MTMed3D: A Multi-Task Transformer-Based Model for 3D Medical Imaging

Fan Li, Arun Iyengar, Lanyu Xu,

***Abstract*—In the field of medical imaging, AI-assisted techniques such as object detection, segmentation, and classification are widely employed to alleviate the workload of physicians and doctors. However, single-task models are predominantly used, overlooking the shared information across tasks. This oversight leads to inefficiencies in real-life applications. In this work, we propose MTMed3D, a novel end-to-end Multi-task Transformer-based model to address the limitations of single-task models by jointly performing 3D detection, segmentation, and classification in medical imaging. Our model uses a Transformer as the shared encoder to generate multi-scale features, followed by CNN-based task-specific decoders. The proposed framework was evaluated on the BraTS 2018 and 2019 datasets, achieving promising results across all three tasks, especially in detection, where our method achieves better results than prior works. Additionally, we compare our multi-task model with equivalent single-task variants trained separately. Our multi-task model significantly reduces computational costs and achieves faster inference speed while maintaining comparable performance to the single-task models, highlighting its efficiency advantage. To the best of our knowledge, this is the first work to leverage Transformers for multi-task learning that simultaneously covers detection, segmentation, and classification tasks in 3D medical imaging, presenting its potential to enhance diagnostic processes. The code is available at https://github.com/fanlimua/MTMed3D.git.**

***Index Terms*—Multi-task Learning, Swin Transformer, Medical Imaging, Segmentation, Classification, Detection**

I. Introduction

In the real world, medical diagnoses are often complicated and require assessing tumor morphology. For example, developing a treatment plan requires understanding the location, distinguishing between healthy and diseased tissue, and classifying disease type and severity. These steps rely heavily on medical imaging, making tasks like object detection, segmentation, and classification crucial for diagnoses. With advancements in deep learning, its application in the medical field has significantly improved diagnostic accuracy and efficiency. However, most existing models in medical image analysis are single-task models designed to perform only one task at a time. While single-task models are effective for addressing individual tasks, they face several limitations when compared to multi-task models. In particular, they may struggle to generalize when trained on small datasets, which is common in medical imaging where labeled data is often limited. Furthermore, the single-task design isolates them from one field, preventing them from leveraging knowledge from related tasks [1]. In contrast, multi-task models share representations across tasks, enabling them to learn complementary features and utilize information that may not be directly available in the primary task, thereby addressing the data scarcity challenge more effectively. Additionally, single-task models require redundant computations and increased resource consumption for each task, making them inefficient compared to multi-task models. The multi-task models share features and parameters across tasks, allowing them to perform multiple tasks simultaneously. This reduces computational costs, making the model more efficient and practical for medical image analysis, where computational resources are often limited.

In recent years, Transformer-based models have shown promising results for medical image analysis [2], [3], [4], [5], [6]. Some of these models leverage self-attention and positional encodings, enabling them to effectively capture long-range dependencies. However, traditional transformers like Vision Transformer (ViT) [7] suffer from high computational complexity due to the quadratic scaling of global self-attention, making them inefficient. To address this, Swin Transformers [8] apply self-attention within smaller, localized windows and then use the shift window-based multi-head self-attention module to establish connections between neighboring non-overlapping windows. This enables Swin Transformers to capture local and global context efficiently without the heavy computational cost. Building on Swin Transformer advancements, we propose an end-to-end Transformer-based multi-task model. While brain tumors serve as the test case for evaluating MTMed3D's performance, its architecture can be adapted to other medical imaging tasks. In summary, our main contributions are as follows:

- We propose MTMed3D, the first end-to-end Multi-task Transformer-based model for 3D medical image segmentation, detection, and classification.
- We evaluated MTMed3D on the Multimodal Brain Tumor Segmentation Challenge (BraTS) [9] 2018 and 2019, showing its promising performance across all three tasks, with detection performance surpassing that of existing methods. We have open-sourced MTMed3D at https://github.com/fanlimua/MTMed3D.git.
- We show that MTMed3D outperforms single-task models in efficiency with lower computation and latency. Additionally, we conduct a comprehensive ablation study, validating the effectiveness of our model design.

This work is supported by the National Science Foundation (NSF) under Grant No. 2245729. Submitted on November 18, 2025.

Fan Li is with the Department of Computer Science and Engineering, Oakland University, Rochester Hills, MI USA (e-mail: fanli@oakland.edu).

Arun Iyengar is with Intelligent Data Management and Analytics, LLC, Yorktown Heights, NY USA (e-mail: aki@akiyengar.com).

Lanyu Xu is with the Department of Computer Science and Engineering, Oakland University, Rochester Hills, MI USA (e-mail: lxu@oakland.edu).

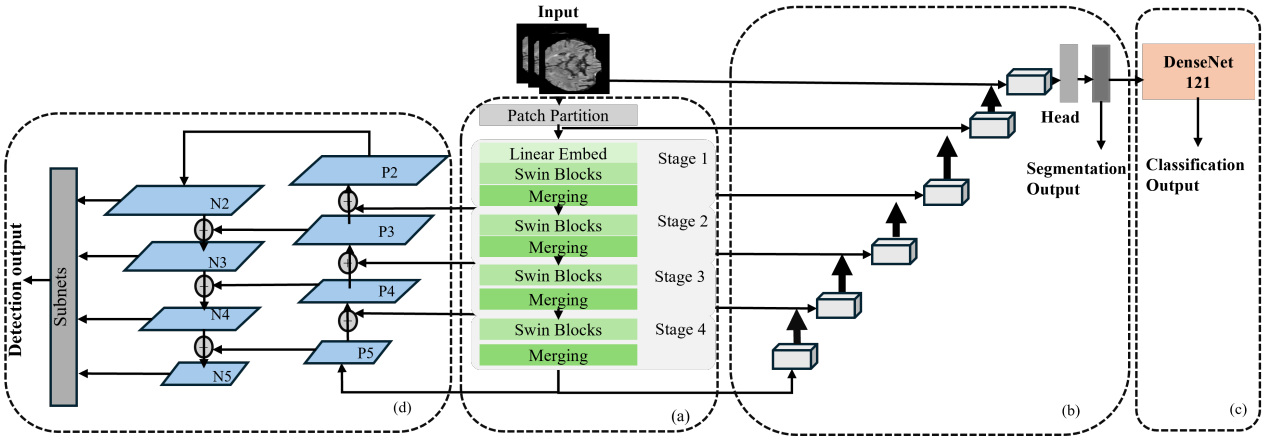

Fig. 1. Overview of MTMed3D: (a) Swin Transformer-based encoder, (b) Segmentation Decoder, (c) Classification Decoder, (d) Detection Decoder. The input to our model is 3D multi-modal MRI images with 4 channels. The data first passes through a shared Swin Transformer-based encoder and then flows through three task decoders to generate results for all three tasks.

II. RELATED WORK

A. Multi-task Learning

Multi-task learning (MTL) is a paradigm in machine learning that aims to jointly learn multiple related tasks, leveraging the knowledge from one task to benefit others. This approach hopes to improve the generalization performance of all tasks involved [10]. Currently, multi-task learning is applied in many fields of machine learning, such as autonomous driving. YOLOP [11] proposed an efficient multi-task model based on CNN, which can simultaneously handle traffic object detection, drivable area segmentation, and lane detection. Multi-task learning has also been applied to the field of medical images. Park et al. [12] proposed a multi-task model based on a ViT, which can perform three tasks of lung image classification, target detection, and segmentation at the same time. Their approach differs from ours in several key aspects. They utilize federated learning and split learning, training the shared transformer body centrally and task-specific heads and tails on separate machines with different datasets. In contrast, our model is trained end-to-end on a single machine using the same image for all tasks. Additionally, they employ a ViT as the shared encoder, which has a significantly larger parameter count of 66.367 million compared to the 8.063 million parameters in our Swin Transformer encoder. Amyar et al. [13] present an automatic classification segmentation tool for helping to diagnose COVID-19 pneumonia using chest CT imaging. Yan et al. [14] introduced a multi-task universal lesion analysis network designed for the joint detection, tagging, and segmentation of lesions across various body parts. This network is built upon an improved Mask R-CNN framework featuring three head branches and a 3D feature fusion strategy.

B. Transformer Models and Their Applications

Transformer was proposed by [15] and was initially used for machine translation and natural language processing (NLP) tasks. Its success has been widely attributed to the self-attention mechanism, which effectively models long-range dependencies [16], [17]. Given this capability, Transformer has been increasingly adopted in computer vision tasks. ViT [7] processes images as tokenized patches for classification but suffers from high computational complexity at high resolutions. To address this, Swin Transformer [8] introduces window attention, applying local attention within windows. This method effectively reduces complexity. Transformer-based models have gained increasing attention in medical image analysis [18], [19]. For segmentation, TransUNet [2] was the first Transformer-based framework for medical image segmentation, integrating a Transformer layer into the bottleneck of a U-Net for multi-organ segmentation. Swin UNETR [6] leverages a Swin Transformer encoder to extract rich semantic features and a CNN decoder to reconstruct structured outputs. Transformer-based architectures have also been explored for classification tasks. Lu et al. [20] proposed a two-stage framework that first employs contrastive pre-training to extract meaningful representations and then aggregates features using a Transformer-based sparse attention module for label prediction. Zhang et al. [21] introduced a Transformer-based two-stage framework for COVID-19 diagnosis in 3D CT scans. Their approach involves using a UNet for lung segmentation, followed by Swin Transformer feature extraction from each CT slice, which is then aggregated into a 3D volume-level representation for classification. For object detection tasks, Shen et al. [22], inspired by the Detection Transformer (DETR), proposed a Convolutional Transformer (COTR) network for end-to-end polyp detection. COTR consists of a CNN for feature extraction, Transformer encoder layers interleaved with convolutional layers for feature encoding and recalibration, Transformer decoder layers for object querying, and a feed-forward network for detection prediction. This architecture effectively leverages the strengths of both CNNs and Transformers, enabling COTR to outperform DETR on two different datasets. Building on previous research, we explore the potential of the Transformer as a shared backbone for multi-task medical imaging.

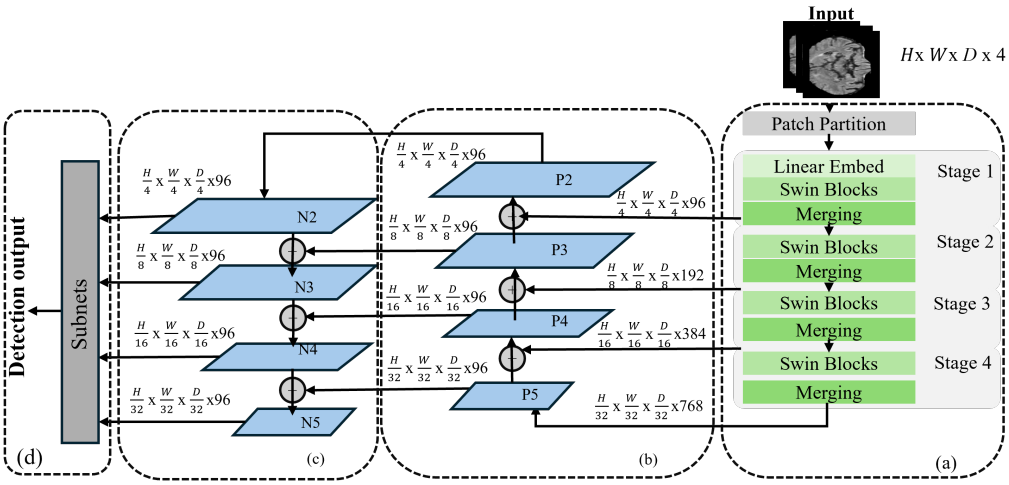

Fig. 2. Detection Network: (a) Swin Transformer-based encoder, (b) FPN, (b) + (c) PANet, (d) Subnets for final box prediction are composed of Convs, GroupNorm, and ReLU.

III. MTMed3D Model Design

A. Design Overview

We propose MTMed3D, a Multi-task Transformer-based model for 3D medical image analysis that consists of a shared encoder and three decoders. An overview of the proposed MTMed3D is presented in Figure 1. In designing this multi-task model, we solved the following challenges. ***Challenge 1:*** Selection of appropriate model architecture. **Solution:** We adopted hard parameter sharing [23], enabling tasks to use a shared encoder. ***Challenge 2:*** Task balancing among multiple tasks. **Solution:** We used Gradient Normalization (GradNorm) [24] to adjust loss weights across tasks. ***Challenge 3:*** The absence of detection labels. **Solution:** We generated detection labels from existing segmentation annotations.

B. Encoder

Our multi-task model uses a shared Transformer encoder implemented with Swin Transformer blocks, which consist of four hierarchical stages. Each stage consists of Swin blocks followed by a patch merging layer. Given a 3D image, we first apply a patch partition layer to create flattened 3D patches, followed by a linear embedding layer to project the patches into a latent space. The patch embeddings then pass through four Swin Transformer blocks and patch merging layers to generate hierarchical feature representations. In the Swin Transformer block, the window-based multi-head self-attention (W-MSA) module computes self-attention within local windows, which helps to reduce computational complexity. However, W-MSA lacks connections across windows, so the shift window-based multi-head self-attention (SW-MSA) module is applied to establish connections between neighboring non-overlapping windows in the previous layer. After SW-MSA, mutual communication between windows can be achieved. So Swin Transformer block [8] can be formulated as:

$$z'_l = W - MSA(LN(z_{l-1})) + z_{l-1}, \quad (1)$$

$$z_l = MLP(LN(z'_l)) + z'_l, \quad (2)$$

$$z'_{l+1} = SW - MSA(LN(z_l)) + z_l, \quad (3)$$

$$z_{l+1} = MLP(LN(z'_{l+1})) + z'_{l+1}. \quad (4)$$

z'_l and z'_{l+1} denote the outputs of W-MSA and SW-MSA; MLP and LN denote Multi-Layer Perceptron and Layer Normalization, respectively. Similar to prior works [15], self-attention is computed using queries, keys, and values. After each Swin Transformer block, a patch merging layer reduces the resolution of feature maps, enabling the encoder to generate multi-scale features. This shared encoder is used across tasks to address Challenge 1, where tasks share parameters and representations to reduce redundant computation.

C. Decoders

In this section, we introduce three decoders, each designed to handle a different task.

1) Detection Decoder: For the detection task, we utilize a modified RetinaNet [25] framework. In our modified RetinaNet, we adopt the Swin Transformer encoder as the backbone to extract features and replace the Feature Pyramid Networks (FPN) [26] architecture with Path Aggregation Network (PANet) [27], as shown in Figure 2. This approach has not been proposed in previous works. The Swin Transformer encoder extracts multi-scale feature maps at different stages i (i=2,3,4) and the bottleneck (i=5), which are reshaped to a uniform channel size using convolutional blocks. These reshaped features are progressively upsampled and fused with the previous layer's reshaped features through skip connections. Applying convolution filters on these features to generate the feature maps P_i (i=2,3,4,5) for PANet. PANet extends FPN by introducing an additional pathway as (c) in Figure 2 to enhance low-level feature propagation. In this path, Each feature map N_i first goes through a convolutional layer to

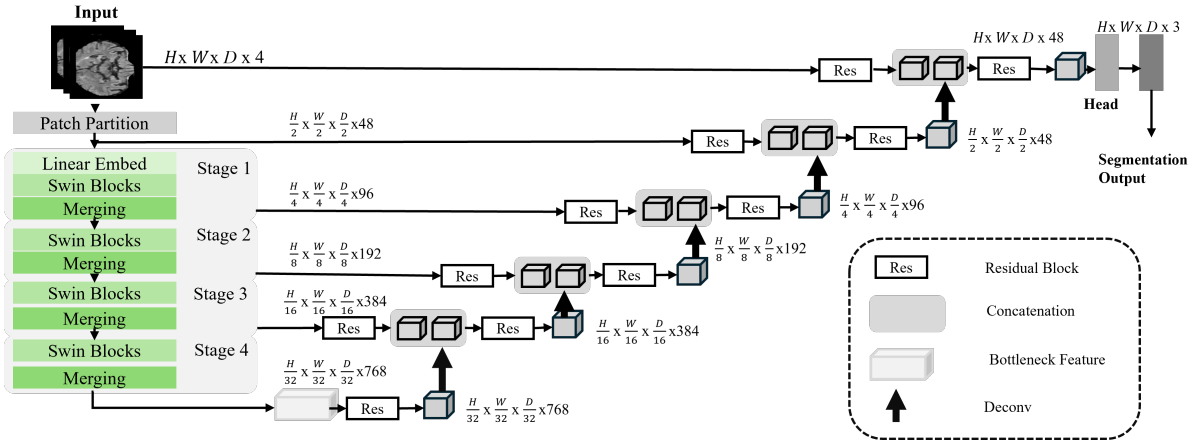

Fig. 3. Segmentation Network: The Swin Transformer-based encoder and CNN-based decoder form a U-shape segmentation network.

reduce the spatial size and concatenate with feature map P_{i+1} via a skip connection. The fused feature map is processed by a convolutional layer to generate N_{i+1} for the following sub-networks. The box regression subnet uses convolutional layers, GroupNorm, and ReLU to predict object coordinates.

2) Segmentation Decoder: The U-Net architecture has shown promising results in medical image segmentation, leading many recent studies [28], [29], [30], [31], [32] to design their architectures in a U-Net shape. So in our segmentation decoder, we follow the Swin UNETR [6] network to achieve segmentation. Swin UNETR extracts the feature representation using the Swin Transformer encoder and feeds them to the decoder via skip connection at each stage, as shown in Figure 3. Features from each encoder stage i (i=0,1,2,3,4,5) are processed through residual blocks. The decoder progressively increases resolution by deconvolution layers, concatenating features from previous stages to refine segmentation. A final convolutional layer with a sigmoid activation generates the segmentation output. This U-shaped architecture effectively integrates multi-scale features, enhancing segmentation detail and performance.

3) Classification Decoder: Tripathi et al. [33] used segmentation outputs as inputs for classification, leveraging detailed tumor information to improve performance. Following this approach, we place our classification branch after segmentation, using DenseNet-121 [34] for tumor grading. DenseNet improves information flow by connecting each layer directly to all subsequent layers. Hence, the lth layer receives the feature maps of all preceding layers, and the formula is computed as follows:

$$x_l = H_l([x_0, x_1, \ldots, x_{(l-1)}]) \tag{5}$$

where $[x_0, x_1, \ldots, x_{(l-1)}]$ represents the concatenation of the feature maps produced in layers $0, 1, 2, \ldots, l-1$, and H_l can be a composite function of operations, x_l denotes the output of the lth layer. DenseNet can alleviate gradient vanishing and reduce model size.

D. Optimization

Our network uses three decoders, with the total loss formulated as a weighted sum of their respective loss components:

$$L_{\text{total}} = w_1 L_{\text{seg}} + w_2 L_{\text{cls}} + w_3 L_{\text{det}} \tag{6}$$

where $Lseg$ is DiceLoss, $Lcls$ is FocalLoss, and $Ldet$ is SmoothL1Loss, w_1, w_2, and w_3 are the weights correspond to the three tasks. A key challenge in multi-task learning is balancing task performance, as one task may dominate training due to larger gradients. To address the imbalance described in Challenge 2, we employ GradNorm [24], which directly adjusts gradient magnitudes by optimizing multi-task loss. A multi-task loss function can be expressed as follows:

$$L(t) = \sum w_i(t) L_i(t) \tag{7}$$

where the sum runs over all T tasks at training step t. This method first normalizes gradient norms across tasks by defining the $L2$ norm of the gradient for the weighted single-task loss $w_i(t) \times L_i(t)$ with respect to the chosen weights w as $G_w^{(i)}(t)$. The mean task gradient $\bar{G_w}$ averaged across all task gradients $G_w^{(i)}$ at training step t is considered as a common basis from which the relative gradient sizes across tasks can be measured:

$$\overline{G}_w(t) = \mathbb{E}_{\text{task}} \left[G_w^{(i)}(t) \right] \tag{8}$$

To identify which tasks are lagging in the current training process, the inverse training rate L_i of task i at training step t is defined as follows:

$$\widetilde{L_i}(t) = L_i(t)/L_i(0) \tag{9}$$

To compare the training progress of different tasks, the relative inverse training rate of task i at training step t is defined as follows:

$$r_i(t) = \widetilde{L_i}(t)/\mathbb{E}_{\text{task}} \left[\widetilde{L_i}(t) \right] \tag{10}$$

When a task's loss decreases slowly, its inverse training rate $\widetilde{L_i}(t)$ will be high. This results in an increase in the relative inverse training rate $r_i(t)$. Therefore, the desired gradient norm for each task i is simply:

$$G_w^{(i)}(t) \rightarrow \overline{G}_w(t) \cdot r_i(t) \tag{11}$$

TABLE I
SEGMENTATION, CLASSIFICATION, AND DETECTION RESULTS ON FIVE-FOLD CROSS-VALIDATION.

Task	**Segmentation**						**Classification**			**Detection**			
Metric	**Dice**			**HD**			**Acc**	**Sen**	**Spe**	**mAP@**	**mAP@**	**mAR@**	**mAR@**
Region	**WT**	**TC**	**ET**	**WT**	**TC**	**ET**				0.1:0.5	0.5	0.1:0.5	0.5
BraST 2018 (Best)	0.9082	0.8183	0.8038	5.4346	6.9773	5.6794	0.9298	0.9286	0.9333	0.9711	0.8650	0.9844	0.9298
BraST 2018 (Avg.)	0.8793	0.8019	0.7755	9.7864	8.3361	6.7452	0.9193	0.9761	0.7634	0.9082	0.7628	0.9357	0.8421
BraST 2019 (Best)	0.8814	0.8084	0.7879	13.1085	9.9521	7.7802	0.9701	0.9804	0.9375	0.8820	0.7368	0.9187	0.8209
BraST 2019 (Avg.)	0.8875	0.8311	0.8020	9.2851	7.5083	5.5534	0.8955	0.9769	0.6153	0.8793	0.7250	0.9257	0.8269

TABLE II
DETECTION PERFORMANCE OF DIFFERENT METHODS.

Methods	**mAP@0.1:0.5**	**mAP@0.5**	**mAR@0.1:0.5**	**mAR@0.5**
RetinaNet [25]	0.8664	0.6768	0.8850	0.7193
nnDetection [35]	–	0.8360	–	–
MedYOLO [35]	–	0.8610	–	–
MTMed3D (Best)	**0.9711**	**0.8650**	**0.9844**	**0.9298**

Equation (11) gives a target for each task i's gradient norms, and GradNorm updates loss weights to move gradient norms towards this target for each task. GradNorm is implemented as an L_1 loss function L_{grad} between the actual and target gradient norms at each timestep for task i:

$$L_{\text{grad}}(i) = \left| G_w^{(i)}(t) - \overline{G}_w(t) \cdot r_i(t) \right| \tag{12}$$

L_{grad} is then differentiated only with respect to the w_i, as the w_i directly control gradient magnitudes per task. The computed gradients are then applied via optimizer to update w_i. In practice, these weights, w_1, w_2, and w_3, are updated iteratively during training via backpropagation.

IV. EXPERIMENT

A. Setting

1) Dataset: We evaluated our proposed model on BraTS 2018 and BraTS 2019 [9], which offers different image modalities, including T1, T2, T1ce, and FLAIR. We combined four MRI modalities into a single four-channel input. The BraTS datasets provide segmentation labels for necrosis (label 1), edema (label 2), and enhancing tumor (label 4), which are grouped into three sub-regions: Whole Tumor (WT, labels 1, 2, 4), Tumor Core (TC, labels 1, 4), and Enhancing Tumor (ET, label 4). The BraTS 2018 dataset includes 210 High-Grade Glioma (HGG) and 75 Low-Grade Glioma (LGG) cases, while BraTS 2019 contains 259 HGG and 76 LGG cases. To address Challenge 3, as BraTS lacks official object detection labels, we generate bounding boxes by extracting the minimum and maximum coordinates from the segmentation annotations.

2) Data Augmentation: The input image size is $4 \times 240 \times 240 \times 155$, We arranged the channels as the first dimension and used the RAS (Right, Anterior, Superior) orientation. We normalized all input images and performed a center crop from the 3D image volumes to $4 \times 96 \times 96 \times 96$ to reduce the computational load. We applied a random axis mirror flip with a probability of 0.5 for all 3 axes and random rotation with a probability of 0.75. Additionally, we applied data augmentation transformers for a random per-channel intensity shift in the range (-0.1, 0.1) and random scale of intensity in the range (-0.1, 0.1) to the input channel.

3) Implementation Details: We conducted experiments on a machine equipped with a 13th Gen Intel Core i7-13700F CPU and an NVIDIA GeForce RTX 4070 Ti GPU. For the segmentation task, we set the learning rate to 1e-4, while for the detection and classification tasks, we set it to 1e-5. During training, we utilized a cosine annealing learning rate scheduler and the AdamW optimizer. The batch size was set to 1 to prevent excessive memory usage. Our model was trained using a five-fold cross-validation scheme with a 4:1 ratio.

4) Evaluation Metrics: We used task-specific metrics to evaluate our model's performance. Segmentation performance is evaluated using the Dice Coefficient (Dice) and Hausdorff Distance (HD). We used mean Average Precision (mAP) and mean Average Recall (mAR) for detection. For classification, we utilized Accuracy (Acc), Sensitivity (Sen), and Specificity (Spe). Additionally, we compared the efficiency of the multi-task model with three single-task models using metrics such as Multiply-Accumulate Operations (MACs), Floating Point Operations (FLOPs), Parameters (Params), model size, and latency.

B. Results

1) Detection Results: Table I reports the best and average results over five-fold cross-validation on BraTS 2018 and 2019 for detection, segmentation, and classification, where the best fold is defined as the one with the highest average performance across all three tasks. mAP@0.1:0.5 refers to the mAP computed over IoU thresholds from 0.1 to 0.5 with a step size of 0.05, while mAP@0.5 denotes the result at a fixed IoU of 0.5. Since the BraTS dataset was originally designed for segmentation and classification tasks, there are limited existing 3D detection methods evaluated on this dataset. To establish a fair comparison, we implemented a 3D RetinaNet with a ResNet-FPN backbone and trained it using the fold that achieved the best performance in our cross-validation experiments. This model served as our detection baseline. We also compared our proposed model against nnDetection and MedYOLO [35], though their reported results are available at mAP@0.5, a dash ("-") indicates that no relevant data were available. Table II presents the detailed results, showing that MTMed3D outperforms these methods, possibly due to the hierarchical Transformer encoder's ability to effectively capture spatial context. The strong mAP and mAR values demonstrate

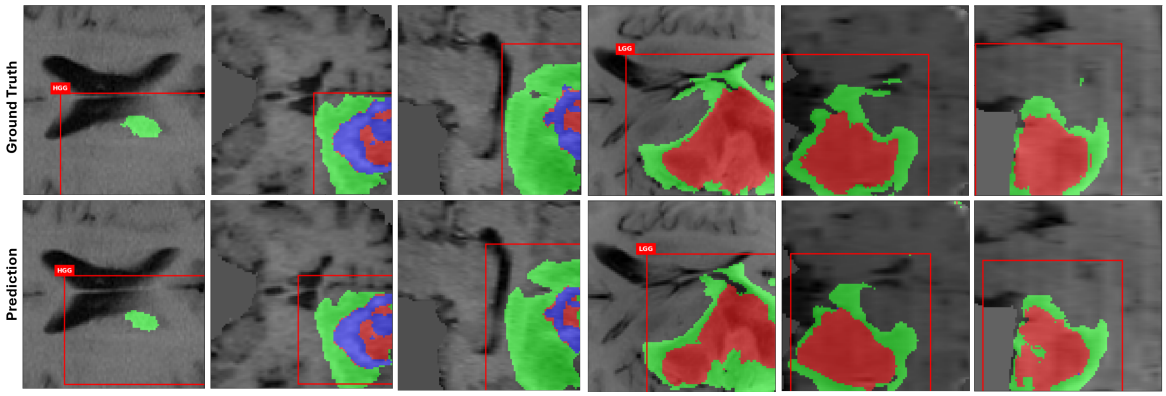

Fig. 4. Qualitative results for MTMed3D: The top row represents the Ground Truth, while the bottom row shows the outputs of MTMed3D. The left three columns depict HGG samples, representing three brain directions, and the right three columns show LGG samples. Red indicates the Tumor Core, green represents the Whole Tumor, and blue denotes the Enhanced Tumor.

the effectiveness of our model in accurately localizing brain tumors in 3D medical images. While our model outperforms existing methods on BraTS, further validation on additional datasets is necessary to assess its generalization capability in broader detection scenarios. In Figure 4, the red rectangular boxes represent the predicted detection bounding boxes.

TABLE III
TUMOR SEGMENTATION PERFORMANCE OF DIFFERENT METHODS.

Methods	Dice				HD			
	WT	TC	ET	Avg.	WT	TC	ET	Avg.
AutoReg [36]	0.8839	0.8154	0.7664	0.8219	5.9044	**4.8091**	3.7731	4.8289
MIC-DKFZ [37]	0.8781	0.8062	0.7788	0.8210	6.0300	5.0800	**2.9000**	**4.6700**
DeepSCAN [38]	0.8859	0.7992	0.7318	0.8056	5.5185	5.5347	3.4808	4.8447
MultiDeep [39]	0.8842	0.7960	0.7775	0.8192	5.4681	6.8773	2.9366	5.0940
MTMed3D (Best)	**0.9082**	**0.8183**	**0.8038**	**0.8434**	**5.4346**	6.9773	5.6794	6.0304

2) Segmentation Results: In Table I, the model shows strong segmentation performance with high Dice and low HD across all tumor regions. Table III compares the segmentation performance of our model with several state-of-the-art (SOTA) methods leading in the BraTS 2018 challenge rankings. Since BraTS requires server submission for test evaluation and lacks detection labels, we report five-fold cross-validation results to ensure consistency across the three tasks. MTMed3D achieves the highest Dice scores. Although our model slightly lags behind some methods in HD for specific components, the overall performance of MTMed3D remains competitive. Figure 4 presents the output results of our multi-task model. Comparing the ground truth with the segmentation results, while there are little differences in some areas, such as the boundary of the WT and TC, the overall results are closely aligned with the ground truth.

[**Insight 1**] The compared methods from the ranking are designed for single tasks and rely on ensembling to improve the performance. In contrast, MTMed3D achieves competitive results using a single end-to-end architecture, offering better efficiency and practicality for clinical deployment.

3) Classification Results: Table I details MTMed3D's classification performance on the BraTS 2018 and 2019 datasets. Our average 5-fold cross-validation results show higher sensitivity but lower specificity, suggesting a bias toward the majority HGG class. We applied Focal Loss to address this class imbalance, but it did not lead to improved performance. Table IV compares the classification performance of our model with SOTA methods on the BraTS 2018 dataset. According to Table IV, other methods outperform ours. This is primarily because these methods operate on 2D slices rather than directly on 3D volumes and rely on dedicated preprocessing designed for classification. For instance, Mask R-CNN[40], CNN [41], and SE-ResNeXt [42] are all based on 2D inputs. However, our model is a multi-task model that processes all tasks on the same 3D data, without separate 2D preprocessing or classification steps. In addition, ConvNet [40] uses a two-stage pipeline that segments the tumor and then applies additional data augmentation to the segmented regions before training a classification model. MTMed3D is an end-to-end model, eliminating the need for manually designed intermediate steps and simplifying the overall training and inference pipeline. Moreover, SE-ResNeXt applies transfer learning to enhance generalization. In contrast, our multi-task model not only ensures the performance of each task but also maintains a balance across tasks, preventing any single task from dominating training. This constraint limits the use of task-specific optimization or pre-trained weights that could disproportionately favor one task over others. Figure 4 shows classification results above the red bounding boxes, with both HGG and LGG correctly classified.

TABLE IV
CLASSIFICATION PERFORMANCE OF DIFFERENT METHODS.

Methods	Accuracy	Sensitivity	Specificity
Mask R-CNN (2D) [40]	0.9630	0.9350	0.9720
ConvNet (3D) [40]	0.9710	0.9470	0.9680
CNN (2D) [41]	0.9715	0.9818	**0.9855**
SE-ResNeXt (2D) [42]	**0.9745**	**0.9835**	0.9493
MTMed3D (Best)	0.9298	0.9286	0.9333

[**Insight 2**] While MTMed3D performs well on segmentation and detection tasks, classification remains more challenging, possibly due to the absence of a learnable class token, which may limit its ability to represent features critical

TABLE V
PERFORMANCE COMPARISON OF MULTI-TASK AND SINGLE-TASK MODELS AND ABLATION STUDY ON MULTI-TASK COMPONENTS.

Task	Segmentation								Classification			Detection		Task Avg.
	Dice				HD				Acc	Sen	Spe	mAP@	mAR@	
Region	WT	TC	ET	Avg.	WT	TC	ET	Avg.				0.1:0.5	0.1:0.5	
MTMed3D (Avg.)	0.8793	0.8019	0.7755	0.8189	9.7864	8.3361	6.7452	8.2892	0.9193	0.9761	0.7634	0.9082	0.9357	**0.8955**
Single-task (Avg.)	0.8837	0.7972	0.7348	0.8052	6.7715	6.2509	5.4199	6.1474	0.9017	0.9594	0.7687	0.8217	0.8608	0.8474
FPN + GradNorm (Avg.)	0.8803	0.8060	0.7582	0.8148	10.2574	8.7848	6.9169	8.6530	0.9123	0.9859	0.7056	0.8963	0.9310	0.8886
FPN + MGDA (Avg.)	0.8826	0.8070	0.7837	0.8244	9.1462	8.0395	6.4389	7.8749	0.8666	0.9712	0.5760	0.9172	0.9399	0.8870
PANet + MGDA (Avg.)	0.8772	0.8036	0.7763	0.8190	10.1070	8.2520	6.6129	8.3240	0.8632	0.9812	0.5401	0.9070	0.9380	0.8818

TABLE VI
EFFICIENCY COMPARISON BETWEEN MTMED3D AND THREE SINGLE-TASK MODELS (↓ INDICATES LOWER IS BETTER).

Model Name	MACs ↓	FLOPs ↓	Params ↓	Inference (s) ↓	Size (MB) ↓
Single-task Models	4×10^{11}	7×10^{11}	1×10^{8}	0.2248	587.81
MTMed3D	$\mathbf{2 \times 10^{11}}$	$\mathbf{4 \times 10^{11}}$	$\mathbf{8 \times 10^{7}}$	**0.0996**	**306.79**

for distinguishing LGG from HGG. Although our classification performance does not surpass other methods, it suggests Swin Transformer may be insufficient in glioma grading classification without additional structural enhancements, which is also reflected in the ResMT [43], a model trained on BraTS 2019 that incorporates a parallel CNN branch alongside Swin UNETR and is further supplemented by additional modules to improve classification.

4) Multi-task vs Single-task Models: Table V compares the performance of MTMed3D and single-task models. The Task Avg. column represents the average of Dice, accuracy, and detection performance, providing an overall measure across all tasks. Each single-task model follows the same architecture as its multi-task counterpart but is trained independently. While the single-task model achieves better segmentation results in terms of HD, MTMed3D remains competitive. In classification, MTMed3D demonstrates higher accuracy and sensitivity, while for detection, it has higher mAP and mAR. We also compared the efficiency of our MTMed3D with three single-task models, as illustrated in Table VI. MTMed3D significantly reduces computational costs, with approximately 46.64% fewer MACs and FLOPs. It also requires 47.80% fewer parameters and less size. The inference time is reduced by 55.70%, enabling faster decision-making. This significant reduction in computational demand, along with its strong performance, makes MTMed3D a practical choice for real-world clinical settings where efficiency and accuracy are crucial under limited computing resources. Further analysis of the computation breakdown and inference time of different components is provided in Supplementary Section 3. Additionally, MTMed3D's flexible design lets users activate tasks independently based on resources and clinical needs, optimizing efficiency without compromising accuracy. It directly processes 3D images, eliminating the need to convert 3D data into 2D slices and thus avoiding the overhead of repetitive forward propagation for each slice and the computational burden of reassembling the final output.

[**Insight 3**] Comparing the multi-task and single-task models, we find that multi-task learning enhances detection performance by leveraging shared representations across tasks. Features crucial for detection, which may be difficult to learn in a single-task model, are more easily learned in a multi-task setting, as they also contribute to segmentation and classification. This shared learning process improves overall model performance.

C. Ablation Studies

In our ablation studies, we analyze the impact of different decoder designs and the task-balancing optimizer on performance. Additional analyses, including the effect of varying encoder depth and task interactions when training without the optimizer, are provided in Supplementary Sections 1 and 2.

1) Detection Branch: To evaluate the efficiency of PANet, we conducted an ablation study on different detection branches. Table V shows results obtained using FPN as the detection branch. Although FPN (FPN + GradNorm) achieves a slightly higher Dice score for the WT and TC regions, MTMed3D (PANet + GradNorm) provides better performance in Dice for ET and consistently lower HD values. MTMed3D shows improved classification accuracy, specificity, and detection performance. Overall, PANet outperforms FPN.

2) Optimizers: In the optimizer, we utilize the GradNorm method to adjust gradient magnitudes by tuning multi-task loss functions. To evaluate the optimization strategy, we design an ablation study comparing GradNorm with the Multiple Gradient Descent Algorithm (MGDA) proposed by [44]. MGDA formalizes multi-task learning as a multi-objective optimization problem, aiming for Pareto optimality where no task's loss can be improved without harming others. Our comparative analysis reveals how these distinct approaches impact task performance trade-offs. Table V compares MTMed3D with PANet + MGDA, showing that MGDA slightly outperforms in segmentation metrics but suffers from lower classification accuracy, highlighting an imbalance across tasks. In contrast, MTMed3D excels in classification accuracy and specificity, achieving the best balance and ensuring competitive results in segmentation and detection without compromising classification. Similarly, FPN + MGDA performs well in segmentation but struggles with classification. Overall, MTMed3D handles all tasks effectively without sacrificing performance.

V. CONCLUSION

We present MTMed3D, a novel end-to-end multi-task model for 3D detection, segmentation, and classification. Our multi-task design improves resource efficiency compared to single-task models that train tasks separately. MTMed3D shows

promising results on all three tasks, demonstrating not only its effectiveness but also the potential of the Transformer framework for multi-task learning in medical imaging applications. In particular, our detection results not only outperform existing methods on BraTS, but also show improvement over the single-task model under joint training, suggesting that multi-task learning with Transformer backbones offers benefits for detection. While we primarily validated it on brain tumor datasets, it is adaptable to other datasets, making it suitable for various medical imaging scenarios and tasks. Overall, MTMed3D presents an efficient solution for multi-task medical image analysis, with the potential to serve as an assistive tool for radiologists to help alleviate workload.

REFERENCES

[1] S. Vandenhende, S. Georgoulis, W. Van Gansbeke, M. Proesmans, D. Dai, and L. Van Gool, "Multi-task learning for dense prediction tasks: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 7, pp. 3614–3633, 2021.

[2] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A. L. Yuille, and Y. Zhou, "Transunet: Transformers make strong encoders for medical image segmentation," *arXiv preprint arXiv:2102.04306*, 2021.

[3] A. Hatamizadeh, Y. Tang, V. Nath, D. Yang, A. Myronenko, B. Landman, H. R. Roth, and D. Xu, "Unetr: Transformers for 3d medical image segmentation," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2022, pp. 574–584.

[4] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian, and M. Wang, "Swin-unet: Unet-like pure transformer for medical image segmentation," in *European conference on computer vision*. Springer, 2022, pp. 205–218.

[5] W. Wenxuan, C. Chen, D. Meng, Y. Hong, Z. Sen, and L. Jiangyun, "Transbts: Multimodal brain tumor segmentation using transformer," in *International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer*, 2021, pp. 109–119.

[6] A. Hatamizadeh, V. Nath, Y. Tang, D. Yang, H. R. Roth, and D. Xu, "Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images," in *International MICCAI Brainlesion Workshop*. Springer, 2021, pp. 272–284.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[8] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.

[9] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest *et al.*, "The multimodal brain tumor image segmentation benchmark (brats)," *IEEE transactions on medical imaging*, vol. 34, no. 10, pp. 1993–2024, 2014.

[10] Y. Zhang and Q. Yang, "A survey on multi-task learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 12, pp. 5586–5609, 2021.

[11] D. Wu, M.-W. Liao, W.-T. Zhang, X.-G. Wang, X. Bai, W.-Q. Cheng, and W.-Y. Liu, "Yolop: You only look once for panoptic driving perception," *Machine Intelligence Research*, vol. 19, no. 6, pp. 550–562, 2022.

[12] S. Park, G. Kim, J. Kim, B. Kim, and J. C. Ye, "Federated split vision transformer for covid-19 cxr diagnosis using task-agnostic training," *arXiv preprint arXiv:2111.01338*, 2021.

[13] A. Amyar, R. Modzelewski, H. Li, and S. Ruan, "Multi-task deep learning based ct imaging analysis for covid-19 pneumonia: Classification and segmentation," *Computers in biology and medicine*, vol. 126, p. 104037, 2020.

[14] K. Yan, Y. Tang, Y. Peng, V. Sandfort, M. Bagheri, Z. Lu, and R. M. Summers, "Mulan: multitask universal lesion analysis network for joint lesion detection, tagging, and segmentation," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part VI 22*. Springer, 2019, pp. 194–202.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[16] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[17] J. Cheng, L. Dong, and M. Lapata, "Long short-term memory-networks for machine reading," *arXiv preprint arXiv:1601.06733*, 2016.

[18] F. Shamshad, S. Khan, S. W. Zamir, M. H. Khan, M. Hayat, F. S. Khan, and H. Fu, "Transformers in medical imaging: A survey," *Medical Image Analysis*, p. 102802, 2023.

[19] Z. Liu, Q. Lv, Z. Yang, Y. Li, C. H. Lee, and L. Shen, "Recent progress in transformer-based medical image analysis," *Computers in Biology and Medicine*, vol. 164, p. 107268, 2023.

[20] M. Lu, Y. Pan, D. Nie, F. Liu, F. Shi, Y. Xia, and D. Shen, "Smile: Sparse-attention based multiple instance contrastive learning for glioma sub-type classification using pathological images," in *MICCAI Workshop on Computational Pathology*. PMLR, 2021, pp. 159–169.

[21] L. Zhang and Y. Wen, "A transformer-based framework for automatic covid19 diagnosis in chest cts," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 513–518.

[22] Z. Shen, R. Fu, C. Lin, and S. Zheng, "Cotr: Convolution in transformer network for end to end polyp detection," in *2021 7th International Conference on Computer and Communications (ICCC)*. IEEE, 2021, pp. 1757–1761.

[23] E. Meyerson and R. Miikkulainen, "Beyond shared hierarchies: Deep multitask learning through soft layer ordering," *arXiv preprint arXiv:1711.00108*, 2017.

[24] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *International conference on machine learning*. PMLR, 2018, pp. 794–803.

[25] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[26] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[27] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8759–8768.

[28] X. Li, H. Chen, X. Qi, Q. Dou, C.-W. Fu, and P.-A. Heng, "H-denseunet: hybrid densely connected unet for liver and tumor segmentation from ct volumes," *IEEE transactions on medical imaging*, vol. 37, no. 12, pp. 2663–2674, 2018.

[29] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*. Springer, 2018, pp. 3–11.

[30] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: Redesigning skip connections to exploit multiscale features in image segmentation," *IEEE transactions on medical imaging*, vol. 39, no. 6, pp. 1856–1867, 2019.

[31] D. Jha, M. A. Riegler, D. Johansen, P. Halvorsen, and H. D. Johansen, "Doubleu-net: A deep convolutional neural network for medical image segmentation," in *2020 IEEE 33rd International symposium on computer-based medical systems (CBMS)*. IEEE, 2020, pp. 558–564.

[32] J. Shang and S. Zhou, "Lk-unet: Large kernel design for 3d medical image segmentation," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 1576–1580.

[33] P. C. Tripathi and S. Bag, "An attention-guided cnn framework for segmentation and grading of glioma using 3d mri scans," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2022.

[34] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[35] J. Sobek, J. R. Medina Inojosa, B. J. Medina Inojosa, S. Rassoulinejad-Mousavi, G. M. Conte, F. Lopez-Jimenez, and B. J. Erickson, "Medyolo: a medical image object detection framework," *Journal of Imaging Informatics in Medicine*, pp. 1–9, 2024.

[36] A. Myronenko, "3d mri brain tumor segmentation using autoencoder regularization," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and*

Traumatic Brain Injuries: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part II 4. Springer, 2019, pp. 311–320.

[37] F. Isensee, P. Kickingereder, W. Wick, M. Bendszus, and K. H. Maier-Hein, "No new-net," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part II 4*. Springer, 2019, pp. 234–244.

[38] R. McKinley, R. Meier, and R. Wiest, "Ensembles of densely-connected cnns with label-uncertainty for brain tumor segmentation," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part II 4*. Springer, 2019, pp. 456–465.

[39] C. Zhou, S. Chen, C. Ding, and D. Tao, "Learning contextual and attentive information for brain tumor segmentation," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part II 4*. Springer, 2019, pp. 497–507.

[40] Y. Zhuge, H. Ning, P. Mathen, J. Y. Cheng, A. V. Krauze, K. Camphausen, and R. W. Miller, "Automated glioma grading on conventional mri images using deep convolutional neural networks," *Medical physics*, vol. 47, no. 7, pp. 3044–3053, 2020.

[41] H. A. Hafeez, M. A. Elmagzoub, N. A. B. Abdullah, M. S. Al Reshan, G. Gilanie, S. Alyami, M. U. Hassan, and A. Shaikh, "A cnn-model to classify low-grade and high-grade glioma from mri images," *IEEE Access*, vol. 11, pp. 46 283–46 296, 2023.

[42] J. Linqi, N. Chunyu, and L. Jingyang, "Glioma classification framework based on se-resnext network and its optimization," *IET Image Processing*, vol. 16, no. 2, pp. 596–605, 2022.

[43] H. Cui, Z. Ruan, Z. Xu, X. Luo, J. Dai, and D. Geng, "Resmt: A hybrid cnn-transformer framework for glioma grading with 3d mri," *Computers and Electrical Engineering*, vol. 120, p. 109745, 2024.

[44] O. Sener and V. Koltun, "Multi-task learning as multi-objective optimization," *Advances in neural information processing systems*, vol. 31, 2018.