via MONAI) and TensorFlow, allowing researchers to easily integrate the toolkit into their existing workflows.

By providing a fully containerized and standardized environment, the `brats-toolkit` ensures computational reproducibility, a cornerstone of modern scientific inquiry. This ecosystem not only aids challenge participants but also serves the broader neuro-oncology imaging community by making high-quality data and benchmark models more accessible for novel research endeavors. However, the toolkit is still under development and has some limitations: it currently runs only on Docker, which can be impractical on shared clusters, and executing heavy workloads with extensive I/O can be challenging.

3.6 BraTS 2025 Key Dates

The following timeline outlines the main milestones for the BraTS 2025 Challenge:

- **3 March: Registration opens**
  Participants can register on `synapse.org` until the short paper submission deadline.

- **10 March: Training and validation data release**
  Training data with ground truth labels and validation data without ground truth labels become available.

- **31 July: Short paper submission deadline**
  Participants submit reports of methods and results on training and validation data. The final paper will later include testing results.

- **28 July – 7 August: Containerized algorithm submission**
  Organizers evaluate algorithms on testing data for participants with submitted short papers. Methods are ranked after statistical significance assessment via multiple permutation testing.

- **15 August: Invitation to participate**
  Participants with valid submissions (paper + container) are invited to present at the conference; presentation type is determined within two weeks.

- **23 August: Contacting top-performing teams**
  Preparation of slides for oral presentations.

- **10 September – 6 October: Camera-ready and copyright submission**
  Includes results on testing data for inclusion in the LNCS proceedings.

- **23–27 September: Challenge at MICCAI**
  Final top 3 ranked teams are announced during the conference.

3.7 Submission Protocol

The submission to the challenge followed a structured protocol, as summarized in the list above. First, a short paper (see B) had to be submitted, detailing the key components and implementation specifics of the proposed framework, along with the results on the validation set. In addition, participants were required to review two or three other submitted papers to provide constructive feedback on their approaches. Finally, the complete solution, including evaluation code, trained models, checkpoints and any necessary post-processing pipelines, was containerized into a Docker[21] image to ensure reproducibility and facilitate evaluation by the challenge organizers.

3.8 Task 8: Brasyn

Participation to this specific task will be the main focus of the thesis. Most state-of-the-art brain tumor segmentation algorithms rely on four MRI modalities: T1, T1 with contrast, T2, and FLAIR. In practice, some sequences are often missing. Building upon the Brain MR Image Synthesis Benchmark (BraSyn) organized at MICCAI 2023 and 2024, the 2025 edition continues to evaluate algorithms designed for the synthesis of entire MRI volumes, with an emphasis on robustness across varying acquisition protocols and pathological conditions. From a participant's perspective, this task is particularly relevant, as successful synthesis enables the direct application of established BraTS segmentation networks in clinical environments with limited imaging protocols and supports the analysis of archival tumor datasets where modalities may be missing.

3.9 Last Year's Winners: HF-GAN

HF-GAN [22] was adopted as the baseline model, utilizing a lightweight 2D pipeline to enhance training and inference efficiency. The framework comprises two stages: a first stage that generates individual 2D brain slices from preprocessed 3D volumes, and a second stage that refines the stacked slices to enforce volumetric consistency. In the present work, only the first stage is employed for slice-wise synthesis.

Preprocessing

In the first stage, the preprocessing pipeline employed a linear scaling of each slice to the range $[-1, 1]$. While this normalization ensured a bounded input space, it also introduced noticeable inter-slice discrepancies, as the scaling was sensitive to variations in intensity across different slices. To mitigate this issue, the second stage adopted a single-sample z-normalization, standardizing each slice by subtracting its mean intensity and dividing by its standard deviation. This approach reduced intensity inconsistencies and improved the overall homogeneity of the input data.

First Stage

In the first stage of the framework, the generator is designed to synthesize missing modalities across all possible scenarios using a unified network (Figure 3.1). It consists of four modality-specific late-fusion encoders (one for each modality), an early-fusion encoder that processes all available modalities simultaneously, a channel-attention feature fusion module, a modality-infuser, and a decoder. The overall encoder–decoder architecture follows the standard U-Net structure.

Each late-fusion encoder, composed of residual convolutional blocks with SiLU activation [23] and group normalization [24], extracts modality-specific features. The early-fusion encoder, architecturally identical to the late-fusion encoders, accepts a stacked four-channel input (with missing modalities masked) to capture complementary information across all modalities. The feature fusion module integrates both global and modality-specific information, employing channel-wise attention computed via global average pooling followed by softmax. A modality-infuser, implemented using Transformer blocks [32], incorporates information about the missing modality into the latent space.

The decoder reconstructs the output using upsampling blocks consisting of nearest-neighbor interpolation followed by a 3×3 2D convolution for smoothing. Consistent with standard U-Net design, skip connections link corresponding encoder and decoder layers, preserving spatial information and facilitating gradient flow.

One of the main concerns with this first-stage design lies in its overall complexity. Another critical issue is the reliance on seemingly arbitrary loss coefficients, whose selection lacks theoretical justification and may strongly influence the final performance. Finally, the use of the simplistic $[-1, 1]$ linear scaling as a normalization strategy proves to be particularly problematic: it amplifies inter-slice inconsistencies and fails to provide a stable statistical grounding, ultimately compromising the quality of the synthesized outputs.

Second Stage

While the first stage synthesizes slices in a purely 2D manner, volumetric consistency and tumor representation can be suboptimal due to the lack of 3D context. To address this, a

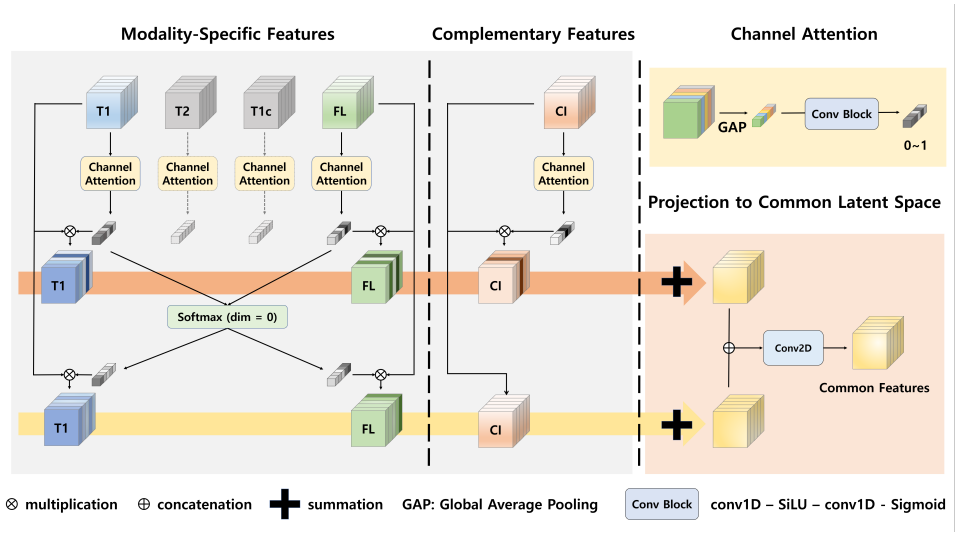


FIGURE 3.1: A visual representation of the original first stage implementation.

3D Refiner can be incorporated to enhance the slice-wise output by integrating structural information across the entire volume.

The Refiner is composed of three main components: an encoder, an element-wise cross-attention module, and a decoder (Figure 3.2) . The encoder processes both the synthesized MRI sequence and the available modalities individually, producing a set of feature representations. Specifically, given a generated volume $\hat{X}_{m_i} \in \mathbb{R}^{1\times D\times H\times W}$ and available MR sequences $X_{m_j} \in \mathbb{R}^{1\times D\times H\times W}$, with $m_j \in \{\text{T1, T2, FLAIR, T1ce}\}$, modality-specific encoders (based on simplified U-Net architectures) generate corresponding features $\hat{F}_{m_i}$ and $F_{m_j} \in \mathbb{R}^{C\times D\times H\times W}$.

The element-wise cross-attention module refines the synthesized feature maps by performing cross-attention in a voxel-wise manner. Unlike conventional attention mechanisms that rely on dot products across vectors, this module computes the similarity between features at the same spatial location using an absolute difference followed by a negative exponential, ensuring maximal similarity when voxel intensities match. Formally:

where Q_{m_i}, K_{m_j}, and V_{m_j} denote query, key, and value projections, respectively, and FF is a feed-forward layer implemented with 1×1 convolutions. The process is applied sequentially across all available modalities in a fixed order.

The decoder reconstructs the refined MRI volume from the updated features. To mitigate voxel-wise inconsistencies introduced by the cross-attention mechanism, convolutional layers with kernel size 3 are first applied to enforce spatial smoothness. A final 1×1 convolution maps the features back to image space, and the result is added residually to the slice-wise output from the first stage, yielding a volumetrically consistent MRI synthesis.

Despite its conceptual appeal, such a refinement stage introduces significant practical challenges. The Refiner employs a full U-Net (including both encoder and decoder) for each modality to extract voxel-wise features, which already makes the architecture computationally demanding. On top of this, the element-wise cross-attention mechanism further increases the memory load, as it requires processing and matching features at every voxel location across multiple 3D volumes. The combination of several heavy U-Net backbones with voxel-level attention results in an extremely large model, both in terms of parameters and runtime cost. Training such a system becomes impractical. Moreover, the computational burden is disproportionate to the expected performance gains, making this approach unsuitable for realistic large-scale or resource-constrained scenarios.

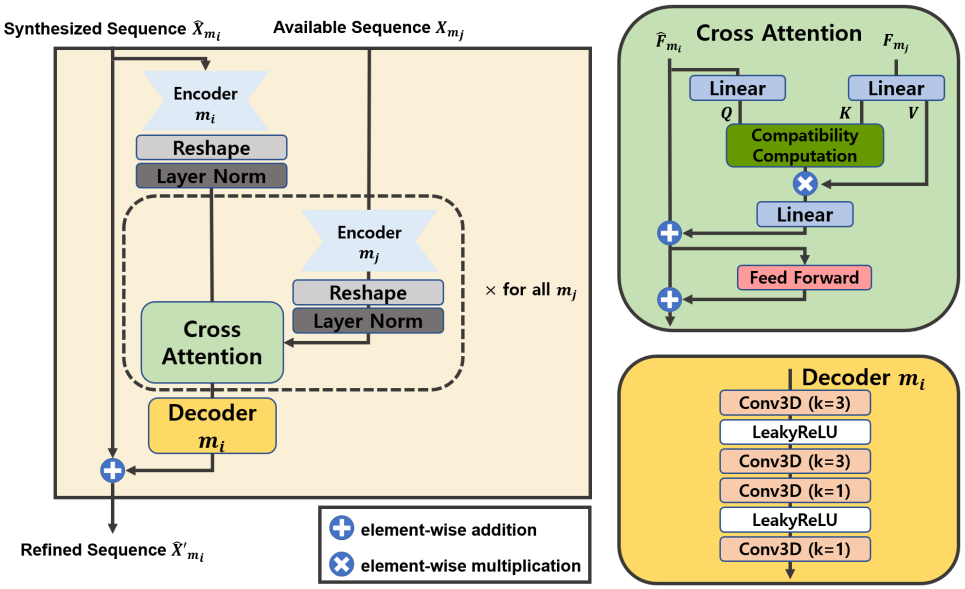


FIGURE 3.2: Visual representation of the Refiner network, which was discarded in the solution proposed in this thesis.

The Dataset

The BraSyn-2025 dataset is derived from BraTS-GLI 2023[25], BraTS-METS 2023[26], and BraTS-Meningioma[27]. It consists of a retrospective collection of brain tumor multi-parametric MRI (mpMRI) scans acquired from multiple institutions under routine clinical conditions, using different scanners and acquisition protocols. This results in a heterogeneous dataset that reflects the variability of real-world clinical practice. Ground-truth annotations of all tumor subregions were reviewed and approved by expert neuroradiologists. All scans were programmatically skull-stripped, removing cranial bone and non-brain tissue while preserving brain and cerebellar structures. During the validation and test stages, segmentation masks are not available, and for each subject one of the four modalities is randomly excluded ("dropout") to simulate incomplete clinical acquisitions. The training set includes 1,251 glioma cases and 238 newly added metastases cases. The metastases dataset was provided in two separate folders: the main set (165 samples) and an additional set (73 samples). The latter is considered "additional" by the organizers and is sometimes listed separately as *Train Metastases Add.*. The validation set consists of 219 glioma and 31 metastasis cases. The private test set comprises 219 glioma, 59 metastasis, and 283 meningioma cases. A visual representation of the dataset is shown in Figure 3.9. The inclusion of meningioma exclusively in the test set is intended to assess the generalizability of methods to previously unseen tumor types.

Test Metrics

In the inference task, each submitted algorithm is required to handle test cases in which one of the four MRI modalities is randomly excluded. For each subject, only three modalities are provided, and the algorithm must predict a plausible brain tumor image for the missing modality. The resulting synthesized image is then evaluated based on both its visual realism and its utility for downstream tumor segmentation.

Two complementary sets of metrics are employed for ranking the submissions (see formulas in Chapter 2, Eqs. 2.7–2.7).

- **Image quality metrics.** The structural similarity index measure (SSIM) is computed to quantify the realism of the synthesized images relative to the clinically acquired images.

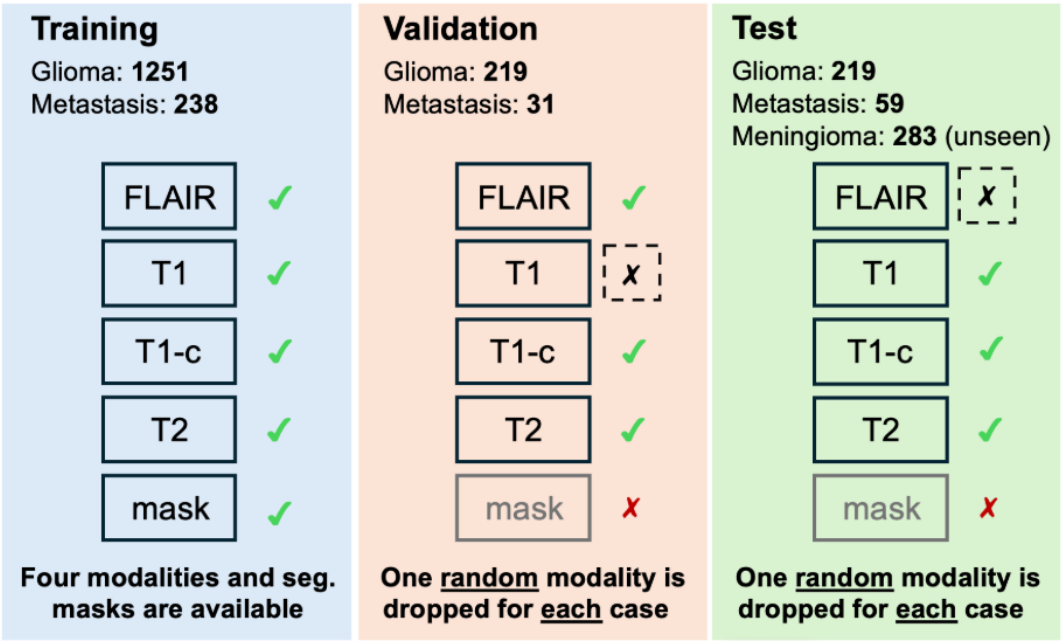


FIGURE 3.3: Dataset structure

SSIM is evaluated separately within the tumor region and the healthy brain, yielding two scores per test subject.

- **Segmentation metrics.** To assess the practical usefulness of the synthesized images, a state-of-the-art BraTS segmentation algorithm is applied to the completed image volumes. Performance is measured using the Dice similarity coefficient and the normalized surface distance (NSD) for three tumor substructures. Initially, the Hausdorff Distance was employed as the boundary-based metric, but it was changed to NSD in the final days before the challenge deadline. A pre-trained segmentation model is provided to participants to allow optimization under the evaluation conditions. For each subject, Dice and NSD scores are combined equally to produce a single ranking score per tumor substructure, resulting in three scores per test subject.

Chapter 4

Method

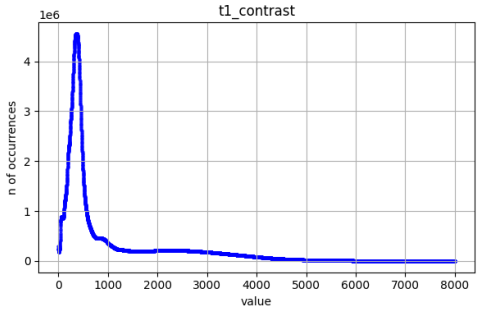
4.1 Summary

The main limitation of the previous HF-GAN framework was the use of [-1, +1] normalization, which is suboptimal compared to z-score normalization widely adopted by state-of-the-art models such as nnU-Net. The approach presented in this thesis adapts the HF-GAN pipeline by integrating z-score normalization, providing a more stable standardization of MR intensities and allowing the model to preserve absolute intensity information during de-normalization, thereby eliminating the need for additional intensity encoding. The methodology focuses on streamlining HF-GAN by employing only the first stage for 2D slice-wise synthesis. This design reduces computational cost, simplifies training, and maintains a lightweight framework, with final 3D volumes reconstructed by stacking the generated 2D slices, achieving competitive performance with a considerably leaner architecture.

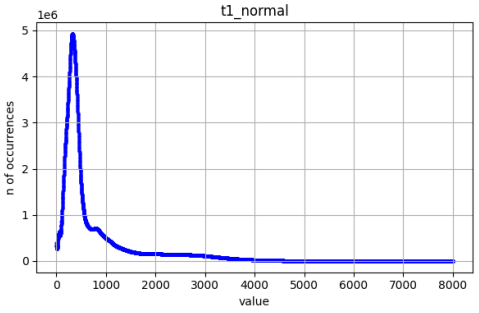
4.2 Preprocessing

Magnetic resonance (MR) images are notoriously challenging to preprocess effectively. Unlike natural images, where pixel intensities are bounded and can be normalized with simple linear scaling (e.g., mapping [0–255] to [0–1]), MR voxel values represent magnetic field intensities and are inherently unbounded. Consequently, standard scaling techniques are inadequate. To address this, the data distribution was first analyzed and visualized. A small number of

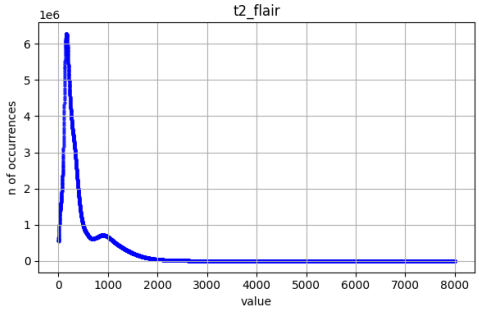
voxels with extremely high intensities were observed, introducing severe outliers that inflated both the mean and standard deviation in a non-representative manner. In addition, some samples contained negative values, considered unrepresentative artifacts originating from acquisition errors. These issues were mitigated by clamping all negative values to zero, after which histograms were computed to characterize the data distribution. For both visualization purposes and accurate statistical estimation, background voxels were excluded, since the distributions would otherwise be heavily dominated by background values. As the four modalities encode distinct types of information and exhibit markedly different distributions, a separate histogram was constructed for each modality.



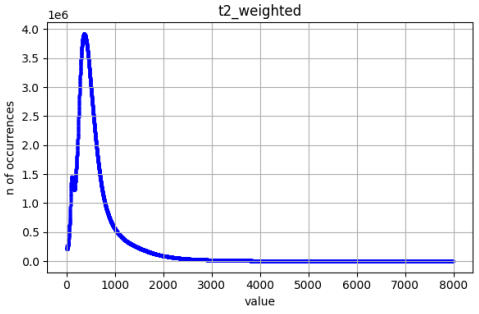
(a) T1 Contrast histogram



(b) T1 histogram



(c) FLAIR histogram

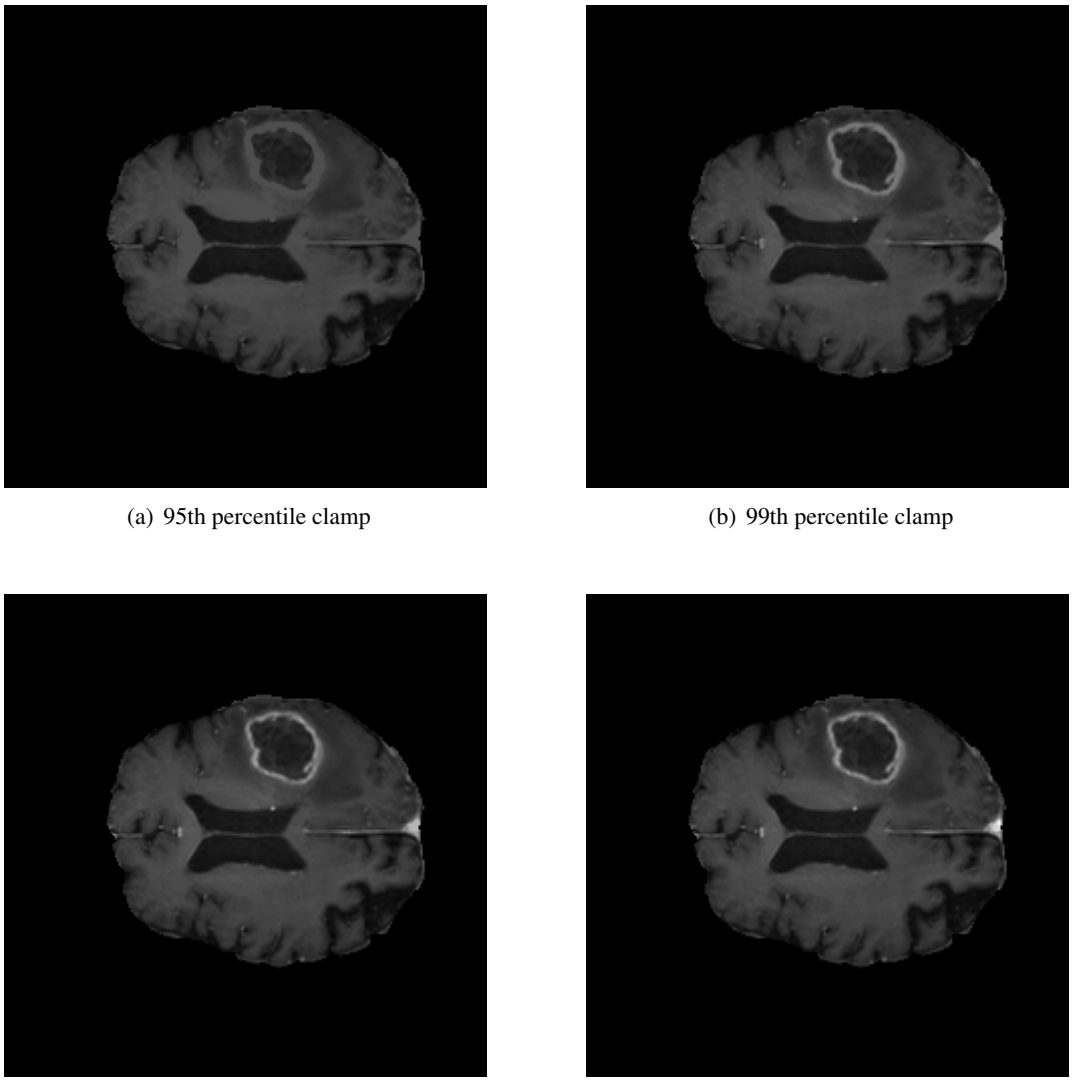


(d) T2 histogram

FIGURE 4.1: The four histograms representing data distribution. Background voxels (value 0) are excluded, and intensities are clamped to a maximum of 8000 to enhance qualitative visualization of the distributions.

After computing the histograms (see Figure 4.1), it became clear that clamping was required to mitigate the effect of extreme outliers while preserving the overall statistical characteristics of each modality. When clamping unbounded distributions, care must be taken to preserve the statistical properties of the data and avoid excessive information loss. As illustrated in

Figure 4.2, clamping results in a flattening of high-intensity areas, particularly in the occipital lobe and around the tumor. Extreme values are compressed into a uniform plateau, which obscures variation and flattens the distribution tails.



(a) 95th percentile clamp

(b) 99th percentile clamp

(c) 99.5th percentile clamp (best)

(d) Original

FIGURE 4.2: Different visualizations of the axial slice 90 from the sample BraTS-GLI-00494-000. The original contained values in the range [0, 12060]. For visualization, the differently clamped slices were linearly scaled using the same scale factor.

The 95th, 98th, 99th, and 99.5th percentiles were evaluated, resulting in different maximum values for clamping, as summarized in Table 4.1.

Perc.	T1c	T1n	T2f	T2w
95th	3,696	2,799	1,403	1,707
98th	4,623	3,503	1,797	2,532
99th	5,695	4,158	2,332	3,839
99.5th	8,664	7,315	8,842	8,233

TABLE 4.1: Max values in relation with different percentiles

Value	Clipp.	Norm.	T1c	T1n	T2f	T2w
Max.	✗	✗	2,120,538	155,724	612,368	4,563,634
Max.	✓	✗	8,664	7,315	8,842	8,233
Avg.	✓	✗	1,066.34	781.22	510.99	673.44
Std.	✓	✗	1,301.70	944.34	769.42	804.39
Min.	✓	✓	-0.8192	-0.8273	-0.6641	-0.8372
Max.	✓	✓	5.8367	6.9189	10.8277	9.3979

TABLE 4.2: MRI intensity values of the training set before and after applying the 99.5th percentile clipping and normalization strategies.

The final stats for the chosen method are reported in Table 4.2.

After computing the dataset statistics, 3D volumes were sliced along all three spatial dimensions—axial (top-down), coronal (front-back), and sagittal (left-right)—to generate a dataset of 2D brain slices. Following the approach used in HF-GAN, slices containing fewer than 2000 pixels per modality were discarded to avoid expending computational resources on nearly empty slices and to prevent the model from focusing on less informative regions. Generating slices along all three axes provided multiple views, enabling models to be trained on different perspectives and to select the best-performing one. The resulting number of slices is summarized in Table 4.3. Finally, all background pixels were set to a value of -1 to provide a consistent representation across modalities, ensuring coherence and full compatibility with the HF-GAN framework.

4.3 The Models

The framework consists of three models: a generator, a discriminator, and a segmenter. The generator is responsible for producing missing modalities, while the discriminator guides the

Slicing	Train Gli.	Val. Gli.	Train Met.	Train Met. Add.	Val. Met.
Axial	158,867	27,830	20,723	9,156	3,843
Coronal	194,472	34,034	25,857	11,541	4,791
Sagittal	158,976	27,906	20,765	9,246	3,865

TABLE 4.3: 2D dataset sample counts

generation in a GAN-style adversarial setup. The segmenter provides additional guidance to the generator, encouraging the production of slices with more accurately defined tumor structures, which are easier to segment.

4.3.1 The Generator

As anticipated in Section 4.1, the generator is based on the first stage of the HF-GAN framework, shown in Figure 4.3. It follows a U-Net architecture and consists of four main components: the encoders, the channel attention module, the modality infuser, and the decoder. The generator operates by accepting four input slices corresponding to the different modalities, in which one, two, or three modalities may be masked (i.e., unavailable) and set to a value of -1. The network then reconstructs a single target modality from these inputs. The following sections provide a detailed description of each component.

Encoders

The generator employs four modality-specific encoders and one early-fusion encoder, all sharing the same overall architecture. While each modality-specific encoder processes a single slice from its corresponding modality, the early-fusion encoder operates on all four modalities simultaneously, by receiving them as a stacked input.

Since feeding data sequentially into five separate encoders is computationally inefficient, the previous implementation was replaced with a Group Convolution approach. This module effectively functions as four independent convolutional networks—corresponding to the modality-specific encoders—executing in parallel. By processing all modalities simultaneously within a single operation, the Group Convolution module preserves the behavior of

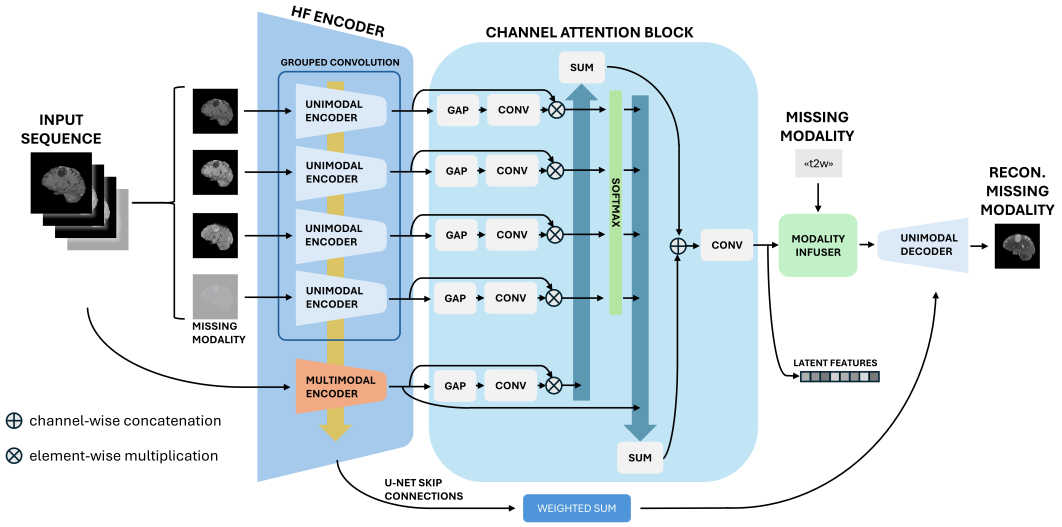


FIGURE 4.3: Architecture of the generator

the original modality-specific encoders while significantly reducing computation time and memory overhead, enabling faster feature extraction from multi-modal input slices.

Each encoder processes its input through five downsampling steps, progressively reducing spatial resolution while increasing feature depth. The channel dimensions across the encoder are as follows:

$$\mathbf{4\ (input)} \rightarrow \mathbf{64} \rightarrow \mathbf{128} \rightarrow \mathbf{256} \rightarrow \mathbf{512} \rightarrow \mathbf{640}$$

Correspondingly, the spatial dimensions are reduced as follows:

$$\mathbf{240{\times}240\ (input)} \rightarrow \mathbf{120{\times}120} \rightarrow \mathbf{60{\times}60} \rightarrow \mathbf{30{\times}30} \rightarrow \mathbf{15{\times}15}$$

Each downsampling step consists of two alternating blocks: a ResNet block followed by a downsampling block (see Figure 4.4). The downsampling block is implemented as a 2D convolution with kernel size 3 and stride 2, with padding chosen to reduce the spatial dimensions to exactly half of the input. The final downsampling step is an identity, preserving spatial resolution at the deepest level. After each ResNet block, a skip connection is extracted and later fed to the decoder, allowing high-resolution features to be combined with the upsampled representations during reconstruction.

A ResNet block in our encoder is composed of the following sequence of operations:

- Group Normalization

- 2D Convolution (stride 1, kernel size 3, padding 1)
- SiLU activation
- Group Normalization
- 2D Convolution (stride 1, kernel size 3, padding 1)
- SiLU activation

A skip connection takes the input of the ResNet block and adds it element-wise to the output of the second convolution, ensuring the block can learn residual mappings while preserving information from the input.

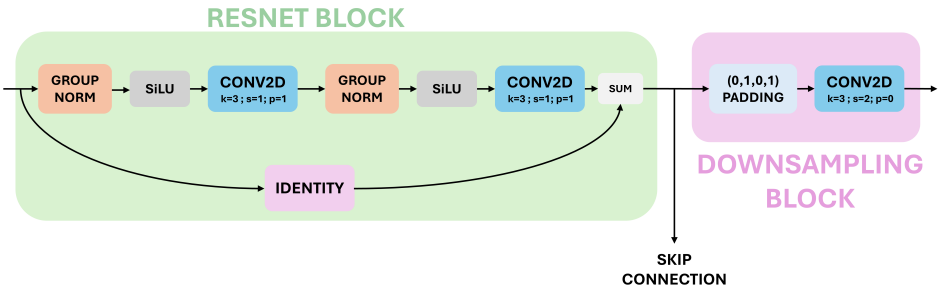


FIGURE 4.4: Architecture of a downsampling stage.

After each ResNet block, a skip connection is extracted and later fed to the decoder, allowing the network to combine high-resolution features with the upsampled representations during reconstruction. The modality-specific encoders isolate features for each input modality, while the early-fusion encoder simultaneously processes all available modalities, masking missing ones, to capture complementary cross-modality information. A complete representation of the encoder architecture is shown in Figure 4.5.

Since the architecture employs five encoders and a single decoder, it is necessary to aggregate the skip-connection tensors from the different encoders in a meaningful way. The chosen strategy is to compute a weighted sum of the skip features, defined as follows:

$$\mathbf{F}_{\text{skip}} = \mathbf{F}_{\text{early}} + \sum_{m \in \mathcal{A}} \beta_m \, \mathbf{F}_m, \tag{4.1}$$

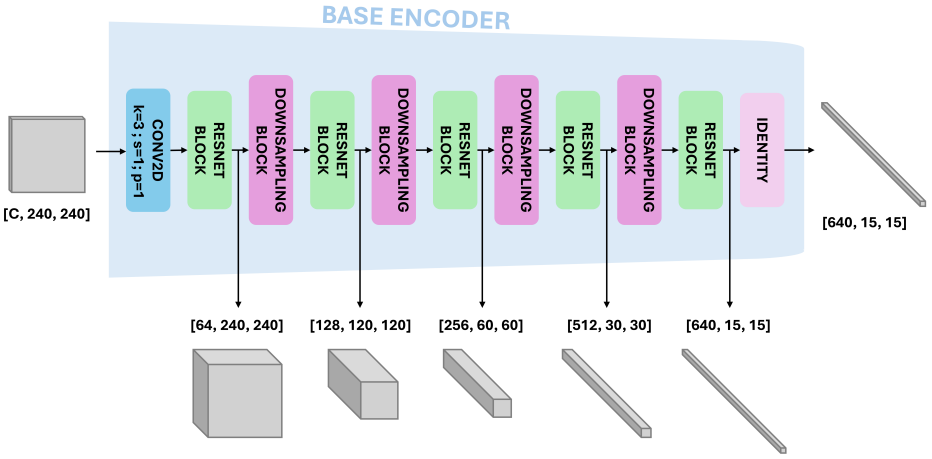


FIGURE 4.5: Architecture of one of the five encoders

where $\mathbf{F}_{\text{early}}$ denotes the skip features from the early-fusion encoder, $\mathbf{F}_m$ the skip features from the m-th modality-specific encoder, and $\mathcal{A}$ the set of available modalities. The coefficients are constrained so that

$$\sum_{m \in \mathcal{A}} \beta_m = 1, \tag{4.2}$$

ensuring that the modality-specific contributions are evenly distributed across the available inputs, while the early-fusion features are always included with weight one.

Channel Attention Module

The fusion module operates through two distinct processing paths, designed to handle the dual nature of modality-specific and complementary information. In the first path, feature representations corresponding to individual MR sequences are processed independently, with importance maps computed for each channel and normalized using a softmax operation across the available modalities. This normalization ensures that the relative contribution of each modality is balanced, even when certain sequences are missing, preventing over-reliance on any single input.

The second path becomes active only when multiple MR sequences are present, focusing on extracting complementary information that arises from their joint availability and cannot be captured by any modality in isolation. Importantly, despite its name, the module does not implement attention in the classical sense of query–key–value operations; instead, it performs channel-wise weighting to emphasize informative features and suppress redundant or irrelevant ones.

By integrating the outputs of the modality-specific and complementary paths, the module transforms the heterogeneous set of feature representations into a consistent latent space, enabling robust downstream processing regardless of the specific composition of MR inputs. While the design appears to improve performance, future ablation studies could clarify whether these benefits stem from the functional contribution of the module or primarily from the additional learnable parameters. The entire channel attention module is represented in Figure 4.3.

Modality Infuser

The modality infuser plays a crucial role in conditioning the generator on the specific target MR sequence. It transforms the common latent features into a target-specific representation through a structured processing path, as shown in Figure 4.6. First, the latent features are flattened and projected into a sequence of 1D embedding tokens using a convolutional operation, with each token having a hidden size of 640. Positional information is encoded via sinusoidal embeddings, which are element-wise added to the token sequence to preserve spatial context.

Modality-specific guidance is incorporated through a separate modality embedding, computed using a sinusoidal function followed by two fully connected layers with a SiLU activation in between, and then element-wise added to the sequence. The combined token sequence is processed through four transformer layers, each employing multi-head self-attention with 16 heads, capturing complex interactions across channels and spatial locations while integrating the target modality information. Finally, the sequence is reshaped back into the original spatial dimensions of the latent feature maps, forming a latent representation specific to the target modality, which is then passed to the decoder to reconstruct the synthesized MR image.

The main modification and optimization of this module consisted in replacing the custom attention blocks, which in the original implementation were built from scratch using `nn.Module` components. This design, although functional, was relatively inefficient and not well suited for large-scale training. In the revised implementation, the standard PyTorch attention module was adopted, allowing compatible architectures to take advantage of FlashAttention [28]. This change improves computational efficiency, reduces memory usage, and makes training more stable. In some experimental settings, an additional encoder module was also introduced to inject view-specific information, providing view-aware embeddings. This addition was useful when training the model to generate slices from different anatomical planes. The extra encoder was only used in selected runs, as described in Chapter 5.

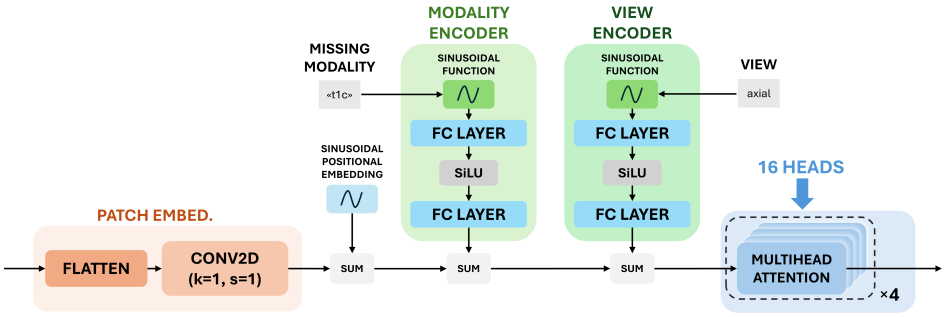


FIGURE 4.6: Architecture of the modality infuser module

Decoder

The decoder reconstructs 2D slices from the fused and infused features, mirroring the encoder structure through five upsampling stages. Each stage alternates between ResNet blocks (identical to those in the encoder) and upsampling blocks, where nearest-neighbor interpolation is followed by 3×3 convolutions to mitigate artifacts. The architecture of an upsampling stage is shown in Figure 4.7. This design choice follows established practice, as transpose convolutions are known to introduce checkerboard patterns that can compromise reconstruction accuracy. Skip connections between encoder and decoder layers preserve spatial information and facilitate gradient propagation, consistent with the U-Net paradigm. Aggregated skip connections from the encoders are concatenated with the outputs of the ResNet blocks at each stage. The final upsampling block is replaced with an identity operation, followed by a concluding ResNet block, group normalization, SiLU activation, and a final convolutional layer,

yielding the reconstructed modality slice. The complete decoder architecture is illustrated in Figure 4.8. The model was trained using cross-entropy loss for the classification head and binary cross-entropy loss for the discriminator head.

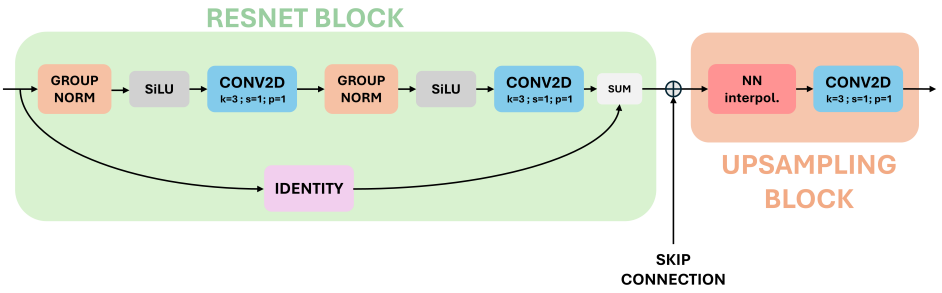


FIGURE 4.7: Architecture of an upsampling stage.

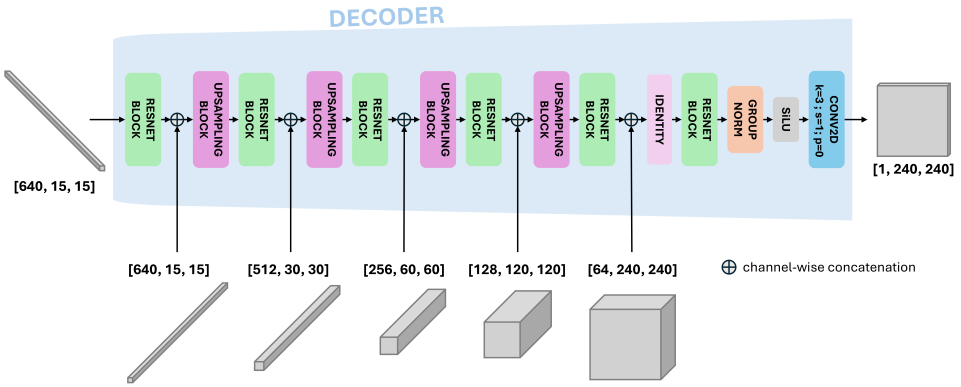


FIGURE 4.8: Architecture of the decoder

4.3.2 The Discriminator

The discriminator follows the PatchGAN design [29] and is implemented as a convolutional network. It begins with a convolutional layer followed by a LeakyReLU activation. This is followed by a sequence of three convolutional blocks, each consisting of a convolution, group normalization, and a LeakyReLU non-linearity, with progressively increasing feature

dimensionality and spatial downsampling. The feature maps produced by these blocks are passed through a final convolutional layer to obtain the real/fake discrimination output, which guides the generator toward producing anatomically faithful images and discourages the appearance of synthesis artifacts. In parallel, a classifier head with a large receptive field is applied to the same features to predict class labels, thereby encouraging the generator to produce modality-consistent images. The complementary roles of the two heads in the overall training scheme are further detailed in Section 4.4. The architecture of the discriminator is illustrated in Figure 4.9.

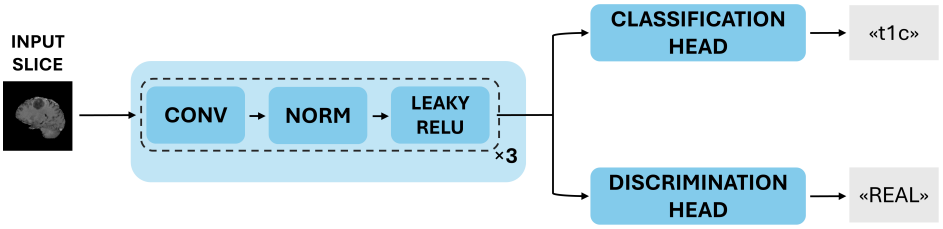


FIGURE 4.9: Architecture of one of the five encoders

4.3.3 The Segmenter

Segmenter. To support the generation process and improve tumor segmentation accuracy, a lightweight 2D segmentation network was employed to delineate brain tumor subregions. The architecture was derived from nnU-Net, consisting of four downsampling–upsampling stages with feature dimensions of [32, 64, 128, 256], and using SiLU activations throughout.

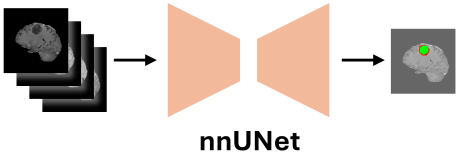


FIGURE 4.10: Simple representation of the segmenter.

Training was performed on a subset of slices containing at least 0.1% tumor tissue, ensuring that the model focused on informative regions. To promote generalization, the network was trained across all anatomical views (axial, coronal, and sagittal) and tumor types (gliomas and metastases), with view and tumor type information incorporated into the input through one-hot encoding. This design resulted in a compact yet effective model with only 1.79 million parameters.

The segmenter achieved validation Dice scores of 0.74, 0.80, and 0.82 on the NETC, ED, and ET classes, respectively. Training was carried out using the Focal Tversky loss [30], which has shown advantages in handling highly imbalanced class distributions. The model was trained independently and subsequently frozen during the training of the generation pipeline.

The segmenter contributed to the training of the generator through a dedicated term in the loss function, as described in Section 4.4.

A multi-class approach was adopted rather than a multi-label formulation, meaning that each voxel is assigned to a single class; however, the two formulations are conceptually equivalent. Optimal performance was achieved using the Tversky focal loss [30], which effectively mitigates class imbalance. Specifically, while the background (comprising both background pixels and healthy tissue) is always present in each sample, tumor structures may be entirely absent. The Tversky focal loss addresses this imbalance by emphasizing harder-to-classify examples, thereby improving the model's focus on challenging samples. The stability and effectiveness of this loss render it a natural choice for guiding the generator in the segmentation component of the loss, as illustrated below.

4.4 The Loss Function

Our model leverages a combination of complementary loss functions designed to guide the network toward learning robust and balanced representations across both anatomical fidelity and modality consistency. The overall loss function is defined as:

$$\mathcal{L}_{\text{total}} = 10 \cdot \mathcal{L}_{\text{recon}} + 0.25 \cdot \mathcal{L}_{\text{adv}} + 0.25 \cdot \mathcal{L}_{\text{class}} + \mathcal{L}_{\text{feat}} + \mathcal{L}_{\text{cycle}} + 5 \cdot \mathcal{L}_{\text{SSIM}} + 5 \cdot \mathcal{L}_{\text{Tversky}}$$

Not all components were used in every experimental configuration. A complete analysis of all tested combinations is provided in Chapter 5.

The coefficients assigned to each loss term in $\mathcal{L}_{\text{total}}$ were not obtained through an exhaustive hyperparameter search. Due to the high computational cost and time constraints, the weighting scheme was largely guided by the choices adopted in the original HF-GAN implementation.

The reconstruction loss $\mathcal{L}_{\text{recon}}$, implemented as the mean absolute error (MAE) between the generated and ground-truth images, constitutes the primary loss component. This term enforces pixel-level fidelity and ensures that synthesized images closely approximate the real modalities. To complement this, Structural Similarity Index (SSIM) losses $\mathcal{L}$SSIM are included to enhance perceptual quality, focusing on preserving local structure and contrast. $\mathcal{L}_{\text{SSIM}}$, computed as 1-SSIM on a per-pixel basis and aggregated only over the regions of interest. Two variants are employed: one considering the entire volume and another producing separate contributions for healthy tissue and tumor tissue, weighted equally during training to balance the network's attention across anatomically and clinically relevant regions.

The adversarial loss $\mathcal{L}_{\text{adv}}$ drives the generator to produce outputs that are sufficiently realistic to be indistinguishable from real images according to the discriminator, thereby reducing common synthesis artifacts and improving image fidelity. Similarly, the classification loss $\mathcal{L}$class leverages the discriminator's auxiliary head to enforce modality coherence, guiding the generator to produce outputs consistent with the target modality.

Cycle consistency is enforced through a two-step generation procedure. In the first step, the generator synthesizes the missing modality using the available ground-truth modalities. In the second step, this generated output replaces its corresponding placeholder, while one of the remaining available modalities is randomly masked (with uniform probability across maskable modalities), and the generator is applied again using the partially masked inputs along with the previously generated slice. The cycle loss $\mathcal{L}_{\text{cycle}}$ is defined as the MAE between the second-pass output and the original ground-truth modality, ensuring that the generator produces outputs that are consistent across multiple passes and robust to missing input scenarios.

To further encourage representational consistency, a feature-level loss $\mathcal{L}_{\text{feat}}$ is included, defined as the cosine similarity between the bottleneck features extracted during the first and second generation passes. This term promotes stability in the latent space, ensuring that generated features remain coherent even when inputs vary.

Finally, a Tversky-focal loss [30], $\mathcal{L}_{\text{Tversky}}$, is incorporated to facilitate the generation of outputs that are more easily segmentable, particularly in the presence of class imbalance. This loss is computed between the predictions of the frozen segmenter and the ground-truth segmentation labels, encouraging the generator to emphasize structures corresponding to clinically relevant subregions and thereby improving downstream segmentation performance.

Consistent with the HF-GAN framework, forward propagation is executed twice per target modality for cycle consistency, resulting in a total of eight forward passes on the same network (note that these do not follow separate network paths). Gradients from these multiple forwards are aggregated by combining the corresponding losses into a single scalar, with backpropagation applied once per batch. This approach ensures stable gradient updates while capturing both pixel-level and feature-level constraints across the entire synthesis pipeline.

4.5 The Training

Due to the complexity of the model training and the multiple loss components involved, we provide pseudo-code in Algorithm 1 for clarity. During training, each batch is composed of 2D slices randomly sampled from the entire dataset, without enforcing that all slices belong to the same patient. For each batch, the entire training pipeline is repeated four times, once for each modality to be generated. Four subsets of available are generated independently for each sample, corresponding to the four modalities. The missing modalities are then masked with value -1 (background). In the first step, a random modality is selected for generation; in the next steps, the remaining modalities are generated sequentially. This procedure ensures that the network learns to reconstruct each modality from the available information while being exposed to diverse combinations of missing inputs.

Once the batch is formed, and the inputs are correctly masked, the generator performs a forward pass to predict the missing modality. The reconstruction losses, specifically the pixel-wise reconstruction loss L_{recon} and the structural similarity loss L_{SSIM}, are computed to quantify the fidelity of the generated images with respect to the ground truth. Subsequently, the discriminator evaluates the generator's output, producing both the adversarial loss L_{adv} and the classification loss L_{class}, which encourage realistic and semantically consistent outputs.

The frozen segmenter is then applied to the generated images, producing segmentation maps. The Tversky loss, computed over all four classes (including background), quantifies the discrepancy between the predicted and ground-truth segmentations. To enforce cycle consistency, a second forward pass through the generator is performed: in this pass, another modality is requested to be generated using the previously generated modality as part of the input, while the other three modalities are fully available. This second forward contributes to

the cycle consistency loss L_{cycle} on the output and to the feature-level loss L_{feat}, promoting coherent reconstructions and consistent latent representations.

All the individual losses are then aggregated into the total loss:

$$\mathcal{L}_{\text{total}} = 10L_{\text{recon}} + 0.25L_{\text{adv}} + 0.25L_{\text{class}} + L_{\text{feat}} + L_{\text{cycle}} + 5L_{\text{SSIM}} + 5L_{\text{Tversky}}.$$

The generator parameters are updated with a backward pass, an optimization step, and a learning rate scheduler step. Finally, the discriminator is trained separately: each of its batches is composed of the generated missing modalities and an equal number of real images, and it is updated using the binary cross-entropy (BCE) loss. This procedure is repeated for all modalities, batches, and epochs, gradually improving both reconstruction quality and segmentation performance.

Training was performed using the Adam optimizer with a learning rate of 0.0005 for a batch size of 64, which was linearly adjusted when the batch size changed. A cosine learning rate scheduler with a linear warmup phase was employed, but only for a single cycle, and the scheduler was stepped immediately after each optimizer update. Gradient clipping was applied with a maximum norm of 0.1 to stabilize the SSIM loss. No data augmentation was applied, as it was deemed unnecessary and could introduce artifacts not present in the original slices, and gradient accumulation was not used. All training metrics, losses, and system stats were tracked using Weights & Biases[31] to ensure reproducibility and facilitate monitoring of model convergence.

Algorithm 1: Training Loop for Multi-Modal Segmentation and Reconstruction

for each epoch **do**
  for each batch in dataloader **do**
    for each modality **do**
      Modality Dropout: Drop one modality per sample and mask it
      Generator Forward:
        Forward pass through generator
        Compute reconstruction loss L_{recon}
      Discriminator and Classification:
        Forward pass through discriminator
        Compute adversarial loss L_{adv}
        Compute classification loss L_{class}
      SSIM Loss:
        Compute L_{SSIM}
      Segmentation: Forward pass through segmenter
        Compute Tversky loss: $L_{\text{Tversky}} = \sum_{i=0}^{3} L_{\text{Tversky\_class\_i}}$
      Cycle Consistency: Drop another modality, replace the previous missing one
        Forward pass through generator
        Compute feature loss L_{feat} and cycle consistency loss L_{cycle}
      Total Loss:
        Aggregate all losses into $\mathcal{L}_{\text{total}}$
        Backward pass, optimizer step, scheduler step (Generator)
      Discriminator training:
        Forward through the Discriminator
        Compute BCE loss
        Backward pass, optimizer step, scheduler step (Discriminator)
    end for
  end for
end for

4.6 The Stacking Pipeline

Since the proposed model operates exclusively on 2D slices, while the input data is inherently 3D, a dedicated post-processing pipeline was developed to correctly reassemble the slices into volumetric outputs. This pipeline is applied only at inference time, as the training is entirely performed in 2D. The complete pipeline consists of the following steps:

Slicing. The input volume is sliced along one or more spatial axes. The number of slices depends on the chosen axis: 240 for sagittal and coronal slicing, and 155 for axial slicing.

Padding. While axial slices are already 240×240, sagittal and coronal slices have dimensions of 240×155. To obtain square slices, zero-padding is applied by adding 42 pixels at the top and 43 pixels at the bottom of each slice.

Preprocessing. Standard 2D preprocessing operations are applied to each slice, as described in Section 4.2.

Model inference. The preprocessed slices are batched and fed to the generator. Batch size is set to 31 for axial slices and 40 for sagittal and coronal slices. Batching does not influence the model's output but accelerates inference.

Stacking. The generated slices are stacked to reconstruct the intermediate 3D volume.

Cropping. The padding added for sagittal and coronal slices is removed, restoring the original spatial dimensions of the input volume.

Denormalization. Intensity values are transformed back to their original scale.

Background determination. To ensure consistency in background regions, a majority voting strategy is applied across the available input modalities. Since three modalities are always available during testing, a voxel is labeled as background if at least two modalities agree on its classification. In such cases, the voxel value in the reconstructed volume is set to 0.

Volume aggregation. When inference is performed across multiple views, the reconstructed volumes are aggregated by computing the voxel-wise mean, producing the final output volume.

Figure 4.11 shows a graphical representation of the complete stacking pipeline.

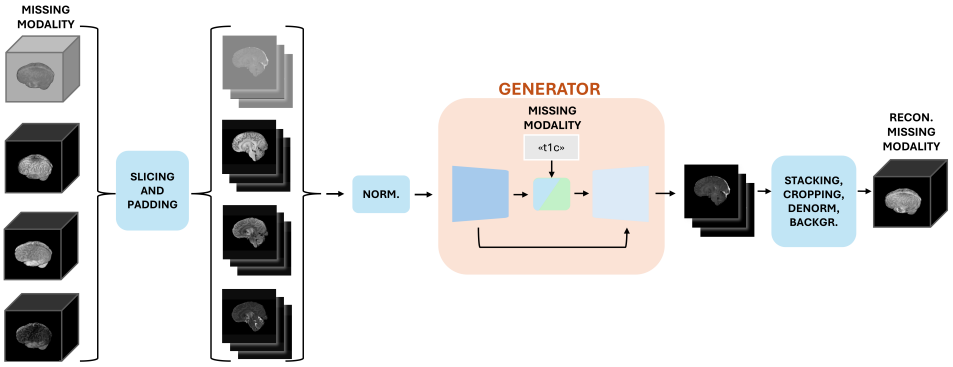


FIGURE 4.11: The full 3d generation pipeline

The entire pipeline, including the slice reassembly and post-processing steps, was packaged into the Docker container that we submitted to the challenge, ensuring that the model can be executed reproducibly in any compatible environment.

Chapter 5

Experiments and Results

A total of 25 different experiments were conducted to evaluate various strategies, loss functions, hyperparameters. To maintain clarity and focus, only the most important and noteworthy runs are reported in this work. The Structural Similarity Index Measure (SSIM) was computed by masking irrelevant regions with zeros. This approach introduces significant limitations when comparing scores across different samples, as the background dominates the metric and produces artificially high values. While methods that completely avoid this issue exist (see Appendix B for details), the challenge adopted this conventional procedure. Consequently, the reported tables follow this methodology.

Table 5.1 provides a comprehensive evaluation of all metrics. Each score is reported on the Glioma (GLI), Metastasis (MET), and combined (ALL) validation sets. In addition, SSIM is computed by masking all regions except for either the Whole Tumor (WT) or the Healthy Tissue (HT). Finally, segmentation metrics (DICE, NSD) are evaluated on the three standard tumor subregions: Enhancing Tumor (ET), Tumor Core (TC), and Whole Tumor (WT). Further details on these subregion definitions are given in Section 2.2. All scores were computed locally in order to reproduce, as closely as possible, the official validation pipeline. This was necessary because the challenge platform opened for submissions only a few days before the deadline, leaving insufficient time to rely on the official validation phase.

At the time of writing this thesis, the final scores on the hidden test sets have not yet been released. They will be made available at the following link:
`https://www.synapse.org/Synapse:syn64153130/wiki/633062`.

TABLE 5.1: Experimental results obtained combining different losses, leveraging different training sets and 2D views. The Run IDs are explained in Table 5.2.

Metric	Val.Set	Class	Run ID										
			R1	**R2**	**R3**	**R4**	**R5**	**R6**	**R7**	**R8A**	**R8S**	**R9**	**R10**
SSIM	ALL	WT	99.73	99.77	99.76	99.74	99.71	99.76	99.75	99.76	99.77	99.75	99.76
		HT	93.31	94.07	94.05	93.82	93.15	94.09	93.91	93.79	93.89	94.07	93.79
	GLI	WT	99.72	99.76	99.76	99.74	99.70	99.75	99.74	99.75	99.75	99.74	99.75
		HT	93.69	94.47	94.44	94.22	93.52	94.49	94.28	94.10	94.20	94.47	94.11
	MET	WT	99.79	99.81	99.82	99.80	99.78	99.82	99.82	99.84	99.84	99.82	99.84
		HT	90.62	91.23	91.23	90.94	90.52	91.29	91.27	91.59	91.73	91.27	91.59
DICE	ALL	ET	53.62	57.13	54.16	54.16	52.43	57.02	57.70	55.98	57.86	57.00	56.42
		TC	73.86	76.49	75.48	75.48	73.89	76.88	75.11	75.28	77.55	76.88	75.28
		WT	71.74	73.63	71.89	71.89	70.41	74.07	74.97	73.57	74.45	74.07	73.58
	GLI	ET	52.94	56.67	53.82	53.82	51.87	56.58	57.40	55.90	57.14	56.58	55.90
		TC	72.28	75.27	74.24	74.24	72.55	75.59	73.55	73.88	76.16	75.58	73.85
		WT	69.75	71.91	70.01	70.01	68.47	72.25	73.26	71.79	72.50	72.25	71.79
	MET	ET	58.44	60.36	56.51	56.51	56.40	60.11	59.77	56.52	62.92	59.98	60.09
		TC	85.03	85.06	84.28	84.28	83.35	86.05	86.14	85.17	87.38	86.00	85.33
		WT	85.78	85.83	85.15	85.15	84.16	86.93	87.03	86.13	88.26	86.92	86.18
NSD	ALL	ET	54.06	56.48	57.66	54.38	52.55	56.96	58.03	56.81	57.92	56.96	56.81
		TC	63.27	66.80	68.55	65.83	63.06	67.46	65.73	65.79	68.30	67.48	65.79
		WT	52.80	55.54	56.51	53.02	51.32	56.11	57.30	55.41	56.64	56.11	55.41
	GLI	ET	52.68	55.41	56.86	53.43	51.34	55.84	57.08	55.51	56.40	55.85	55.51
		TC	61.11	65.04	66.94	64.22	61.26	65.32	63.42	63.65	66.25	65.35	63.65
		WT	49.19	52.17	53.20	49.60	47.90	52.35	53.79	51.78	52.88	52.36	51.78
	MET	ET	63.82	64.05	63.25	61.12	61.10	64.87	64.74	66.00	68.64	64.81	66.00
		TC	78.47	79.24	79.88	77.22	75.73	82.62	82.08	80.89	82.82	82.58	80.89
		WT	78.30	79.34	79.91	77.15	75.48	82.65	82.13	81.12	83.23	82.64	81.12

The final validation set was generated by randomly dropping one modality per sample, with a fixed seed to ensure reproducibility. Contrary to what was initially assumed, it did not involve iteratively dropping each of the four modalities for every sample. More about the evaluation pipeline at:

`https://github.com/hongweilibran/BraSyn.git`

5.1 Runs

Each run is identified by a unique progressive identifier. All runs were trained on nodes equipped with L40S or A40 48 GB NVIDIA GPUs. Training times vary due to differences in GPU speed: although the two models have the same memory capacity, the L40S is faster.

TABLE 5.2: Characterization of settings employed in different experiments. Unmentioned losses are always present.

RunID	Losses					Train		View	
	$\mathcal{L}_{\text{cycle}}$	$\mathcal{L}_{\text{feat}}$	$\mathcal{L}_{\text{SSIM\_whole}}$	$\mathcal{L}_{\text{SSIM\_dual}}$	$\mathcal{L}_{\text{Tversky}}$	GLI	MET	Axial	Sagit.
R1	✓	✓	✗	✗	✗	✓	✗	✓	✗
R2	✓	✓	✗	✓	✗	✓	✗	✓	✗
R3	✓	✓	✗	✓	✗	✓	✗	✓	✗
R4	✓	✓	✓	✗	✗	✓	✗	✓	✗
R5	✗	✗	✗	✗	✗	✓	✗	✓	✗
R6	✓	✓	✗	✓	✓	✓	✗	✓	✗
R7	✓	✓	✗	✓	✓	✓	✗	✗	✓
R8A	✓	✓	✗	✓	✗	✓	✓	✓	✓
R8S	✓	✓	✗	✓	✗	✓	✓	✓	✓
R9	✓	✓	✗	✓	✓	✓	✗	✓	✓
R10	✓	✓	✗	✓	✗	✓	✓	✓	✓

Additionally, jobs were automatically scheduled on the AImageLab cluster, which further contributed to variability in training duration. Checkpointing was performed only at the end of each epoch. Consequently, if a job stopped mid-epoch, the compute performed during that epoch had to be repeated. While checkpointing mid-epoch may seem more efficient, the end-of-epoch scheme was chosen to avoid misalignment with scheduler steps, which occur at every optimizer step. Therefore, reported training times should be considered indicative only and not as a measure of network performance. A description of the main characteristics of each run is provided below. Exhaustive details regarding the components included or omitted in each experiment are reported in Table 5.2.

R1

This run serves as a baseline. The model was trained on the training set for 10 epochs, a number chosen because 2D validation scores plateaued. The loss function was the original formulation from the HF-GAN paper; no SSIM or segmentation terms were included. As shown in the tables, the baseline scores are already solid.
Precision: FP16. Total training time: 38 hours on two GPUs. Dataset: GLI.

R2

This run incorporates an SSIM term into the loss, divided into separate components for healthy and tumor regions, weighted equally. This formulation proved effective, as it encouraged the model to focus on the tumor regions, aligning with the evaluation criteria at test time. The SSIM computation was implemented as a non-trainable `nn.Module`, after programmatically masking background pixels to avoid unnecessary or harmful gradients. Due to mathematical instability, the loss occasionally diverged to NaN under FP16 precision, likely due to dynamic range and precision issues. Gradient clipping and a small epsilon (on the order of 10^{-3}) were added to mitigate overflow and underflow. Training was therefore performed in FP32 precision. Performance improved relative to R1.
Precision: FP32. Total training time: 61 hours on two GPUs. Dataset: GLI.

R3

This run modifies the original procedure by restricting the model to generate only a single missing modality, instead of an arbitrary subset, as required by the challenge. This change did not affect final results but enabled earlier convergence, facilitating monitoring of early-stage performance. The dual healthy-tumor SSIM loss from R2 was retained for direct comparison.
Precision: FP32. Total training time: 90 hours on two GPUs. Dataset: GLI.

R4

This run is a counterpart to R2. The difference lies in the SSIM formulation: here, the loss is a single term encompassing both healthy and tumor tissue. This caused the network to focus more on healthy brain regions, which dominate the image. Consequently, healthy SSIM did not improve significantly, while tumor SSIM decreased, as expected.
Precision: FP32. Total training time: 87 hours on two GPUs. Dataset: GLI.

R5

This run is similar to R1, with the L_{feat} and L_{cycles} terms removed to evaluate their impact on performance. Removing cycle consistency reduced the computational cost by half (4

forwards per batch instead of 8). Scores decreased slightly, confirming that cycle consistency contributes positively to performance.
Precision: FP32. Total training time: 39 hours on two GPUs. Dataset: GLI.

R6

This run resumes from the checkpoint of R3 and continues training for three additional epochs, introducing the segmentation loss L_{Tversky}. Starting from a checkpoint was motivated by prior observations that adding this term too early negatively affected both training stability and performance. The model was fine-tuned using a learning rate equal to one-tenth of the original starting rate. Both the segmenter from the previous training and the generator in this run were trained exclusively on axial slices, as in all prior iterations.
Precision: FP32. Total fine-tuning time: 30 hours on two GPUs. Dataset: GLI.

R7

This run mirrors R6 but operates on sagittal slices.
Precision: FP32. Total fine-tuning time: 20 hours on two GPUs. Dataset: GLI.

R8A, R8S, R10

These identifiers correspond to the same model, evaluated under different settings. The model was trained using axial, sagittal, and coronal slices simultaneously, aiming to generalize across multiple views. Training was performed for a number of steps equivalent to 10 epochs, resulting in approximately four effective epochs. Dataset: GLI and MET.

Evaluation details:

- **R8A.** Generation on the axial view.
- **R8S.** Generation on the sagittal view; this model was the overall best-performing.
- **R10.** Generation on both axial and sagittal views, followed by stacking and averaging of the two post-processed volumes.

This approach, which did not include the segmenter loss, demonstrates that knowledge learned across multiple views can transfer effectively to single-view generation. Coronal slices were excluded during evaluation, as their smaller per-slice area significantly reduced performance. *Precision: FP32. Total training time: 88 hours on two GPUs.*

R9

This run aggregates predictions from R6 and R7, similarly to R10.

Other Runs

Additional runs were conducted to explore various aspects of the training pipeline. Some experiments focused on hyperparameter tuning, such as comparing training for 10 versus 20 epochs, which demonstrated that 10 epochs were sufficient for convergence. Other runs investigated model performance under different compilation strategies, memory layouts, and mixed-precision settings. While these runs are not noteworthy enough to be listed individually, they provided valuable insights into model stability, training efficiency, and the sensitivity of segmentation performance to various design choices.

These exploratory runs also revealed that the model was extremely memory-bound, with over 95% of GPU time spent on memory operations, resulting in significantly slower performance than hardware capabilities would allow. Adjustments to memory layout and model compilation yielded only marginal improvements, suggesting that a fundamental redesign of the architecture may be necessary to achieve better efficiency.

5.2 Discussion

This section aims to consolidate the observations from the individual experiments and provide an overall perspective on the outcomes. The results collectively demonstrate strong MRI synthesis performance and good generalization to unseen data. An example of real-versus-reconstructed comparisons computed on four different samples from the validation set is shown in Figure 5.1. Notably, models trained exclusively on the GLI dataset maintained

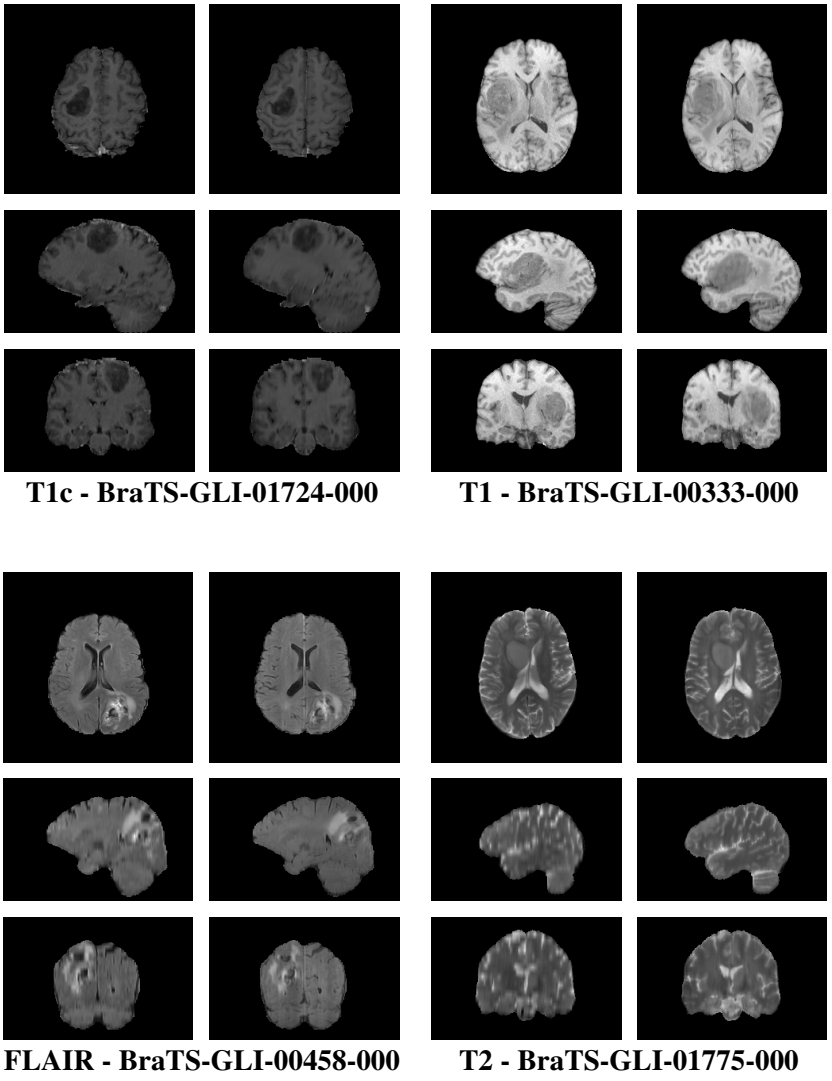


FIGURE 5.1: Comparisons of real (left) and reconstructed (right) images for four different patients sampled from the GLI validation dataset.

satisfactory performance when applied to MET volumes, highlighting the robustness of the proposed framework across different tumor types.

Run R1, which did not include the SSIM loss term, already achieved high perceptual quality. Subsequent runs explored incremental modifications. Introducing the SSIM loss in R2, formulated as two separate terms for healthy and tumor tissue, improved structural similarity metrics by encouraging the model to balance attention between the two regions. In contrast, R3 applied the SSIM loss across the entire volume without differentiating between tissue

types, which degraded tumor-specific performance and confirmed the importance of spatially targeted perceptual losses.

The original framework supports scenarios with 1, 2, or 3 missing modalities. Runs R2 and R3 illustrate the impact of this choice: while R2 allowed variable numbers of known modalities and R3 fixed the number at three, the final quantitative scores were comparable. However, convergence in R3 was faster, suggesting that fixing the number of input modalities can facilitate training stability without substantially affecting final performance.

Run R5 investigated the role of auxiliary losses by removing the cycle-consistency and feature-matching terms. This modification led to degraded performance, confirming their contribution to both stability and reconstruction quality. Nevertheless, the inclusion of cycle consistency approximately doubled computational cost, requiring eight forward passes per batch instead of four, and also increased memory usage.

The introduction of the Tversky loss in R6 and R7, aimed at improving segmentation quality, did not result in consistent improvements. Similarly, aggregating predictions from multiple views by simple averaging (R9, R10) failed to produce performance gains, suggesting that more advanced fusion strategies may be necessary to fully exploit multiview information.

The best-performing model was obtained in R8S. This configuration was trained on both axial and sagittal views of the combined GLI+MET dataset but evaluated on sagittal slices only. Notably, it did not include the Tversky loss, yet benefited from the increased diversity of training views, leading to improved generalization and the strongest overall results.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This thesis investigated the problem of missing-modality MRI synthesis in the context of brain tumor imaging, with a focus on improving both perceptual quality and clinical relevance. Building upon the HF-GAN framework, a series of experiments were conducted to analyze the impact of different loss functions, input configurations, and training strategies. The study demonstrated that perceptual guidance through SSIM losses, particularly when applied separately to healthy and tumor regions, significantly improves structural similarity metrics and tumor-specific reconstruction. The experiments also confirmed the importance of auxiliary objectives such as cycle consistency and feature-matching for stable training, despite their higher computational cost. In contrast, the introduction of the Tversky loss did not yield consistent benefits, and simple aggregation strategies across multiple views proved ineffective.

The results further highlighted the robustness of the proposed framework, as models trained exclusively on the GLI dataset generalized effectively to MET volumes. Among the tested configurations, the multiview model trained on combined axial and sagittal representations (R8S) achieved the strongest overall performance, underscoring the value of exploiting complementary anatomical perspectives for cross-modality synthesis.

Overall, the contributions of this work are twofold: (i) a systematic evaluation of architectural and training design choices for MRI synthesis, and (ii) the identification of strategies that balance reconstruction quality with computational efficiency. These findings provide a solid

basis for future research on multimodal medical image synthesis, suggesting that further exploration of advanced fusion strategies and integration with 3D refinement techniques could further enhance the clinical applicability of the framework.

6.2 Future Work

Several promising directions emerge from this work. A first avenue is the extension of the framework to full 3D generation. While current experiments were performed slice-wise to preserve efficiency, moving to volumetric models could improve spatial coherence and reduce slice-level artifacts, provided that computational cost can be contained. This would require careful architectural design to balance the trade-off between model complexity and training speed.

Another promising direction is the exploration of alternative generative backbones. Recent architectures such as Mamba and transformer-based models have shown remarkable performance in vision and medical imaging tasks. Their ability to capture long-range dependencies could enhance the modeling of cross-modality relationships in MRI synthesis, potentially surpassing the capabilities of GAN-based approaches.

Guiding generation with segmentation signals also remains an open challenge. The integration of segmentation, either by stabilizing the segmenter already used in this work or by introducing an auxiliary segmentation head to guide the feature space, could encourage anatomically meaningful reconstructions and improve the clinical reliability of the outputs.

Further progress may also be achieved by scaling to larger and more diverse datasets. While this year's challenge restricted training to the GLI and MET cohorts, datasets such as Meningioma could provide additional variability and improve generalization. In parallel, advanced preprocessing and filtering techniques could be applied to the BRATS dataset, which contains images of variable and sometimes low quality. More rigorous data curation would likely benefit both training stability and final model performance.

Overall, these directions highlight opportunities to improve both the methodological and practical aspects of cross-modality MRI synthesis, paving the way toward more robust and clinically useful generative models.

Appendix A

The Flaws of SSIM on Background Dominated Images

A.1 Introduction

The Structural Similarity Index (SSIM) is an image quality assessment metric introduced by Wang et al. [32] to overcome the limitations of traditional error-based measures such as Mean Squared Error (MSE) or Peak Signal-to-Noise Ratio (PSNR). While these metrics quantify pixel-wise differences, they often fail to capture perceptually important aspects of image quality, such as structural information, texture, and local contrast. Small shifts in intensity or slight blurring can dramatically affect MSE or PSNR, even when the visual appearance remains largely unchanged. SSIM was designed to address this shortcoming by explicitly modeling image degradation in terms of changes in structural information, considering three components: luminance, which captures the brightness differences; contrast, which evaluates variations in intensity; and structure, which assesses the correlation of local patterns between images.

By combining these components, SSIM provides a more perceptually meaningful measure of similarity between images, making it particularly useful in applications where visual fidelity is crucial. These include image compression, denoising, super-resolution, and image reconstruction. In medical imaging, for example, SSIM has been widely used to evaluate

MRI and CT reconstructions, as it better reflects clinically relevant features such as tissue boundaries and fine anatomical structures compared to purely pixel-wise metrics.

Furthermore, SSIM is computationally efficient and can be applied locally using sliding windows, which allows it to capture spatial variations in image quality. Its values range from -1 to 1, where higher values indicate greater similarity. This perceptually-oriented approach has made SSIM a standard tool for both natural and medical image quality assessment, complementing traditional metrics and providing insight into structural fidelity that MSE and PSNR cannot convey:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{A.1}$$

where:

- μ_x, μ_y are the local means of x and y,
- σ_x^2, σ_y^2 are the local variances,
- σ_{xy} is the local covariance between x and y,
- $C_1 = (K_1 L)^2$ and $C_2 = (K_2 L)^2$ are small constants to stabilize the division, where L is the dynamic range of the pixel values, and K_1, K_2 are small scalars (commonly $K_1 = 0.01$, $K_2 = 0.03$).

A.2 SSIM in medical imaging

The limitations of SSIM become particularly evident in the context of medical imaging. In MRI and similar modalities, a large portion of the volume often corresponds to non-relevant background, typically set to zero after deskulling, while relevant tissue occupies only a fraction of the voxels. A common strategy in the literature is to mask the background by explicitly setting it to zero before computing SSIM. However, this approach does not resolve the problem; rather, it exacerbates it. By artificially increasing the dominance of zero-valued voxels, the metric becomes biased: volumes with larger background regions naturally yield higher SSIM scores, independent of the quality of tissue reconstruction. While this does not

affect the relative ordering of scores within a single dataset, higher SSIM still indicates better reconstructions, it renders comparisons across samples or datasets unreliable. In such cases, SSIM is influenced not only by reconstruction accuracy but also by the relative size of the structures of interest, limiting its interpretability for cross-sample or cross-dataset evaluations.

A.3 Consequences

As shown in Table 5.1, the reported SSIM scores suggest that all models perform almost identically, with some differences on the order of 0.01. This phenomenon is a direct consequence of the previously discussed limitations of SSIM in medical imaging. Computing SSIM alone on masked images, even after setting the background to zero, does not alleviate the issue. Although SSIM ranges between -1 and 1 (commonly rescaled to -100 to 100 for visualization), much of this range is effectively unused due to the dominance of background voxels. Computed statistics on the GLI training set further illustrate this imbalance: only 17.61% of the voxels correspond to brain tissue, and just 1.08% to tumoral tissue. Controlled experiments also demonstrate the effect: generating a tumor region by randomly sampling voxel values within the same range as the original yields SSIM scores between 98.4 and 99.5, depending on the sample. These values are misleadingly high, despite the fact that the generated tissue is purely random, highlighting that SSIM alone on masked images can fail to reflect meaningful reconstruction quality in scenarios dominated by large background regions.

A.4 A Simple Solution

The solution to the background bias in SSIM is straightforward. Since SSIM, like MAE, is computed on a pixel- or voxel-wise basis, it is sufficient to exclude the values associated with background regions and aggregate the metric only over meaningful tissue voxels. By ignoring the background, the full dynamic range of the metric is restored, and comparisons across samples or datasets become meaningful and reliable. Implementations for modules that output the SSIM image are available, and a custom module was developed for this thesis using a `nn.Module`, enabling direct computation of voxel-wise SSIM maps within the training and evaluation pipeline.

This approach was employed during the training for loss calculation, as well as for the computation of both 2D and 3D initial evaluation scores. However, for the purpose of publication, the less accurate version of SSIM, which includes background voxels, was reported to maintain consistency with previously established evaluation conventions in the literature.

Appendix B

Challenge Paper Publication

The research activity conducted in the context of this thesis has led to the submission and acceptance of a short paper for the BraTS 2025 challenge, along with an oral presentation at MICCAI 2025. For the sake of completeness and to provide the reader with direct access to the core results and methodology, the paper is included in full in the following pages.

Omar Carpentiero, Kevin Marchesini, Costantino Grana, and Federico Bolelli.
"No More Slice Wars: Towards Harmonized Brain MRI Synthesis for the BraSyn Challenge."

No More Slice Wars: Towards Harmonized Brain MRI Synthesis for the BraSyn Challenge

Omar Carpentiero*, Kevin Marchesini*,
Costantino Grana, and Federico Bolelli ✉

University of Modena and Reggio Emilia, Italy
{*name.surname*}@unimore.it

Abstract. The synthesis of missing MRI modalities has emerged as a critical solution to address incomplete multi-parametric imaging in brain tumor diagnosis and treatment planning. While recent advances in generative models, especially GANs and diffusion-based approaches, have demonstrated promising results in cross-modality MRI generation, challenges remain in preserving anatomical fidelity and minimizing synthesis artifacts. In this work, we build upon the Hybrid Fusion GAN (HF-GAN) framework, introducing several enhancements aimed at improving synthesis quality and generalization across tumor types. Specifically, we incorporate z-score normalization, optimize network components for faster and more stable training, and extend the pipeline to support multi-view generation across various brain tumor categories, including gliomas, metastases, and meningiomas. Our approach focuses on refining 2D slice-based generation to ensure intra-slice coherence and reduce intensity inconsistencies, ultimately supporting more accurate and robust tumor segmentation in scenarios with missing imaging modalities. Our source code is available at https://github.com/AImageLab-zip/BraSyn25.

Keywords: Image Synthesis · MRI · Multimodal · BraTS · Brain Tumor Imaging · GANs · Medical Imaging

1 Introduction

In recent years, deep learning has significantly advanced medical image analysis, particularly for tasks such as segmentation and classification across various imaging modalities [7, 8, 18, 21, 37, 39]. Moreover, generative models have also emerged as a powerful technique to produce fully synthetic datasets or expand existing ones, thereby increasing data variability, mitigating class imbalance, and supporting the development of more robust and generalizable deep learning models [9, 15, 22, 31, 36]. In this context, while certain applications involve distinct anatomical structures that can be accurately analyzed using a single image modality [6, 27], many clinical scenarios require multi-modal imaging to

*Equal contribution. Authors are allowed to list their name first on their CVs.
✉ Corresponding authors: federico.bolelli@unimore.it.

effectively capture complex anatomical and pathological variations, lesion heterogeneity, and enhance tissue contrast [35]. Among the latter, the diagnosis and monitoring of brain tumors rely on multi-parametric Magnetic Resonance Imaging (MRI), considered the standard due to its superior capability in delineating tumor boundaries, quantifying tumor volumes, and guiding therapeutic decisions [3, 6]. Specifically, clinical practice typically employs four complementary MRI sequences: T1-weighted images (T1), T1-weighted images with contrast enhancement (T1c), T2-weighted images (T2), and Fluid-Attenuated Inversion Recovery (FLAIR). Each modality highlights distinct tumor sub-regions, facilitating comprehensive analysis. However, acquiring all four MRI modalities is not always feasible in clinical practice due to constraints such as differing acquisition protocols, scanner limitations, or patient-specific issues like allergies to contrast agents (in the case of T1c modality). This absence of modalities, which can compromise the accuracy of diagnostic tasks, in particular tumor segmentation, has motivated extensive research into synthesizing missing MRI modalities from available ones. Recently, generative approaches spanning from Generative Adversarial Networks (GANs) to diffusion models, have been proposed to preserve the informative characteristics of each modality [20].

GAN-based modality synthesis. GANs have been widely used for cross-modality MRI translation, yielding promising results in producing realistic missing scans. Early works focused on paired image-to-image translation, adapting state-of-the-art general frameworks, such as Pix2Pix [5,12,38,43], to the MRI domain. Several other works demonstrated that GANs can produce anatomically plausible MRI sequences, if integrated with specific losses, such as a cycle-consistency loss [10, 26], an edge-aware loss [43], a frequency loss [5], or masked versions of common losses to penalize more the errors in tumor regions [5]. Authors proved that synthetic modalities produced by GAN methods retain critical tumor information, leading to improved segmentation performance [32, 40].

Diffusion models. Diffusion models have recently emerged as a strong alternative to GANs for cross-modality MRI synthesis, offering higher fidelity via explicit likelihood modeling and gradual denoising [33]. Approaches include latent-space diffusion [19, 44], which conditions on compressed representations to save memory, and modality-masked diffusion, like M2DN [28], which treats missing channels as noise for inpainting. The second and third place teams in the BraSyn 2024 challenge [14, 16] used wavelet-domain diffusion, showing that denoising in wavelet space improves full-volume reconstruction and reduces 3D artifacts.

Hybrid and Multi-Stage Methods. Recent work explored hybrid architectures and cascades to improve synthesis quality [17, 20, 33, 34]. Hybrid Fusion GAN (HF-GAN) [20], the basis of our model, uses a hybrid generator with attention-based fusion to integrate modality-specific features, which are then mapped to the target sequence via a modality infuser. The BraSyn 2024 winner [20] extended HF-GAN with an intensity encoder for global context and a 3D Refiner to reduce artifacts and improve tumor segmentation.

In this work, we refine the HF-GAN framework by adding z-score normalization, optimizing network components, and adapting the training pipeline for

multiview generation of tumors such as gliomas, metastases, and meningiomas. We focus on improving 2D generation to produce coherent slices and reduce intra-slice artifacts like intensity discrepancies.

2 Method

2.1 Preliminaries

We adopted HF-GAN [20] as our baseline model, using a lighter 2D pipeline to improve training and inference performance. The framework consists of a generator that synthesizes 2D brain slices from preprocessed 3D volumes and a discriminator for GAN-style adversarial learning. To enable the synthesis using a unified network independently from the missing modality scenario, the generator is composed of 4 modality-specific late-fusion encoders (one for each modality), an early fusion encoder that takes as input all the available modalities, a channel attention feature fusion module, a modality infuser, and a decoder. The encoder-decoder architecture is based on a U-Net structure.

The late fusion encoders, composed of residual convolutional blocks, with SiLU activation [13] and group normalization [42], are used for modality-specific feature extraction. The early-fusion encoder, architecturally identical to the specific encoders, accepts a stacked 4-channel image to extract complementary information from all the modalities, masking the missing ones.
Then, a feature fusion module integrates global and modality-specific information, with channel attention using global average pooling and softmax. A modality-infuser, made of Transformer blocks [41], infuses information about the missing modality into the hidden space. The decoder expands the feature maps using upsampling blocks characterized by a nearest neighbor interpolation layer followed by a 3x3 2D-convolution layer to smooth the image.

As for the standard U-Net architecture, the model incorporates skip connections between corresponding layers of the encoder and decoder to preserve spatial information and facilitate gradient flow.

2.2 Dataset & Preprocessing

Data. The Brain Tumor Segmentation (BraTS) challenge series has been organized annually since 2012, providing standardized multimodal MRI datasets and benchmarks that have driven progress in AI-based brain tumor analysis [2, 25, 29]. The BraSyn-2025 dataset is based on datasets containing different tumor cases, i.e., the BraTS-GLI 2023 (Glioma, GLI), BraTS-METS 2023 (Metastasis, MET) [30], and BraTS-MEN (Meningioma, MEN) [24]. The resulting dataset contains a retrospective collection of brain tumor mpMRI (multiparametric MRI) scans acquired from multiple institutions under standard clinical conditions but with different equipment and imaging protocols, resulting in a vastly heterogeneous image quality reflecting diverse clinical practice across different institutions. The training set is composed of 1,251 complete sequences from

Table 1: MRI intensity values of the training set before and after applying the 99.5th percentile clipping and normalization strategies.

Value	Clipp.	Norm.	T1c	T1n	T2f	T2w
Max.	✗	✗	2,120,538	155,724	612,368	4,563,634
Max.	✓	✗	8,664	7,315	8,842	8,233
Avg.	✓	✗	1,066.34	781.22	510.99	673.44
Std.	✓	✗	1,301.70	944.34	769.42	804.39
Min.	✓	✓	-0.8192	-0.8273	-0.6641	-0.8372
Max.	✓	✓	5.8367	6.9189	10.8277	9.3979

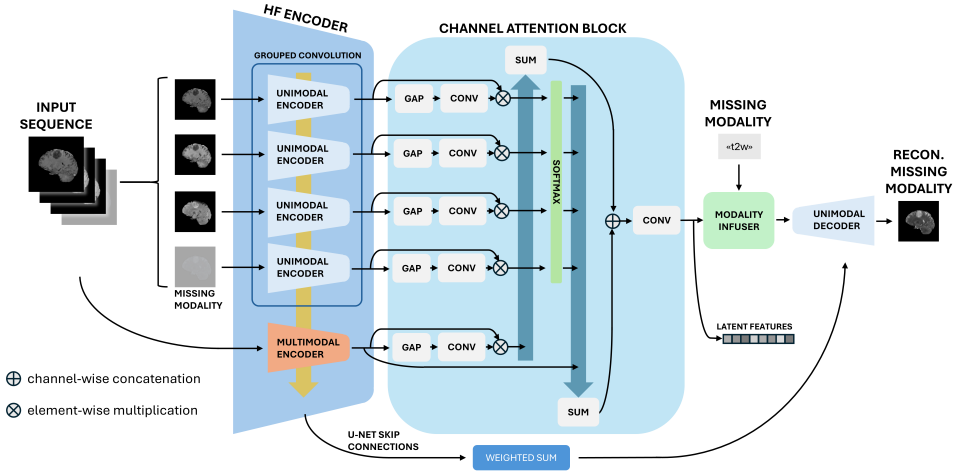
the BraTS-GLI dataset and 238 complete sequences from the BraTS-METS. All samples are annotated with a segmentation mask with 3 tumor structures: Enhancing Tumor (ET), Non-Enhancing Tumor Core (NETC), and peritumoral EDema (ED) [4]. The evaluation is always performed on an aggregation of these classes, namely Whole Tumor (WT) = ET + NETC + ED and Tumor Core (TC) = NETC + ET, as well as on the Enhancing Tumor (ET) region individually. The validation set is composed of 219 complete sequences from the BraTS-GLI dataset and 31 complete sequences from the BraTS-METS. Finally, the test set is composed of 219 complete sequences from the BraTS-GLI dataset, 59 complete sequences from the BraTS-METS, and 283 sequences from BraTS-MEN.

In contrast to previous editions, this year's challenge subtask introduces MET cases into both the training and validation sets and includes GLI, MET and MEN cases in the test set. This design aims to evaluate the models' ability to generalize across different tumor types. During the validation and test phases, ground-truth segmentation masks are not provided, and one of the four imaging modalities is randomly withheld ("modality dropout") for each subject.

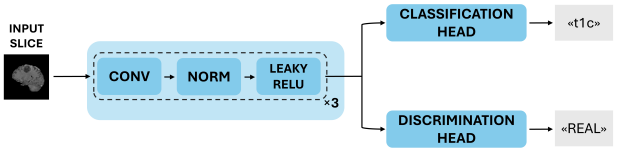
Data Preprocessing. Our contribution begins by modifying the original HF-GAN data preprocessing pipeline. Since MRI intensities are unbounded and often contain extreme outliers, we first applied intensity clipping at the 99.5th percentile after setting all negative values to zero and excluding zero-valued background pixels from the calculation. This step helps mitigate the influence of outlier voxels while preserving the meaningful dynamic range of the brain tissue. The original maximum values across modalities in the training before and after the clipping are reported in Tab. 1.

Moreover, instead of linearly projecting the data into the $[-1, 1]$ range as done in the original work [20], we adopt *dataset-wise* z*-score normalization*, computing the global mean (Avg.) and standard deviation (Std.) across all training volumes from both the GLI and MET datasets. These statistics are computed after applying a 99.5th percentile intensity clipping and are reported in Tab. 1, along with the resulting voxel intensity ranges post-normalization.

To maintain compatibility with the original HF-GAN framework, we standardized background voxel values across all modalities. Specifically, all background voxels (i.e., those originally equal to zero) were reassigned a constant value of -1, which also serves as the placeholder for masked input slices.



(a) Generator



(b) Discriminator

Fig. 1: Architecture of the proposed model.

To adapt our 2D framework to the dataset's 3D nature, we extracted axial, sagittal, and coronal slices, discarding those with fewer than 2,000 foreground pixels per modality ($< 3.47\%$ brain tissue). Sagittal and coronal slices were symmetrically padded to 240×240; padding was precomputed for training and applied on-the-fly at inference.

2.3 The Proposed Solution

As a GAN-based architecture, our framework consists of a Generator and a Discriminator, with the addition of a 2D Segmenter to guide the generation process. The generator receives three available modalities and a masked placeholder to reconstruct the missing one; this synthetic image is then combined with the originals to form a four-channel input for the tumor segmenter, which performs tumor segmentation. The resulting segmentation is used to compute a task-specific loss to enhance segmentation metrics on the reconstructed volumes. Following the typical GAN setup, the reconstructed modality is also passed to the Discriminator, which distinguishes between real and generated images, but also classifies the modality type to enforce modality-specific feature learning. The details regarding each module are reported below.

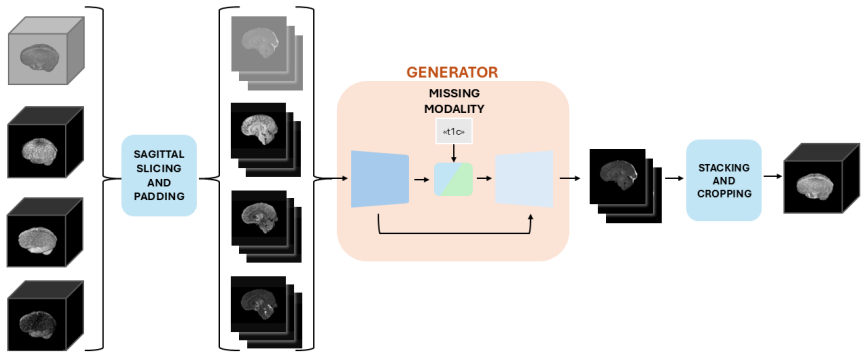


Fig. 2: Full inference pipeline. The input is split into slices, processed by the generator, then stacked to form the 3D volume. Padding and cropping are applied only to non-square sagittal slices.

Generator. Our model incorporates four modality-specific encoders and a shared fusion encoder. Each encoder consists of five downsampling stages with channel dimensions of $[64, 128, 256, 512, 640]$. Each stage includes a convolutional residual block, composed of normalization, SiLU activation, and a 3×3 convolution (stride 1, padding 1), followed by a downsampling block implemented as a 3×3 convolution with stride 2 and no padding. The fusion encoder is implemented using grouped convolutions to parallelize modality processing, replacing the previous sequential approach. Feature fusion, performed by the channel attention module is followed by a FlashAttention-compatible multi-head attention module, composed of four stages, replacing the previous custom implementation. This module is responsible for infusing modality information into the hidden space and can optionally incorporate view-specific information for multi-view generation tasks. Due to the use of z-score normalization, which preserves intensity structure while improving robustness to inter-slice variation, the intensity encoding modules were removed. The decoder mirrors the encoder with five upsampling stages, each comprising a convolutional residual block (as above) and an upsampling block using nearest-neighbor interpolation followed by a 3×3 convolution. Skip connections link each encoder stage to its corresponding decoder stage, merged by a weighted sum of the feature vectors: encoders corresponding to available modalities share the contribution equally, so that the total sum of coefficients is 1; encoders of missing modalities contribute 0, while the early-fusion encoder always contributes with full weight (1). The complete architecture of the generator is represented in Fig. 1a.

Segmenter. To guide the generation process and improve segmentation accuracy, we trained a lightweight 2D segmentation model to segment brain tumor subregions. Specifically, we adopted a nnU-Net-based architecture with four downsampling-upsampling stages and feature dimensions of [32, 64, 128, 256], employing SiLU as the activation function.

The model was trained on a subset of the training data consisting exclusively of slices containing at least 0.1% tumor tissue, thereby focusing the learning

process on informative regions. To enhance generalizability, the segmenter was trained across all anatomical views (axial, coronal, and sagittal) and tumor types (GLI, MET), with the view and tumor type information incorporated into the input via one-hot encoding. This design enabled a compact yet effective model comprising only 1.79 million parameters. The resulting model achieved robust performance, with validation Dice scores of 0.74, 0.80, and 0.82 on the NETC, ED, and ET classes, respectively. The model was trained using the Focal Tversky loss [1], due to its ability to provide better performance in contexts where the different classes are highly imbalanced. The segmenter was independently trained, and then frozen during the training of the generation pipeline.

Discriminator. The discriminator, shown in Fig. 1b, is based on the PatchGAN architecture [11], which classifies local image patches rather than the whole image. It uses three downsampling layers and starts with 32 filters. Each block includes a 4×4 convolution with stride 2 (except the last, which uses stride 1), group normalization, and LeakyReLU activations. The final convolution outputs a single-channel map for real/fake classification. An auxiliary classifier branch predicts 4 class logits per image, one for each modality.

Final Generation. To correctly reconstruct a volume from 2D slices, the generator processes batches of slices, and then the outputs stacked (Fig. 2). After stacking, the image is denormalized, and background pixels are set to zero. Padding is removed for 3D volumes generated from the sagittal view. The entire process, including data loading, preprocessing, and saving, requires, on average, 13.4 seconds per sample on an RTX5000 16GB.

Table 2: List of loss components.

Term	Role
$\mathcal{L}_{\text{recon}}$	Enforces pixel-level accuracy.
$\mathcal{L}_{\text{adv}}$	Encourages realistic outputs.
$\mathcal{L}_{\text{class}}$	Encourages modality specific features.
$\mathcal{L}_{\text{feat}}$	Promotes consistency in the latent space.
$\mathcal{L}_{\text{cycle}}$	Promotes cycle consistency.
$\mathcal{L}_{\text{SSIM}}$	Improves structural fidelity.
$\mathcal{L}_{\text{Tversky}}$	Promotes better downstream segmentation.

2.4 The loss

Our model leverages a combination of different loss functions used to guide the architecture toward learning more robust and balanced representations. The overall loss function is:

$$\mathcal{L}_{\text{total}} = 10 \cdot \mathcal{L}_{\text{recon}} + 0.25 \cdot \mathcal{L}_{\text{adv}} + 0.25 \cdot \mathcal{L}_{\text{class}} + \mathcal{L}_{\text{feat}} + \mathcal{L}_{\text{cycle}} + 5 \cdot \mathcal{L}_{\text{SSIM}} + 5 \cdot \mathcal{L}_{\text{Tversky}}$$

Each loss targets a specific aspect of the task: a summary is reported in Tab. 2 and detailed in the following. The reconstruction loss $\mathcal{L}_{\text{recon}}$, based on the mean absolute error (MAE), serves as the dominant loss term, prioritizing pixel fidelity between the synthesized and ground-truth images. To enhance perceptual quality, we incorporate Structural Similarity Index (SSIM) losses $\mathcal{L}_{\text{SSIM}}$, computed as 1-SSIM on a per-pixel basis, and aggregated only over the regions of interest. We implement two variants of this loss: one that considers the entire volume, and another that produces two separate terms, one for healthy brain tissue and one for tumor tissue, which are equally weighted during training. The

Table 3: Characterization of settings employed in different experiments. Unmentioned losses are always present.

RunID	Losses					Train		View	
	$\mathcal{L}_{\text{cycle}}$	$\mathcal{L}_{\text{feat}}$	$\mathcal{L}_{\text{SSIM\_whole}}$	$\mathcal{L}_{\text{SSIM\_dual}}$	$\mathcal{L}_{\text{Tversky}}$	GLI	MET	Axial	Sagit.
R1	✓	✓	✗	✗	✗	✓	✗	✓	✗
R2	✓	✓	✗	✓	✗	✓	✗	✓	✗
R3	✓	✓	✗	✓	✗	✓	✗	✓	✗
R4	✓	✓	✓	✗	✗	✓	✗	✓	✗
R5	✗	✗	✗	✗	✗	✓	✗	✓	✗
R6	✓	✓	✗	✓	✓	✓	✗	✓	✗
R7	✓	✓	✗	✓	✓	✓	✗	✗	✓
R8A	✓	✓	✗	✓	✗	✓	✓	✓	✓
R8S	✓	✓	✗	✓	✗	✓	✓	✓	✓
R9	✓	✓	✗	✓	✓	✓	✗	✓	✓
R10	✓	✓	✗	✓	✗	✓	✓	✓	✓

adversarial loss $\mathcal{L}_{\text{adv}}$ is used to encourage realism in the generated outputs and to suppress common synthesis artifacts. Cycle consistency is enforced through a two-step generation process. In the first step, the generator synthesizes a missing modality using ground truth modalities. In the second step, this previously generated output replaces the corresponding placeholder, one of the available modalities is randomly masked (selected from a uniform distribution, so each maskable modality has equal probability of being chosen), and the generator is applied again using the two remaining ground truth modalities along with the generated one. The cycle loss $\mathcal{L}_{\text{cycle}}$ is defined as the MAE between the output of the second generation and the ground truth of the corresponding modality. To further promote consistency at the representational level, a feature loss $\mathcal{L}_{\text{feat}}$ is included, defined as the cosine similarity between the hidden features of the bottleneck layer extracted during the first and second generation passes. Finally, the Tversky-focal loss [1] $\mathcal{L}_{\text{Tversky}}$ is introduced to guide the generator to produce outputs that are easier to segment, particularly in the presence of class imbalance. As in the default HF-GAN framework, forward propagation is performed twice, for cycle consistency, to generate each modality, resulting in a total of 8 forward passes on the same network (not along different paths). Gradients from these multiple forwards are aggregated by combining the corresponding losses into a single scalar and performing backpropagation once.

3 Experiments & Results

3.1 Assessment Metrics

Different algorithms are assessed using a combination of image quality and segmentation metrics. The evaluation relies on multiple quantitative metrics.

Structural Similarity Index Measure (SSIM), as an image quality metric, it is employed to measure the realism of the reconstructed volume. It is indepen-

dently computed on the Whole Tumor (WT) area and on the Healthy Tissue (HT) part of the brain, resulting in two scores for each test subject.

Dice score and Normalized Surface Distance (NSD) are used to evaluate the effect of reconstructed volumes on segmentation masks and boundaries. Metrics are computed for three tumor structures: Enhancing Tumor (ET), Tumor Core (TC), and Whole Tumor (WT). Segmentation pseudo-labels for the validation set are generated using state-of-the-art algorithms from the BraTS python package [23], then compared to segmentations from sequences where one modality was masked and reconstructed.

3.2 Results

All experimental results from our analysis, on the validation set, are summarized in Tab. 4, where each run is identified by a unique identifier (e.g., R1, R2) and corresponds to a specific configuration of the generator's training process. The details of each configuration are provided in Tab. 3. All runs are trained for an equivalent number of steps, corresponding to 10 epochs on the GLI dataset, with the exception of R6 and R7. Notably, R1, R2, R4, and R6 differ from the others in how the input modalities are provided to the network, since the number of available modalities varies between 1, 2, or 3, whereas in the other runs it is fixed to 3. Runs R6 and R7 resume training from the checkpoint of R3 and are further optimized for 3 additional epochs, adding the Tversky loss. Runs R8A, R8S, and R10 share the same model, which is trained on multiple views and on the combined GLI+MET dataset, but are evaluated under three different settings: R8A generates axial views, R8S generates sagittal views, and R10 combines both views by averaging the outputs. Similarly, R9 aggregates the axial predictions from R6 and the sagittal predictions from R7. All models were evaluated on the combined GLI+MET validation set to assess their generalization capabilities across different tumor types. Table XY reports the results on the hidden test set, provided by the challenge organizers. Additionally, Fig. 3 reports a comparison of the original and reconstructed slices sampled from all four modalities of random patients. The figure shows that although no 3D refinement was used, the slices show little to no striping artifacts, which are typical of 2D generation approaches.

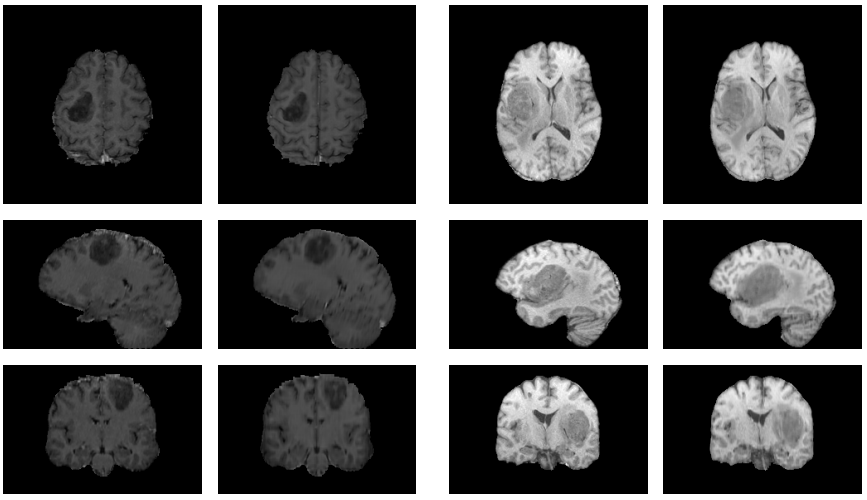
3.3 Discussion

All of our experiments demonstrate strong MRI synthesis performance and good generalization to unseen data. In particular, models trained exclusively on the GLI dataset still perform reasonably well when tasked with generating MET volumes, highlighting the robustness of the framework. R1, even without the SSIM loss term, achieves strong perceptual quality. Subsequent runs focus on incremental improvements. Introducing the SSIM loss on R2, split into two terms for healthy and tumor tissue, improves structural similarity metrics. In contrast, R3 applies the SSIM loss over the entire volume without differentiating between tissue types, which leads to worse performance, highlighting the importance of

Table 4: Experimental results obtained combining different losses, leveraging different training sets and 2D views. The Run IDs are explained in Tab. 3.

Metric	Val.Set	Class	Run ID										
			R1	**R2**	**R3**	**R4**	**R5**	**R6**	**R7**	**R8A**	**R8S**	**R9**	**R10**
SSIM	ALL	WT	99.73	99.77	99.76	99.74	99.71	99.76	99.75	99.76	99.77	99.75	99.76
		HT	93.31	94.07	94.05	93.82	93.15	94.09	93.91	93.79	93.89	94.07	93.79
	GLI	WT	99.72	99.76	99.76	99.74	99.70	99.75	99.74	99.75	99.75	99.74	99.75
		HT	93.69	94.47	94.44	94.22	93.52	94.49	94.28	94.10	94.20	94.47	94.11
	MET	WT	99.79	99.81	99.82	99.80	99.78	99.82	99.82	99.84	99.84	99.82	99.84
		HT	90.62	91.23	91.23	90.94	90.52	91.29	91.27	91.59	91.73	91.27	91.59
DICE	ALL	ET	53.62	57.13	54.16	54.16	52.43	57.02	57.70	55.98	57.86	57.00	56.42
		TC	73.86	76.49	75.48	75.48	73.89	76.88	75.11	75.28	77.55	76.88	75.28
		WT	71.74	73.63	71.89	71.89	70.41	74.07	74.97	73.57	74.45	74.07	73.58
	GLI	ET	52.94	56.67	53.82	53.82	51.87	56.58	57.40	55.90	57.14	56.58	55.90
		TC	72.28	75.27	74.24	74.24	72.55	75.59	73.55	73.88	76.16	75.58	73.85
		WT	69.75	71.91	70.01	70.01	68.47	72.25	73.26	71.79	72.50	72.25	71.79
	MET	ET	58.44	60.36	56.51	56.51	56.40	60.11	59.77	56.52	62.92	59.98	60.09
		TC	85.03	85.06	84.28	84.28	83.35	86.05	86.14	85.17	87.38	86.00	85.33
		WT	85.78	85.83	85.15	85.15	84.16	86.93	87.03	86.13	88.26	86.92	86.18
NSD	ALL	ET	54.06	56.48	57.66	54.38	52.55	56.96	58.03	56.81	57.92	56.96	56.81
		TC	63.27	66.80	68.55	65.83	63.06	67.46	65.73	65.79	68.30	67.48	65.79
		WT	52.80	55.54	56.51	53.02	51.32	56.11	57.30	55.41	56.64	56.11	55.41
	GLI	ET	52.68	55.41	56.86	53.43	51.34	55.84	57.08	55.51	56.40	55.85	55.51
		TC	61.11	65.04	66.94	64.22	61.26	65.32	63.42	63.65	66.25	65.35	63.65
		WT	49.19	52.17	53.20	49.60	47.90	52.35	53.79	51.78	52.88	52.36	51.78
	MET	ET	63.82	64.05	63.25	61.12	61.10	64.87	64.74	66.00	68.64	64.81	66.00
		TC	78.47	79.24	79.88	77.22	75.73	82.62	82.08	80.89	82.82	82.58	80.89
		WT	78.30	79.34	79.91	77.15	75.48	82.65	82.13	81.12	83.23	82.64	81.12

spatially targeted perceptual losses. The original framework supports cases with 1, 2, or 3 missing modalities. By always providing three known modalities as input, convergence is faster, but final scores showed minimal variation, as evidenced by the comparison between R2, which used variable input modalities, and R3, where three known modalities were always provided. We also experiment with removing the cycle consistency and feature losses on R5. This leads to performance degradation, confirming their contribution to training stability and reconstruction quality. However, the inclusion of cycle consistency roughly doubles training time and memory utilization. Incorporating the Tversky loss, intended to improve segmentation quality, did not yield significant gains. Similarly, using a simple mean of multiview predictions did not enhance performance. Our best-performing model, R8S, is trained using both axial and sagittal views and does not include the Tversky loss. It produces slices in the sagittal plane and benefits from the increased diversity of input representations.



T1c - BraTS-GLI-01724-000 **T1 - BraTS-GLI-00333-000**

Fig. 3: Comparisons of real (left) and reconstructed (right) for two different patients sampled from the GLI validation dataset.

4 Conclusion

We proposed an enhanced 2D MRI synthesis framework based on HF-GAN, introducing z-score normalization, architectural optimizations, and multi-view training to improve synthesis quality and generalization across tumor types. Our method achieves high structural fidelity and supports robust tumor segmentation, even with missing modalities. Results show strong cross-dataset performance, confirming the effectiveness of our design.

Acknowledgments. This work was supported by the University of Modena and Reggio Emilia and Fondazione di Modena through the "Fondo di Ateneo per la Ricerca - FAR 2024" (CUP E93C24002080007) and FARD-2024.

Disclosure of Interests. The authors have no conflicts of interest to declare.

References

1. Abraham, N., Khan, N.M.: A Novel Focal Tversky loss function with improved Attention U-Net for lesion segmentation. In: IEEE 16th International Symposium on Biomedical Imaging (2019)
2. Baid, U., Ghodasara, S., Mohan, S., Bilello, M., Calabrese, E., Colak, E., Farahani, K., Kalpathy-Cramer, J., Kitamura, F., Pati, S., et al.: The RSNA-ASNR-MICCAI BraTS 2021 Benchmark on Brain Tumor Segmentation and Radiogenomic Classification. arXiv preprint arXiv:2107.02314 (2021)
3. Baid, U., et al. (eds.): Brain Tumor Segmentation, and Cross-Modality Domain Adaptation for Medical Image Segmentation, Lecture Notes in Computer Science, vol. 14669 (2023)

4. Bakas, Spyridon and Akbari, Hamed and Sotiras, Aristeidis and Bilello, Michel and Rozycki, Martin and Kirby, Justin S and Freymann, John B and Farahani, Keyvan and Davatzikos, Christos: Advancing The Cancer Genome Atlas glioma MRI collections with expert segmentation labels and radiomic features. Scientific data **4**(1) (2017)
5. Baltruschat, I.M., Janbakhshi, P., Lenga, M.: BraSyn 2023 Challenge: Missing MRI Synthesis and the Effect of Different Learning Objectives. In: International Challenge on Cross-Modality Domain Adaptation for Medical Image Segmentation (2023)
6. Bolelli, F., Marchesini, K., van Nistelrooij, N., Lumetti, L., Pipoli, V., Ficarra, E., Vinayahalingam, S., Grana, C.: Segmenting Maxillofacial Structures in CBCT Volumes. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2025)
7. Bontempo, G., Bolelli, F., Porrello, A., Calderara, S., Ficarra, E.: A Graph-Based Multi-Scale Approach with Knowledge Distillation for WSI Classification. IEEE Transactions on Medical Imaging (2023)
8. Cipriano, M., Allegretti, S., Bolelli, F., Di Bartolomeo, M., Pollastri, F., Pellacani, A., Minafra, P., Anesi, A., Grana, C.: Deep Segmentation of the Mandibular Canal: a New 3D Annotated Dataset of CBCT Volumes. IEEE Access **10** (2022)
9. Cipriano, M., Allegretti, S., Bolelli, F., Pollastri, F., Grana, C.: Improving Segmentation of the Inferior Alveolar Nerve through Deep Label Propagation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)
10. Dar, S.U., Yurt, M., Karacan, L., Erdem, A., Erdem, E., Cukur, T.: Image Synthesis in Multi-Contrast MRI With Conditional Generative Adversarial Networks. IEEE Transactions on Medical Imaging **38**(10) (2019)
11. Demir, U., Unal, G.: Patch-Based Image Inpainting with Generative Adversarial Networks. arXiv preprint arXiv:1803.07422 (2018)
12. Eker, A.G., Pehlivanoğlu, M.K., Duru, N., Duendar, T.T.: BrainPixGAN: Generating intraoperative MRI images with mask-based generative networks. Engineering Science and Technology, an International Journal **58** (2024)
13. Elfwing, S., Uchibe, E., Doya, K.: Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. Neural networks (2018)
14. Ferreira, A., Luijten, G., Puladi, B., Kleesiek, J., Alves, V., Egger, J.: Brain Tumour Removing and Missing Modality Generation using 3D WDM. arXiv preprint arXiv:2411.04630 (2024)
15. Frid-Adar, M., Diamant, I., Klang, E., Amitai, M., Goldberger, J., Greenspan, H.: GAN-based Synthetic Medical Image Augmentation for Increased CNN Performance in Liver Lesion Classification. Neurocomputing (2018)
16. Friedrich, P., Durrer, A., Wolleb, J., Cattin, P.C.: cWDM: Conditional Wavelet Diffusion Models for Cross-Modality 3D Medical Image Synthesis. arXiv preprint arXiv:2411.17203 (2024)
17. Hamghalam, M., Lei, B., Wang, T.: High Tissue Contrast MRI Synthesis Using Multi-Stage Attention-GAN for Segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34 (2020)
18. Huang, S.C., Pareek, A., Jensen, M., Lungren, M.P., Yeung, S., Chaudhari, A.S.: Self-supervised learning for medical image classification: a systematic review and implementation guidelines. NPJ Digital Medicine **6**(1) (2023)
19. Jiang, L., Mao, Y., Wang, X., Chen, X., Li, C.: CoLa-Diff: Conditional Latent Diffusion Model for Multi-modal MRI Synthesis. In: International Conference on Medical Image Computing and Computer-Assisted Intervention (2023)

20. Jihoon, C., Jonghye, W., Jinah, P.: A Unified Framework for Synthesizing Multi-sequence Brain MRI via Hybrid Fusion. Medical Image Analysis (2024)
21. Karargyris, A., Umeton, R., Sheller, M.J., Aristizabal, A., George, J., Wuest, A., Pati, S., Kassem, H., Zenk, M., Baid, U., et al.: Federated benchmarking of medical artificial intelligence with MedPerf. Nature machine intelligence **5**(7) (2023)
22. Kazeminia, S., et al.: GANs for Medical Image Analysis. AIME **109** (2020)
23. Kofler, F., Rosier, M., Astaraki, M., Baid, U., Möller, H., Buchner, J.A., Steinbauer, F., Oswald, E., de la Rosa, E., Ezhov, I., et al.: BraTS orchestrator : Democratizing and Disseminating state-of-the-art brain tumor image analysis. arXiv preprint arXiv:2506.13807 (2025)
24. LaBella, D., Adewole, M., Alonso-Basanta, M., Altes, T., Anwar, S.M., Baid, U., Bergquist, T., Bhalerao, R., Chen, S., Chung, V., et al.: The ASNR-MICCAI Brain Tumor Segmentation (BraTS) Challenge 2023: Intracranial Meningioma. arXiv preprint arXiv:2305.07642 (2023)
25. Li, H.B., Conte, G.M., Hu, Q., Anwar, S.M., Kofler, F., Ezhov, I., van Leemput, K., Piraud, M., Diaz, M., Cole, B., et al.: The Brain Tumor Segmentation (BraTS) Challenge 2023: Brain MR Image Synthesis for Tumor Segmentation (BraSyn). ArXiv (2024)
26. Li, Hongwei and Paetzold, Johannes C and Sekuboyina, Anjany and Kofler, Florian and Zhang, Jianguo and Kirschke, Jan S and Wiestler, Benedikt and Menze, Bjoern: DiamondGAN: Unified Multi-modal Generative Adversarial Networks for MRI Sequences Synthesis. In: International Conference on Medical Image Computing and Computer-Assisted Intervention (2019)
27. Lumetti, L., Capitani, G., Ficarra, E., Grana, C., Calderara, S., Porrello, A., Bolelli, F.: U-Net Transplant: The Role of Pre-training for Model Merging in 3D Medical Segmentation. In: 28th International Conference on Medical Image Computing and Computer Assisted Intervention (2025)
28. Meng, X., Sun, K., Xu, J., He, X., Shen, D.: Multi-Modal Modality-Masked Diffusion Network for Brain MRI Synthesis With Random Modality Missing. IEEE Transactions on Medical Imaging **43**(7) (2024)
29. Menze, B.H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., et al.: The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). IEEE Transactions on Medical Imaging **34**(10) (2014)
30. Moawad, A.W., Janas, A., Baid, U., Ramakrishnan, D., Saluja, R., Ashraf, N., Maleki, N., Jekel, L., Yordanov, N., Fehringer, P., et al.: The Brain Tumor Segmentation - Metastases (BraTS-METS) Challenge 2023: Brain Metastasis Segmentation on Pre-treatment MRI. ArXiv (2024)
31. Morelli, N., Marchesini, K., Lumetti, L., Santi, D., Grana, C., Bolelli, F.: Enhancing Testicular Ultrasound Image Classification Through Synthetic Data and Pretraining Strategies. In: Image Analysis and Processing – ICIAP 2025 (2025)
32. Osuala, R., Joshi, S., Tsirikoglou, A., Garrucho, L., Pinaya, W.H., Diaz, O., Lekadir, K.: Pre- to Post-Contrast Breast MRI Synthesis for Enhanced Tumour Segmentation. In: Medical Imaging 2024: Image Processing. vol. 12926 (2024)
33. Özbey, M., Dalmaz, O., Dar, S.U., Bedel, H.A., Özturk, Ş., Güngör, A., Cukur, T.: Unsupervised Medical Image Translation With Adversarial Diffusion Models. IEEE Transactions on Medical Imaging **42**(12) (2023)
34. Pan, S., Eidex, Z., Safari, M., Qiu, R., Yang, X.: Cycle-guided Denoising Diffusion Probability Model for 3D Cross-modality MRI Synthesis. In: Medical Imaging 2025: Clinical and Biomedical Imaging. vol. 13410 (2025)

Bibliography

[1] BraTS Challenge. BraTS-Lighthouse 2025 Challenge, 2025. URL `https://www.synapse.org/Synapse:syn64153130/wiki/631053`.

[2] Vittorio Pipoli, Alessia Saporita, Kevin Marchesini, Costantino Grana, Elisa Ficarra, Federico Bolelli, et al. IM-Fuse: A Mamba-based Fusion Block for Brain Tumor Segmentation with Incomplete Modalities. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2025*. 2025.

[3] Hong Liu, Dong Wei, Donghuan Lu, Jinghan Sun, Liansheng Wang, and Yefeng Zheng. M3AE: Multimodal Representation Learning for Brain Tumor Segmentation with Missing Modalities. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 1657–1665, 2023.

[4] Yao Zhang, Nanjun He, Jiawei Yang, Yuexiang Li, Dong Wei, Yawen Huang, Yang Zhang, Zhiqiang He, and Yefeng Zheng. mmFormer: Multimodal Medical Transformer for Incomplete Multimodal Learning of Brain Tumor Segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 107–117. Springer, 2022.

[5] Tongxue Zhou, Stéphane Canu, Pierre Vera, and Su Ruan. Feature-enhanced generation and multi-modality fusion based deep neural network for brain tumor segmentation with missing MR modalities. *Neurocomputing*, 466:102–112, 2021.

[6] Florian Kofler, Marcel Rosier, Mehdi Astaraki, Ujjwal Baid, Hendrik Möller, Josef A Buchner, Felix Steinbauer, Eva Oswald, Ezequiel de la Rosa, Ivan Ezhov, et al. BraTS orchestrator : Democratizing and Disseminating state-of-the-art brain tumor image analysis. *arXiv preprint arXiv:2506.13807*, 2025.

[7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. Convolutional Networks for Biomedical Image Segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[8] Erena Siyoum Biratu, Friedhelm Schwenker, Taye Girma Debelee, Samuel Rahimeto Kebede, Worku Gachena Negera, and Hasset Tamirat Molla. Enhanced region growing for brain tumor mr image segmentation. *Journal of Imaging*, 7(2):22, 2021.

[9] Anil Kumar Mandle, Satya Prakash Sahu, and Govind Gupta. Brain Tumor Segmentation and Classification in MRI using Clustering and Kernel-Based SVM. *Biomedical and Pharmacology Journal*, 15(2):699–716, 2022.

[10] M Bach Cuadra, Mathieu De Craene, Valerie Duay, Benoît Macq, Claudio Pollo, and J-Ph Thiran. Dense deformation field estimation for atlas-based segmentation of pathological MR brain images. *Computer methods and programs in biomedicine*, 84(2-3): 66–75, 2006.

[11] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3D U-Net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016.

[12] Fabian Isensee, Paul F Jaeger, Stefan SA Kohl, Jens Petersen, and Lutz Maier-Hein. nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation. *arXiv preprint arXiv:2011.00848*, 2020. URL https://arxiv.org/abs/2011.00848.

[13] Rehan Raza, Usman Ijaz Bajwa, Yousaf Mehmood, Muhammad Wasim Anwar, and Muhammad H Jamal. 3D Deep Residual U-Net Based Brain Tumor Segmentation from Multimodal MRI. *Biomedical Signal Processing and Control*, 74:103497, 2023. doi: 10.1016/j.bspc.2022.103497. URL https://www.sciencedirect.com/science/article/pii/S1746809422003809.

[14] L Zhao, J Zhang, L Wang, Y Liu, and Y Zhang. MM-UNet: A Multimodality Brain Tumor Segmentation Network in MRI Images. *Frontiers in Oncology*, 12:950706, 2022. doi: 10.3389/fonc.2022.950706. URL https://www.frontiersin.org/articles/10.3389/fonc.2022.950706/full.

[15] Mohammad Kharaji and et al. Brain Tumor Segmentation with Advanced nnU-Net: Pediatric and Adult Tumors. *arXiv preprint arXiv:2406.16848*, 2024. URL https://arxiv.org/abs/2406.16848.

[16] Qiran Jia and Hai Shu. BiTr-Unet: A CNN-Transformer Combined Network for MRI Brain Tumor Segmentation. *Frontiers in Neuroscience*, 16:951, 2022. doi: 10.3389/fnins.2022.100951. URL https://www.frontiersin.org/articles/10.3389/fnins.2022.100951/full.

[17] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks. *Advances in neural information processing systems*, 27, 2014.

[18] Nelson, a cute mountain cat, 2024.

[19] Prafulla Dhariwal and Alexander Nichol. Diffusion Models Beat GANs on Image Synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[20] Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2025. URL https://www.miccai.org/. Annual international conference on medical image computing and computer-assisted intervention.

[21] Docker, Inc. Docker. URL https://www.docker.com/.

[22] Jihoon Cho, Jonghye Woo, and Jinah Park. A Unified Framework for Synthesizing Multisequence Brain MRI via Hybrid Fusion. *arXiv preprint arXiv:2406.14954*, 2024.

[23] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107:3–11, 2018.

[24] Xilai Li, Wei Sun, and Tianfu Wu. Attentive Normalization. In *European Conference on Computer Vision*, pages 70–87. Springer, 2020.

[25] Ujjwal Baid, Satyam Ghodasara, Suyash Mohan, Michel Bilello, Evan Calabrese, Errol Colak, Keyvan Farahani, Jayashree Kalpathy-Cramer, Felipe C Kitamura, Sarthak Pati, et al. The RSNA-ASNR-MICCAI BraTS 2021 Benchmark on Brain Tumor Segmentation and Radiogenomic Classification. *arXiv preprint arXiv:2107.02314*, 2021.

[26] Ahmed W Moawad, Anastasia Janas, Ujjwal Baid, Divya Ramakrishnan, Rachit Saluja, Nader Ashraf, Nazanin Maleki, Leon Jekel, Nikolay Yordanov, Pascal Fehringer, et al. The Brain Tumor Segmentation - Metastases (BraTS-METS) Challenge 2023: Brain Metastasis Segmentation on Pre-treatment MRI. *ArXiv*, pages arXiv–2306, 2024.

[27] Dominic LaBella, Ujjwal Baid, Omaditya Khanna, Shan McBurney-Lin, Ryan McLean, Pierre Nedelec, Arif Rashid, Nourel Hoda Tahon, Talissa Altes, Radhika Bhalerao, et al. Analysis of the BraTS 2023 Intracranial Meningioma Segmentation Challenge. *arXiv preprint arXiv:2405.09787*, 2024.

[28] T Dao, DY Fu, S Ermon, A Rudra, and C FlashAttention Ré. FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness. *URL http://arxiv.org/abs/2205.14135*.

[29] Ugur Demir and Gozde Unal. Patch-Based Image Inpainting with Generative Adversarial Networks. *arXiv preprint arXiv:1803.07422*, 2018.

[30] Nabila Abraham and Naimul Mefraz Khan. A Novel Focal Tversky loss function with improved Attention U-Net for lesion segmentation. In *IEEE 16th International Symposium on Biomedical Imaging*, 2019.

[31] URL `https://www.wandb.com/`.

[32] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.