

Original Article

Tumor Segmentation for Brain Tumor Using Combination of Deep Learning and Machine Learning Algorithms.

Pravin Kumar , Sandeep Singh , Soniya Pal- , Monini Manav, Anuj Tyagi.

Abstract

Purpose

The purpose of this study was to accurately segment a brain tumor with as little subjectivity as feasible. In this study, three different types of convolutional neural networks (CNNs)—3D U-Net, 3D-UNet using VGG16 as an encoder, and 3D-UNet using VGG19 as an encoder—were used to separate various tumor regions on brain 3D MRI scans.

Materials and Methods

The dataset is taken from the website of Cancer Imaging Achieve (TCIA). The BRATS 2021 dataset has multi-parametric magnetic resonance imaging (mpMRI) scans of 611 patients with their manually segmented labels for several histologically different sub-regions of the tumor. The NIfTI format is used for all five series of the mpMRI scan (T1, GD-T1, Flair, T2, and T2) and the segmented mask file (seg). For this, we used three different models for image segmentation.

Result

The accuracy of the UNet model was 97.3 %, 97.5 % for UNet-VGG16, and 98.3 % for UNet-VGG19. The mean IOU score for UNet was 0.8131, 0.7831 for UNet-VGG16 and 0.7931 for UNet-VGG19. The mean dice score was 0.9721 for UNet, 0.9521 for UNet-VGG16, and 0.9621 for UNet-VGG19.

Conclusion

The UNet model outperforms the other two in terms of mean IOU score and mean dice coefficient, while UNet-VGG19 outperforms in terms of accuracy.

K-Practitioner2024;29(4):48-54.

Introduction

Magnetic resonance imaging (MRI) produces 3D images of live tissue non-invasively. MRI captures radio-frequency signals released by hydrogen atoms following the application of electromagnetic (radio-frequency) waves, localising the signal using spatially variable magnetic gradients, and measuring different tissue parameters based on the pulse sequence[1], mpMRI[2], stands for multi-parametric magnetic resonance imaging, is a type of MRI that uses a combination of different sequences and contrasts to provide detailed information about the tissue being imaged. The different sequences and contrasts used in mpMRI can include T1-weighted, T2-weighted, and diffusion-weighted imaging (DWI), as well as dynamic contrast-enhanced imaging (DCE-MRI). T1-weighted imaging is sensitive to tissue density and can provide detailed information about the structure of the tissue being imaged. T2-weighted imaging is sensitive to tissue water content and can provide detailed information about the tissue's microstructure. DWI is used to measure the diffusion of water molecules in the tissue and can provide information

Author Affiliations

Pravin Kumar ,Department of Information Technology ,SRIET Chaudhary Charan Singh University, Meerut (UP); **Sandeep Singh** , Department of Physics, GLA University, Mathura, UP; **Monini Manav** , Department of Radiation Oncology, Andromeda Cancer Hospital, Sonapat, Haryana; **Anuj Tyagi**-,Department of Radiation Oncology, LLRM Medical college, Meerut, UP.

Correspondence

Pravin Kumar- Department of Information Technology ,SRIET Chaudhary Charan Singh University, Meerut (UP)
E-Mail pravinpanwar.ccs@gmail.com
(m) 9719340340

Indexed

EMBASE, SCOPUS , IndMED, ESBCO, Google Scholar besides other national and International databases.

Cite this article as

Kumar P, Singh S , Pal S , Manav M, Tyagi A. Tumor Segmentation for Brain Tumor Using Combination of Deep Learning and Machine Learning Algorithms.JK Pract2024;29(4):48-54..

Full length article available at jkpractitioner.com one month after publication

Keywords

Convolutional neural networks, deep learning, transfer learning, 3D image processing, VGG16, VGG19

about the tissue being imaged. The different sequences and contrasts used in mpMRI can include T1-weighted, T2-weighted, and diffusion-weighted imaging (DWI), as well as dynamic contrast-enhanced imaging (DCE-MRI). T1-weighted imaging is sensitive to tissue density and can provide detailed information about the structure of the tissue being imaged. T2-weighted imaging is sensitive to tissue water content and can provide detailed information about the tissue's microstructure. DWI is used to measure the diffusion of water molecules in the tissue and can provide information about the tissue's cellularity and architecture. DCE-MRI is used to measure the uptake of a contrast agent by the tissue and can provide information about the tissue's blood flow and vascularization. mpMRI is used in many medical applications, including the diagnosis and management of prostate cancer, the assessment of brain tumors, and the evaluation of liver and kidney disorders. It is considered as a non-invasive imaging technique and can be used to detect, diagnose, and monitor the progression of disease and response to treatment. mpMRI is particularly useful in the diagnosis and management of prostate cancer, as it can provide detailed information about the size, location, and aggressiveness of the tumor, as well as information about the surrounding tissue and the prostate gland as a whole. It is considered as a highly accurate imaging technique and is being increasingly used as an alternative to traditional biopsy methods. In addition, mpMRI is also being used for the assessment and treatment planning of brain tumors. It can be used to identify areas of abnormal tissue, provide information about the tumor's location and size, and help to determine the best treatment options.

Medical image analysis has significant importance in clinical studies [3] and medical image analysis using machine learning and deep learning is a rapidly growing field that aims to improve the diagnosis and treatment of diseases using advanced image processing techniques. Medical images, such as X-ray, CT, MRI [4,5] and ultrasound images, contain a wealth of information about the human body and its internal structures. However, this information is often difficult for human observers to extract and interpret, especially in cases where the images are complex or the disease is difficult to diagnose. Machine learning and deep learning [6] algorithms provide a powerful tool for automating the analysis of medical images. These algorithms can be trained to recognize patterns and features in the images that are indicative of specific diseases or conditions. This can be achieved by using large amounts of labelled data, which is used to train the algorithms to recognize patterns of interest. One of the most common applications of machine learning and deep learning in medical image analysis is image segmentation, which involves isolating specific structures or regions of interest in an image. This can be used to identify and isolate tumors, vessels, organs, or other structures in the image. U-Net, DeepLab and

Mask R-CNN are some of the most popular architectures used for this purpose [7]. Another important application of machine learning and deep learning in medical image analysis is image classification, which involves identifying the presence or absence of a specific disease or condition in an image. This can be achieved by training a classifier to recognize patterns in the image that are indicative of the disease or condition of interest. Convolutional Neural Networks (CNNs) are commonly used for this purpose. Another popular approach is to use pre-trained models like U-Net. U-Net is a type of CNN that is specifically designed for image segmentation tasks. It is trained on a large dataset and then fine-tuned to the specific task of tumor segmentation. This approach can be more efficient than training a CNN from scratch, as it allows the algorithm to leverage the knowledge gained from the pre-training process. In recent years, Generative Adversarial Networks (GANs) [8] has also been explored in the field of tumor segmentation. GANs consists of two neural networks, a generator, and a discriminator, that work against each other in a zero-sum game. The generator generates new images, while the discriminator attempts to distinguish the generated images from real images. By training GANs on a large dataset of brain MRI images, both with and without tumors, GANs are able to learn the features and patterns associated with GBM tumors and generate new images that can be used for segmentation.

Overall, deep learning algorithms have the potential to significantly improve the accuracy and speed of tumor segmentation, which could lead to better treatment outcomes for patients. However, it is important to note that further research and development is needed before these techniques can be widely adopted in clinical practice [9]. Additionally, it is important that these algorithms are validated and tested on large, diverse patient populations to ensure their generalizability and fairness. In addition to image segmentation and classification, medical image analysis using machine learning and deep learning also includes other tasks such as registration, super-resolution, denoising and anomaly detection. Variability in picture quality, contrast, signal-to-noise ratio, or capture technology limits these algorithms' application in research and clinical contexts [10,11,12].

The approach put forward in this paper, was based on a blend of different artificial neural network models. The suggested model's main goals were to improve segmentation accuracy and other segmentation parameters. The 3 models used for segmentation were a 3D-UNet, second a combination of UNet and Visual Geometry Group 16 (VGG16), and the third was a combination of UNet and Visual Geometry Group 19 (VGG19).

Materials and Methods

Glioblastoma [13,14] (GBM) is a highly aggressive and malignant brain tumor that is notoriously difficult to treat and most frequently found in the frontal or

temporal lobes [15]. Glioblastoma has an incidence of 3.21 per 100,000 people. The median age of diagnosis is 64 years old, and men are more likely than women to receive one. The prognosis for survival is bleak, with survival rates of only 40% in the first year following diagnosis and 17% in the second [16]. Segmentation of GBM tumors is a crucial step in the diagnosis and treatment of the disease, as it allows doctors to precisely identify the location and size of the tumor. However, manual segmentation of GBM tumors can be time-consuming and prone to human error. To address this challenge, researchers have been exploring the use of machine learning algorithms for the automation of GBM tumor segmentation. Machine learning algorithms, such as convolutional neural networks (CNNs) and deep learning algorithms, have been shown to be highly effective in image segmentation tasks. These algorithms are able to learn the features and patterns associated with tumors, allowing them to accurately segment them in new images.

Data Collection and Processing For this task of segmenting brain tumors, the BRATS 2021[17,18,19] dataset with 3D multi-parametric MRI images, was used. This dataset contains tumor label information along with four different MRI image types (T1 weighted, T2 weighted, GD-T1-weighted (post contrast), and FLAIR). The Enhancing Tumor Core (ET), Peritumoral Edema, and Peritumoral Edema are the three sub-components of the tumor. There have been 611 scans in total. The code was executed using the online tool Google Colab Pro [20], which uses Python version V3.7. Out of 611 patients, 30 were chosen for testing, while the remaining 581 were divided into 75 and 25% for training and validation purposes, respectively. The dataset was processed to remove slices with a lot of background pixels since it is too large for the system to handle at 240 by 240 by 155 pixels. ReLU is a type of activation function that will output the input directly if it is positive or one; otherwise, it will give an output of zero. In single-label issues, where the output is either 1 or 0, we thus employ ReLU [21]. SoftMax is utilised in the neural network's output layer for multiclass labelling. It is a mathematical function that turns a vector of integers into a vector of probabilities, where the probability of each value in the vector is proportional to their relative sizes.

Model parameters

A neural network works in the same way as the human brain does. Neurons in the network received information from previous phases with each learning step. The activation function transports this data and decides whether or not to activate the neuron. It also assists the network in learning complicated patterns in the input. Without the activation function, the data would move through the layers of the network by simply passing through the linear function. The convolution block in the model forms the foundation of feature extraction. An encoder block typically consists of several convolution blocks followed by max pooling

down sampling to encode the input image into feature representations at various levels. This abstract representation is sent into the decoder network, which produces a semantic segmentation mask. A 2-by-2 transpose convolution is the first step in the decoder block. The relevant skip connection feature map from the encoder block is then concatenated with it.

Optimizers are algorithms or methods that are used to adjust the characteristics of the neural network, such as weights and learning rate, in order to minimise losses. The learning rate is a hyper-parameter that governs how the weights of our neural network are calculated in relation to the loss gradient. It specifies how quickly the neural network's learned notions are updated. In general, a high learning rate permits the model to learn faster, but at the expense of producing a suboptimal final set of weights. A smaller learning rate may allow the model to learn a more optimal or even globally optimal set of weights, but it may take substantially longer to train.

In image segmentation tasks, a Convolutional Neural Network (CNN) is trained to predict a segmentation mask for an input image. The loss function is a critical component of the CNN training process as it measures the difference between the predicted segmentation mask and the ground-truth segmentation mask (i.e., the mask provided by an expert annotator). The goal of the training process is to minimize this difference, or loss, in order to generate accurate predictions. One of the main benefits of using a loss function in CNN for image segmentation is that it allows for the optimization of the CNN's parameters, such as the weights and biases of the neurons. By comparing the predicted segmentation mask with the ground-truth segmentation mask, the loss function can guide the CNN to adjust its parameters in order to improve its predictions. Another benefit of using a loss function is that it allows for the comparison of different CNN architectures and hyperparameters. By comparing the loss on a validation set, researchers can determine which CNN architecture and hyperparameters perform best on a given task. Additionally, using a loss function specific to image segmentation, such as the Intersection over Union (IoU) or Dice coefficient loss, can improve the accuracy and robustness of the CNN model. These loss functions are designed to take into account the specific characteristics of image segmentation tasks, such as class imbalance and the need for pixel-level accuracy and can lead to better performance compared to using general-purpose loss functions such as Mean Squared Error (MSE). In summary, the loss function plays a crucial role in the training of CNNs for image segmentation. It allows for the optimization of the CNN's parameters, enables the comparison of different CNN architectures and hyperparameters, and improves the accuracy and robustness of the CNN model by using specific loss functions for image segmentation task.

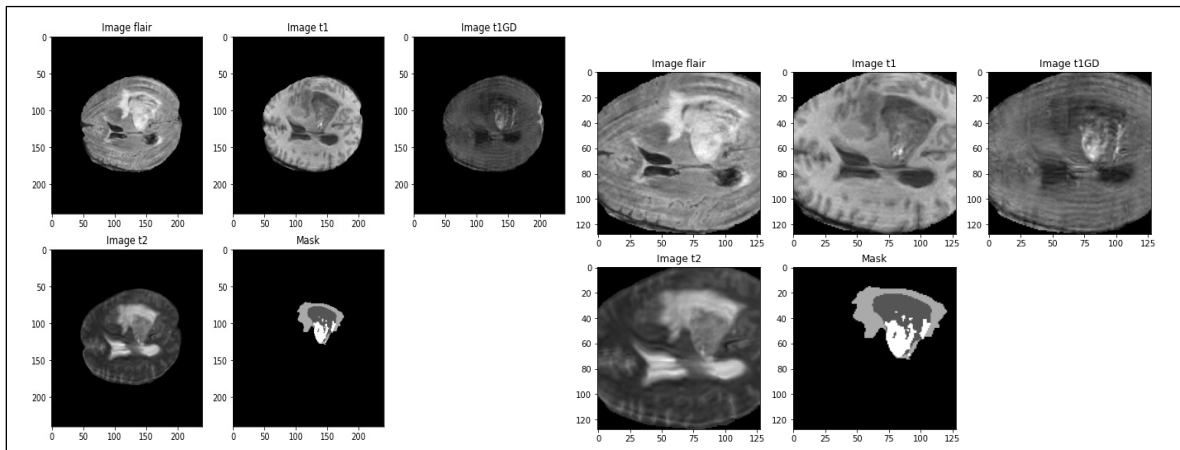


Figure 1- Left (Image before processing) and Right (Image after processing)

Used Model Architecture

U-Net is a popular deep learning architecture that is specifically designed for image segmentation tasks. It is composed of an encoder and a decoder, where the encoder is used to extract features from the input image, and the decoder is used to generate a segmentation mask. VGG16 and VGG19 are both convolutional neural network (CNN) models developed by the Visual Geometry Group (VGG) at the University of Oxford. Both models were trained on the ImageNet dataset, which contains more than a million images and 1000 classes. The main difference between VGG16 and VGG19 is the number of layers in each model. VGG16 has 16 layers, including 13 convolutional layers and 3 fully connected layers, while VGG19 has 19 layers, including 16 convolutional layers and 3 fully connected layers. Because VGG19 has more layers, it is able to learn more complex features from the input images compared to VGG16. This means that VGG19 may perform better on image classification tasks than VGG16. However, VGG19 also requires more computational resources and memory to run, so it may not be the best choice for tasks with limited resources. Additionally, VGG16 and VGG19 have different numbers of parameters, VGG16 has about 138 million parameters, while VGG19 has about 143 million parameters. In summary, VGG19 has more complex features and requires more computational resources and memory to run, and it may not be the best choice for tasks with limited resources. Additionally, VGG19 has more parameters to optimize during training and it may be more prone to overfitting.

One variation of U-Net that has been used in several research studies is U-Net with a VGG16 encoder. VGG16 is a pre-trained convolutional neural network (CNN) that was trained on the ImageNet dataset. It is known for its ability to extract powerful features from images and has been used as the encoder in several image segmentation architectures, including U-Net. When U-Net is combined with a VGG16 encoder, the VGG16 network is used to extract features from the

that corresponds to the input image. The VGG16 encoder allows U-Net to leverage the features and patterns learned by the VGG16 network, which can improve the performance of the U-Net on image segmentation tasks. Similarly other variation of U-Net that has been used in several research studies is U-Net with a VGG19 encoder. VGG19 is a pre-trained convolutional neural network (CNN) that was trained on the ImageNet dataset. It is an improvement over VGG16, which is known for its ability to extract powerful features from images and has been used as the encoder in several image segmentation architectures, including U-Net. In practice, the pre-trained VGG16 and VGG19 network is first frozen and the U-net architecture is trained using these frozen network as the encoder. This is done to prevent the encoder from learning new features and only use the pre-trained weights. U-Net with a VGG16 and VGG19 encoder has been used in several research studies and has shown promising results in the segmentation of GBM brain tumors. The combination of the powerful feature extraction capabilities of the VGG16 and VGG19 encoder and the segmentation capabilities of the U-Net decoder allows for improved segmentation performance compared to using U-Net alone.

The U-Net contains two blocks: encoders and decoders. The encoder and decoder blocks are made up of many convolution layers with hidden sub-layers. In this study, three alternative model combinations are used. The first is a simple U-Net, as illustrated in model2. The encoder block is made up of two convolutional routes that use the rectified linear unit (ReLU) [21] activation function, which is followed by a 2-by-2 max pooling layer. The identical procedure is conducted in reverse in the decoder path. ADAM [22,23] is employed as a training optimizer, with dice loss [24] as the loss function. 0.00001 is the learning rate employed. To avoid memory issues, a batch size of four images is used. The model is trained for a total of 100 epochs. Similarly, in the second model, all of the model's parameters remain

the same except that the encoder block has been swapped out for a VGG16 network (as shown in Figure 3), and in the third, the encoder block has been swapped out for a VGG19 network (as shown in Figure 4).

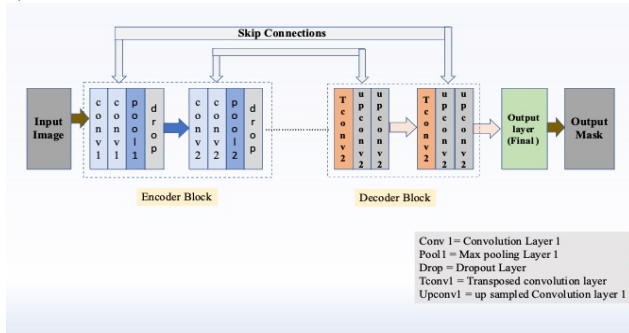


Figure 2: Block Diagram of U-Net with Encoder and decoder blocks

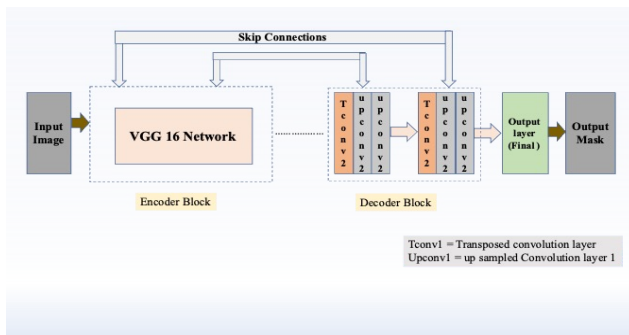


Figure 4: Block Diagram of U-Net with Encoder block changed with a VGG19 network and decoder blocks

Results

In image segmentation, accuracy is a measure of how well a model's predicted segmentation mask aligns with an expert annotator's ground-truth segmentation mask. It is a popular metric for assessing the performance of image segmentation models. In general, higher accuracy indicates better model performance, while lower accuracy indicates the model's inability to generate accurate predictions. It is important to note that accuracy alone is not a sufficient measure of performance for image segmentation models because it does not account for the quality of the predicted segment boundaries. As a result, other metrics, such as the intersection-over-Union score and dice coefficient, which are more sensitive to segmentation errors, are commonly used.

Intersection over Union (IoU) is another popular metric in image segmentation for evaluating model performance. It is also known as the Jaccard index, and it calculates the degree of similarity between predicted and ground-truth segments. The IoU score is calculated by dividing the predicted and ground-truth segments' intersection by their union. The intersection is the number of pixels that appear in both the predicted and ground-truth segments, whereas the union is the number of pixels that appear in only one of the two segments. The IoU score is a number between 0 and 1, with 1 indicating a perfect match between the predicted and ground-truth segments and 0 indicating no overlap between the segments. In general, a higher IoU score indicates that the model is performing well, while a lower IoU score indicates that the model is unable to generate accurate predictions. In a multi-class segmentation problem, the IoU score for each class can be calculated, and the mean IoU score can be used to evaluate the model's overall performance. It can also be used to assess the model's performance on various regions of the image, such as the background, objects, and boundaries.

The Dice coefficient, also known as the Sørensen-Dice similarity coefficient, measures the similarity between the predicted and ground-truth segments and ranges from 0 to 1, with 1 indicating a perfect match and 0 indicating no overlap. In a multi-class segmentation problem, the Mean Dice Score (MDS) is calculated by averaging the Dice coefficients for each class.

$$\text{Dice} = 2 * (\text{intersection of predicted and ground-truth segments}) / (\text{number of pixels in predicted segment} + \text{number of pixels in ground-truth segment})$$

Comparison of the models:

The results of the three-segmentation model were presented in Table 1. As can be seen from the table, the accuracy of the UNet-VGG19 was higher than that of the other two models, while the mean dice score and mean IOU score were both higher for UNet

Model	Accuracy of Model	Mean IOU	Mean Dice
UNet	0.9734	0.8131	0.9721
UNet - VGG16	0.9754	0.7831	0.9521
UNet-VGG19	0.9825	0.7931	0.9621

Table 1: Segmentation results for UNet, UNet-VGG16 and UNet-VGG19

Prediction on different test images

The results of applying the model to some of the test images were shown in Table 2. As can be seen, the UNet model's accuracy was 97.3%, 97.5% for UNet-VGG16, and 98.3% for UNet-VGG19. UNet had a

Image No.	Accuracy of Model			Mean IOU			Mean Dice Score		
	UNet	UNet - VGG16	UNet-VGG19	UNet	UNet - VGG16	UNet-VGG19	UNet	UNet - VGG16	UNet-VGG19
1	0.9709	0.9729	0.9804	0.8283	0.7983	0.8083	0.9708	0.9508	0.9608
2	0.9774	0.9794	0.9864	0.8109	0.7809	0.7909	0.9774	0.9574	0.9674
3	0.9727	0.9747	0.9817	0.7596	0.7296	0.7396	0.9725	0.9525	0.9625
4	0.9760	0.9780	0.9850	0.8161	0.7861	0.7961	0.9759	0.9559	0.9659
5	0.9709	0.9729	0.9801	0.8083	0.7783	0.7883	0.9708	0.9508	0.9608
6	0.9667	0.9687	0.9757	0.8220	0.7920	0.8020	0.9664	0.9464	0.9564
7	0.9744	0.9764	0.9834	0.8175	0.7875	0.7975	0.9743	0.9543	0.9643
8	0.9739	0.9759	0.9829	0.8326	0.8026	0.8126	0.9735	0.9535	0.9635
9	0.9779	0.9799	0.9872	0.8251	0.7951	0.8051	0.9677	0.9477	0.9577

Table 2: Segmentation results on test dataset

mean IOU score of 0.8131, UNet-VGG16 had a score of 0.7831, and UNet-VGG19 had a score of 0.7931. The mean dice score for UNet was 0.9721, 0.9521 for UNet-VGG16, and 0.9621 for UNet-VGG19. Figure 5 depicts the segmented mask generated with UNet, UNet-VGG16, and UNet-VGG19, along with the corresponding ground truth.

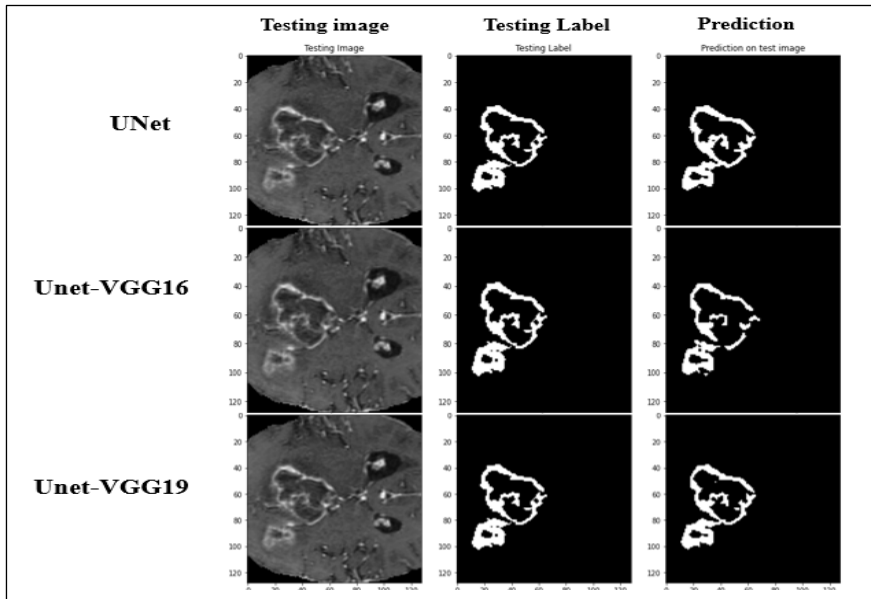


Figure 5: Segmented tumor using UNet, UNet-VGG16 and UNet-VGG19 with their ground truths

Conclusion

In conclusion, U-Net is a popular deep learning architecture for image segmentation tasks, and it has been shown to be effective in the segmentation of GBM brain tumors. The use of a pre-trained encoder, such as VGG16 or VGG19, can further improve the performance of U-Net on image segmentation tasks. In comparison, the use of VGG16 as the encoder in U-Net has been shown to achieve promising results in the segmentation of brain tumors. The combination of the powerful feature extraction capabilities of the VGG16 encoder and the segmentation capabilities of the U-Net

decoder allows for improved segmentation performance compared to using U-Net alone. Similarly, using VGG19 as the encoder in U-Net had also been shown to achieve promising results in the segmentation of brain tumors. VGG19 is an improvement over VGG16, and the combination of the powerful feature extraction capabilities of the VGG19 encoder and the

segmentation capabilities of the U-Net decoder allows for improved segmentation performance compared to using U-Net alone. The UNet model outperforms the other two models in terms of mean dice coefficient and mean IOU score, based on the results. In terms of accuracy, UNet-VGG19 exceeds the other two. So, we can say that utilising machine learning, we can automate brain tumor segmentation with a few clicks and greater accuracy, saving a lot of time. It's worth noting that the choice of encoder can depend on the specific task or dataset, and the results can also vary depending on the quality and size of the dataset used for

training. It is also important to consider other factors such as the computational resources and time available, as well as the specific requirements and constraints of the task at hand.

Financial support and sponsorship: Nil.

Conflicts of interest: There are no conflicts of interest.

References

1. Lerch JP, van der Kouwe AJW, Raznahan A, Paus T, Johansen-Berg H, Miller KL, et al. Studying neuroanatomy using MRI. *Nat Neurosci.* 2017 Feb 23;20(3):314–26.

2. Caulo M, Panara V, Tortora D, Mattei PA, Briganti C, Pravata E, et al. Data-driven Grading of Brain Gliomas: A Multiparametric MR Imaging Study. *Radiology*. 2014 Aug;272(2):494–503.
3. MRI brain classification using support vector machine | Semantic Scholar [Internet]. [cited 2022 Aug 1].
4. MRI based brain tumor detection using wavelet packet feature and artificial neural networks. *Proceedings of the International Conference and Workshop on Emerging Trends in Technology* [Internet]. [cited 2022 Aug 1].
5. Kazmi JH, Qureshi K, Rashid H. Enhanced MRA Images Quality Using Structure Adaptive Noise Filter and Edge Sharpening Methods. *Malaysian Journal of Computer Science*. 2007 Dec 1;20(2):99–114.
6. Bardis MD, Houshyar R, Chang PD, Ushinsky A, Glavis-Bloom J, Chahine C, et al. Applications of Artificial Intelligence to Prostate Multiparametric MRI (mpMRI): Current and Emerging Trends. *Cancers*. 2020 May;12(5):1204.
7. Sultana F, Sufian A, Dutta P. Evolution of Image Segmentation using Deep Convolutional Neural Network: A Survey. *Knowledge-Based Systems*. 2020 Aug 9;201–202:106062.
8. Zhai D, Hu B, Gong X, Zou H, Luo J. ASS-GAN: Asymmetric semi-supervised GAN for breast ultrasound image segmentation. *Neurocomputing*. 2022 Jul 7;493:204–16.
9. Hesamian MH, Jia W, He X, Kennedy P. Deep Learning Techniques for Medical Image Segmentation: Achievements and Challenges. *J Digit Imaging*. 2019 Aug 1;32(4):582–96.
10. Othman MFB, Abdullah NB, Kamal NFB. MRI brain classification using support vector machine. In: *Simulation and Applied Optimization 2011 Fourth International Conference on Modeling*. 2011. p. 1–4.
11. Jovicich J, Czanner S, Greve D, Haley E, van der Kouwe A, Gollub R, et al. Reliability in multi-site structural MRI studies: Effects of gradient non-linearity correction on phantom and human data. *NeuroImage*. 2006 Apr 1;30(2):436–43.
12. Chen J, Liu J, Calhoun VD, Arias-Vasquez A, Zwiers MP, Gupta CN, et al. Exploration of scanning effects in multi-site structural MRI studies. *Journal of Neuroscience Methods*. 2014 Jun 15; 230:37–50.
13. Ladomersky E, Scholtens DM, Kocherginsky M, Hibler EA, Bartom ET, Otto-Meyer S, et al. The Coincidence Between Increasing Age, Immunosuppression, and the Incidence of Patients with Glioblastoma. *Frontiers in Pharmacology* [Internet]. 2019 [cited 2022 Aug 1];10.
14. Philips A, Henshaw DL, Lamburn G, O’Carroll MJ. Brain Tumors: Rise in Glioblastoma Multiforme Incidence in England 1995–2015 Suggests an Adverse Environmental or Lifestyle Factor. *Journal of Environmental and Public Health*. 2018 Jun 24;2018: e7910754.
15. Larjavaara S, Mäntylä R, Salminen T, Haapasalo H, Raitanen J, Jääskeläinen J, et al. Incidence of gliomas by anatomic location. *Neuro Oncol*. 2007 Jul;9(3):319–25.
16. McKinnon C, Nandhabalan M, Murray SA, Plaha P. Glioblastoma: clinical presentation, diagnosis, and management. *BMJ*. 2021 Jul 14;374: n1560.
17. Bakas S, Sako C, Akbari H, Bilello M, Sotiras A, Shukla G, et al. The University of Pennsylvania glioblastoma (UPenn-GBM) cohort: advanced MRI, clinical, genomics, & radiomics. *Scientific Data*. 2022 Jul 29;9.
18. Clark K, Vendt B, Smith K, Freymann J, Kirby J, Koppel P, et al. The Cancer Imaging Archive (TCIA): Maintaining and Operating a Public Information Repository. *J Digit Imaging*. 2013 Dec 1;26(6):1045–57.
19. Bakas, S., Sako, C., Akbari, H., Bilello, M., Sotiras, A., Shukla, G., Rudie, J. D., Flores Santamaria, N., Fathi Kazerooni, A., Pati, S., Rathore, S., Mamourian, E., Ha, S. M., Parker, W., Doshi, J., Baid, U., Bergman, M., Binder, Z. A., Verma, R., ... Davatzikos, C. (2021). Multi-parametric magnetic resonance imaging (mpMRI) scans for Glioblastoma (GBM) patients from the University of Pennsylvania Health System [Data set].
20. Bisong E. Google Colaboratory. In: *Building Machine Learning and Deep Learning Models on Google Cloud Platform* [Internet]. Berkeley, CA: Apress; 2019 [cited 2022 Jun 14]. p. 59–64.
21. Agarap AF. Deep Learning using Rectified Linear Units (ReLU) [Internet]. arXiv; 2019 [cited 2022 Aug 2].
22. Nwankpa C, Ijomah W, Gachagan A, Marshall S. Activation Functions: Comparison of trends in Practice and Research for Deep Learning. 2018 [cited 2022 Jun 14];
23. Kingma DP, Ba J. Adam: A Method for Stochastic Optimization [Internet]. arXiv; 2017 [cited 2022 Aug 2].
24. Sudre CH, Li W, Vercauteren T, Ourselin S, Cardoso MJ. Generalised Dice overlap as a deep learning loss function for highly unbalanced segmentations. In 2017 [cited 2022 Aug 2]. p. 240–8. Available from: <http://arxiv.org/abs/1707.03237>