

Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)



End Semester Project Report

CS517: Digital Image Processing



Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

Full Name	Enrollment No.
Malya Singh	2022CSZ0015
Seema	2022CSZ0006
Anshika	2021CSB1069

Course Instructor: Dr. Puneet Goyal



Department of Computer Science and Engineering
Indian Institute of Technology Ropar
Rupnagar, Punjab India
May 14, 2023



Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

Contents

- 1 Introduction
- 2 Objectives
- 3 Tools Used
- 4 Methodology
- 5 Details of the Final Model
- 6 Results
- 7 Learning Outcomes
- 8 Limitations and future scopes
- 9 References



1 | Introduction

This project is based on **The Brain Tumor Segmentation (BraTS)** Continuous Challenge seeks to identify the current, state-of-the-art segmentation algorithms for brain diffuse glioma patients and their sub-regions.

This continuous challenge will culminate in an annual, state-of-the-field evaluation where we will ask participants to submit containerized versions of their best models which will be run against a held out testing data set, and the results presented at the annual MICCAI conference



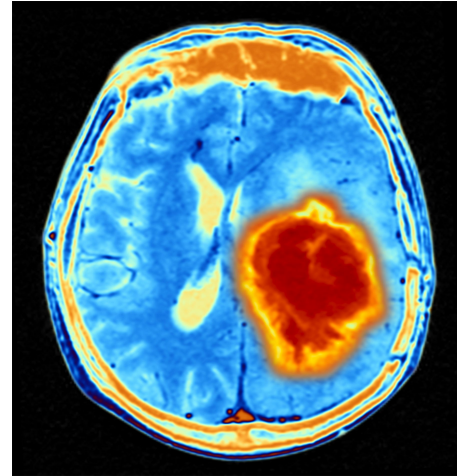


Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

2 | Objectives

The following are the objectives of the End Semester project challenge:

- Develop a method and produce segmentation labels of the different Gliomas.
- Identify various subregions of deadly Brain Tumour called Glioma.
- Test the developed methodology on provided BraTS21 dataset.



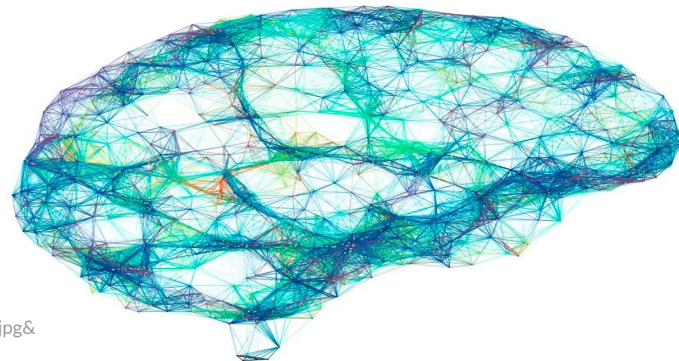


3 | Tools Used

For the implementation of this project, we used and explored various tools and technologies. Most of the work is done using the Python programming language in Google Colab.

Although, the project began with a simple segmentation model, various deep learning based models were utilized and explored in the implementation of this project. The models are enlisted below:

- **UNet Model:** based on Convolutional Neural Network architecture.
- **Swin UNETR Model:** Transformer-based Encoder-Decoder model
- **DeepLab Model:** based on Fully CNN architecture





4 | Methodology

Here is the step-by-step description of our work done on this project

- We began with a simple model for segmentation. This model is based on the simple methodologies that we understood throughout the class and lab assignments.

After reading the image, we notice by plotting its histogram that it needs to be enhanced. Hence, we apply some standard techniques for its enhancement.

Gaussian blurring filter is applied on entire input image to obtain blurred version of it.

For enhanced homogeneity in the input image, the blurred image is added to the original grayscale image.

Further, owing to the edge-preserving and denoising properties of the median filter, we apply it on the image.

Then a kernel is generated as a morphological gradient. This improves the edge contrast across the entire image.

Then thresholding is applied.

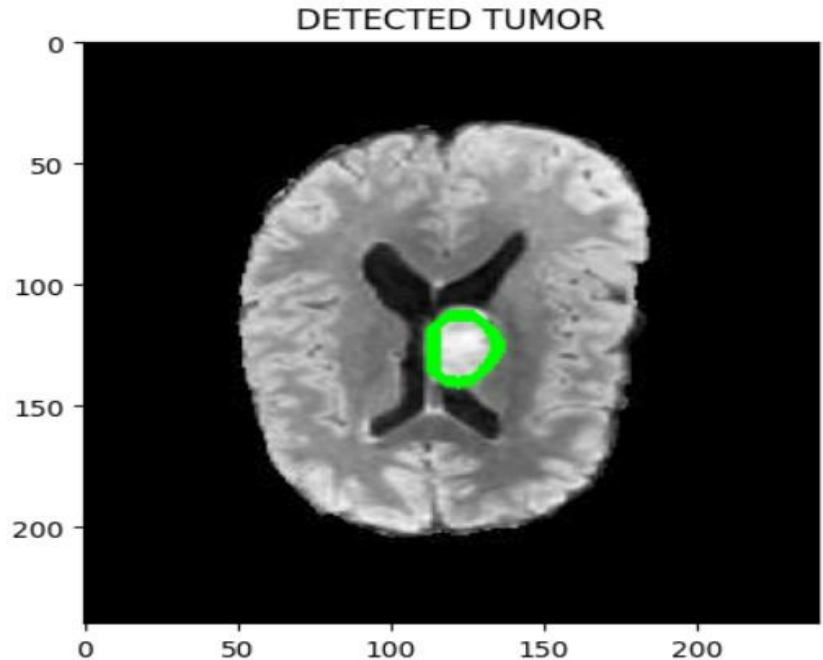
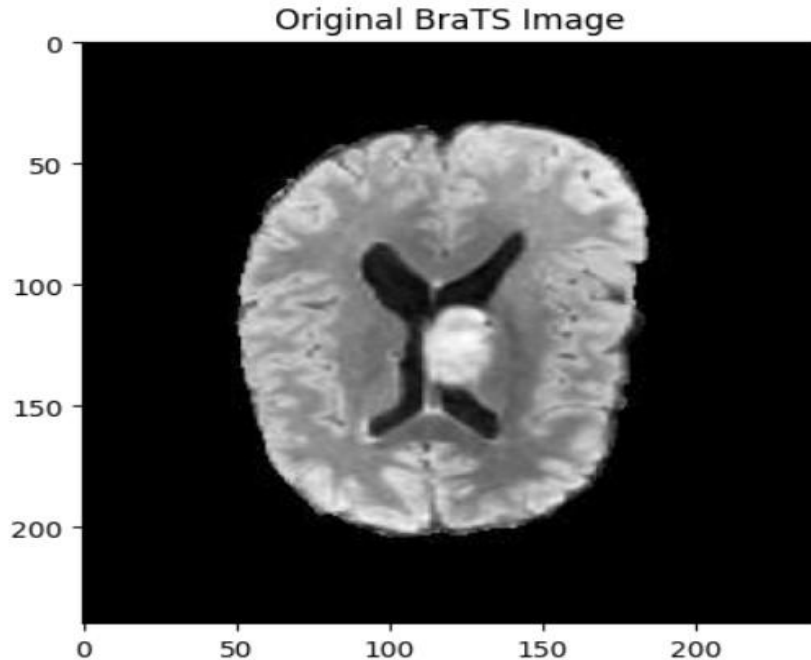
Then we apply various morphological operations to develop a better region of interest after thresholding. Finally, we obtain the segmented model.



Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

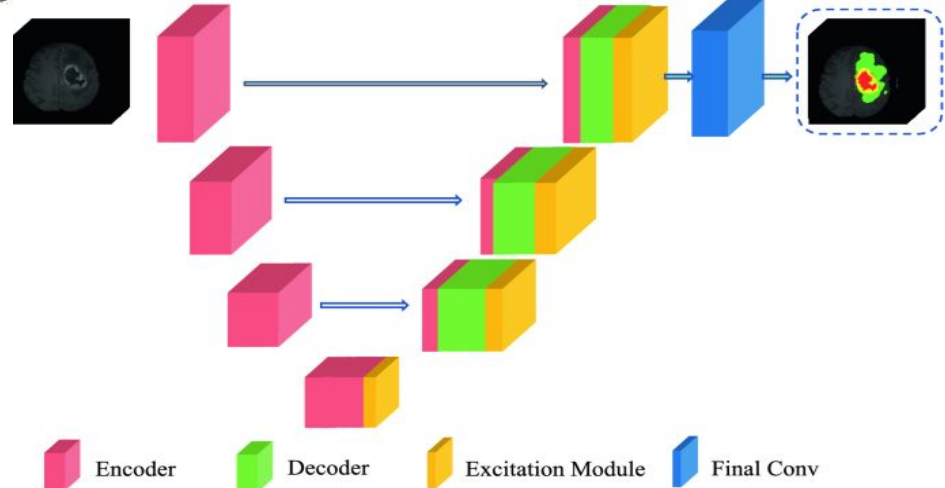
Limitations: This method is case specific as for different data, it needs different thresholding values and morphological operations. This can lead to unreliable results in real life problems.

Hence we moved towards more deep learning based models.



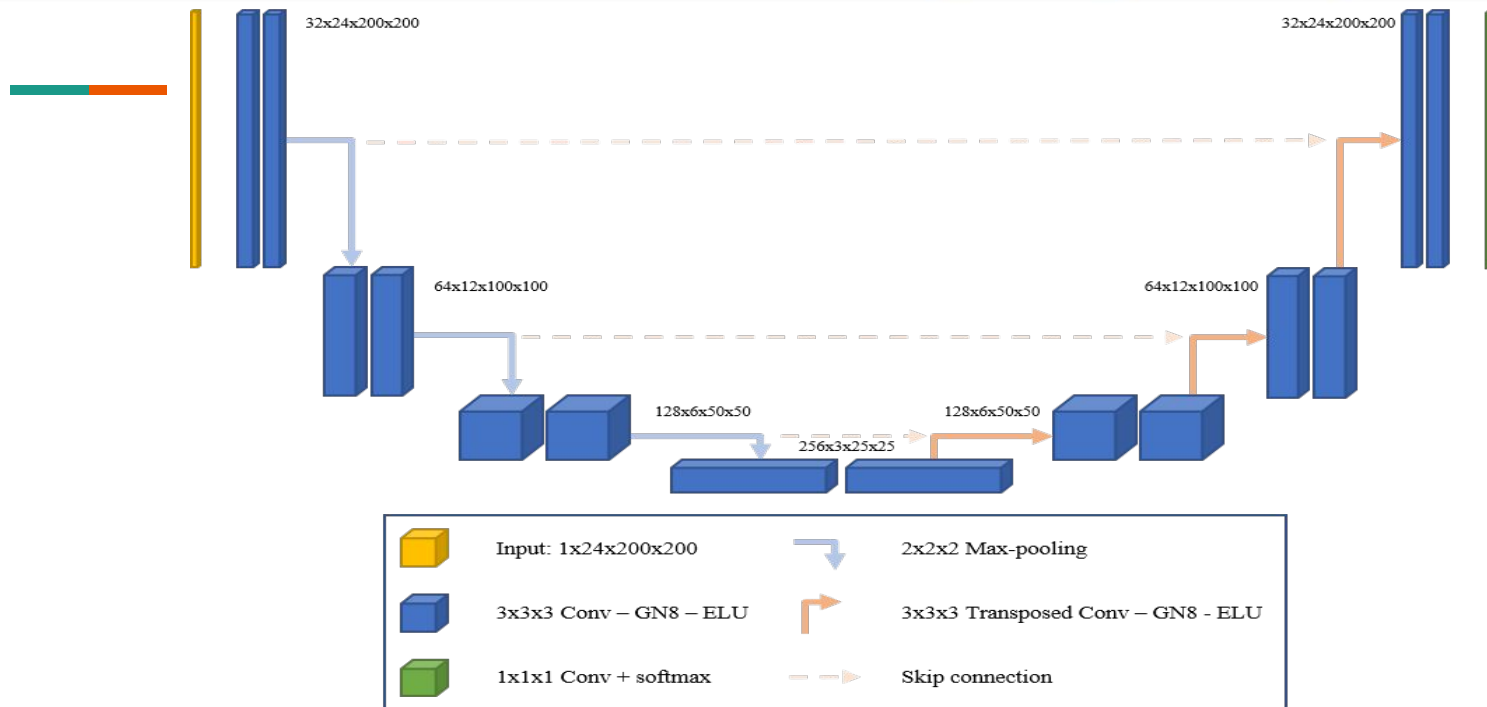
UNET MODEL -

- As the simple segmentation using thresholding and Morphological Operations had various shortcomings, we moved to the deep learning based model called UNet Model.
- UNet Model gets its name after its architecture. It is a U-shaped model that comprises of convolutional layers and two networks. It consists of an encoder followed by decoder. With the UNet model it is possible to solve two problems:
 - What part of image is the tumor?
 - Where is the tumor in the image?





Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)



UNet Model Architecture



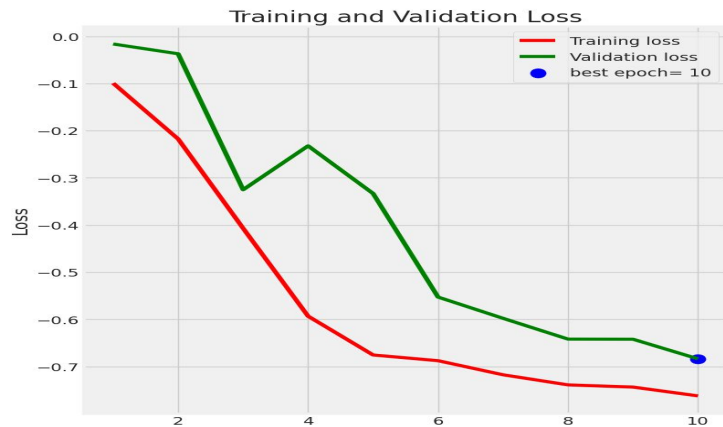
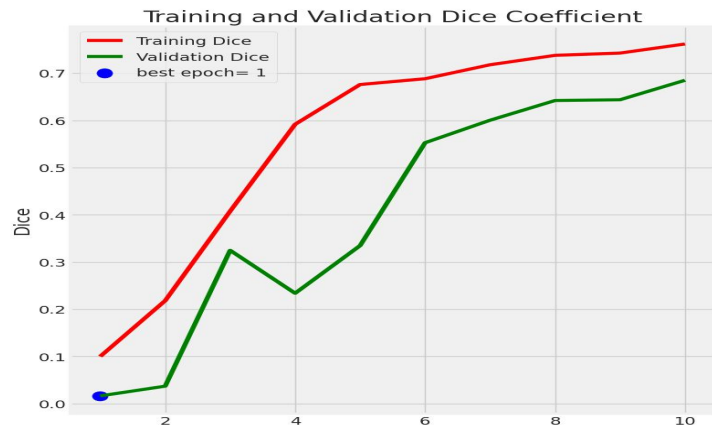
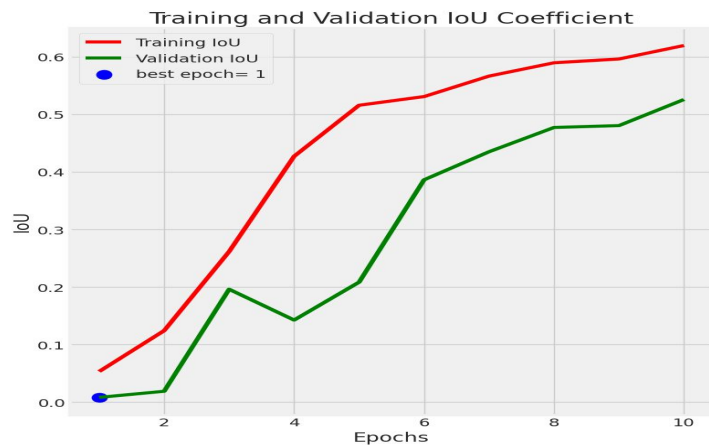
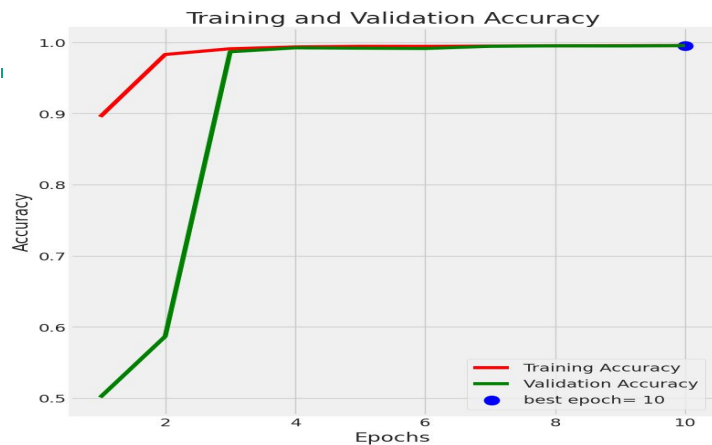
5 | Details of the Models

1. On training the BraTS21 Dataset on UNet Model, promising results were observed with great accuracy. The structure of the attached model included:

- Functions to create data frame from dataset :
 - *create_df* : Function to create data frame
 - *split_df* : Function to split data frame into train, valid, test
- Function to create image generators and augmentation : *create_gens*
- Functions that have UNet Structure: *unet*
- Functions for coefficients and loss :
 - *dice_coef* : function to create dice coefficient
 - *dice_loss* : function to create dice loss
 - *iou_coef* : function to create iou coefficient
- Function to show images sample : *show_images*
- Function to display training history : *plot_training*



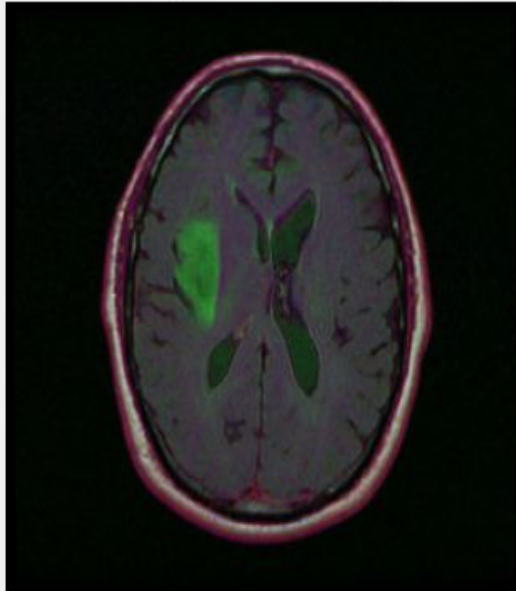
Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)



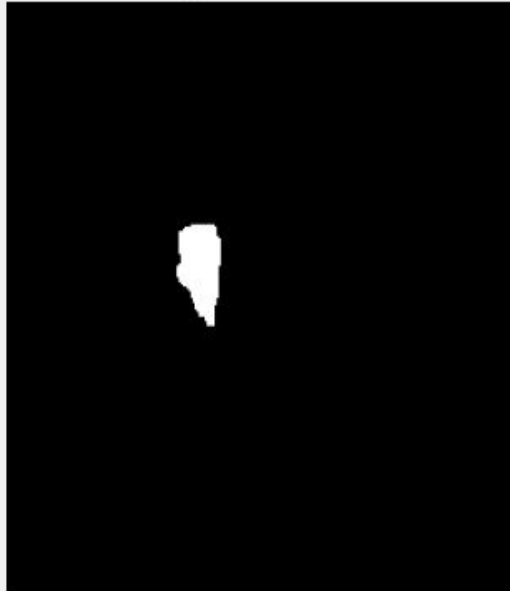


Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

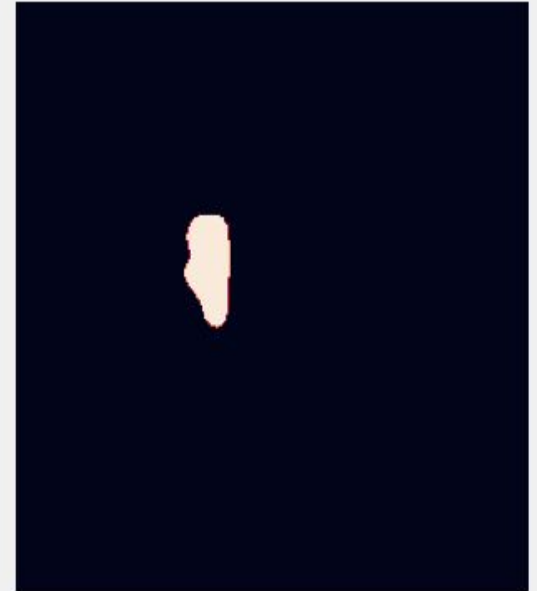
Original Image



Original Mask



Prediction

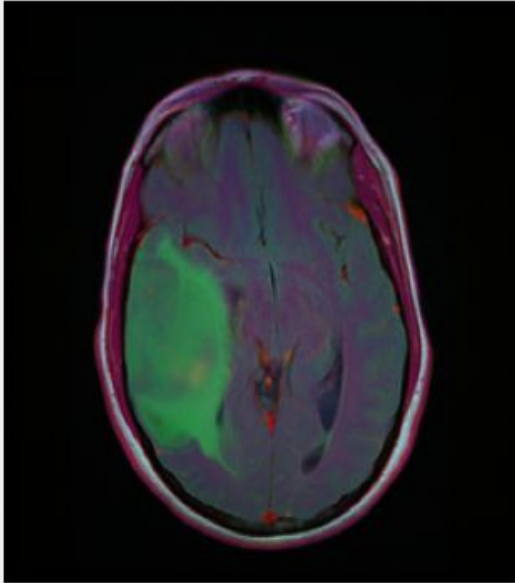


UNet Result 1

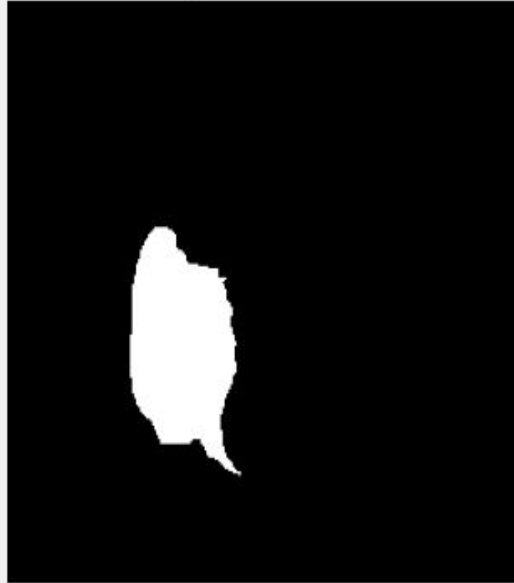


Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

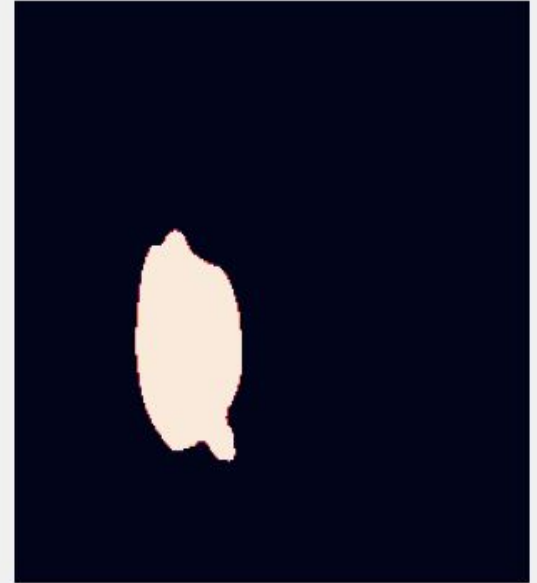
Original Image



Original Mask



Prediction

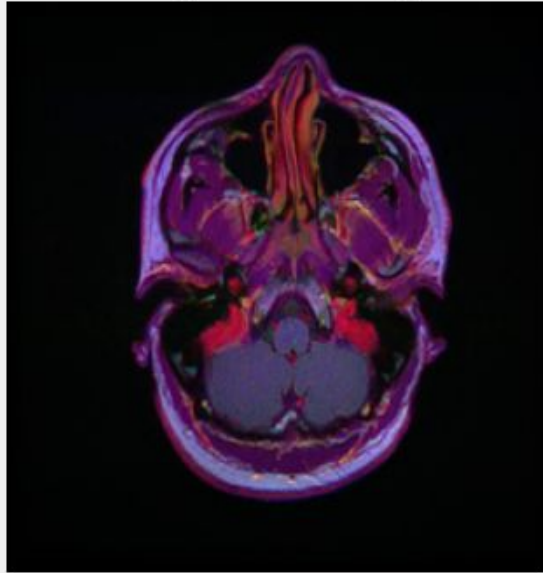


UNet Result 2



Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

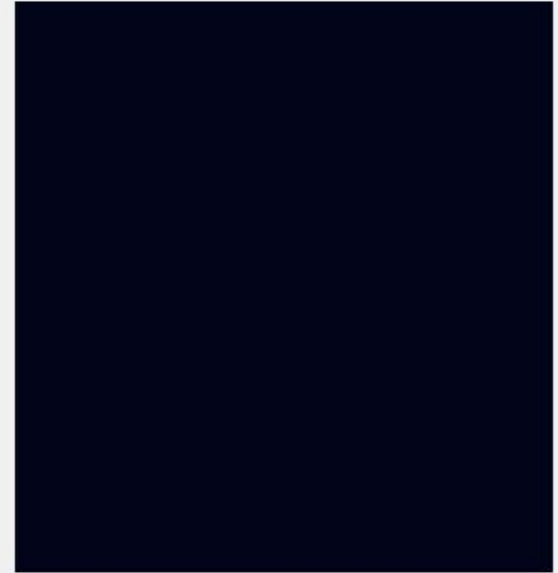
Original Image



Original Mask



Prediction



UNet Result 3

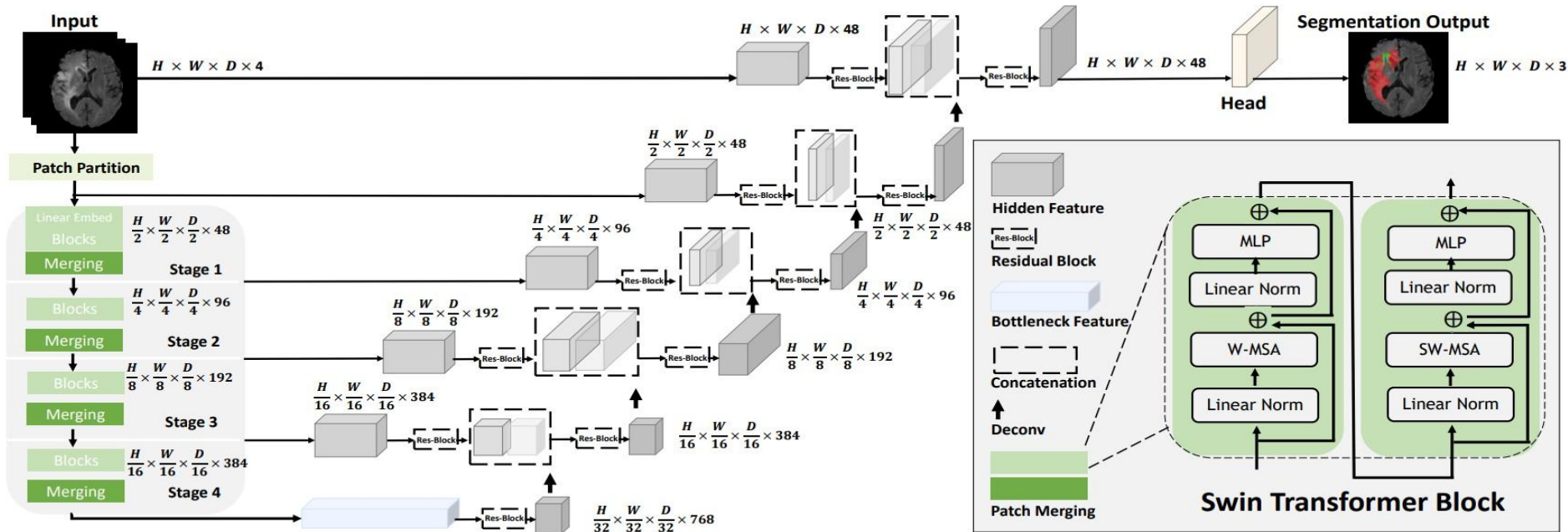


SWIN UNETR MODEL -

- **Swin-UnetR** is a variant of the **Swin Transformer architecture** that has been adapted for brain tumor segmentation in medical images. **Swin-UnetR combines the Swin architecture with the U-Net architecture**, which is a widely used deep learning architecture for image segmentation.
- The **Swin-UnetR architecture consists of an encoder and a decoder**. The encoder uses the Swin architecture to extract features from the input medical images, while the decoder uses the U-Net architecture to generate a segmentation mask that labels each pixel in the image as either tumor or non-tumor.
- The encoder in Swin-UnetR consists of several stages, with each stage processing the input image at a progressively lower spatial resolution. Each stage consists of multiple Swin blocks that apply the shifted window strategy to capture spatial dependencies between adjacent image patches. The output of the final stage is a set of feature maps that represent the high-level features of the input image.
- The decoder in Swin-UnetR takes the feature maps from the encoder as input and uses a series of up-sampling and convolutional layers to generate a segmentation mask. The up-sampling layers gradually increase the spatial resolution of the feature maps, while the convolutional layers use the high-level features to generate a pixel-wise segmentation mask.



Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)



Ref-<https://developer.nvidia.com/blog/novel-transformer-model-achieves-state-of-the-art-benchmark-s-in-3d-medical-image-analysis/>



5 | Details of the Models

2. On training the BraTS21 Dataset on SWIN UNeTR Model, promising results were observed with great accuracy.

The structure of the attached SWIN UNetR model has following dependencies included:

- MONAI version: 0.9.0rc3
- Numpy version: 1.22.2
- Pytorch version: 1.10.0

The structure of the attached model included:

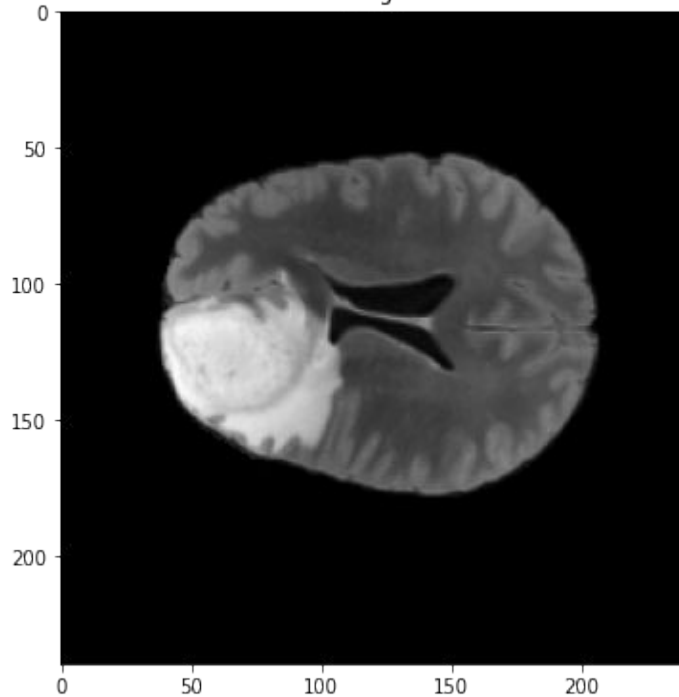
- Setup average meter, fold reader, checkpoint saver : Class AverageMeter
 - ☐ *datafold_read*
 - ☐ *save_checkpoint*
- Setup dataloader : *get_loader*
- Set dataset root directory hyperparameters
- Checking data shape and visualize
- Creating SWIN UNetR Model
- Optimizer and Loss Functions



Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

Image shape: (240, 240, 155)

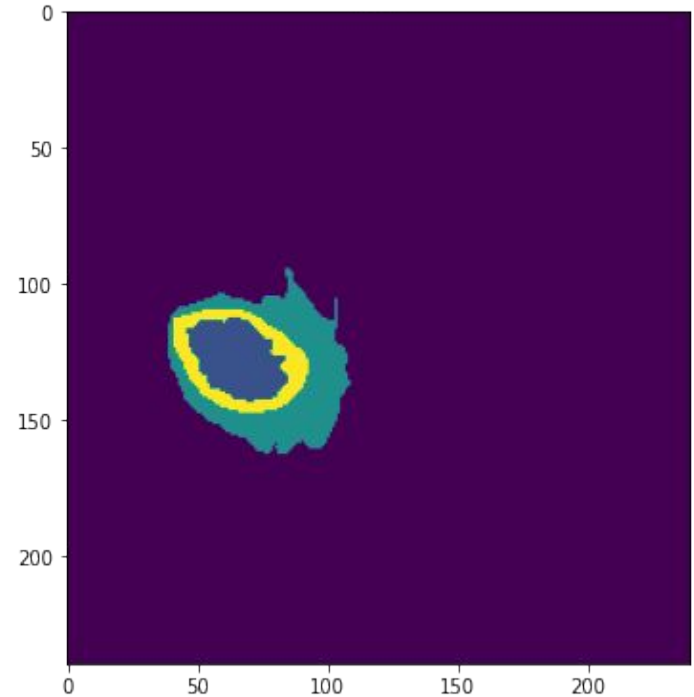
image



Input Image

label shape: (240, 240, 155)

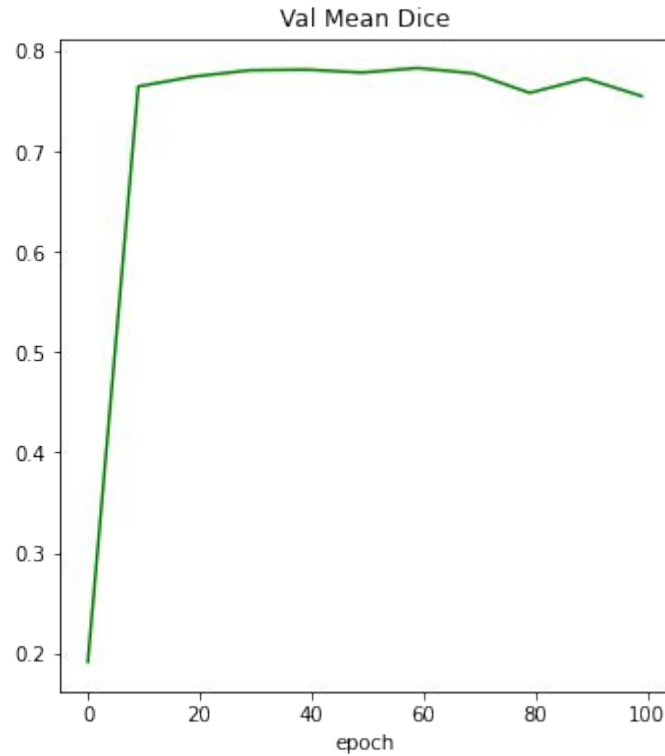
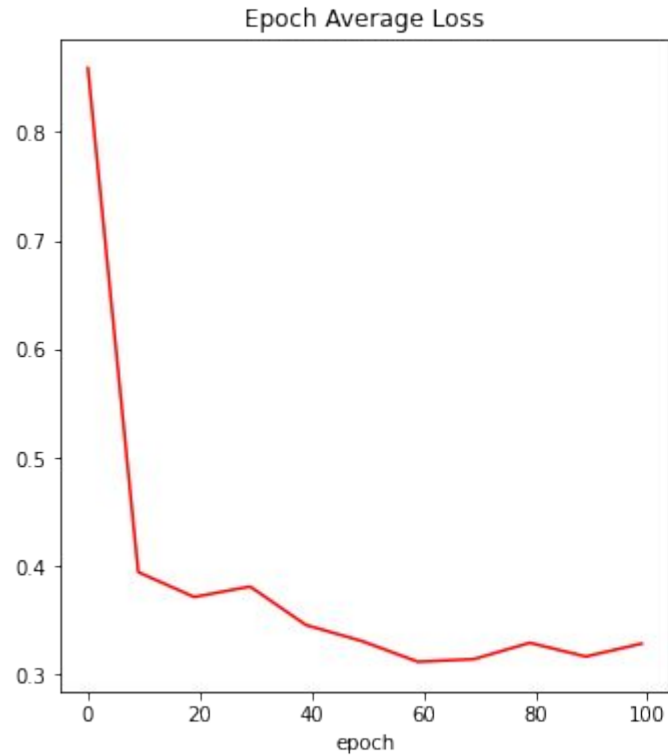
label



Input Label



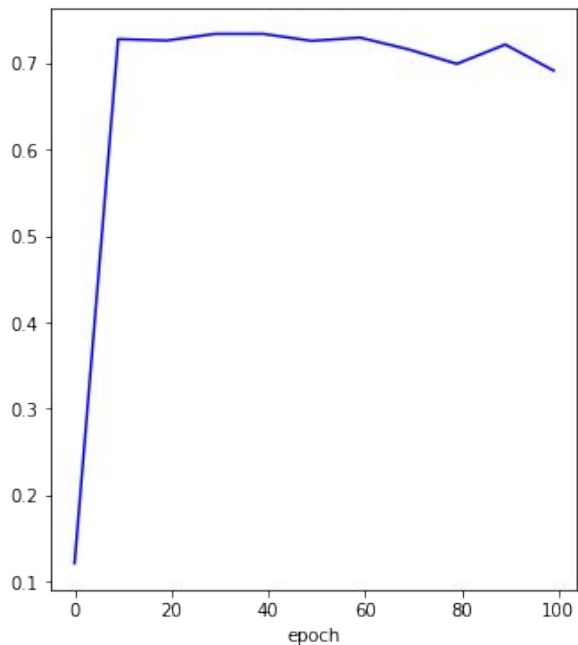
Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)



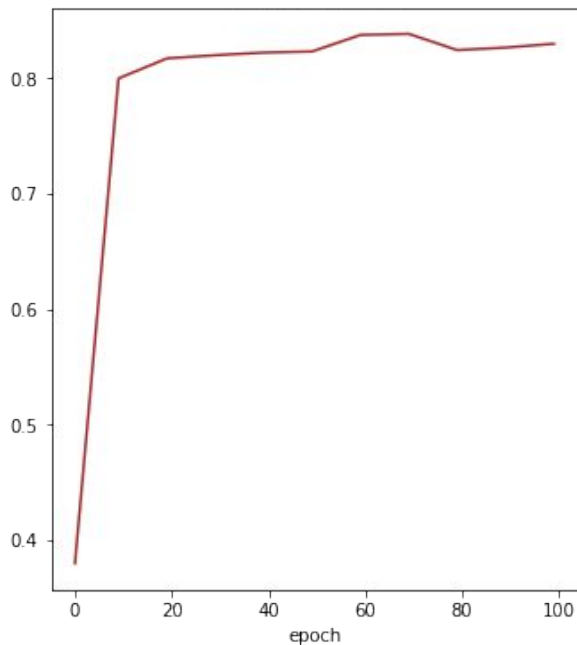


Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

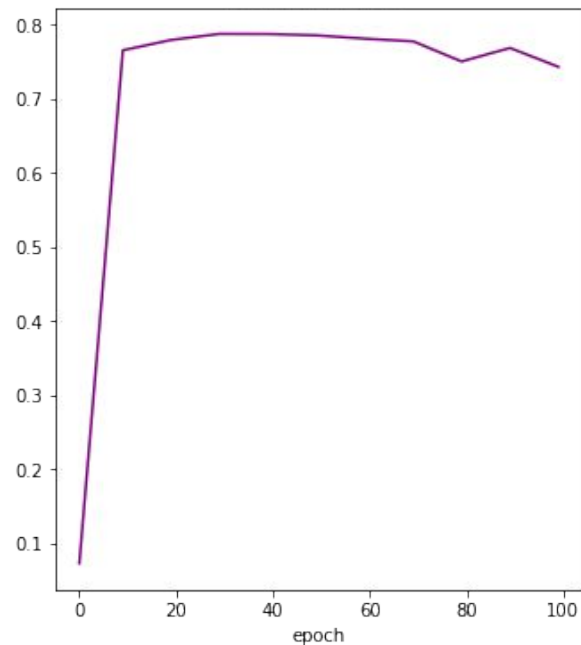
Val Mean Dice TC



Val Mean Dice WT



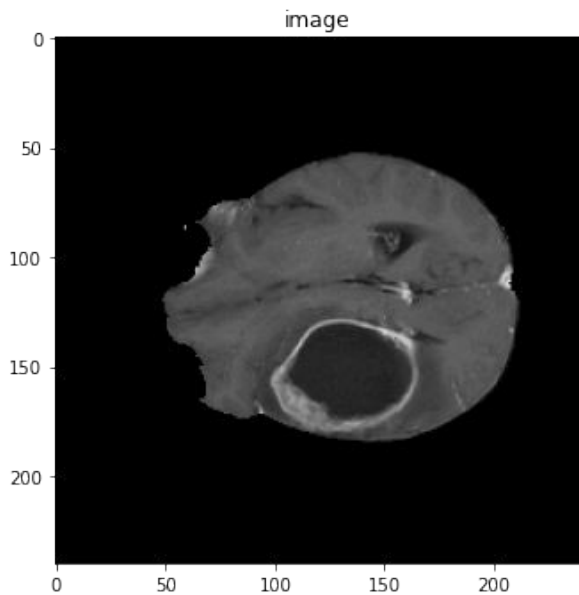
Val Mean Dice ET



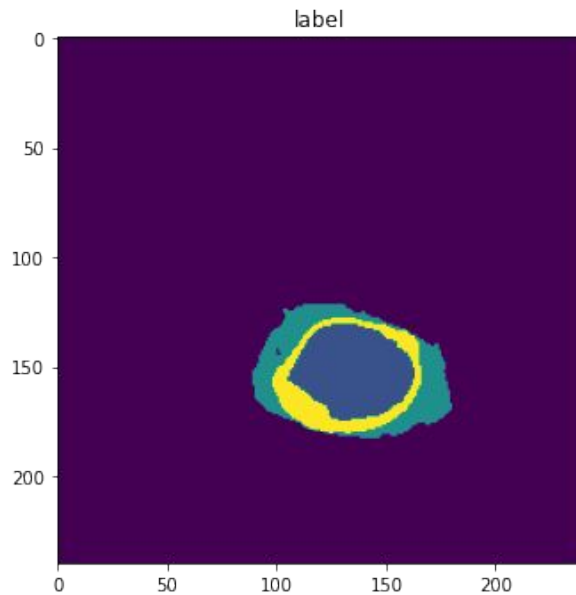


Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

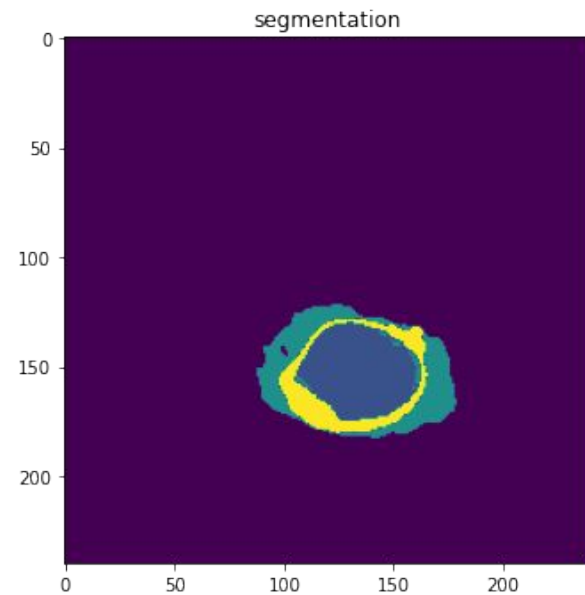
Best average dice: 0.7828



Input Image



Input Label



Predicted Mask



DEEPLABV3+ MODEL -

5 | Details of the Models

3. To test the accuracy of various deep learning models in order to develop a better model, we tested various models and simultaneously compared them as well.

The structure of the attached DeepLab model has following dependencies included:

- `sklearn.model_selection`
- `Tensorflow`
- Keras package: `keras.utils`, `keras.preprocessing`, `keras.models`, `keras.optimizers`, `keras.layers`

The structure of the attached model Deeplabv3+ is as follows On checking the metrics and results, we implemented DeepLab V3+.

- Preparation
- Data Processing
- Image Segmentation
 - *Helper Functions*
 - *Deeplabv3+*
- Result : *Training Loss and Metrics*
- Image Results



Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

DeepLab Versions	Key Features
DeepLabv1	Utilizes atrous (dilated) convolutions to capture multi-scale information.
DeepLabv2	Builds upon DeepLabv1 by incorporating the Spatial Pyramid Pooling (SPP) module, which captures multi-scale context information.
DeepLabv3	Uses Atrous Spatial Pyramid Pooling (ASPP), which employs parallel dilated convolutions at multiple rates to capture multi-scale features.
DeepLabv3+	Extends DeepLabv3 with skip connections from earlier layers to recover spatial information and improve boundary details.



Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

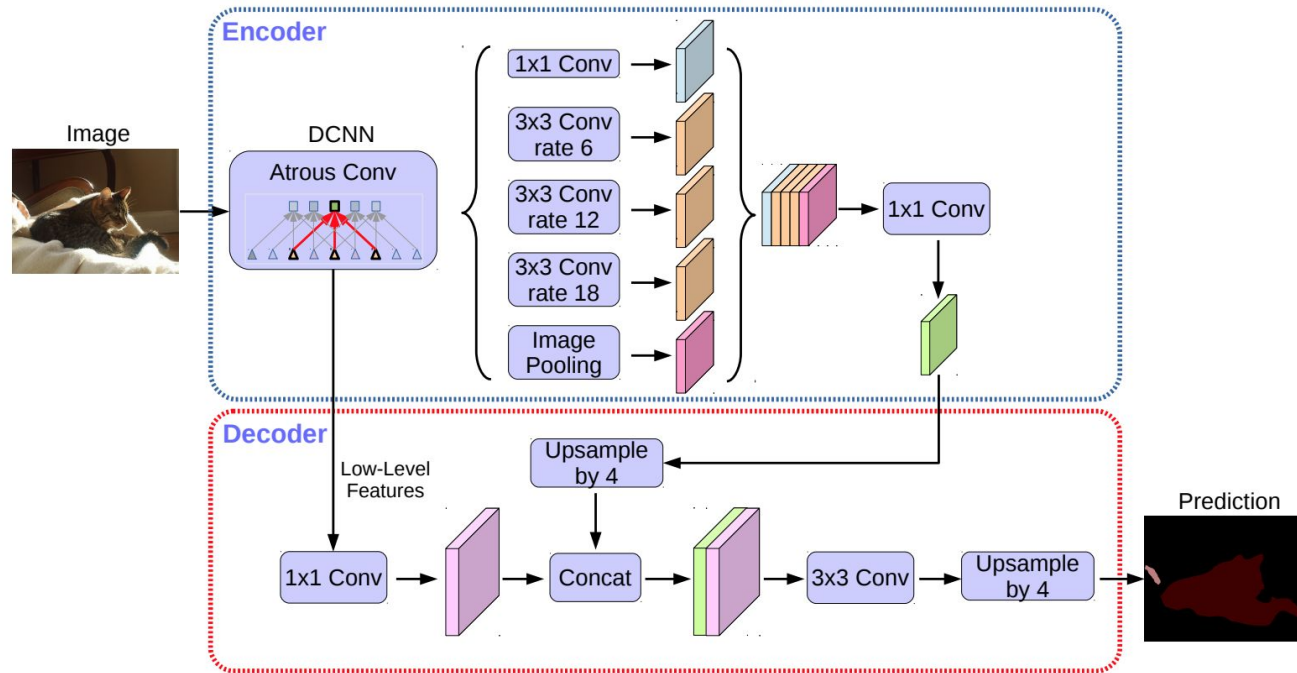
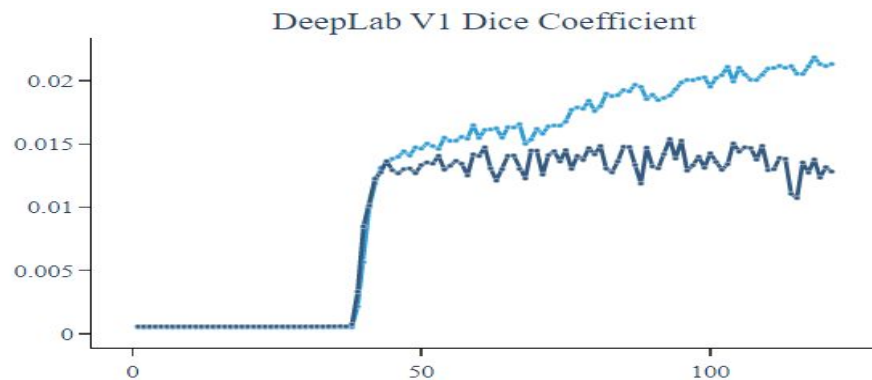
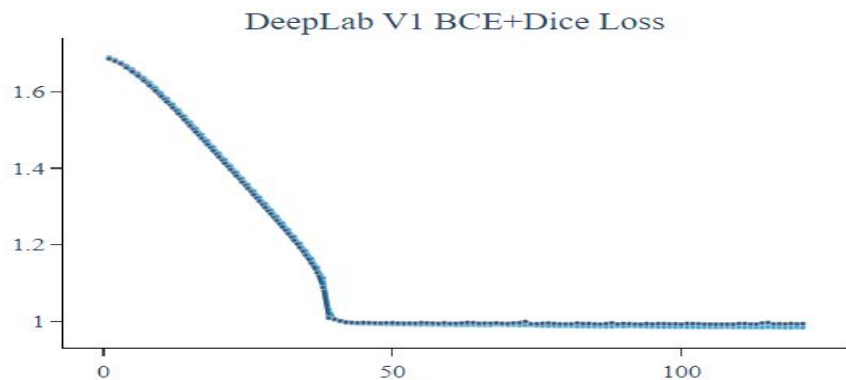
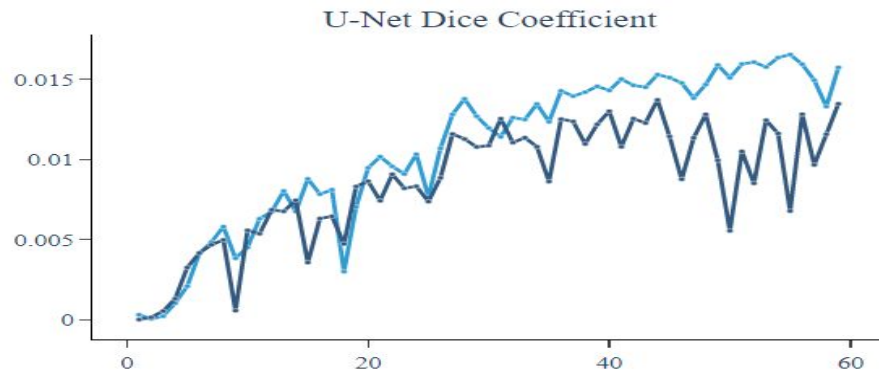
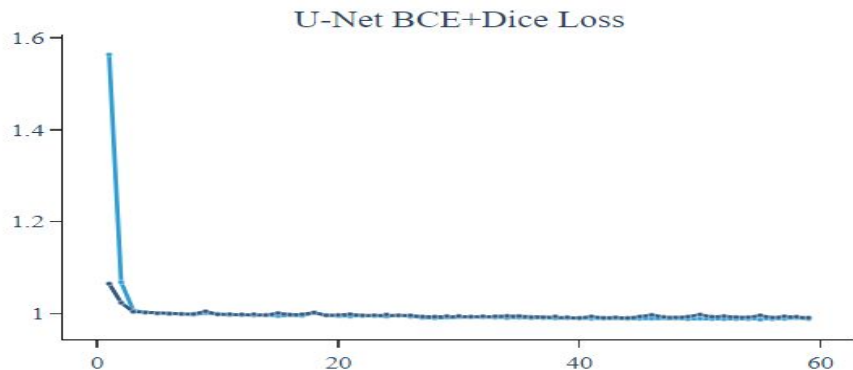


Figure: Deeplab model. [ref: <https://arxiv.org/abs/1802.02611>]



Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

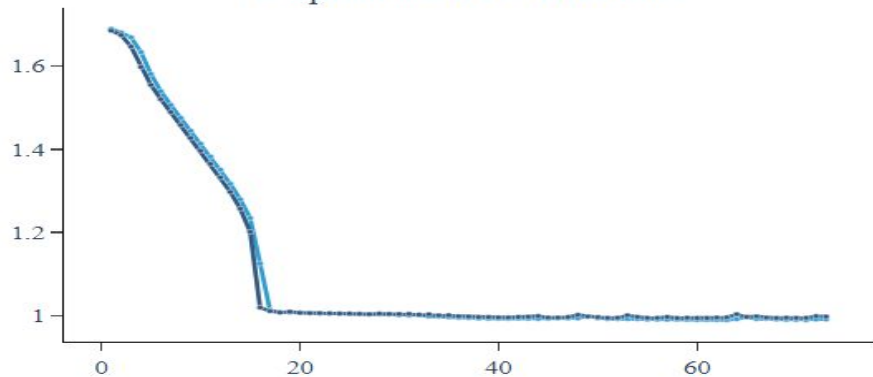
6.1 | U-Net and Deeplab family Training Metrics



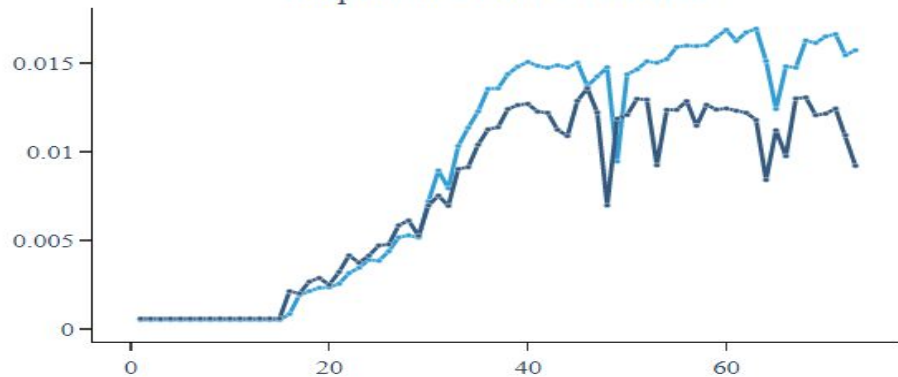


Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

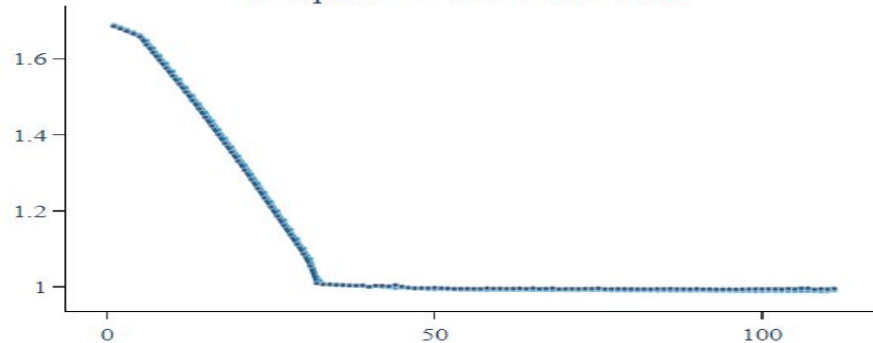
DeepLab V2 BCE+Dice Loss



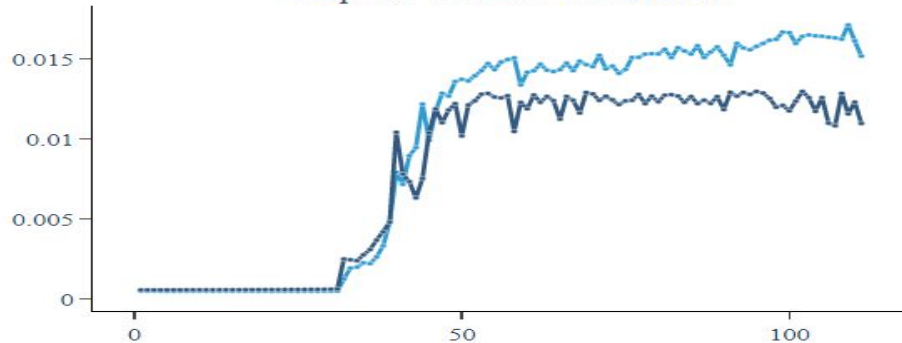
DeepLab V2 Dice Coefficient



DeepLab V3 BCE+Dice Loss



DeepLab V3 Dice Coefficient

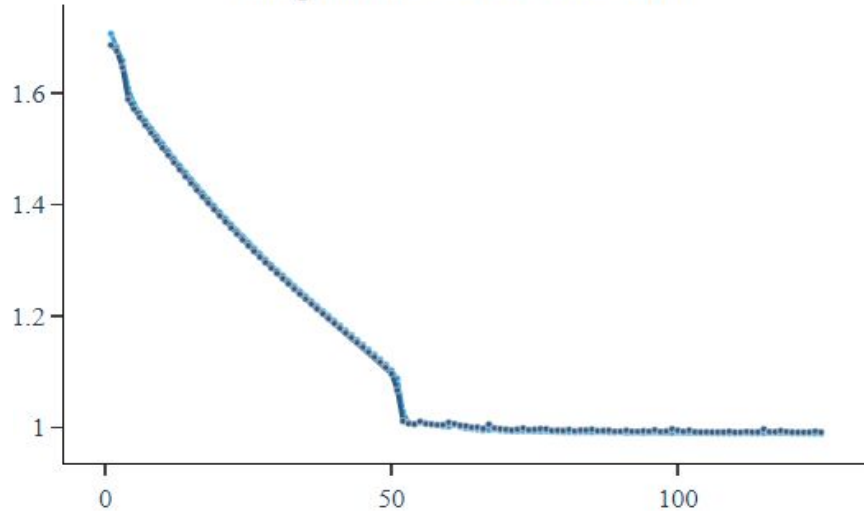




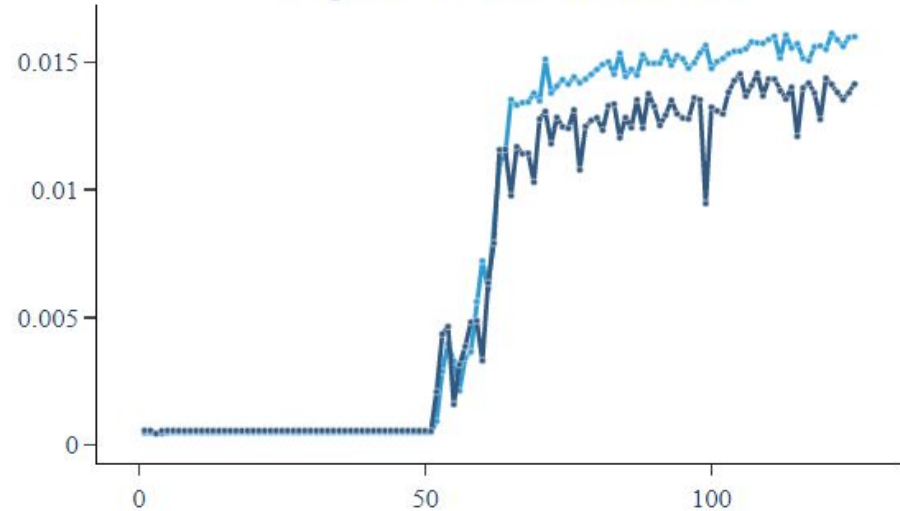
Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

6.1 | DeeplabV3+ model and Training Metrics

DeepLab V3+ BCE+Dice Loss



DeepLab V3+ Dice Coefficient

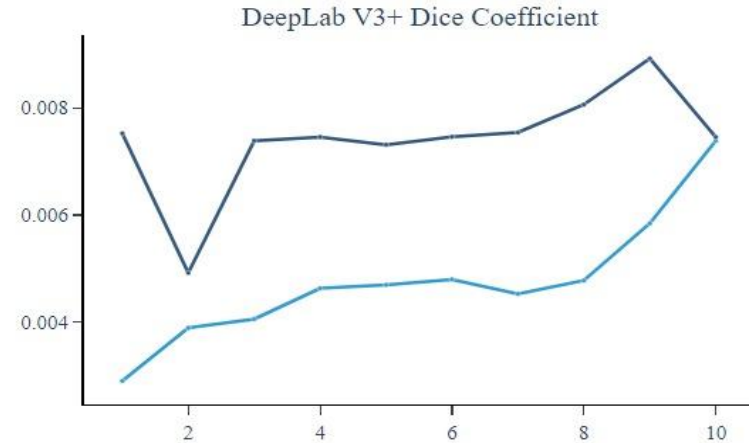
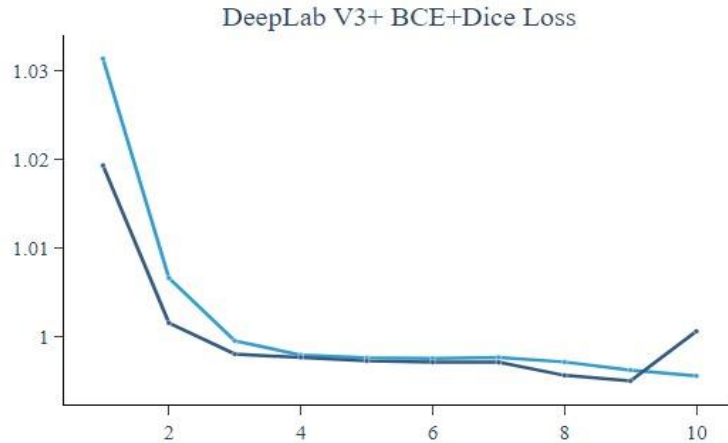




Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

6.1 | DeeplabV3+ model and Training Metrics

Training Loss and Metrics



Dice Coefficient = 0.231846 for epochs=10



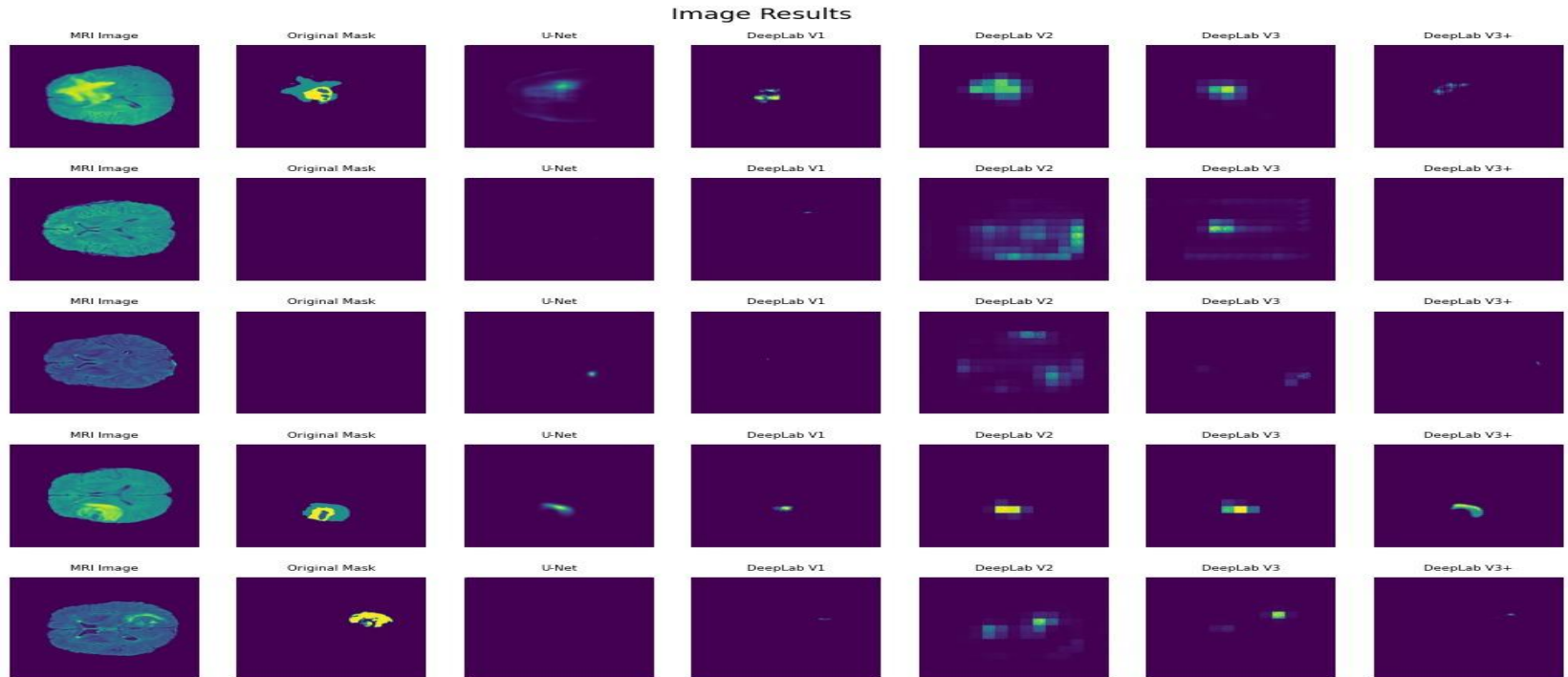
6.2 | Testing Metrics

	model	dice_coef
0	DeepLab V3	0.013697
1	DeepLab V2	0.024340
2	DeepLab V3+	0.035036
3	U-Net	0.087385
4	DeepLab V1	0.152041



Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

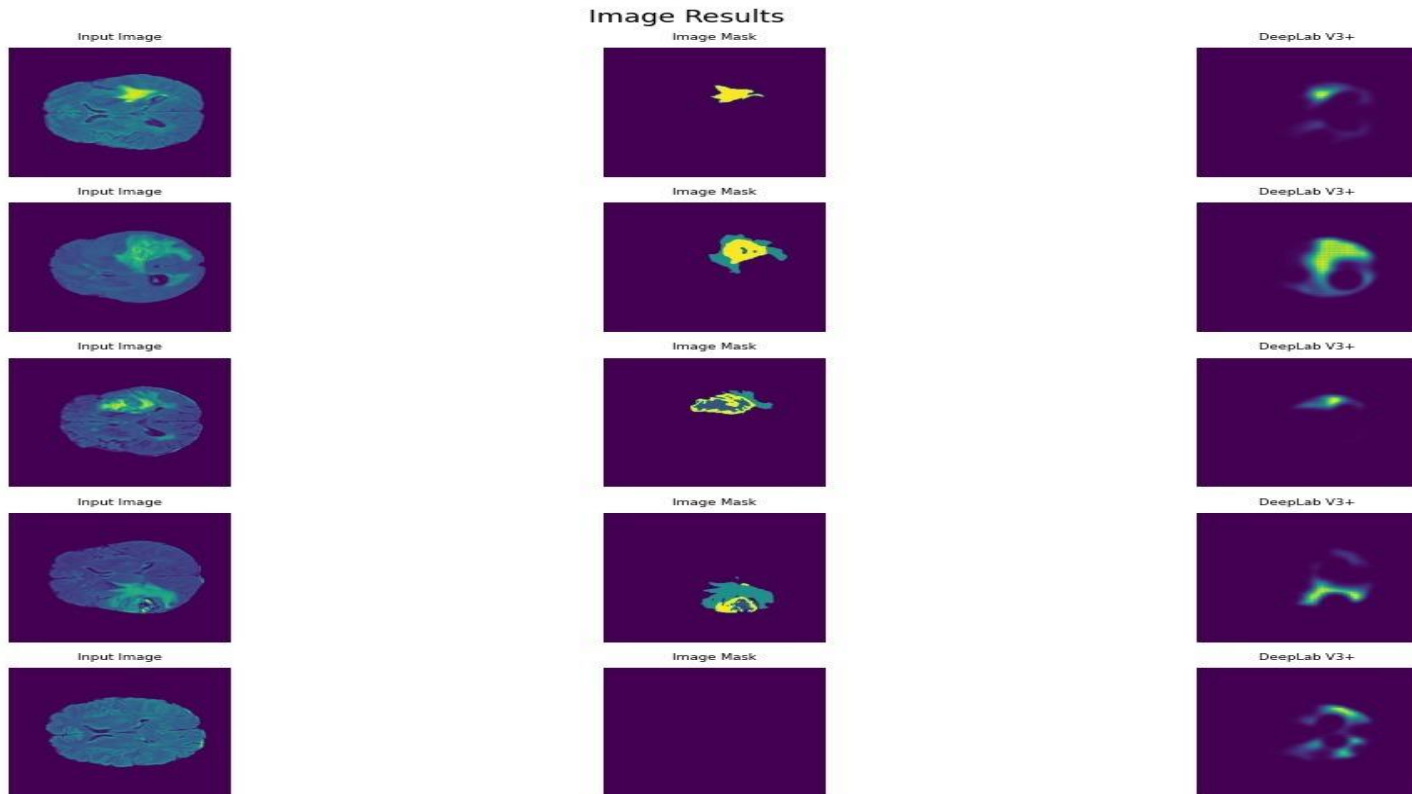
6 | Comparison of Deeplab family





Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

6 | Results of DeepLab V3+





7 | Learning Outcomes

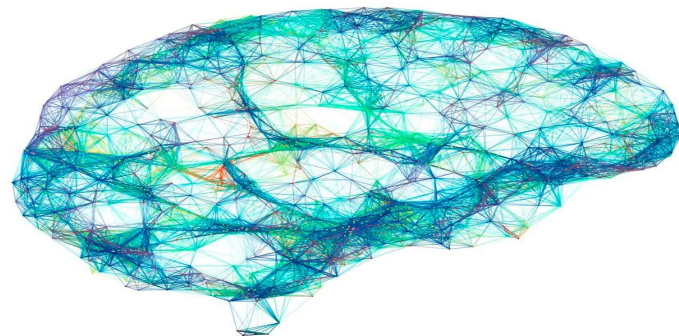
In this project we tried to develop a model to solve one of the prominent medical problems, which is Brain Tumor Sub-region segmentation.

Starting from a simple model based on simple thresholding and morphological operations.

Further, we explored UNet model, with which we could segment simply the tumor region.

On obtaining the testing metrics, we implemented the DeepLabV3+ Model for the sub-region segmentation

Finally, we explored a transformer based model, SWIN UNetR model, with which we could segment the subregions to a great accuracy.

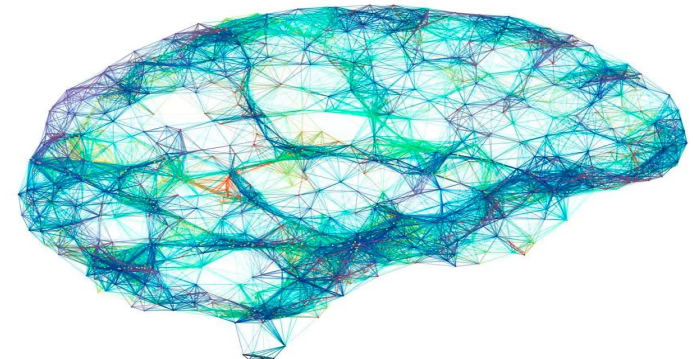




Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

8 | References

1. Hatamizadeh, A., Nath, V., Tang, Y., Yang, D., Roth, H., Xu, D. (2022). Swin UNETR: Swin Transformers for Semantic Segmentation of Brain Tumors in MRI Images. ArXiv. /abs/2201.01266
2. Nasim, M. A., Munem, A. A., Islam, M., Palash, M. A., Haque, M. M., Shah, F. M. (2022). Brain Tumor Segmentation using Enhanced U-Net Model with Empirical Analysis. ArXiv. /abs/2210.13336
3. Chen, L., Papandreou, G., Schroff, F., Adam, H. (2017). Rethinking Atrous Convolution for Semantic Image Segmentation. ArXiv. /abs/1706.05587





Brain Tumour Sub-Region Segmentation (BraTS21 Dataset)

THANK YOU!

