Machine Learning in Medical Image Processing

Kévin Ferreira (500653) Veit Laule (412647) Mohammad Orabe (402831)

Final Report of the project *Machine Learning in Medical Image Processing* at the TU Berlin

This report is divided into two primary sections, addressing distinct tasks in brain tumor analysis. The first section explores segmentation, covering the SegResNet approach and two U-Net architectures (2.5D and 3D models) for delineating tumor regions in MRI scans. Both approaches include discussions on data preprocessing, model training, and result presentation. The second section focuses on the classification task, specifically predicting the genetic subtype of brain tumors based on MGMT protein presence. It encompasses challenges, ResNet10 model selection, data preprocessing strategies, and an innovative use of the 99.9th percentile for improved tumor representation.

Task 1 — Segmentation of Glioma Sub-Regions

Introduction to the Problem. When diagnosing a patient with glioma, which are the most common primary brain tumor, the patient typically undergoes an MRI scan to help understand the size and development of the tumor and aid clinician in the further treatment of patients (1). Historically the task of segmenting the tumor areas is done by radiology professionals which manually mark the specific tumor (sub-)regions in the scan. However, with the advent of deep learning (DL) in the image processing field, this task can be done by a DL model instead.

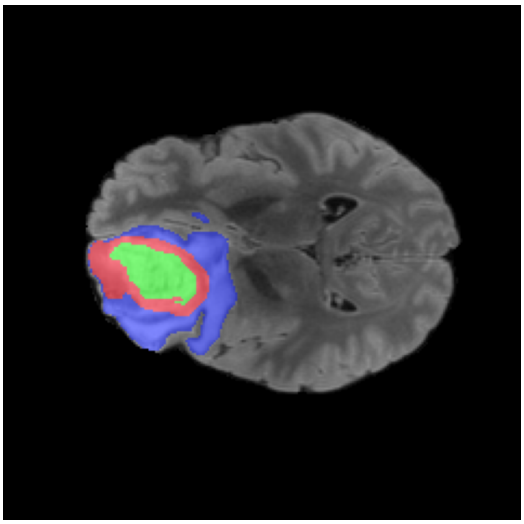

Fig. 1. MRI and segmentation mask

Such models for segmenting specific tumor sub-regions are implemented and described in the following sections. The trained models are based on stacked MRI scans from patients with gliomas with accompanying segmentation masks (see figure 1) and are later evaluated on the Dice score and Hausdorff Distance (HD) for the specific labels. The MRI scans for each subject are available in four different contrasts (FLAIR, T1w, T2w and T1wCE), to make the data available to the model more diverse.

Technical Background. The training data provided for this task is a set of 240×240×155 MRI scans with segmentation masks and 4 scan contrasts (Flair, T1, T1ce, T2) for 1000 patients. The accompanying segmentation masks represent the following segmentation labels: Background (Label 0), Necrotic Parts of Tumor (NCR) (Label 1), Edamatous/Invaded Tissue (ED) (Label 2), Enhancing Tumor (ET) (Label 4).

The relative share of the given labels over the whole dataset is relatively even, meaning it should not introduce noticeable biases to the trained model, except for the background label, which accounts for 98% of the voxels in the segmentation masks. However, this bias is dealt with in later sections of this report.

For the model architectures, we first tested a selection of well performing models from the BRATS challenge (1), which poses a similar challenge to the one posed in this project. After testing, we decided to train UNets (2) and a SegResNet model (3). The decision for this selection is based on their relative memory efficiency [1] and comparatively good performance (3). The SegResNet architecture used in this project is taken from the MONAI framework (4) and the UNet models are created from the ground up using TensorFlow (5).

The UNet model architecture, elaborated by Ronneberger et al. in their biomedical image segmentation paper (2), serves as an extension of the Fully Connected Network tailored for precise segmentations in scenarios with limited datasets. Featuring a U-shaped encoder-decoder design, the model comprises four interconnected encoder and decoder blocks, with the encoder reducing spatial dimensions and increasing feature channels, while the decoder restores spatial dimensions and refines features. The final output undergoes a 1x1 convolution, employing softmax activation to create a segmentation mask for pixel-wise classification. UNet avoids fully connected layers and opts for valid convolution to maintain essential contextual information (2). Its use of upsampling layers and the seamless fusion of high-resolution features contribute to precise localization, making UNet a powerful tool for smooth segmentation, especially when handling large images through an overlap-tile strategy.

The SegResNet architecture was introduced by Myronenko in (3) as a contender in the BRATS 2018 challenge. It is structured like a standard UNet (2) and consists of ResNet blocks which follow the specified structure: Group Norm → ReLu → 3×3×3 Convolution → Group Norm → ReLu →

[1] The compute available in this project is restricted to a 12GB Nvidia 2080ti GPU

3×3×3 Convolution. These blocks are arranged in 4 levels consisting of 1, 2, 2 and 4 blocks respectively for the encoder part, where the image size is decreased by a factor of two between each level, while the number of features is increased by the same amount. The encoder then increases the feature size again between each level and decreases the number of features. The original model introduced in (3) also includes a VAE to aid the model in training, which was left out here due to memory constraints.

Data Pre-Processing and Training. For the data pre-processing and training of the SegResNet and the UNet model architectures we decided on implementing and comparing a selection of different approaches described in detail in this section.

SegResNet Model. For the SegResNet model, we chose the configuration as depicted in table 1 as our baseline. This also included Intensity normalization, random flips along all image axes, spatial cropping to an area of 128×128×128 and random image shifting for data augmentation, as to not let the model overfit too easily on the given dataset. Additionally, all the training of the SegResNet model was done using mixed precision to make the model more memory efficient when compared to training with full precision. Lastly, all the model variations described in this section were trained on labels 1, 2 and 4 only to avoid the class imbalance, that the background label (0) would introduce. This label was then later calculated by taking the negative of the other label masks, which resulted in a dice-score of 99% for all model variations.

Hyperparameter	Value
Learning Rate	1e-4
Learning Rate Scheduler	CosineAnnealingLR (4)
Batch Size	1
Loss	Dice Loss (4)
Optimizer	Adam
Number of Convolutional Filters	16

Table 1. SegResNet Baseline Hyperparameters

We then first trained the model with the data augmentations mentioned above and the hyperparameters from table 1 for 300 epochs on a (60, 20, 20) split. This yielded an average dice score of 0.846 for our baseline. As this configuration already showed promising results, we then continued training with the variations descibed below.

Reducing the amount of slices from 128 to 80 firstly resulted in shorter training time and secondly eliminated noise from the dataset, as most MRI-scans only included actual brain scans in 80 of the total 155 slices 2. This resulted in a minor improvement with respect to the average Dice score (0.846 for the baseline vs. 0.847), but a considerable reduction in training time. As the models' performance did not degrade when reducing the amount of slices, we opted for training the rest of the model variations with this slice range as well.

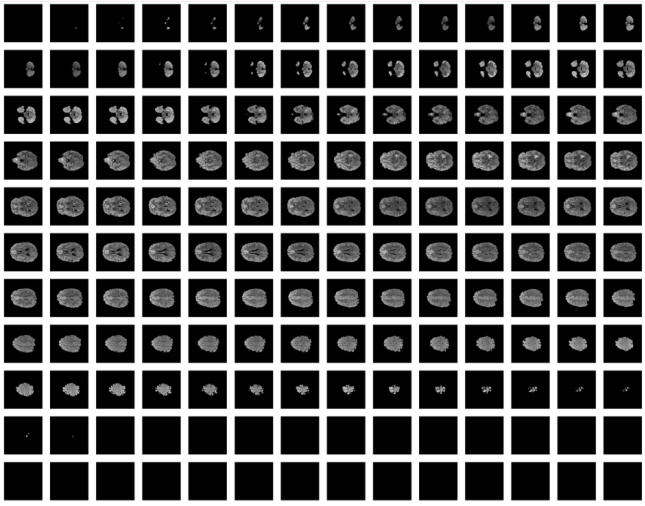

Fig. 2. Flair MRI slices of subject 00006

The second variation we introduced was in regard to the loss applied during training. For the baseline model we used the Dice Score supplied by Monai (4). The losses we tested the SegResNet model with were all taken from successful competitors in the BRATS challenge. The Generalized Wasserstein Dice Loss introduces a distance matrix to supplement the standard dice loss with known relationships between segmentation labels in a multilabel segmentation task (6), as in this project. However, applying the distance matrix from the original paper to the dataset used in this project did not yield satisfactory results, even though the labels used in the original paper and this project are similar. Another loss that was tested but did not improve the models performance is the Hausdorff Distance Loss (7), which aims to train the model to directly reduce the Hausdorff loss, but led to out of memory errors when tested on the compute available in this project. The last loss that was applied to the model is the DiceCE loss (4), which is a combination of the Cross Entropy loss often used in training of segmentation models and the Dice loss used for the baseline. This loss finally gave a slight improvement when compared to the baseline (0.846 for the baseline vs. 0.848).

With two promising model variations, we continued training for an additional 500 epochs in both cases. This yielded an average dice score of 0.855 for both models which is a definite improvement over the baseline, but does not result in any meaningful differentiation between the model variations.

UNet Models. In the UNet model architecture, we employed two model architectures: a 2.5D UNet and a 3D UNet. These models were strategically chosen to strike a balance between computational efficiency and the comprehensive utilization of three-dimensional spatial context. Each model is discussed in detail in the subsequent sub-paragraphs.

2.5D UNet Model. The 2.5D UNet model is designed to process image slices along with limited context information to optimize computational efficiency. The efficiency of this model surpasses that of a 3D UNet, exhibiting faster processing speed and lower memory requirements.

The approach involves a strategic choice in data representation. While experts often benefit from the four modalities (FLAIR, T1w, T2w and T1wCE) for comprehensive tumor analysis, our focus on computational efficiency led us to employ only two modalities, specifically, we excluded the T1 modality in favor of its improved version, T1wCE. The exclusion of the T2w modality is motivated by potential degradation in predictions caused by fluid presence. The T1wCE and FLAIR modalities provide complementary information about the anatomy and tissue contrast of the patient's brain.

Additionally, we noticed a substantial number of slices that exhibit limited information. This trend persists across all the modalities. In response to this pattern, we excludes these less informative slices. Specifically, we found the range (60:135) to be particularly influential in refining our dataset for analysis.

The 2.5D UNet model unfolds in the axial plane, leveraging the square shape of the images (240x240) within this plane. Another crucial step we made involves resizing each image slice from (240x240) to a (128x128) format. This resizing is imperative as it aligns with the requirement for image shapes to be powers of two. This necessity arises from the incorporation of pooling layers (MaxPooling2D) in our convolutional neural network (CNN), which systematically reduces spatial resolution by a factor of 2.

We also deployed One-Hot Encoding to the segmented regions to transform the categorical regions into a numerical representation, making it compatible with our neural network, which operates on mathematical equations.

We employed a distribution ratio of 73:15:12 for the training, validation, and testing sets, resulting in 680, 200, and 120 images, respectively, in each set. The training process is set to run for 30 epochs, with each epoch processing a batch size of 1. This means that during each epoch, the model receives 680 samples, each comprising a volumetric shape of (128, 128, 75, 2), representing the input medical image data. The corresponding ground truth segmentation for each sample is shaped as (128, 128, 75, 4), encapsulating the segmented regions across four classes. The model is optimized using the Adam optimizer with a learning rate of 0.001. For the loss function, categorical cross-entropy is employed, allowing the model to evaluate the difference between predicted and true segmentation. Additionally, the output layer of the neural network is equipped with a softmax activation function, facilitating the transformation of raw scores into probability distributions across the four output classes.

In the evaluation of model performance (figure 3.A), both training and validation accuracies exhibit notable improvement over epochs, reaching a plateau and suggesting successful learning and generalization without signs of overfitting. Simultaneously, the loss graphs (figure 3.B) demonstrate a consistent decrease in both training and validation losses, emphasizing the model's learning dynamics. The optimal model state is observed around epoch 22 , as indicated by the training logs.

We utilizied the best model weights obtained at epoch 21. To facilitate predictions and evaluations, we visualized and

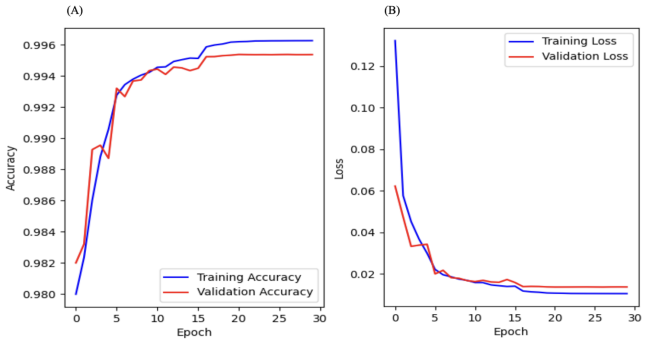

Fig. 3. Training and validation accuracy (A) and training and validation loss (B) for the 2.5D-UNet model over epochs. The plots illustrate the learning dynamics, showcasing consistent improvements in accuracy and decreases in loss. An optimal state is reached around epoch 22.

compared the model's predictions with the original segmentations in the axial plane. While examining the results, we observed that some false positives occur, indicating tumor detection where none is present originally (figure 4). To address this issue, we performed a post-processing techniques such as argmax decoding, thresholding, and morphological operations to refine the predicted segmentations. The argmax function is applied to assign a single label to each pixel based on the class with the highest probability. This step not only refines predictions but also ensures consistent display colors between the original segmentation and the predictions.

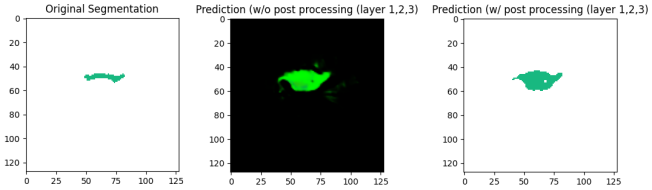

Fig. 4. Comparison of ground truth segmentation (left), model prediction on the test set (middle), and post-processed image using argmax decoding (right). The post-processed image refines the predictions by eliminating some of the artifacts introduced during the initial inference.

In summary, the 2.5D UNet model demonstrated a good performance on the test dataset, achieving an average Dice coefficient of 0.73 (0.82, 0.79, 0.70 for enhancing, edema, necrotic classes). However, while our results demonstrate high accuracy, occasional false positives in our predictions highlight the need for a nuanced evaluation in medical imaging. To address this, we explore potential improvements with the 3D UNet model in the next subsection, leveraging its ability to capture volumetric information for enhanced segmentation precision.

3D UNet Model. A 3D UNet model can leverage the 3D spatial context of the images, which means it can reduce the risk of false positives and false negatives that can occur due to partial or incomplete information in individual 2D slices.

In contrast to the 2.5D UNet model, we utilized all four modalities (FLAIR, T1w, T2w, and T1wCE) in the 3D UNet model. Image intensity normalization was applied by dividing each image by its maximum intensity to scale them within the range [0,1]. To enhance computational efficiency, we also cropped the images to a size of 128x128x128, specifically

[24:216], [24:216], [13:141] in each dimension. As in the 2.5D UNet model, this decision was motivated by the desire to have dimensions that are powers of 2, aligning with the requirements of the pooling layers (MaxPooling2D) in the convolutional neural network (CNN) as discussd earlier. During data processing, we observed that numerous slices lacked segmentation labels, with much annotated data corresponding to the background (0 value). To address this, we retained only relevant images based on a mask ratio threshold of 1%. If the volume (with all modalities together) has less than 1% mask, the subject was discarded. This led to the exclusion of 79 subjects, resulting in a dataset of 921 subjects. The dataset was then split into training, validation, and test sets with a ratio of 70:15:15. Similar to the 2.5D UNet model, we utilized one-hot encoding to convert categorical classes into a numerical representation. The input shape for the 3D UNet model is (128, 128, 128, 4), and the output shape is (128, 128, 128, 4). We applied a softmax activation function to the output layer.

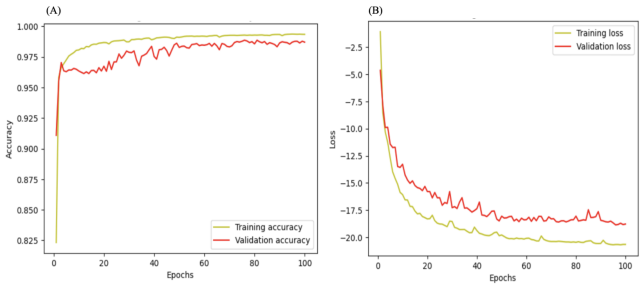

Fig. 5. (A) Training and validation accuracy and (B) training and validation loss curves for the 3D-UNet model across epochs. The plots demonstrate the learning dynamics, revealing consistent accuracy improvements and loss reductions. The model reaches an optimal state around epoch 100.

The 3D UNet model also adopts a contraction-expansion architecture with convolutional and transposed convolutional layers, integrating 3D spatial context to minimize false positives and negatives arising from incomplete 2D information. The 3D model employs dropout rates, max-pooling, and upsampling operations similar to the 2.5D model but accommodates the additional dimension. The chosen hyperparameters for the 3D UNet model include 100 epochs, a batch size of 2 (322 samples per epoch), the Adam optimizer with a learning rate of 1e-4, and a combined loss function of Dice loss and Focal loss. Early stopping was implemented with a patience of 5 and a minimum delta of 1e-3.

In the evaluation of model performance the model showed stability after 100 epochs, with minimal fluctuation in both training and validation losses The optimal model state, minimizing the loss, is achieved at epoch 100 as in Fig. 5.B. In addition, the model attains near-perfect accuracy close to 1, indicating effective learning from the training data as in Fig. 5.A.

In our evaluation, we utilized additional metrics to provide a thorough assessment of the 3D UNet model's segmentation performance. Specifically, we computed Dice coefficients and Hausdorff distances to gauge the accuracy and dissimilarity between predicted and ground truth segmentations (Fig. 6). The Dice coefficients, including values for Dice

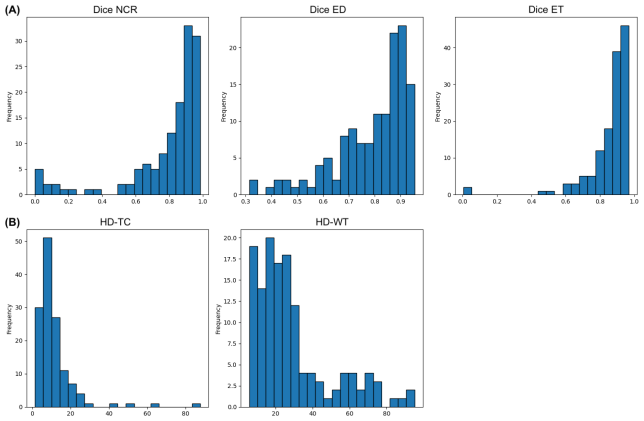

Fig. 6. (A) Histograms depicting the distribution of individual Dice scores for NCR, ED, and ET classes in the 3D U-Net model. (B) Histograms representing the distribution of individual Hausdorff distances (HD-TC and HD-WT).

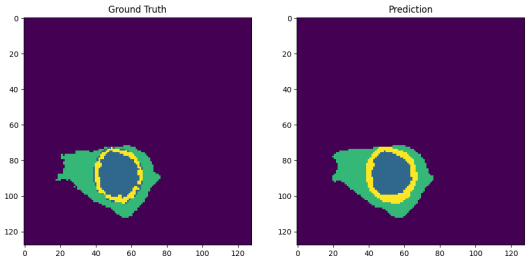

Fig. 7. Comparison of ground truth segmentation (left) and segmentation result using the 3D U-Net on the test set (right).

NCR (0.782), Dice ED (0.789), and Dice ET (0.856), underscore the model's precision in delineating distinct regions within medical images. The calculated average Dice coefficient of 0.809, with a standard deviation of 0.04. Furthermore, the computation of Hausdorff distances, with values of 11.373 for Tumor Core (HD-TC) and 28.916 for Whole Tumor (HD-WT), further highlights the model's effectiveness in accurately outlining tumor regions. Fig. 7 shows a comparison between the ground truth and the predictions on a random sample form the test set illustrating the model's performance and its effectiveness in accurately delineating the tumor regions.

Final Model and Average Output. We finally decided on the best performing SegResNet model, as described in section . This decision was mainly based on the SegResNet model performing slightly better than the UNet model on the validation set, though the difference between both models is relatively small. However, the SegResNet model still performed better in most of the tasks in this challenge (Dice NCR, HD TC and HD WT). The final results on a validation set, that consists of 20% of the available data can be seen in table 2.

The final model performs quite well on the test data and with an average Dice score of 0.809 (excluding the background label), it is close to the actual segmentation masks provided for the data. Still, in some cases, especially for small tumor

Mean Dice	Dice NCR	Dice ED	Dice ET	HD TC	HD WT
0.809	0.768	0.825	0.835	9.786	18.825

Table 2. Final Results of the Segmentation Task

labels, the model does not predict any segmentation masks at all. This relationship between the actual label size and the achieved Dice score can be seen in figure 8 and is especially noticeable for the NCR label.

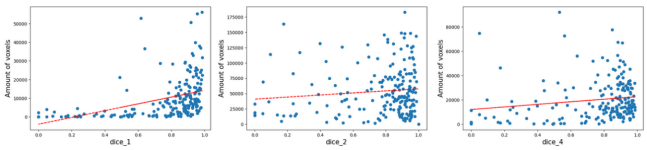

Fig. 8. Relationship between label size and Dice score

Although the model fails to correctly predict small tumor sizes, its overall performance can still be considered good. Most of the predictions we looked at manually place the tumor at the correct position and fit the general outline quite closely, as can be seen in figure 7.

Conclusion. While our results showcase high accuracy, it's crucial to acknowledge the presence of false positives. In the medical imaging domain, carefully weighing the trade-offs between true and false positives becomes pivotal. Evaluating the risks and benefits associated with our approaches underscores the importance of a nuanced interpretation of the model's performance in a medical context.

Task 2 — Genetic subtype of brain tumors (glioblastoma) prediction

Introduction to the Problem. The classification of patients according to MGMT protein from MRI images represents a crucial challenge in the field of early detection of brain tumors. Indeed, the presence of MGMT protein is a significant indicator of tumor development, directly influencing therapeutic strategies and clinical outcomes.

The challenge here is to classify patients according to the presence or absence of the protein, based on MRI images acquired under different contrasts, with each contrast providing specific information. For example, T1w highlights blood vessels, providing valuable clues to brain tissue vascularisation,

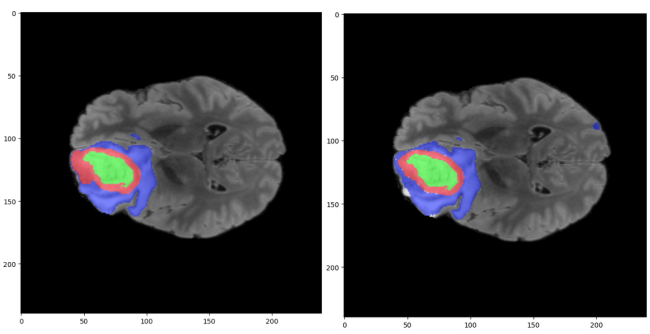

Fig. 9. True labels and SegResNet predictions.

while T2w highlights variations in tissue water content, providing information on cell composition and structure.

A set of MRI images under four different contrasts, as it can be seen in figure 10, (FLAIR, T1w, T2w, and T1wCE) collected from 468 patients is provided. These images are associated with indications of the presence of the MGMT molecule in their bodies, recorded in a CSV file.

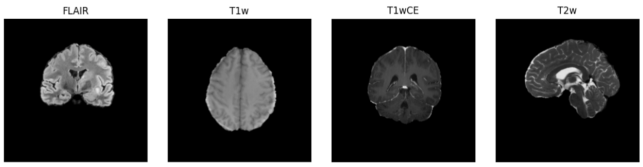

Fig. 10. Central images of a patient's 4 modalities

Technical Background. The dataset includes images and corresponding labels from 468 patients. However, variations remain in the number of images per patient, as well as between two contrasts for the same patient, resulting in a potential fluctuation in the number of images. This disparity represents a significant challenge in terms of standardizing the model to deal effectively with these variations.

Beyond diversity in terms of image quantity, variability extends to the possibility of acquiring images in distinct planes. While this diversity enriches our dataset, it simultaneously increases its complexity. However, the different brain orientations to which the model is exposed play a crucial role in improving its ability to discern patterns across a diverse range of perspectives.

Furthermore, it should be noted that the dataset is balanced, thereby reducing the risk of overfitting. This balance between classes helps to reinforce the robustness of our classification model by avoiding an overfitting on specific classes, which would be prejudicial to its generalization to new data.

The initial decision was to explore two different models, ResNet10 (4) and VGG16 (8), to determine which would perform best on our dataset. The Residual Network with 10 layers was chosen for its ability to overcome the problem of gradient disappearance in deep networks by using residual connections. What's more, ResNet10 offers a good balance between model complexity and performance, making it suitable for our problem. On the other hand, VGG16 is a convolutional neural network known for its simple, deep, 16-layer architecture designed for image classification. Although VGG16 is powerful in image recognition, its drawback lies in the high number of parameters, which can lead to an increase in training time and a risk of over-fitting, especially when training data is limited.

In order to perform the comparison between the two models, we trained and tested the models on the central images of each patient for each mode. We reserved 30% of the dataset for testing and used the remaining 70% for training. The accuracy obtained for ResNet10 was 0.61, while it was 0.57 for VGG16. ResNet10 was then the natural choice for the continuation of the study. This choice was justified by its performance in our classification task. ResNet10, characterized

by direct connections between layers, facilitates deep learning by avoiding gradient vanishing problems. This feature played a crucial role given the complexity of our task, linked to image variability and the need to extract discriminating features.

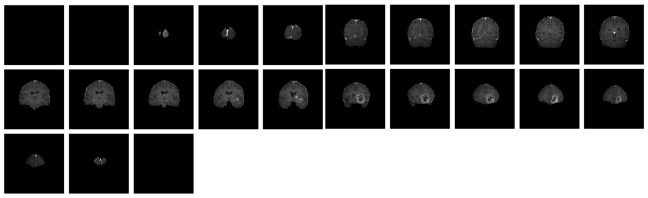

Fig. 11. T1wCE images of a patient

Data Pre-Processing and Training. A close observation of the patient images shown in figure as it can be seen in figure 11, revealed that many images did not contribute significantly to the classification task. Indeed, some images were entirely black, particularly at the beginning and end of the folders. This finding led to consideration of how to optimize data use and reduce model training time. In order to maximize the relevance of information while eliminating superfluous images, we selected a set of images centered on the main image of each MRI and scaled to 255x 255 pixels. The underlying objective is to retain crucial details while eliminating unnecessary information, thereby reducing the model complexity associated with an abundance of images. This strategy also helps to reduce the perturbations caused by fully black images, thus improving the quality and relevance of the training data.

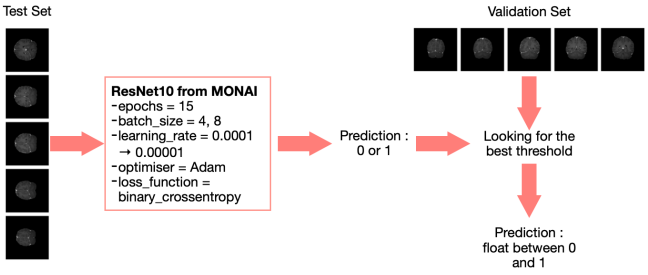

Fig. 12. Training and Validation Process

Meanwhile, in order to determine the optimal parameters for our model, a series of experiments was undertaken. Different ResNet models were trained by varying certain parameters, such as the number of central images used and batch size. The experiments were carried out using the Adam optimizer and the binary cross-entropy loss function. These adjustments aimed to assess the impact of these parameters on model performance, with a test and validation set, both consisting of 15% of the all dataset. After training our models with the training set, the search for the optimal threshold to maximize accuracy was carried out using our validation set. The aim of this crucial step was to determine the classification threshold that would enable our models to perform best on data not seen during training. Finding the threshold involves setting a threshold value on which to base the classification decision. In our context, where we are dealing with probabilistic predictions derived from classification models, threshold adjustment enables us to establish the optimal compromise between model sensitivity and specificity. We thus sought to find the ideal balance that maximizes the overall accuracy of the model on the validation set. The aim was to avoid bias towards a specific class and ensure that the model could generalize effectively to new data.

The results in table 13, were presented by highlighting the area under the ROC curve rather than the accuracy, thus offering a more detailed insight into the model's discriminatory ability. At the end of this stage, it was decided to focus the analysis exclusively on the FLAIR and T1w contrasts. In fact, these two contrasts showed the best performance in the analysis of results, highlighting their ability to discriminate in the context of our binary classification problem.

	FLAIR CONTRAST						T2w CONTRAST					
Number Image	3	7	15	3	7	**15**	3	**7**	15	3	7	15
Batch Size	4	4	4	8	8	**8**	4	**4**	4	8	8	8
Best Threshold	0.530	0.469	0.510	0.694	0.347	**0.735**	0.571	**0.693**	0.714	0.428	0.592	0.510
Loss	0.688	0.680	0.701	0.651	0.831	**0.648**	0.873	**0.832**	0.899	0.994	0.943	0.903
ROC	0.645	0.652	0.635	0.629	0.594	**0.653**	0.553	**0.558**	0.539	0.515	0.546	0.555

	T1wCE CONTRAST						T1w CONTRAST					
Number Image	3	7	**15**	3	7	15	**3**	7	15	3	7	15
Batch Size	4	4	**4**	8	8	8	**4**	4	4	8	8	8
Best Threshold	0.612	0.530	**0.571**	0.592	0.592	0.510	**0.489**	0.367	0.531	0.673	0.653	0.633
Loss	0.782	0.858	**0.859**	1.026	1.015	0.968	**0.752**	0.781	0.840	0.972	0.735	0.958
ROC	0.590	0.565	**0.592**	0.548	0.586	0.539	**0.652**	0.630	0.620	0.592	0.626	0.618

Fig. 13. Classification results table

Improved Selection of Central Images. With the aim of improving the performance of our model, we have undertaken efforts to optimize the input images. Currently, by relying solely on the central images of the file, we are not assured of obtaining representations of the tumor as shown in figure 14, which could potentially reduce the accuracy of the model.

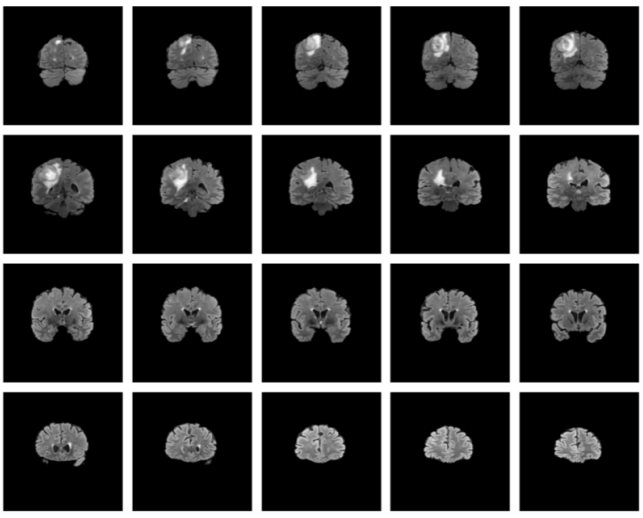

Fig. 14. Flair images for a patient

To address this limitation and better focus the images on the tumor region, we adopted an approach based on the 99.9th percentile of non-zero pixel values as the threshold for image selection. This threshold was determined to identify particularly bright areas, corresponding to the presence of the tumor.

By selecting the image with the maximum number of pixels above this threshold, we guaranteed the capture of an image specifically centered on the tumor. This approach aims to maximize the representation of tumor features in our training set, helping to enhance the model's ability to effectively recognize the presence of the MGMT protein, as observed in figure 15,. However, this approach only work with flair modality.

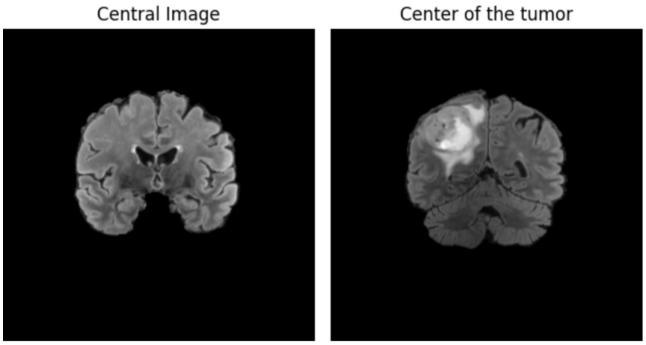

Fig. 15. Central image (left) and tumour-centred image (right)

This approach resulted in a significant improvement in the area under the ROC curve, from 0.653 to 0.694. This evaluation was carried out by testing the model on 30% of the dataset not used for training. The increase in the area under the ROC curve indicates an improvement in the model's ability to discriminate between classes, reinforcing its performance in MGMT protein detection. These encouraging results testify to the effectiveness of the 99.9th percentile approach in optimizing tumor representation in our training set.

Final Models and Average Output. The final step in our approach involved training two separate 3D ResNet10 models, each specialized in a specific image type. The first model was trained on a set of 31 tumor-centered FLAIR images per patient, while the second was trained by taking 15 T1w images centered on the central image of the file per patient.

Predictions generated by the two specialized models were then combined, assigning a weight of 75% to the FLAIR model predictions and 25% to the T1w model. This combination was adjusted carefully to account for the relative contribution of each model, reinforcing the overall consistency and reliability of the final predictions. Finally, the final classification threshold was determined by taking the average of the thresholds calculated during training. By dividing the data sets into 70% for training, 15% for validation and 15% for testing, we obtained ROC results of 0.652 for the FLAIR model, 0.628 for the T1w model, and 0.665 for the average of the two models, justifying taking the average of the predictions as the final prediction.

For the final submission, the model was trained by taking 70% of the dataset for training and 30% for validation, including the search for the optimal threshold. This final approach aims to maximize the model's performance while ensuring a rigorous evaluation of its ability to generalize to new data.

For the Flair model (Fig. 16), it is noticeable that the validation accuracy remains relatively constant throughout training, stabilizing at around 0.53. In contrast, training accuracy evolves positively, rising from 0.53 to 0.67. For the T1w model (Fig. 17), a similar observation is made, with validation accuracy remaining stable at around 0.53. However, training accuracy shows a more pronounced increase, rising from 0.57 to 1. This increase suggests the model's ability to fit training data perfectly, but it also raises the question of the model's possible overfitting to specific training data. However, the models selected are those that maximize accuracy on the validation dataset.

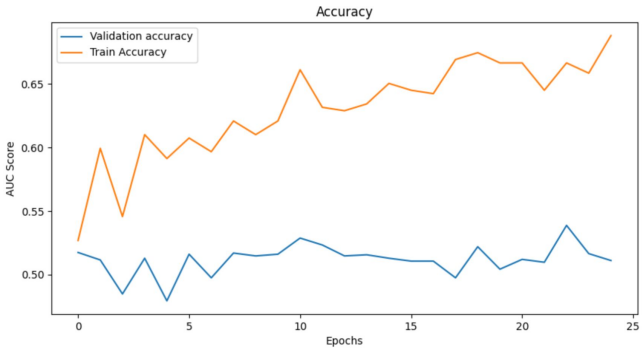

Fig. 16. Learning curves for the Flair model

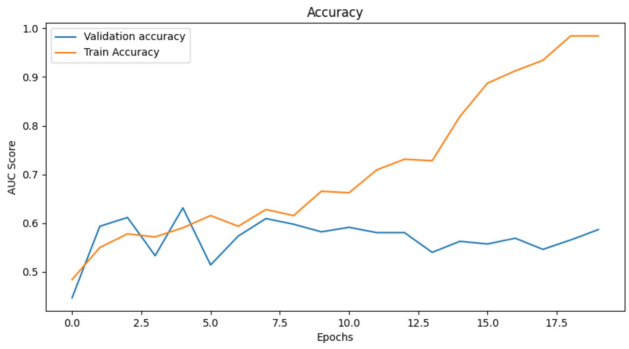

Fig. 17. Learning curves for the T1w model

Explainability. To verify, that what the models are focusing on is actually interpretable and in line with current research, we applied GradCAM++ to the models predictions on the validation set. GradCAM++ is a method introduced by Chattopadhyay et al. in (9), which is an improvement over the existing CAM and GradCAM methods. With this method, we can look at one of the last layers of a given model and extract the feature maps that lead to the later classification result. The resulting saliency maps can give us a visual representation of what areas of an image the models actually focus on.

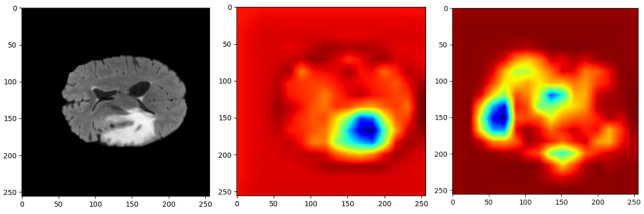

Fig. 18. Results of GradCAM++ with FLAIR image, output of FLAIR model and output of T1cWE model

As can be seen in figure 18, both of the final models focus their attention on the brain itself. The largest amount of attention of the FLAIR model is then focused on the tumor area, that can be seen by the hyper-intense area in the FLAIR image. This is in line with current research (10) in the area of MGMT prediction and provides additional confidence in the model's predictive performance. However, the T1cWE model does not inspire the same amount of confidence, as its main attention is still focused on the brain, but not on the tumor. This pattern of the FLAIR model focusing its attention on tumor areas and the T1cWE model focusing its attention elsewhere can be seen in most other examples as well. Still, the combination of the two models leads to a well performing classifier of MGMT in MRI scans and the areas the T1cWE model focuses on in the scans are potentially still relevant for this prediction.

Conclusion and Considerations. In conclusion, our approach to classifying patients according to the presence of MGMT protein from MRI images resulted in a final model, consolidating two distinct ResNet models, each specialized on FLAIR and T1w images. This involved several crucial steps, from the selection of central images to the use of the 99.9th percentile to optimize tumor presence, to the training of specialized models and the combination of their weighted predictions.

The results obtained on the validation and test sets demonstrate a significant improvement in the area under the ROC curve, rising from 0.543 for random assignment to 0.665 for the final model. This increase attests to the effectiveness of our approach in discriminating between classes, and underlines the relevance of using two specialized models.

However, there are nuances to consider. Observation of the learning curves reveals a certain stability in validation accuracy, suggesting that the model could benefit from additional regularization to avoid over-fitting, particularly in the case of the T1w model. Furthermore, the balance of weights assigned to the FLAIR and T1w models in the final combination could be adjusted to further optimize performance. Further exploration of hyperparameters and regularization techniques could also help improve the generalizability of the model in a variety of clinical contexts.

1. B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, L. Lanczi, E. Gerstner, M.-A. Weber, T. Arbel, B. B. Avants, N. Ayache, P. Buendia, D. L. Collins, N. Cordier, J. J. Corso, A. Criminisi, T. Das, H. Delingette, Demiralp, C. R. Durst, M. Dojat, S. Doyle, J. Festa, F. Forbes, E. Geremia, B. Glocker, P. Golland, X. Guo, A. Hamamci, K. M. Iftekharuddin, R. Jena, N. M. John, E. Konukoglu, D. Lashkari, J. A. Mariz, R. Meier, S. Pereira, D. Precup, S. J. Price, T. R. Raviv, S. M. S. Reza, M. Ryan, D. Sarikaya, L. Schwartz, H.-C. Shin, J. Shotton, C. A. Silva, N. Sousa, N. K. Subbanna, G. Szekely, T. J. Taylor, O. M. Thomas, N. J. Tustison, G. Unal, F. Vasseur, M. Wintermark, D. H. Ye, L. Zhao, B. Zhao, D. Zikic, M. Prastawa, M. Reyes, and K. Van Leemput, "The multimodal brain tumor image segmentation benchmark (brats)," *IEEE Transactions on Medical Imaging*, vol. 34, no. 10, pp. 1993–2024, 2015.
2. O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.
3. 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.
4. M. J. Cardoso, W. Li, R. Brown, N. Ma, E. Kerfoot, Y. Wang, B. Murray, A. Myronenko, C. Zhao, D. Yang, V. Nath, Y. He, Z. Xu, A. Hatamizadeh, W. Zhu, Y. Liu, M. Zheng, Y. Tang, I. Yang, M. Zephyr, B. Hashemian, S. Alle, M. Zalbagi Darestani, C. Budd, M. Modat, T. Vercauteren, G. Wang, Y. Li, Y. Hu, Y. Fu, B. Gorman, H. Johnson, B. Genereaux, B. S. Erdal, V. Gupta, A. Diaz-Pinto, A. Dourson, L. Maier-Hein, P. F. Jaeger, M. Baumgartner, J. Kalpathy-Cramer, M. Flores, J. Kirby, L. A. Cooper, H. R. Roth, D. Xu, D. Bericat, R. Floca, S. K. Zhou, H. Shuaib, K. Farahani, K. H. Maier-Hein, S. Aylward, P. Dogra, S. Ourselin, and A. Feng, "MONAI: An open-source framework for deep learning in healthcare," Nov. 2022.
5. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/
6. L. Fidon, W. Li, L. C. Garcia-Peraza-Herrera, J. Ekanayake, N. Kitchen, S. Ourselin, and T. Vercauteren, "Generalised wasserstein dice score for imbalanced multi-class segmentation using holistic convolutional networks," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: Third International Workshop, BrainLes 2017, Held in Conjunction with MICCAI 2017, Quebec City, QC, Canada, September 14, 2017, Revised Selected Papers 3*. Springer, 2018, pp. 64–76.
7. D. Karimi and S. E. Salcudean, "Reducing the hausdorff distance in medical image segmentation with convolutional neural networks," *IEEE Transactions on medical imaging*, vol. 39, no. 2, pp. 499–513, 2019.
8. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
9. A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," in *2018 IEEE winter conference on applications of computer vision (WACV)*. IEEE, 2018, pp. 839–847.
10. T. Kanazawa, Y. Minami, M. Jinzaki, M. Toda, K. Yoshida, and H. Sasaki, "Predictive markers for mgmt promoter methylation in glioblastomas," *Neurosurgical review*, vol. 42, pp. 867–876, 2019.