Brain Tumor Semantic Segmentation

Melih Berk Yılmaz, *Member, Bilkent University,* Fuat Arslan, *Member, Bilkent University,*

***Abstract*—This study introduces FM-Net, an innovative neural network architecture for brain tumor segmentation in MRI scans, enhancing the conventional U-Net model with attention mechanisms, deep supervision, and an optimized bottleneck design. Focused on improving accuracy and efficiency, the research employs advanced techniques like mixed precision training and GPU-optimized data loading to expedite the training process without compromising performance. Utilizing the BraTS Dataset, the model demonstrates high segmentation accuracy, evidenced by competitive Dice Coefficient scores and low loss values.**

***Index Terms*—Brain Tumor Segmentation, MRI Imaging, U-Net, Attention, Deep Supervision, GPU Optimization, Mixed Precision.**

I. Introduction

THIS paper presents research on brain tumor segmentation using a novel architecture called FM-Net. It leverages state-of-the-art deep learning techniques and optimization methods to achieve high levels of accuracy and efficiency in tumor segmentation. The study builds upon the BraTS Dataset.

The contributions of this paper are twofold: First, it proposes a novel neural network architecture, FM-Net, which incorporates attention mechanisms, deep supervision, and an optimized bottleneck design for improved segmentation performance. Second, it demonstrates the application of advanced training techniques like mixed precision training and GPU-optimized data loading to significantly reduce training time without compromising accuracy.

This paper is organized as follows: Following the introduction, the related works are reviewed, providing context and background for this study. The subsequent sections detail the dataset and preprocessing methods, the methodology including the proposed FM-Net architecture, and the implementation details. The experimental setup, results, and a discussion of these findings are then presented, followed by the conclusion.

II. Related Works

In the realm of brain tumor segmentation, various imaging modalities such as MRI, encompassing FLAIR, T2, T1, and T1c, play a pivotal role in diagnosis. The Brain Tumor Segmentation (BraTS) Challenge Dataset, comprising these modalities and segmentation labels, serves as a fundamental benchmark [1].

In recent studies, brain tumor segmentation has advanced through specialized architectures like the U-Net variants, which excel in handling limited data and achieving high segmentation accuracy. Multi-task approaches [2] that integrate auxiliary tasks, especially those involving image reconstruction and boundary identification alongside segmentation, have shown promise in boosting segmentation performance. Additionally, the integration of GANs in segmentation [3], aiming to generate segmentation maps from images, showcases innovative approaches to enhancing accuracy. These diverse methodologies underscore the significance of leveraging various imaging modalities, evaluation metrics, and model architectures in driving advancements in brain tumor segmentation.

III. Data and Preprocessing

The RSNA-ASNR-MICCAI BraTS Dataset plays a pivotal role in our medical image segmentation research. This dataset comprises Flair, T1, T1CE, and T2 mpMRI views, each crucial for diverse observations [1], [4]. It includes 2D brain slices, with each patient having 155 slices sized at (240, 240) pixels, resulting in a (155, 240, 240) format. Additionally, masks are provided with values 0, 1, 2, 3 representing distinct regions: 0 for black areas, 1 for necrotic tumor parts, 2 for edematous/invaded tissue, and 3 for enhancing tumors [1].

The tumor segmentation involves 3 classes: Whole Tumor (WT), which is the union of mask values 1, 2, and 3; Tumor Core (TC), the union of mask values 1 and 3; and Enhancing Tumor (ET), where mask values are equal to 3.

For each patient, there are four mpMRI scans and a corresponding mask, resulting in (4, 155, 240, 240) images and a (1, 155, 240, 240) segmentation mask pair. With 2,000 patients, the dataset contains 310,000 data points in a 2D approach or 2,000 sets of data in a 3D approach [1]. Figure 1 shows different modalities of random example with its mask.

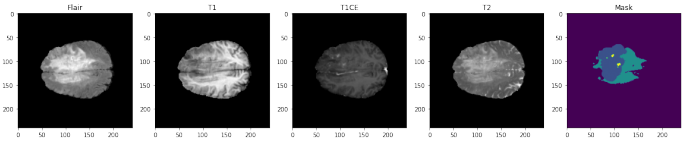

Fig. 1: Modalities of MRI: FLAIR, T2, T1, T1c (from left to right) [5]

Initially, the data is in the Channel-Depth-Height-Width (CDHW) format. To reduce disk and RAM usage during training, the data is cropped to (4, 128, 128, 128) from the DHW axes. Additionally, the depth, representing slices, is cropped because the very first and very last slices are generally empty black images, which do not contain useful information and can lead to excessive RAM usage during training.

Subsequently, the dataset is normalized based on the foreground. This normalization is particularly crucial for medical images, as they often contain substantial black backgrounds unlike natural images. Therefore, their effect on the mean and standard deviation is disregarded during normalization. Following this, data augmentation techniques are employed to enhance the model's generalization performance. The set of augmentations includes Zooming, Flipping, Gaussian Noise

and Blur, and Brightness adjustments. These augmentations are not extreme since medical images need to retain most of their information for optimal results.

IV. Methodology

A. Model Types

Unet: The U-Net architecture is a groundbreaking advancement in biomedical image segmentation, particularly for brain tumor detection. It combines an encoder-decoder structure to capture context and localize details effectively. U-Net's ability to work well with limited training data sets has set a new standard for accuracy in medical imaging[6]. *Attention U-Net*, introduces attention gates to the residual connections between the downscaling and corresponding upscaling blocks of bare bone U-Net [7].

Deep Supervision: Deep supervision in neural networks refers to the practice of adding extra supervision at the hidden layers of the network, aiming to enhance the overall performance of the network. This technique is also known as 'intermediate supervision' or 'auxiliary supervision'. The concept is centered around the integration of additional loss functions into the learning process at various hidden layers [8], [9]. A sample usage for U-Net like architecture given in Figure 2

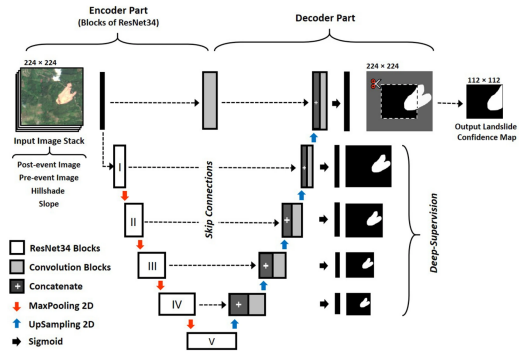

Fig. 2: Sample Deep Supervision Usage [10]

Deep supervision in neural networks offers several benefits: it enhances feature learning for improved prediction accuracy and mitigates overfitting through regularization. By incorporating guidance at hidden layers, it captures a broader range of features, leading to more precise final outputs. Simultaneously, this technique acts as a form of regularization, helping to prevent overfitting by distributing learning across multiple layers. This also improves gradient flow, aiding in efficient and effective training.[8], [9].

B. Implementation Optimization Methods

DALI: The NVIDIA Data Loading Library (DALI) is a GPU-accelerated solution that speeds up data loading and preprocessing for deep learning applications. It replaces built-in data loaders in popular frameworks and tackles the CPU bottleneck by offloading preprocessing to the GPU. DALI's own execution engine optimizes input pipelines with features like prefetching and parallel execution .

Mixed Precision: Mixed precision training is a technique used in deep neural networks that combines lower and higher precision formats, such as FP16 and FP32, to reduce memory consumption and computational time without sacrificing model accuracy. This approach enables training of larger and deeper models on limited GPU memory, accelerates training speed, and often requires minimal hyperparameter tuning. To address models with numerous small gradient values, gradient scaling is introduced, ensuring convergence to the same level of accuracy as models trained with FP32, thereby maintaining the benefits of reduced computational resources. This technique strikes a balance between resource efficiency and model performance, making it a valuable tool for deep learning tasks [11].

V. Experiments

A. Input Modality

The various approaches to processing data for medical imaging models involve utilizing different modalities either separately (such as T1 or FLAIR alone) or by combining them together along the channel dimension. Another aspect involves using 2D slices by rearranging voxel data, changing from CDHW to DCHW, or employing 3D voxel data. These methods can involve single or multiple modalities, and leveraging all modalities is crucial due to data scarcity and the complementary information they offer for brain tumor analysis.

Empirical experiments compared models trained with single-modality versus multi-modality inputs for binary segmentation tasks. Combining all modalities significantly reduced the Dice Loss to 0.08-0.1, while using only one modality resulted in a Dice Loss stuck at 0.2-0.25. Unfortunately, exploring the 3D case was not pursued.

The subsequent investigation compared models trained on 2D slice data versus 3D voxel data. The prevailing literature favors 3D data due to its retention of patient-specific spatial information, while 2D data loses this context. As anticipated, the results, as shown in Figure 3, indicate that 3D data achieved notably lower validation losses than 2D data when using the basic UNet model. Moreover, the 2D case exhibited more instability in both training and validation losses.

B. Full GPU Utilization and Mix Precision

To emphasize the effects of DALI and AMP, experiments were conducted using a sample U-Net as the foundational architecture. This U-Net was employed with both 3D and 4-channel modalities, utilizing symmetric kernel sizes of [32, 64, 128, 256, 512]. Detailed architectural designs will be discussed in a subsequent section. All experiments were conducted on an RTX 3090 GPU. The resulting impact on processing speed is presented in Table I.

	None	AMP	DALI	DALI and AMP
Avg Epoch Time (sec)	1051	640	148	**95**

TABLE I: Comparison of Epoch Times for DALI and AMP

As observed in Table I, it is evident that the inclusion of DALI and AMP separately leads to a significant reduction in

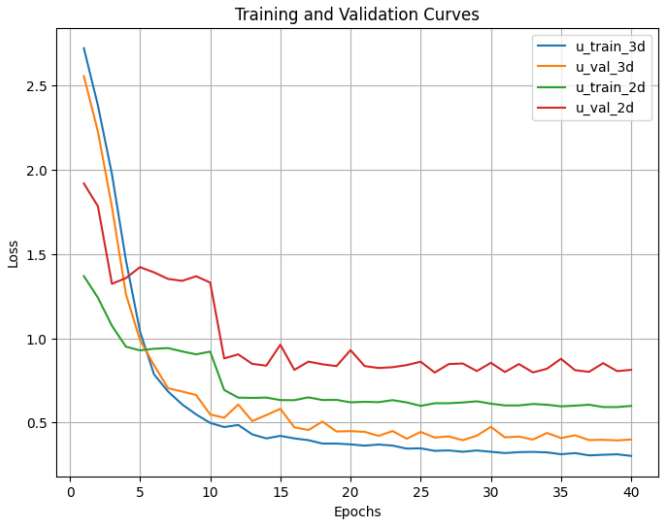

Fig. 3: 2D vs 3D Input Modality Comparison

training time. Remarkably, when both AMP and DALI are employed together, there is an approximately tenfold increase in training speed. Notably, the application of both DALI and AMP did not have a noticeable impact on the obtained results. While there are some minor deviations between them, these differences are negligible and can likely be attributed to other factors such as stochasticity and other sources of variation.

C. Loss Function

A two-part loss function is used for image segmentation. The first part is the Dice loss, which assesses the overlap between predicted and ground truth regions. The Dice loss is defined as:

$$\text{LOSS}_{\text{Dice}} = 1 - \frac{2\sum_{i=1}^{n} p_i y_i}{\sum_{i=1}^{n} p_i^2 + \sum_{i=1}^{n} y_i^2} \tag{1}$$

In this formula, y_i represnets label pixels and p_i are prediction pixels. $1 - \text{LOSS}_{\text{Dice}}$ is the *dice coefficient*.

The second part of the loss is the cross-entropy loss, which quantifies prediction uncertainty and penalizes incorrect predictions logarithmically:

$$\text{LOSS}_{\text{CE}} = -\sum_{i=1}^{n} y_i \log(p_i) \tag{2}$$

This loss is applied to calculate pixel-wise label loss. The overall loss is the sum of the Dice and cross-entropy losses.

D. Architecture Design

Based on insights from [12] and research findings in [13], we chose a U-Net-like encoder-decoder architecture as our primary model due to its computational efficiency while maintaining effectiveness in processing data. The decision to experiment with both 3D and 4-channel modalities stemmed from empirical and theoretical evidence highlighted in [13].

While the core architecture was U-Net-based, we explored alternative designs and state-of-the-art techniques. These explorations involved varying parameters such as kernel and layer numbers, incorporating bottlenecks of different sizes, implementing deep supervision, and integrating attention mechanisms.

Initially, experiments focused on adjusting kernel and layer numbers within the Attention U-Net architecture, considering computational limitations and GPU RAM availability. Using low kernel numbers ([32,64], [8, 16, 32, 64]) led to inadequate model capacity, hindering its ability to capture data complexity. Conversely, extremely high kernel numbers ([32,64,128,256,512,1024,2048]) were infeasible due to GPU RAM limitations. Therefore, we settled on [32, 64, 128, 256, 512] for subsequent experiments, ensuring a balance with 10 convolutional layers in the encoder and a symmetric decoder. This configuration struck a balance between model power, avoiding underfitting, and staying within computational constraints. It enabled effective feature capture and generalization for both Attention U-Net and U-Net models.

U-net and Attention Gated U-Net are compared on the basis of [32,64,128,256,512] architecture. Their resulted comparison graph is given below as Figure 4

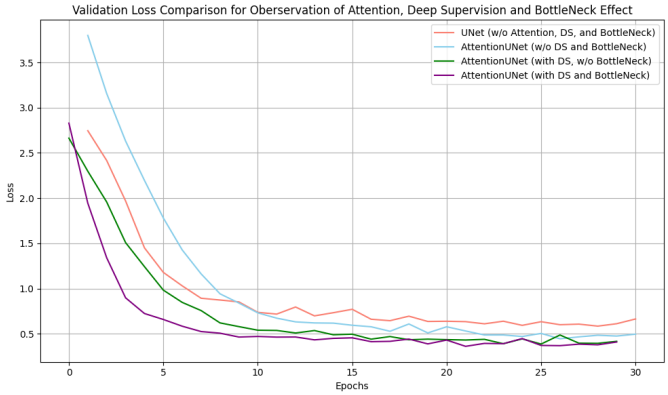

Fig. 4: Validation Loss Comparison for Oberservation of Attention, Deep Supervision and BottleNeck Effect

As evident from Figure 4, the addition of the attention gate to the networks has a positive impact on the results. It is clear that the Attention U-Net demonstrates greater potential in terms of representation power. Consequently, we have selected the Attention U-Net as the foundational architecture for the subsequent phases of experimentation and as the basis for the final architecture.

Following these initial steps, deep supervision was introduced to the network to assess its impact on validation results. As indicated in Figure 4, it is evident that deep supervision plays a crucial role in overcoming the limitations encountered when attention mechanisms are used in isolation. The addition of deep supervision leads to a noticeable improvement in overall results. In the graph depicting the results obtained for the optimal number of deep supervision, which is 2, it becomes evident that increasing the number beyond this threshold has a detrimental effect, imposing an excessive constraint on the decoder and yielding unfavorable results. Conversely, utilizing only a single deep supervision did not yield any observable impact. Therefore, the decision was made to incorporate 2 instances of deep supervision into the subsequent phases

of experimentation, as it struck a suitable balance between regularization and architectural flexibility.

After completing the aforementioned stages of experimentation, the addition of a bottleneck to the network was explored. As illustrated in Figure 4, the inclusion of a bottleneck significantly accelerated the convergence speed. Although the final convergence point remained similar, the bottleneck introduced additional representation power and enhanced the convergence rate. Therefore, the decision was made to incorporate the bottleneck into the final architecture.

The number of bottleneck layers was examined through experimentation. Increasing the number of bottlenecks leads to both loss of information due to numerous transformations and a significant increase in model parameters, especially in the 3D case. Moreover, we experimented with 1, 2, 3, and 4 bottleneck layers and observed the following trends: Having 1 or 2 bottleneck layers improves the results, with 2 yielding the best outcome. However, employing 3 and 4 bottleneck layers increases the number of parameters to a great extent without improving the model. In fact, using 4 bottleneck layers diminishes the model's generalization capability.

E. Final Architecture

The final architecture, referred to as *FM-Net*, is a carefully designed, optimized, and fine-tuned version of the best results obtained during experimentation. An overview of the FM-Net architecture can be found in Appendix A. FM-Net incorporates a combination of various methods and techniques that have been found to be effective.

FM-Net consists of five layers of convolutional blocks in the encoder, symmetrically mirrored by the decoder. At each layer, the convolutional blocks employ convolution layers with 32, 64, 128, 256, and 512 kernels, respectively. Each convolutional block comprises two convolution layers with an instance normalization layer positioned between them. The convolution layers use 3x3x3 kernels, and LeakyReLU activation with a slope of 0.02 is applied. The convolution layers maintain the spatial dimensions of the data. For down-sampling, a 2x2x3 Max Pooling operation is employed, effectively reducing the spatial dimensions by half. For up-sampling, transpose convolutions are utilized, doubling the spatial sizes.

Incorporating insights from experimentation, FM-Net features a two-layered bottleneck with 512 kernels and 3x3x3 kernel size. Additionally, between the encoder and decoder layers, residual connections are introduced using cross-attention mechanisms. The cross-attention calculation occurs between the output of the encoder layer and the previous layer's output, which is then concatenated with the input to the decoder.

To further enhance the architecture, two instances of deep supervision are introduced, one in the third layer and another in the fourth layer of the decoder. The output feature maps from these layers pass through a convolutional layer and are interpolated using trilinear interpolation to match the desired output format. Then all three output prediction used for loss calculation.

FM-Net is designed to produce outputs with three channels, each channel representing the segmentation results for one semantic class.

During the training process, the mixed precision method is employed to enhance training efficiency. The optimization is carried out using the Nesterov Adam optimizer with a learning rate of 0.003 and beta values set to (0.9, 0.999). The training process spans 70 epochs. The model saved is the one that corresponds to the best validation result achieved during training.

VI. Results

A. Benchmark Results

Before proceeding with the results, the benchmark model was trained and tested for comparison purposes. Nvidia's Optimized U-Net for Brain Tumor Segmentation [14] was chosen as the benchmark, given its top ranking in the BraTS Challenge contest.[1] The model was downloaded from the GitHub repository, trained, and tested, achieving a ***0.08 test loss, 0.98, 0.98, 0.97 Dice Coefficient*** for Whole Tumor, Tumor Core, and Enhancing Tumor labels, respectively.

B. Final Architecture Results

After deciding on the final architecture, its desgin, and optimizer parameters, the model is trained again with longer epochs. Figure 5 shows the training and validation losses at the LHS of the plot and shows the validation Dice Coefficient Scores for each segmentation label at RHS of the plot. As seen, the loss progressively decreases whereas the Dice Coefficient for each segmentation labels increases up to around 0.9, which is a very satisfactory results. Then the model is tested on the test set and the Table II shows the results.

Metric	Value
Test Loss	0.325
Test DICE (Whole Tumor)	0.925
Test DICE (Tumor Core)	0.914
Test DICE (Enhancing Tumor)	0.893

TABLE II: Results of the Brain Tumor Segmentation Model

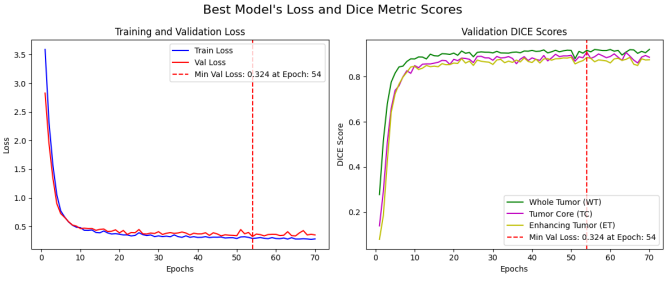

Fig. 5: Best Model's Loss and Dice Metric Scores Plot

C. Sample Test Results

Given the existence of three segmentation labels, every sample is accompanied by three segmentation masks and corresponding predictions. To adequately present these results, the

[1]https://github.com/NVIDIA/DeepLearningExamples/tree/master/PyTorch/Segmentation/nnUNet

outcomes of this section have been relocated to the Appendices section. Appendix B displays the results for 12 samples across three figures.

The obtained results demonstrate a high level of satisfaction, given that the Dice coefficients of the segmentation labels are notably close to those achieved by the champion model. Figures 7, 8, and 9 are structured to represent various scenarios: each row corresponds to different samples, while columns depict the original image, segmentation mask for the first class, prediction mask for the first class, and the same for the second and third classes, respectively.

Figure 7 may seem peculiar; however, it holds paramount significance. It illustrates the model's capability to detect the absence of tumors, hence generating a null mask for normal patients. This particular aspect poses a considerable challenge in medical imaging. Figure 8 showcases instances of incorrect or incomplete results. As depicted, the predictions do not entirely encompass the label, leading to discrepancies. Lastly, Figure 9 presents highly satisfactory outcomes, where the predictions closely align with the label, showcasing a remarkable similarity.

D. The Effect of Deep Supervision Idea

In this section, the outputs of the deep supervision blocks are analyzed to assess their impact. Appendix C demonstrates their results. Figure 10 represents a randomly selected sample from the test set, while Figure 11 is a grid plot. In this grid plot, the rows correspond to different mask labels, and the columns correspond to the segmentation mask, the first and second Deep Supervision output, and the final Prediction, respectively.

As depicted in Figure 11, the Deep Supervision outputs emphasize the segmented regions and mask the non-segmented areas. Progressing from the first to the second Deep Supervision output, the non-segmented parts are further masked out. Eventually, at the final layer, we obtain a refined segmentation mask. Therefore, Deep Supervision confirms our hypothesis, significantly enhancing the accuracy and robustness of the results, while also aiding in refining the segmentation boundaries.

VII. Implementation Details

The model was trained on RTX 3090 GPUs, and both the architecture and training codes were implemented in PyTorch. For reference, all of the code related to this project can be found on the authors' GitHub repository.[2]

VIII. Conclusion

This study presented a comprehensive approach to brain tumor segmentation in MRI images, introducing the FM-Net architecture, a novel adaptation of the U-Net model enhanced with attention mechanisms, deep supervision, and an optimized bottleneck structure. The research demonstrated the efficacy of this architecture in accurately segmenting brain tumors, as evidenced by high Dice Coefficient scores and low loss values.

Key to the success of this approach was not only the architectural innovations but also the application of advanced training techniques such as mixed precision training and GPU-optimized data loading, which significantly reduced training times without sacrificing model performance.

[2]https://github.com/fuat-arslan/MRI-Segmentation

References

[1] U. Baid and S. G. et. al, "The rsna-asnr-miccai brats 2021 benchmark on brain tumor segmentation and radiogenomic classification," 2021.

[2] H. Shen, R. Wang, J. Zhang, and S. McKenna, "Multi-task fully convolutional network for brain tumour segmentation," in *Medical Image Understanding and Analysis*, M. Valdés Hernández and V. González-Castro, Eds. Cham: Springer International Publishing, 2017, pp. 239–248.

[3] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014.

[4] D. Jin, Z. Xu, A. P. Harrison, and D. J. Mollura, "White matter hyperintensity segmentation from t1 and flair images using fully convolutional neural networks enhanced with residual connections," 2018.

[5] J. Hu, X. Gu, and X. Gu, "Dual-pathway densenets with fully lateral connections for multimodal brain tumor segmentation," *International Journal of Imaging Systems and Technology*, vol. 31, 08 2020.

[6] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015.

[7] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz, B. Glocker, and D. Rueckert, "Attention u-net: Learning where to look for the pancreas," 2018.

[8] R. Li, X. Wang, G. Huang, W. Yang, K. Zhang, X. Gu, S. N. Tran, S. Garg, J. Alty, and Q. Bai, "A comprehensive review on deep supervision: Theories and applications," 2022.

[9] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," in *Artificial intelligence and statistics*. Pmlr, 2015, pp. 562–570.

[10] N. Prakash, A. Manconi, and S. Loew, "A new strategy to map landslides with a generalized convolutional neural network," *Scientific Reports*, vol. 11, no. 1, p. 9722, May 2021. [Online]. Available: https://doi.org/10.1038/s41598-021-89015-8

[11] P. Micikevicius, S. Narang, J. Alben, G. Diamos, E. Elsen, D. Garcia, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh, and H. Wu, "Mixed precision training," 2018.

[12] F. Arslan and M. Yilmaz, "Brain tumor segmentation and classification project progress," 2023.

[13] ——, "Literature survey on semantic segmentation of brain tumor," 2023.

[14] M. Futrega, A. Milesi, M. Marcinkiewicz, and P. Ribalta, "Optimized u-net for brain tumor segmentation," 2021.

APPENDIX A
FM-NET ARCHITECTURE

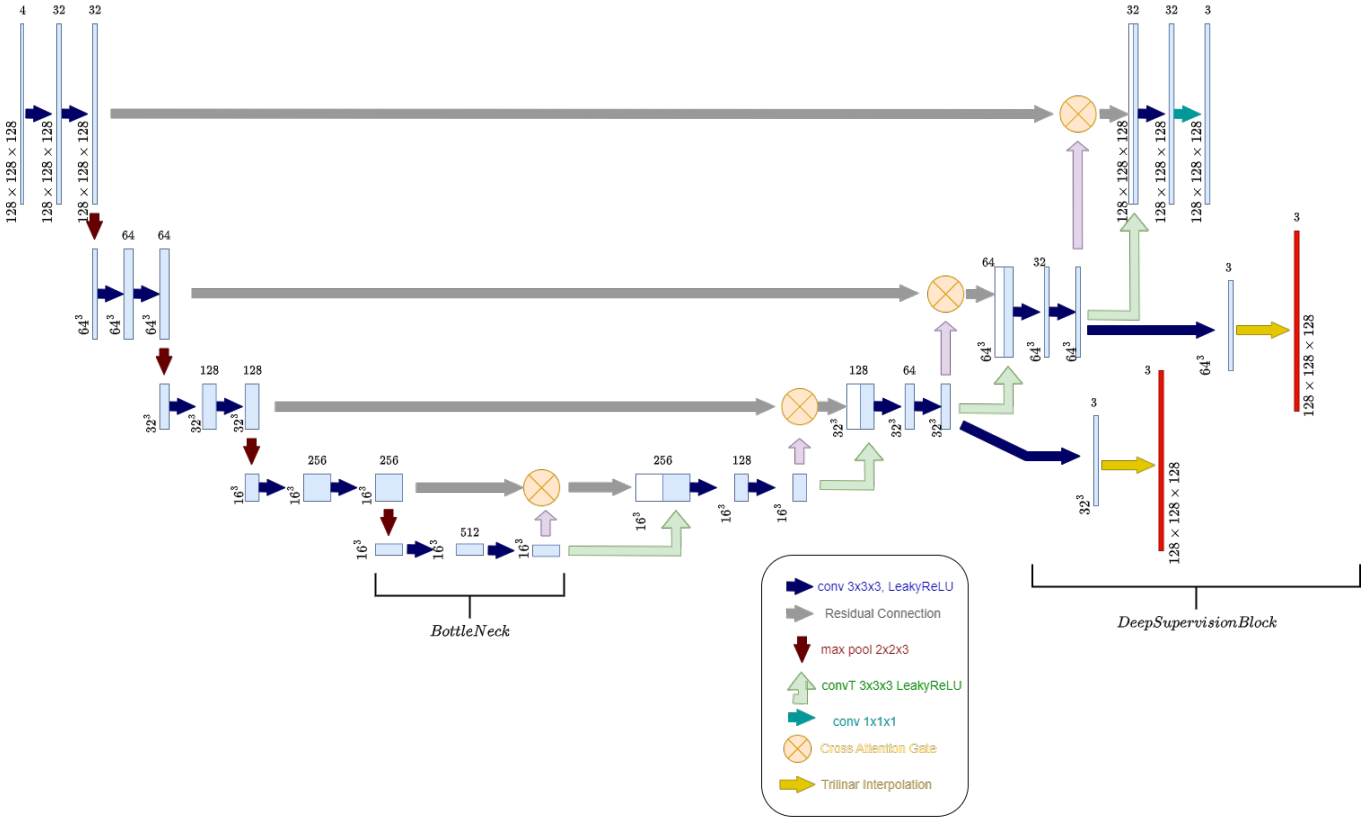

Fig. 6: Final Model Architecture

APPENDIX B
BEST MODELS' TEST SAMPLE RESULTS

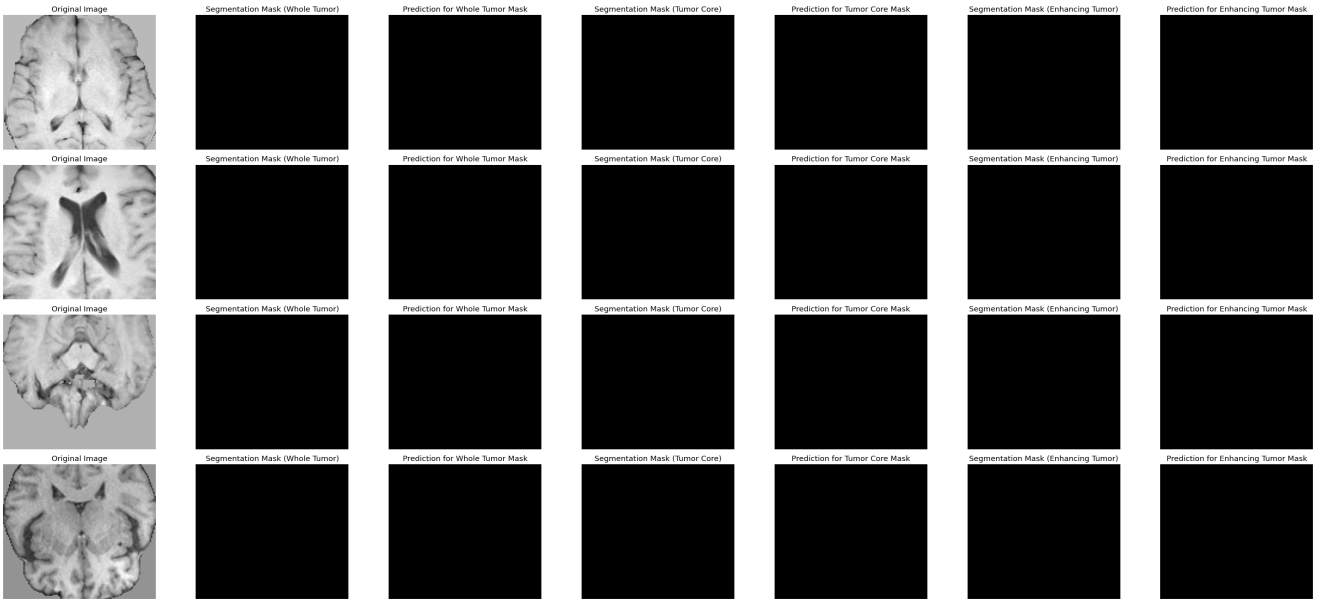

Fig. 7: The Absence of Tumor Prediction

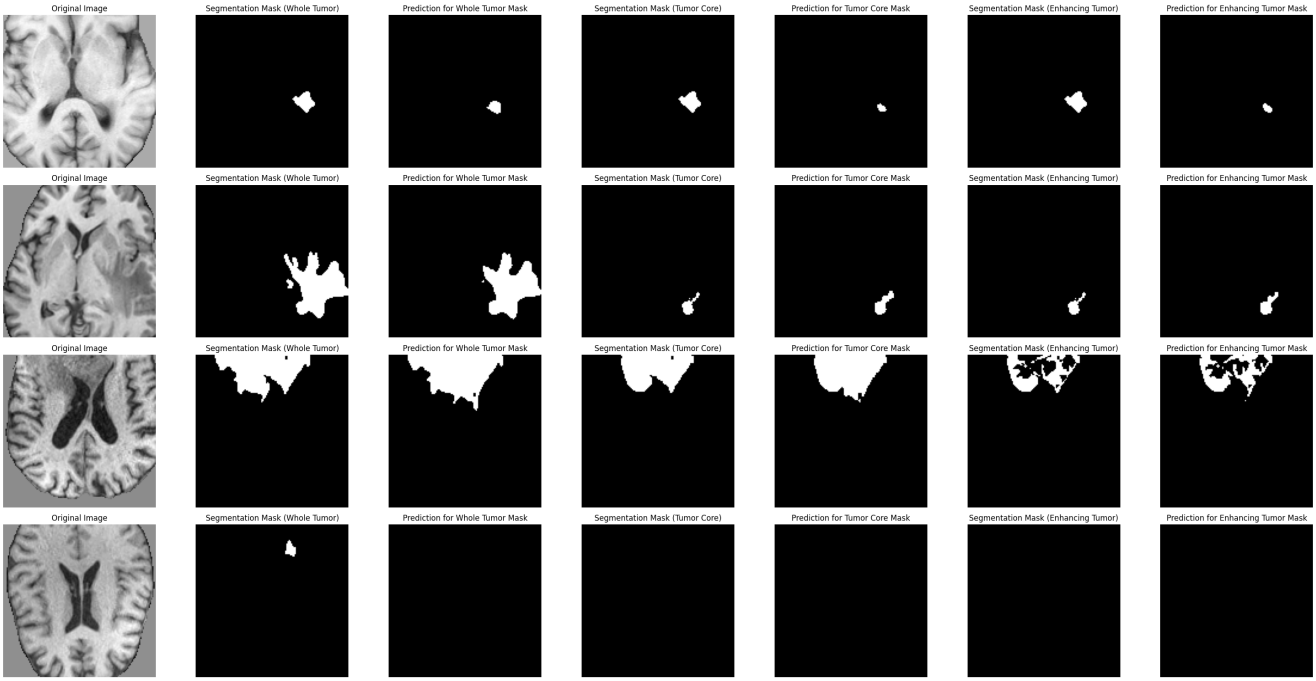

Fig. 8: Incorrect or Incomplete Segmentation

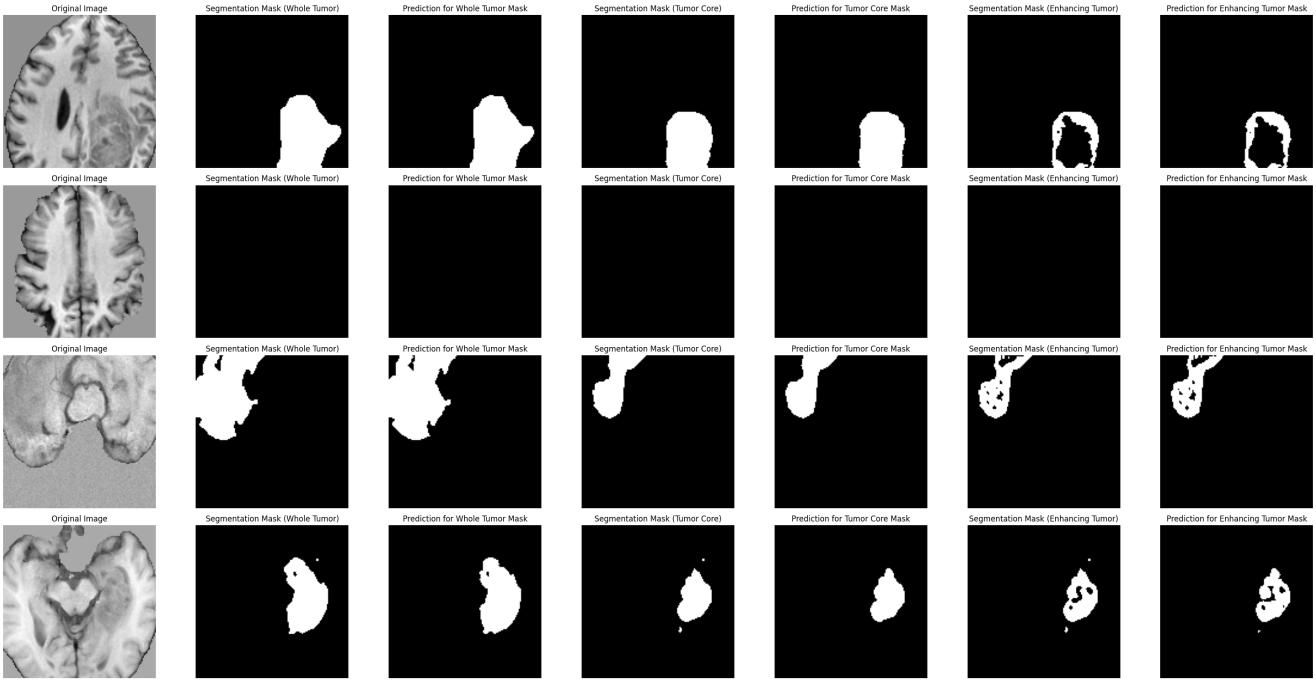

Fig. 9: Best Predictions

APPENDIX C
THE EFFECT OF DEEP SUPERVISION

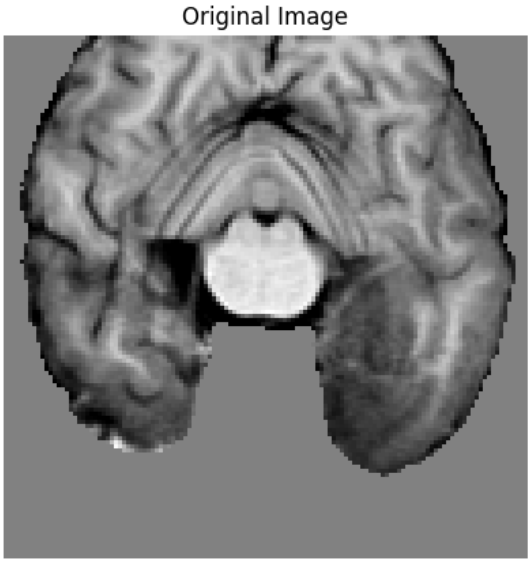

Fig. 10: Random Sample from The Test Set

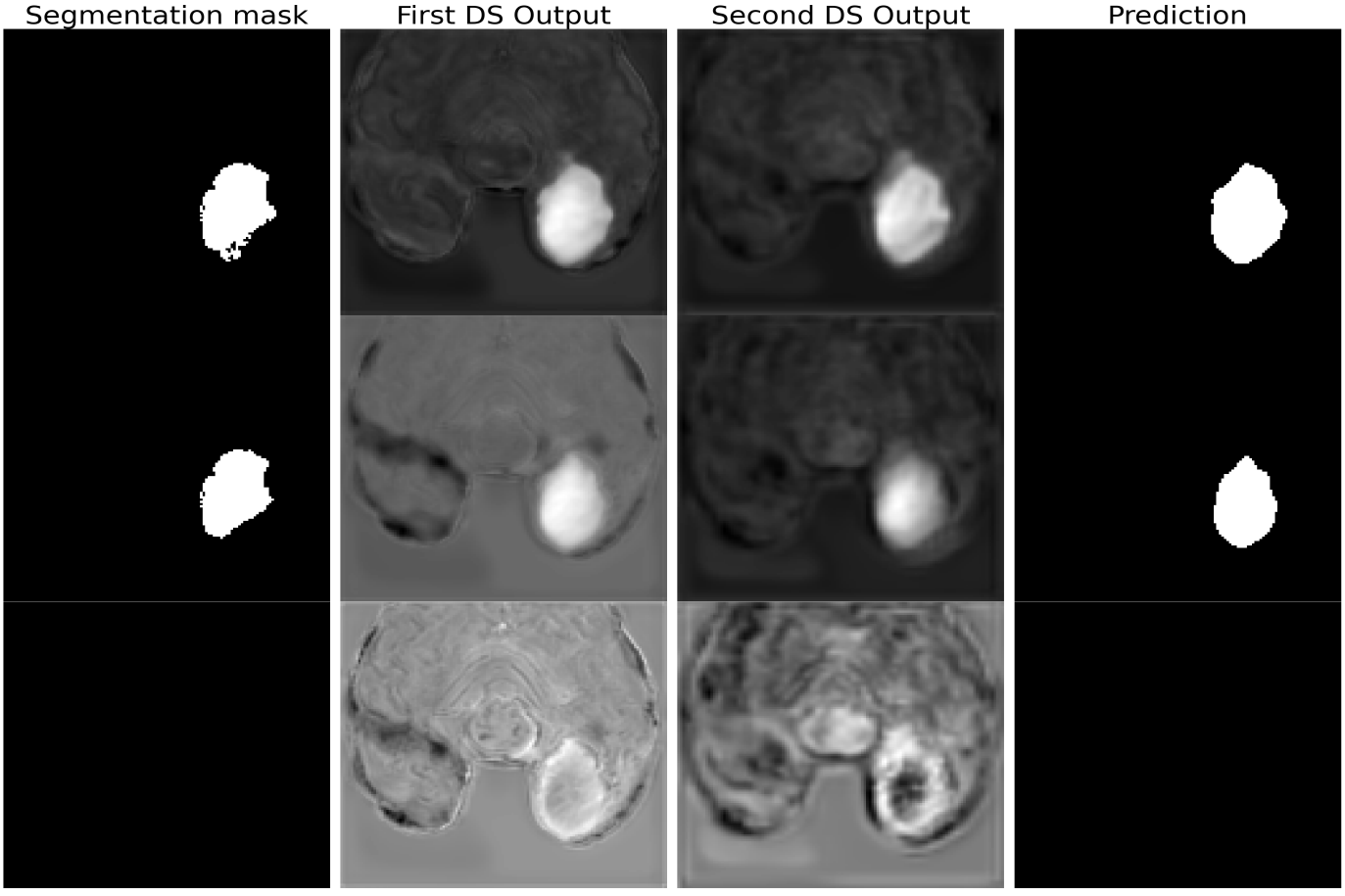

Fig. 11: Segmentation, Deep Supervision Outputs and Prediction of The Random Sample