Summer Internship Project
Report

Brain Tumor Segmentation

Submitted by

Sri Manoj Kondapalli
106118049
National Institute of Technology, Tiruchirappalli

Under the guidance of

R. Mohan
Assistant Professor, NIT Trichy

Department of Computer Science and Engineering
National Institute of Technology
Tiruchirapalli

Summer Internship 2021

Acknowledgment

The internship has been one of my great learning experiences and I would like to express my sincere gratitude to my advisor Prof. Mohan R and my mentor Hari Haran for the continuous support and encouragement throughout the project.I would not have been imagined better advisor and mentor for my internship.I would like to thank for Department of CSE,NIT Trichy for allowing me to do internship under their guidance. It has been my pleasure to work under such a great team.

I would like to thank my teachers for providing me with an exceptional knowledge that helped me complete the project.My completion of this project could not have been accomplished without the support of my parents - thank you for allowing me time away from you to complete my work.

National Institute of Technology Tiruchirappalli-620015

Department of Computer Science and Engineering

To Whom it may concern

This is to certify that **Sri Manoj Kondapalli,** Roll No**. 106118049**, a student of B. Tech Computer Science and Engineering, **National Institute of Technology of Tiruchirappalli (NITT)** worked under my supervision during his internship period. I am pleased to state that he worked hard in preparing this report and he has been able to present a good picture of the concerned works.

I wish him all the best for his future endeavour.

Warm regards,

Dr. R. Mohan
Assistant Professor
Department of Computer Science and Engineering
National Institute of Technology, Tiruchirappalli
Tamil Nadu, India
E-mail: rmohan@nitt.edu

Abstract

Qualitative analysis of brain tumor is critical. Manual segmentation is tedious, time consuming and requires a lot of human expertise, at the same time, this task is challenging for automatic segmentation methods. In past few years CNN based architectures have been proven to be the most successful. Neworks based on U-shaped architectures have been giving state-of-art performance over others. Due to the locality of the convolution operator, it inhibits CNN to learn global and long-range spacial information. We have implemented a recent state-of-art in Brain Tumor segmentation inspired by swin-transformers and U-Net. The network architecture is a modification of classic U-Net architecture by utilizing the power of modern swin-transformers.

Contents

Chapter 1

Introduction about the Industry

The National Institute of Technology Tiruchirappalli, was started as a joint and co-operative venture of the Government of India and the Government of Tamil Nadu during 1964 with a view to catering to the needs of manpower in technology for the Country. The College has subsequently been conferred with autonomy in financial and administrative matters to achieve rapid development. Because of this rich experience, this institution was granted Deemed University Status with the approval of the University Grants Commission, the All-India Council for Technical Education, and the Govt. of India in the year 2003.

NIT-T strives its best to position itself at the forefront of cutting-edge research in pace with global standards. Research activities at NIT-T have been growing in all metrics with respect to the quantity and quality of researchers. There are several sponsored projects currently funded by MHRD, DST, SERB, CSIR, DRDO, ISRO, GTRE, AICTE, RGNIYD, DEITY, DAE. In addition to this, major consultancy projects with agencies like BHEL, CPW, PWD, Airport Authority of India, NLCL, CDAC are also undertaken across different departments of the Institute. The scholarly output of the institute per year is on an average of 700 publications and 10000 citations. In addition to this, the research community of the institute actively engages in translating novel ideas to a product/process and has several published and granted patents to its credit.

Chapter 2

Training Schedule

Week 1

Get to know about BraTS benchmarks and metrics of evaluation.

- BraTS challenge focuses on evaluation of state-of-art methods for the segmentation of brain tumors in multi-parametric magnetic resonance imaging(mpMRI) scans. Evaluation metrics considered are dice-loss, sensitivity, specificity.

Week 2

Review some papers in previous BraTS challenges.

- Reviewed some standard approaches in brain tumor segmentation like Unet [6], Multi-encoder net [10], cross-modality deep feature learning [9], 3D Unet [4], Assembly net , Trans Unet [3] and Swin Unet [2].

Week 3

Review some implemented models in BaraTS.

- Examined some implemented models such as Unet, Trans Unet.

Week 4

Train a simple Unet model on 2017 BraTS dataset.

- Trained a simple Unet model on 2017 BraTS dataset and achieved good results.

Week 5

Preprocessing,metrics modules.

- Preprocessing involves loading all image paths to a data-frame, loading nifti image, resizing and standardizing image.

Week 6

Data Generator module.

- Data generator loads data-frame and preprocesses images and prepares a batch-size of images to the model per iteration.

Week 7

Swin-Transformer block.

- Swin transformer block comprises of 2 swin transformers

Week 8

Swin-Unet module and training the model.

- Coding Swin-Unet model.
- Reviewing the final model and training the model.

Chapter 3

Introduction

3.1 Challenges

Gilomas are one of the most common primary brain tumors. World Health Organization(WHO) reports that gilomas can be graded from level one to four based on the behaviours and severity. Grade I and II are Low-Grade-Gilomas(LGG) which are close to benign and slow growing cases. Grade III and IV are High-GradeGilomas(HGG) which are cancerous and aggressive. Image segmentation plays an important role in diagnosis and treatment. We define brain tumor segmentation as: Given an image frame from one or multiple MRI sequences, the system aims to automate the segmentation of the tumor area from the tissues and to classify each voxel/pixel of input data into a pre-defined category.

The BraTS [1] challenge utilizes multi-institutional pre-operative baseline multi-parametric magnetic resonance imaging (mpMRI) scans, and focuses on the evaluation of state-ofthe-art methods for the segmentation of intrinsically heterogeneous brain glioblastoma sub-regions in mpMRI scans.

These mpMRI scans describe

- native (T1)
- post-contrast T1-weighted (T1Gd)
- T2-weighted (T2)
- T2 Fluid Attenuated Inversion Recovery (T2-FLAIR) volumes

3.2 Background

Despite the extent we have emerged in the field of brain tumor segmentation, even today's state-of-art architectures still experience severe challenges in certain aspects.

- **Location Uncertainity**
  Due to wide spatial distribution of gilomas, tumor can appear in any location inside brain with varying sizes.
- **Low Contrast**
  Due to inability to access direct imaging, scanned images are of low quality and boundaries are hard to detect.
- **Imbalanced number of pixels**
  There exists an unbalanced number of voxels/pixels in different regions which makes the learning algorithm to dominate extracted regions with large tumor size.

3.3 Related Work

3.3.1 Convolution Neural Network based models

Since their first use in hand-written digit recognition and image classification CNNs have been widely used in vision tasks. CNN have strong capability to process and learn features from images and videos. Convolution Neural Networks (CNN) contain stacked convolution and pooling layers, which can effectively capture the translational invariance of the input classes. These can be divided into (1) Single path CNN, (2) Multi path CNN, (3) Fully convolution networks (FCN). Single path neural networks consists of single flow of data processing from input to output classification. It is preferred for its computational efficiency.

Multi path CNN comprises of different paths for processing image and are concatenated at later layers using fully connected layers or CNNs such as hetero-modal image segmentation. Earlier convolution architectures uses fully connected layers as final layers which produces single or n values corresponding to probability of the classes. Later fully convolution layers were introduced which involves deconvolution layer as an output layer, due to computational efficiency. Most highlighted FCN architecture is Unet [6] which involves downsampling and upsampling paths for encoding information and skip connections for information preserving.

3.3.2 Transformer based models

Self-attention [7] based models have been known to preserve information on long sequences. Usage of transformers have been boosted since its excellence in sequential and text based data. Many architectures involved Unet based model induced with selfattention mechanism. TransBTS [8] is one such architecture having self-attention in between the downsampling and upsampling layers in Unet model. TransUnet [3] involves transformers as downsampling layer to encode the data and CNNs to upsample to original size. SwinUnet [2] replaces CNN with swin-transformers making it a pure transformer based architecture.

3.3.3 Generative model based methods

Generative Adversial Networks (GAN) are formed by a generator and a discriminator playing a min-max game to optimize the model. Cross modality deep feature learning based approach uses one modality to generate other and vice-versa using a generator with cycleGAN [11] learning scheme and discriminator distinguishes between original and generated modalities. RescueNet [5] utilizes two generators and discriminators for each scan and label training simultaneously, while one side accounting for adversial loss and other side for cycle-consistensy loss.

Chapter 4

Work Done

4.1 Method

Even though CNNs have been successfull in Brain tumor segmentaion, transformers have been in this field since the introduction of Vision Transformer. We chose Swin-Unet as it holds the power of swin-transformer and method of vision transformer along with tweaks such as cyclic shifted window scheme

4.1.1 Patch Extraction

A standard transformer takes in inputs as a 1D vector of tokens. To input a 2D image to a transformer, image is reshaped into flattened 2D patches. Image of shape

$$R^{HXWXC}$$

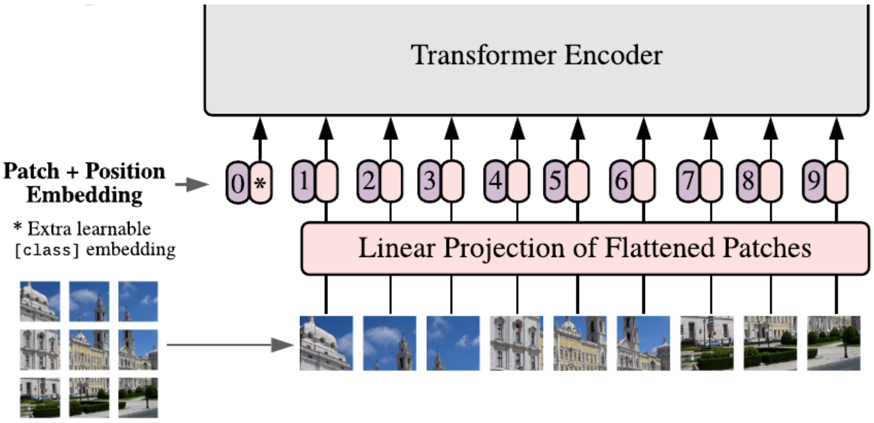
,where (H,W) is dimension of original image, is transformed into

$$R^{NX(PXPXC)}$$

where each image is of dimension (P,P) and

$$N = H.W/P^2$$

which are number of patches produced.

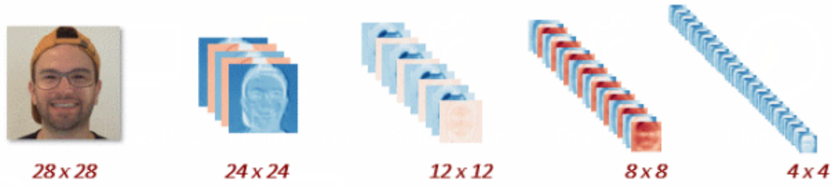

Transformer Encoder
Patch + Position Embedding
* Extra learnable [class] embedding
0
*
1
2
3
4
5
6
7
8
9
Linear Projection of Flattened Patches

4.1.2 Patch Embeddings

Transformer imputs a vector size of D(Embedding Dimension), so all patches are flattened and mapped to a D dimensions with a trainable linear projection. Output of this projections are called Patch Embeddings.

4.1.3 Patch Merging

Starting with high-size image patches, along the layers neibouring patches are merged which serves as compressing the contextual information. Patch merging layer merges neighboring 4 patches into one. Feature resolution will be downsampled by 2x.



4.1.4 Patch Expanding

A linear layer is applied on input features to increase the feature dimension to double the original dimension. Then rearranges to expand the resolution of input features to 2x input resolution and to reduce the feature dimension to quarter its input dimension. So by overall input resolution is doubled in each dimension(a total of 4 times) with feature dimension halved.

- doubling the feature dimension

$$H \times W \times C \rightarrow H \times W \times (2C)$$

- doubling resolution and quartering feature dimension

$$H \times W \times (2C) \rightarrow 2H \times 2W \times (C/2)$$

- Overall,

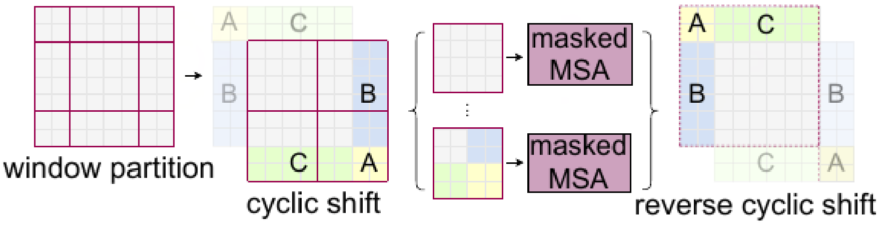
$$H \times W \times C \rightarrow 2H \times 2W \times (C/2)$$

4.1.5 Window partition

Unlike classic transformer, a shifted window scheme is proposed in swin transformer which brings greater efficiency by limiting non-overlapping local windows while also allowing for cross-window connections. While non-overlapping w-msa,window based self-attention which doesnt consider connections between neighboring windows,shiftedw-msa overcame this.

4.1.6 Cyclic and Reverse cyclic shift of windows

While shifted window partitioning, all windows are of different size. A naive solution is to pad smaller windows so as to make all windows same size, but computation complexity increases massively for large images. Thus at each partition, smaller windows are cyclically shifted so as to make all windows same size.

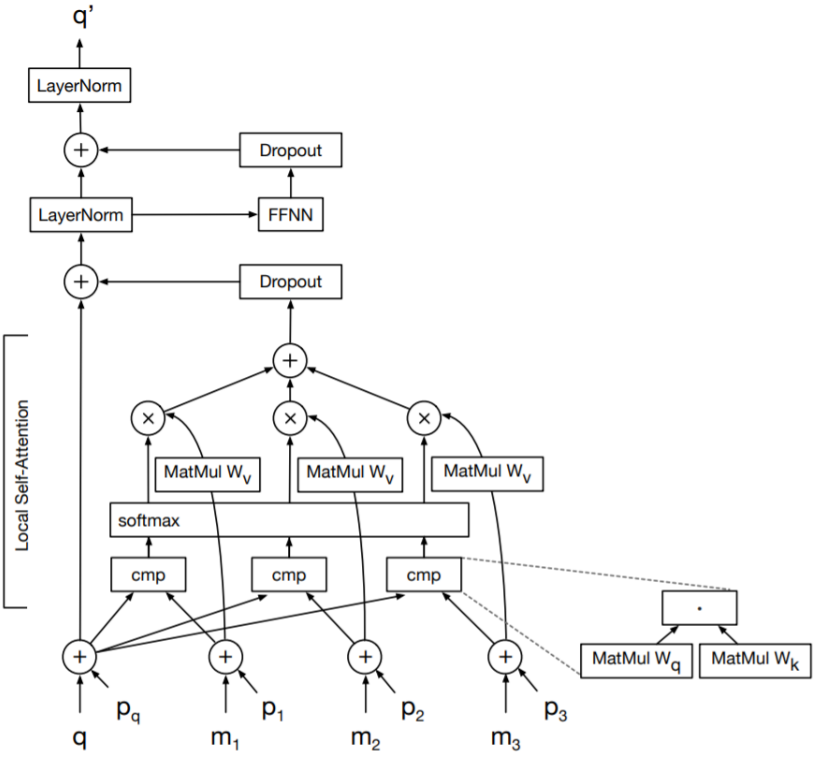


4.1.7 Window Attention

Self-attention is done on windows which are cyclically or reverse cyclically shited. While doing self-attention, a positional bias is added to leverage the positional embedding dependence.

$$Attention(Q, K, V) = SoftMax(QK^T/\sqrt{d} + B)V$$

where Q,K and V are query,key and value matrices.

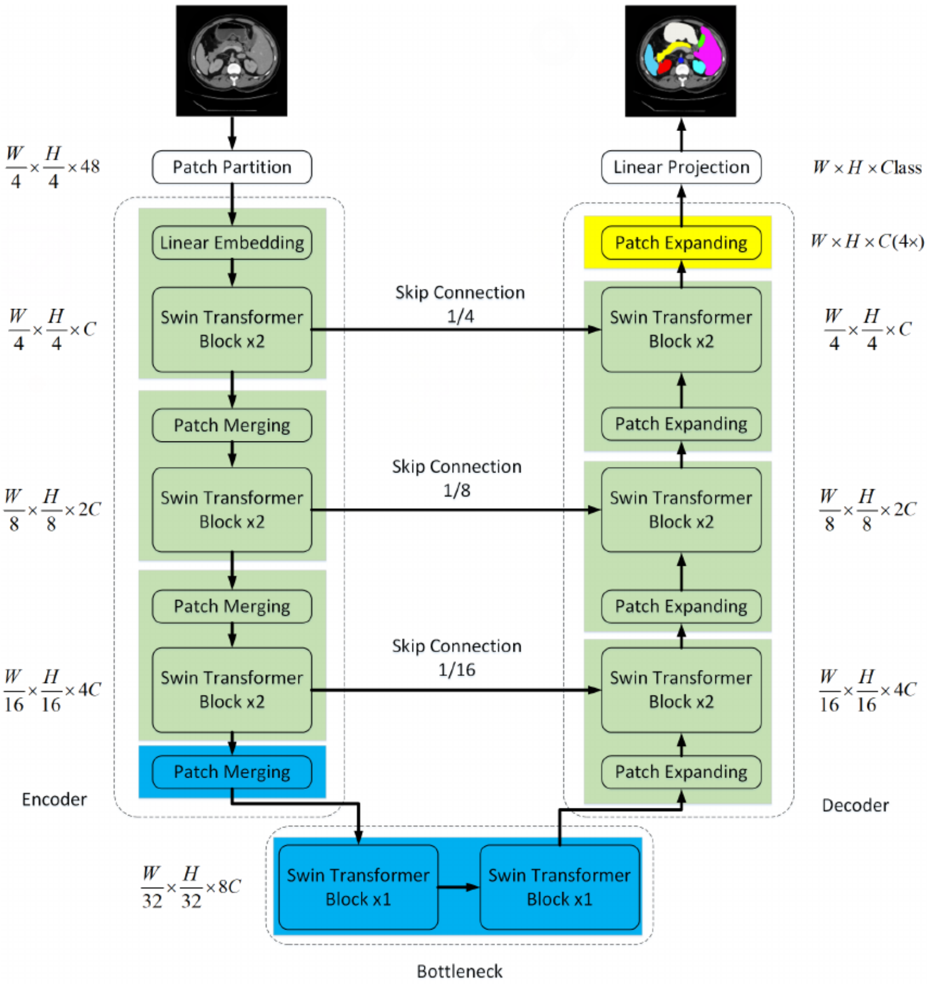


4.1.8 Skip connections

Skip connections serve the same purpose as of Unet architecture i.e to accumulate the spacial information to encoder, which is lost during decoding.

4.1.9 Swin-Transformer Block

Different from a normal transformer architecture, window based multi-head selfattention(W-MSA) and shifted-window based multi-head self-attention(SW-MSA) are applied over two consecutive transformer blocks. Each transformer is preceeded by a Layer Normalization(LN) and a skip connection after self-attention followed by a LN and a linear layer followed by another skip connection.


$\frac{W}{4}\times\frac{H}{4}\times48$
Patch Partition
Linear Embedding
$\frac{W}{4}\times\frac{H}{4}\times C$
Swin Transformer Block x2
Patch Merging
$\frac{W}{8}\times\frac{H}{8}\times2C$
Swin Transformer Block x2
Patch Merging
$\frac{W}{16}\times\frac{H}{16}\times4C$
Swin Transformer Block x2
Patch Merging
Encoder
Skip Connection 1/4
Skip Connection 1/8
Skip Connection 1/16
Linear Projection
$W\times H\times \text{Class}$
Patch Expanding
$W\times H\times C(4\times)$
Swin Transformer Block x2
$\frac{W}{4}\times\frac{H}{4}\times C$
Patch Expanding
Swin Transformer Block x2
$\frac{W}{8}\times\frac{H}{8}\times2C$
Patch Expanding
Swin Transformer Block x2
$\frac{W}{16}\times\frac{H}{16}\times4C$
Patch Expanding
Decoder
$\frac{W}{32}\times\frac{H}{32}\times8C$
Swin Transformer Block x1
Swin Transformer Block x1
Bottleneck

4.1.10 Encoder

Initially inputs are passes through a patch-extraction layer followed by a patch embedding and positional-embedding layers. At each depth, inputs are passed through a patch-merging layer followed by a swin-transformer block.

4.1.11 Decoder

At the encoder part, all features are extracted and are passed through decoder. At each decoding layer, inputs are passed through a patch-expansion layer followed by concatenation of skip connection to preserve the spatial information and then passed through a dense layer followed by swin-transformer block. Finally a convolution layer is added so as to output the segmentation as of our required dimension.

4.1.12 Swin-Unet

Swin-Unet [2] implemented utilises the Unet [6] simplicity followed by transformers power to learn long-range dependences. While patch-expansion and patch-merging layers deal with input dimensions consistensy among transformers, skip connections, window partition and cyclic shifting of windows leverages the model to learn deep features.

4.2 Metrics

Main evaluation metrics are an overlap measure(DSC), similarity and specificity. The commonly used Dice Similarity Coefficient(DSC) measures the overlap between two sets. It can be defined as :

$$DSC = 2TP/2TP + FP + FN$$

where
TP: Correctly segmented or classified brain tumor
FP: Wrong segmentation or classification of an ordinary tissue
FN: Wrong Classification of actual tumor tissue as non-tumor
TN: Correct prediction of non-tumor tissue as non-tumor

Sensitivity and specificity mathematically describe the accuracy of a test which refers the presence or absence of a condition. In a diagnostic test, sensitivity or recall refers to how well the test can identify true positives and specificity is a measure of how well the test identifies true negatives. There is usually a trade-off between sensitivity and specificity, such that higher sensitivities meaning lower specificity and vice versa.

$$Sensitivity = TP/(TP + FN)$$

$$Specificity = TN/(TN + FP)$$

4.3 Contribution

4.3.1 Data Generator

As data is too large to fit in memory, we have devised a data generator to feed the model a batch size of images in each iteration.

```
class CustomDataGen(Sequence):

    ...

    def __getitem__(self,index):
        #get some slices of 2D images of a given image
        X = self._generate_X(flair_list,t1_list,t1ce_list,t2_list)

        #get masks corresponding to X
        y = self._generate_y(mask_list)
        return X,y


    def _generate_X(self,flair_path,t1_path,t1ce_path,t2_path):
        paths = [flair_path,t1_path,t1ce_path,t2_path]
        X = np.empty((10,256,256,4))
        for modality in range(len(paths)):
            ID = paths[modality]
            image_3d = self._load_image(ID)
            for depth in range(image_3d.shape[2]):
                image_2d = image_3d[:,:,depth]
                X[depth,:,:,modality] = image_2d
        return X

    def _generate_y(self,mask_path):
        y = np.empty((10,256,256,1))
        ID = mask_path
        image_3d = self._load_image(ID)
        image_bool = image_3d>0
        image_binary = image_bool.astype(int)
        for depth in range(image_3d.shape[2]):
            image_2d = image_3d[:,:,depth]
            y[depth,:,:,0] = image_2d
        return y
```

4.3.2 Window Attention

```
x_qkv = self.qkv(x)
x_qkv = tf.reshape(x_qkv,
                shape=(-1, N, 3, self.num_heads, head_dim))

x_qkv = tf.transpose(x_qkv, perm=(2, 0, 3, 1, 4))
q, k, v = x_qkv[0], x_qkv[1], x_qkv[2]

# Query rescaling
q = q * self.scale

# multi-headed self-attention
k = tf.transpose(k, perm=(0, 1, 3, 2))
attn = (q @ k)

# Dropout after attention
attn = self.attn_drop(attn)

# Merge qkv vectors
x_qkv = (attn @ v)
x_qkv = tf.transpose(x_qkv, perm=(0, 2, 1, 3))
x_qkv = tf.reshape(x_qkv, shape=(-1, N, C))

# Linear projection
x_qkv = self.proj(x_qkv)

# Dropout after projection
x_qkv = self.proj_drop(x_qkv)

return x_qkv
```

4.3.3 Swin Transformer Block

```
x_skip = x

# Layer normalization
x = self.norm1(x)

# Convert to aligned patches
x = tf.reshape(x, shape=(-1, H, W, C))

# Cyclic shift
if self.shift_size > 0:
    shifted_x = tf.roll(x,
            shift=[-self.shift_size, -self.shift_size], axis=[1, 2])
else:
    shifted_x = x

# Window partition
x_windows = window_partition(shifted_x, self.window_size)
x_windows = tf.reshape(x_windows,
        shape=(-1, self.window_size * self.window_size, C))

# Window-based multi-headed self-attention
attn_windows = self.attn(x_windows, mask=self.attn_mask)

# Merge windows
attn_windows = tf.reshape(attn_windows,
            shape=(-1, self.window_size, self.window_size, C))
shifted_x = window_reverse(attn_windows, self.window_size, H, W, C)

# Reverse cyclic shift
if self.shift_size > 0:
    x = tf.roll(shifted_x,
        shift=[self.shift_size, self.shift_size], axis=[1, 2])
else:
    x = shifted_x

# Convert back to the patch sequence
x = tf.reshape(x, shape=(-1, H*W, C))

# Drop-path
# if drop_path_prob = 0, it will not drop
x = self.drop_path(x)

# Skip connection I (end)
x = x_skip +  x

# Skip connection II (start)
x_skip = x

x = self.norm2(x)
x = self.mlp(x)
x = self.drop_path(x)

# Skip connection II (end)
x = x_skip + x

return x
```

Chapter 5

Results and Analysis

Considering the depth of the model, after around 30 iterations all the metrics started to show as NaN which indicates unstable gradients problem. We tried varoious ways to overcome this situation such as:

- Using ReLU activation
- Tweaking Learning rate
- Changing batch size

We thought of other ways but unlike neural networks, transformers cant use Xavier and He Initialization to overcome this problem. At last our model just halted there with unstable gradients.

Chapter 6

Conclusion

We implemented a pure-transformer based U-shaped architecture using Swin-transformers. Even though fully convolution based architectures such as Unet has proven to be successful. Replacing CNNs with transformers should be the next best advancement due to self-attention mechanism. Vision Transformer boosted the usage of transformers in image classification and segmentation. To leverage the power of transformers, normal transformers have been replaced with Swin transformers utilizing vision transformer approach to handle image segmentation. Local window partitioning is replaced with shifted-window followed by cyclic shifts to enhance cross window connections.[].

References

[1] Brain tumor segmentation (brats) challenge, 2021.

[2] Hu Cao, Yueyue Wang, Joy Chen, Dongsheng Jiang, Xiaopeng Zhang, Qi Tian, and Manning Wang. Swin-unet: Unet-like pure transformer for medical image segmentation, 2021.

[3] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan L. Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation, 2021.

[4] Fabian Isensee, Philipp Kickingereder, Wolfgang Wick, Martin Bendszus, and Klaus H. Maier-Hein. Brain tumor segmentation and radiomics survival prediction: Contribution to the BRATS 2017 challenge. *CoRR*, abs/1802.10508, 2018.

[5] Shubhangi Nema, Akshay Dudhane, Subrahmanyam Murala, and Srivatsava Naidu. Rescuenet: An unpaired gan for brain tumor segmentation. *Biomedical Signal Processing and Control*, 55, 01 2020.

[6] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

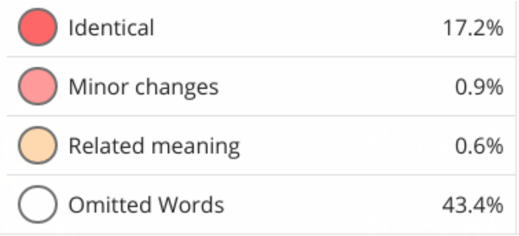
[8] Wenxuan Wang, Chen Chen, Meng Ding, Jiangyun Li, Hong Yu, and Sen Zha. Transbts: Multimodal brain tumor segmentation using transformer, 2021.

[9] Dingwen Zhang, Guohai Huang, Qiang Zhang, Jungong Han, Junwei Han, and Yizhou Yu. Cross-modality deep feature learning for brain tumor segmentation. *Pattern Recognition*, 110:107562, 07 2020.

[10] Wenbo Zhang, Guang Yang, He Huang, Weiji Yang, Xiaomei Xu, Yongkai Liu, and Xiaobo Lai. Me-net: Multi-encoder net framework for brain tumor segmentation. *International Journal of Imaging Systems and Technology*, 03 2021.

[11] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *CoRR*, abs/1703.10593, 2017.

Similarity Check



●	Identical	17.2%
●	Minor changes	0.9%
●	Related meaning	0.6%
○	Omitted Words	43.4%