Segmenting Metastatic Brain Tumor Using Deep Learning

THESIS

Submitted in Partial Fulfillment of

the Requirements for

the Degree of

MASTER OF SCIENCE (Computer Engineering)

at the

NEW YORK UNIVERSITY
TANDON SCHOOL OF ENGINEERING

by

Ankush Pratap Singh

May 2023

Segmenting Metastatic Brain Tumor Using Deep Learning

THESIS

Submitted in Partial Fulfillment of

the Requirements for

the Degree of

MASTER OF SCIENCE (Computer Engineering)

at the

NEW YORK UNIVERSITY
TANDON SCHOOL OF EnGineerinG

by

Ankush Pratap Singh

May 2023

Approved:

Wang Yao

Advisor Signature

5/18/2023

Date

Ivan Selesnick

Department Chair Signature

May 19, 2023

Date

Approved by the Guidance Committee:

Major: Computer Engineering

Wang Yao

Yao Wang

Professor
Electrical and Computer Engineering

Date: 5/18/2023

Anna Choromanska

Assistant Professor
Electrical and Computer Engineering

Date: 05/19/2023

F. Atashzar

S. Farokh Atashzar

Assistant Professor
Electrical and Computer Engineering

Date: 05/19/2023

Microfilm or other copies of this thesis are obtainable from

UMI Dissertation Publishing
ProQuest CSA
789 E. Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Vita

Ankush Pratap Singh was born and raised in the bustling city of New Delhi, India. He began his academic journey in September 2013 at Netaji Subhas Institute of Technology, where he pursued a Bachelor of Engineering in Instrumentation and Control.

After graduating, Ankush worked as a software developer for 3.5 years, gaining experience in developing and implementing software solutions for a variety of industries. However, his passion for learning and research led him back to academia.

In September 2021, Ankush began his graduate studies at New York University (NYU), where he pursued a Master of Science degree in Computer Engineering.

As part of his coursework, he took a keen interest in the field of computer vision and biomedical image segmentation, leading him to join the NYU Video Lab in September 2022 to work on his thesis till May 2023.

Acknowledgments

I express my sincere gratitude to my advisor, Prof. Yao Wang, for her unwavering support, valuable insights, and encouragement throughout my research journey and the writing of this thesis. Her guidance and mentorship have been indispensable in shaping my ideas and improving the quality of my work. Her rigorous attitude towards research has always inspired me to strive for excellence and push my boundaries.

I am also thankful to Prof. Anna Choromanska and Prof. S. Farokh Atashzar for serving on my committee and providing valuable advice and feedback on my work. Their expertise and insights have been instrumental in shaping my research direction and improving the quality of my thesis.

I would like to acknowledge Chris Liu, who had previously worked on this research project and had conducted some of the initial studies. His support, encouragement, and willingness to share his knowledge and expertise have been invaluable in advancing my research.

Lastly, I am grateful to my parents for their unwavering love and support throughout my academic journey. Their constant encouragement and belief in me have motivated me, and I am forever indebted to them.

Dedicated to maa, paa, and bhai

Har Har Mahadev

ABSTRACT

Segmenting Metastatic Brain Tumor Using Deep Learning

by

Ankush Pratap Singh

Advisor: Prof. Yao Wang, Ph.D.

Submitted in Partial Fulfillment of the Requirements for

the Degree of Master of Science (Computer Engineering)

May 2023

The detection of brain metastases in patients with metastatic cancer is crucial for treatment planning and prognosis during radiation therapy. As the disease progresses, brain metastases frequently develop, emphasizing the need for early and accurate detection. This work presents a comprehensive analysis of the NYUMets dataset, which focuses on the dynamics of cancer, specifically metastatic cancer progression over time.

In this study, we evaluated the performance of segmentation-through-time architectures in comparison to a baseline UNet architecture. Our analysis utilized modified architectures with LSTM or transformer encodings (both spatial and temporal), which demonstrated a superior capacity for capturing segmentation as compared to the regular UNet architecture.

The results of our study indicate a notable improvement in segmentation accuracy, with a 25% increase in the Dice Score. These findings suggest the potential utility of our modified architectures for improved segmentation in brain metastases detection, ultimately contributing to more effective treatment planning and better patient outcomes.

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1 Background

Metastatic cancer is a term used to describe cancer that has spread from its original location to other parts of the body. It is also commonly referred to as stage IV cancer and is considered to be a more advanced stage of cancer. Metastasis is the process by which cancer cells break away from the primary tumor and travel through the blood or lymphatic system to other parts of the body, where they can form new tumors.

When cancer cells metastasize to a new location, they retain many of the same features as the primary tumor, which can help doctors identify the type of cancer and the best course of treatment. This is done by examining the cells under a microscope and conducting other tests to determine their characteristics.

One of the common locations where cancer can metastasize is the brain. While any type of cancer can spread to the brain, some types are more likely to do so, including lung, breast, colon, kidney, and melanoma. When cancer cells metastasize to the brain, they can form one or more tumors that can cause a range of symptoms, including headache, memory loss, personality changes, seizures, and more.

Treatment for metastatic brain tumors may involve a combination of surgery, radiation therapy, chemotherapy, immunotherapy, and other treatments. However, the primary focus of treatment is often on managing symptoms and reducing pain and discomfort for the patient.

It is estimated that approximately one-third of patients with another type of cancer will develop one or more metastatic brain tumors. The risk of developing brain metastases typically increases with age, with those over the age of 65 being at the highest risk. It is important for cancer patients and their doctors to be aware of the potential for metastasis and to monitor for any signs or symptoms that may indicate its occurrence.

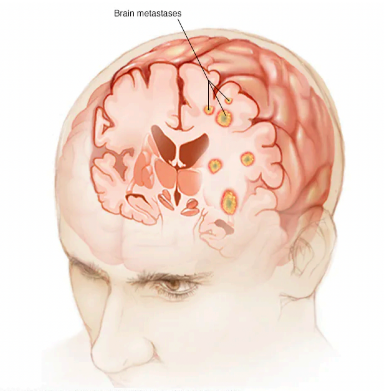


Figure 1.1: Brain metastases (Source: https://www.mayoclinic.org/diseases-conditions/brain-metastases/symptoms-causes/syc-20350136)

1.2 Detecting Brain Metastases using Deep Learning

Deep learning-based algorithms have been gaining attention in the medical field for their potential to improve diagnostic accuracy and treatment outcomes. One area where these algorithms have shown promise is in the automatic detection and segmentation of brain metastases (BMs) in magnetic resonance imaging (MRI) scans. While traditional methods of segmenting BMs rely on manual input from radiologists, deep learning-based algorithms offer the ability to automate this process, potentially reducing human error and increasing efficiency.

Studies have shown that these algorithms can achieve high levels of accuracy in detecting and segmenting BMs. In fact, in some cases, the performance of the algorithm has been shown to be comparable to or even better than that of human experts. Parameters such as the Dice score, which measures the spatial overlap between two segmentation sets of the same region of interest, have been used to quantify the performance of these algorithms compared to expert segmentation.

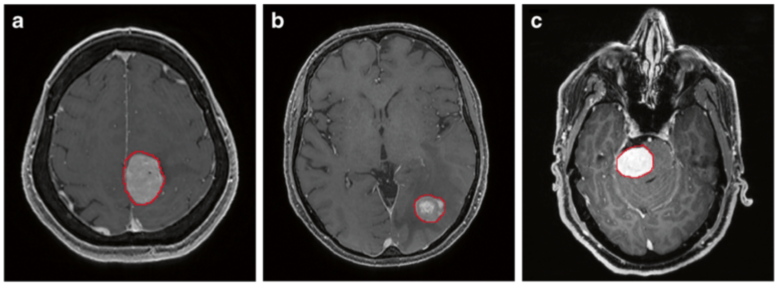


Figure 1.2: Examples of deep learning-based segmentation of brain tumors in the local dataset: (a) meningioma, (b) metastasis, and (c) vestibular schwannoma

1.3 Related Works

Medical image segmentation plays a crucial role in the diagnosis and treatment of brain tumors. With the advent of advanced imaging techniques, such as multiparametric magnetic resonance imaging (mpMRI), the need for accurate and reliable automated image segmentation has become increasingly important. The BraTS challenge has been a major milestone in advancing the field of brain tumor segmentation, providing a standardized dataset for researchers and clinicians to evaluate and compare different algorithms for glioma segmentation.

The dataset used in the BraTS challenge includes a variety of sequences, such as T1-pre and post-gadolinium contrast (T1C), T2-weighted, and T2-FLAIR sequences, which are commonly used in clinical practice. The use of such standardized datasets has allowed researchers to develop and test a range of advanced deep learning algorithms that can automatically detect and segment brain tumors with high accuracy. These algorithms have the potential to significantly improve patient outcomes by enabling earlier detection and more precise treatment planning.

In addition to BraTS, the BrainMetShare dataset has also been extensively studied for the segmentation of brain metastases. BrainMetShare includes 156 whole brain MRIs, each with high-resolution, multi-modal pre- and post-contrast sequences obtained from patients with at least one brain metastasis. These images are segmented by radiologists using ground truth, which provides a benchmark for evaluating the performance of automated segmentation algorithms.

The development of these datasets and algorithms is an exciting area of research with the potential to revolutionize the diagnosis and treatment of brain tumors. By improving the accuracy and speed of image segmentation, clinicians can make more informed treatment decisions, leading to better outcomes for patients.

1.4 Training and Evaluation Methodology

UNet[1] architecture was chosen as a baseline for training due to its straightforward yet effective design. UNet consists of an encoder-decoder network with skip connections, which preserve spatial information and minimize resolution loss during down-sampling and up-sampling. This architecture was originally designed for biomedical image segmentation, and thus considers the unique characteristics of medical images, such as low contrast, noise, and anatomical structure variability.

Moreover, UNet is computationally efficient and has demonstrated impressive performance on a variety of medical image segmentation tasks, including those involving the brain, liver, lungs, and other organs. Overall, UNet's simplicity, effectiveness, and adaptability to medical image segmentation make it a strong candidate for our training approach, all other architectures were modifications of UNet architecture.

In all the experimental architectures that were utilized, encompassing UNet and its various adaptations, the filter size employed for the convolutional layers was set to 3x3x3. Furthermore, the number of input channels was set to 1, indicating a single-channel input, while the number of output channels was set to 2, representing a two-channel output. These configuration choices were made consistently across the different architectural variations explored during the experiments.

In the context of medical image segmentation, evaluating the performance of an algorithm is crucial for its clinical applicability. Therefore, a variety of metrics are employed to assess the quality of the segmentation results obtained from different networks. One of the most used metrics is the Dice score, which provides a quantitative measure of the agreement between a predicted segmentation and its corresponding ground truth.

The Dice coefficient ranges from 0 to 1, with 1 indicating a perfect overlap between the predicted segmentation and the ground truth. It is computed as 2 times the area of overlap between the predicted and ground truth masks divided by the total number of pixels in both images. The Dice score is particularly useful in evaluating the segmentation of structures with irregular shapes and sizes, such as tumors.

In addition to the Dice score, other metrics such as Tumor Volume, Tumor Count, and Small Tumor Count are also evaluated. Tumor Volume refers to the volume of the tumor segmented by the algorithm, while Tumor Count and Small Tumor Count provide information on the number of tumors detected and the number of small tumors detected, respectively with respect to ground truth. These metrics typically do not provide a comprehensive understanding of the comparative superiority of one algorithm over another.

1.5 Outline

The thesis is structured into several chapters to provide a comprehensive analysis of the proposed models. Chapter 2 presents the NYUMets dataset, which is used for all analysis in this thesis. In chapter 3, the UNet architecture is described in detail, including its training and validation results. Chapters 4, 5, 6 introduce three different models that incorporate temporal dependencies of MRI scans. Chapter 7 compares the performances of all proposed architectures on various metrics on validation and test data. Chapter 8 investigates whether adding recurrence at multiple layers improves or decreases the performance through multi-layer recurrence. Finally, in chapter 9, the findings of this thesis are summarized, and future research directions are discussed.

2. NYUMets Dataset

2.1 Background

NYUMets[5] is a remarkable dataset that is assembled from one of the largest clinical registries of patients with metastatic brain cancer in the world. This dataset is now publicly available and will be a valuable resource for researchers and clinicians who are interested in understanding the complex dynamics of metastatic cancer. One of the most intriguing aspects of this dataset is that it provides an opportunity for scientists and healthcare professionals to focus on the changes in metastatic cancer over time, as the disease progresses, which can give insights into the mechanisms that drive the progression of this complex and challenging condition.

The dataset includes information on 1,429 patients who were analyzed over an average period of 17 months, with an average of six imaging studies per patient. In total, the dataset includes 8,003 MRI studies, which encompass a wide range of sequences including segmentation by experts, T1 pre-contrast, T1 post-contrast, high-resolution T1 post-contrast, T2, and FLAIR. The dataset also includes 4,860 clinical follow-up timepoints and 81,562 medication updates.

One of the most notable features of the NYUMets dataset is the extensive imaging data that has been collected. The dataset includes high-resolution MRI studies that are segmented by experts and ML algorithms, which means that the data is highly accurate and provides detailed information on the location and extent of the metastatic tumors.

This information will be invaluable for researchers who are interested in developing new techniques for image segmentation and analysis.

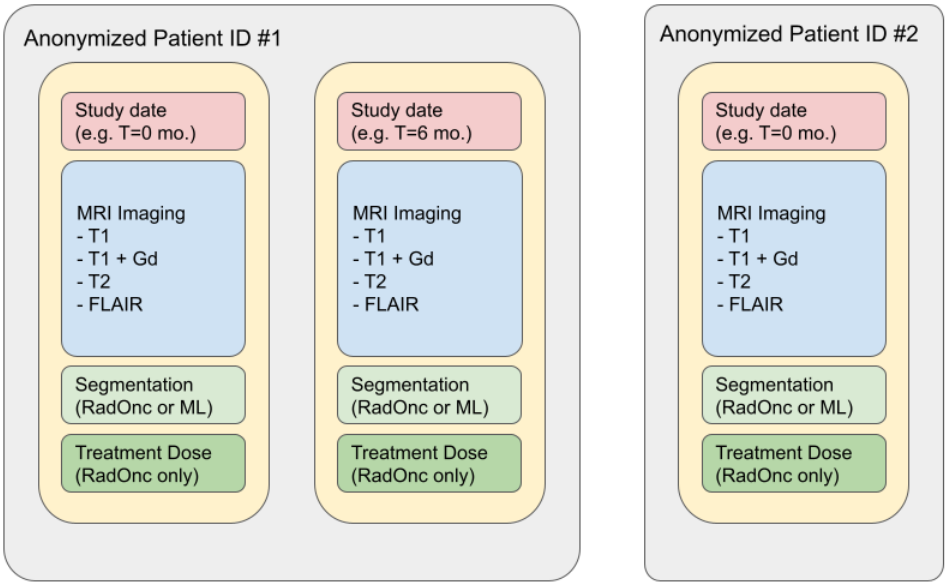


Figure 2.1: NYUMets dataset overview

The dataset has been compiled with a strong focus on patient privacy and confidentiality. There is no disclosure of any patient information at any point in the process, and anonymity is ensured for every patient. This commitment to patient privacy and confidentiality is crucial, as it enables researchers to access the data they need while protecting the rights and interests of the patients who have contributed to the dataset. Overall, the NYUMets dataset represents a major step forward in our understanding of metastatic brain cancer. The extensive imaging data and clinical information included in the dataset will provide researchers and clinicians with a wealth of information that can be used to develop new treatment strategies, improve patient outcomes, and ultimately find a cure for this devastating disease.

2.2 Gamma Knife Surgery

Gamma Knife radiosurgery is a non-invasive and highly precise form of radiation therapy that delivers targeted high-dose radiation to a specific area of the brain. The Gamma Knife machine uses a large number of small radiation beams that intersect at a specific point in the brain, allowing for a concentrated and powerful dose of radiation to be delivered to a specific target while minimizing radiation exposure to the surrounding healthy tissue. This makes it an ideal treatment option for complex and difficult-to-reach brain tumors, vascular malformations, and other brain disorders that may not be suitable for traditional surgery or whole-brain radiation therapy.

Gamma Knife radiosurgery offers many benefits over traditional surgery and whole-brain radiation therapy. Since there is no incision, patients typically experience less pain, scarring, and recovery time. The precise targeting of the radiation also minimizes damage to healthy brain tissue, reducing the risk of side effects such as cognitive impairment and neurological deficits. In addition, Gamma Knife treatment can be repeated if necessary, allowing for greater flexibility in managing complex and recurrent brain disorders.

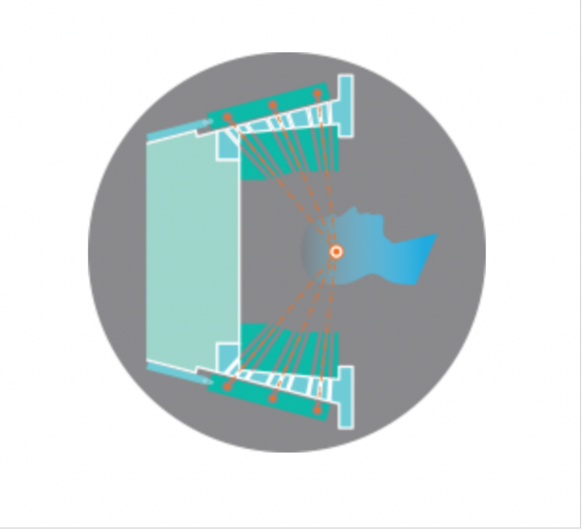


Figure 2.2: Gamma Knife Surgery

2.3 Data Gathering

In the context of medical treatment for brain disorders, a patient may undergo a series of MRI scans and gamma knife surgeries over a period of time, denoted as t=0, t=1, t=2, and so on until t=n. At each time point, the patient goes in for an MRI scan to monitor the progression of the disease and to guide the treatment plan. Following the MRI scan, the patient may receive gamma knife surgery as an alternative to traditional brain surgery or whole-brain radiation therapy, depending on the nature of the brain disorder being treated.

This process may repeat over several time points as the patient continues to receive medical treatment for their condition. The series of MRI scans and gamma knife surgeries can provide valuable information about the disease progression and the effectiveness of the treatment plan, allowing medical professionals to adjust the treatment approach as necessary to optimize outcomes for the patient.

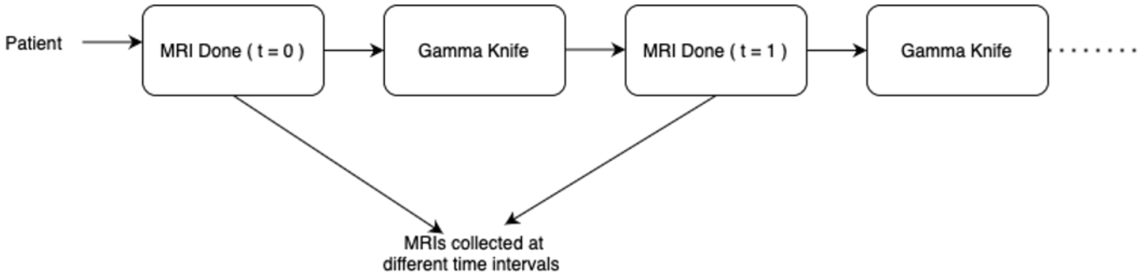


Figure 2.3: Data Gathering

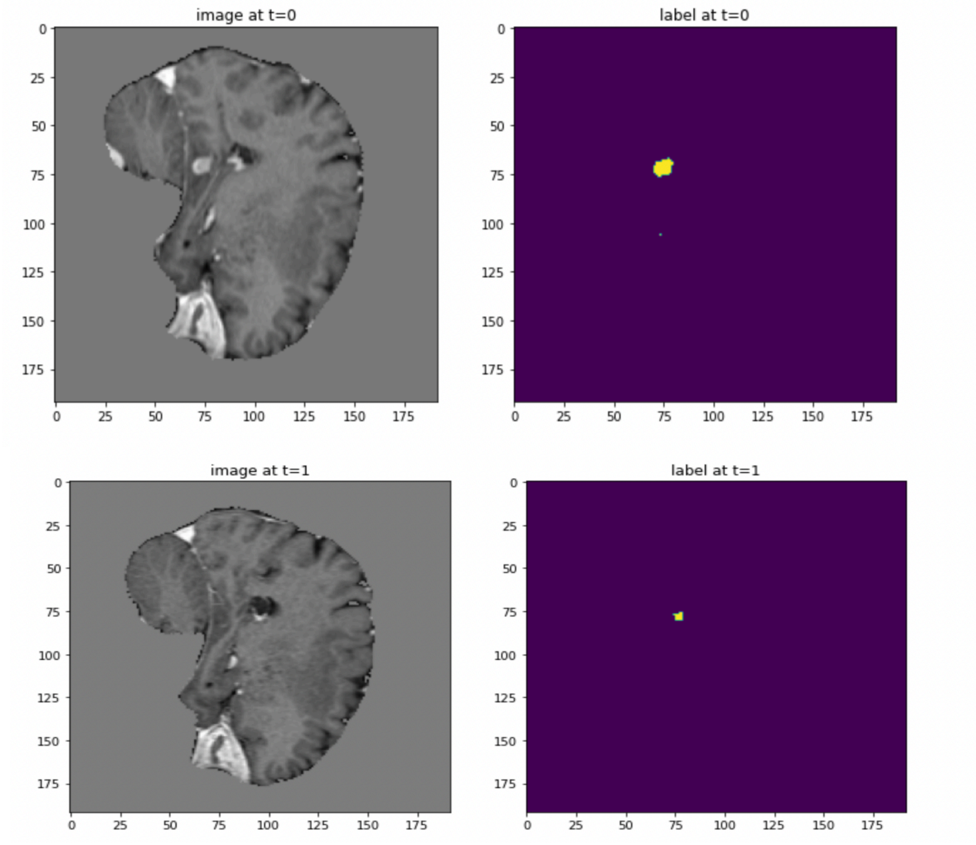


Figure 2.4: Sample Images and Labels

3. UNet Architecture

3.1 Architecture Background

UNet is a deep convolutional neural network (CNN) architecture for image segmentation, particularly in medical image analysis. It was proposed in 2015 by Olaf Ronneberger, Philipp Fischer, and Thomas Brox.

UNet consists of an encoder and a decoder. The encoder is a series of convolutional and pooling layers that downsample the input image to extract features. The decoder then upsamples the output of the encoder to produce a segmentation map with the same size as the input image. The upsampling is performed using transposed convolutional layers that gradually increase the spatial resolution of the feature maps.

The UNet architecture also includes skip connections between the encoder and decoder, which allow the decoder to use information from earlier stages of the encoder to improve the segmentation results. The skip connections concatenate the feature maps from the encoder with the feature maps from the corresponding decoder layers, allowing the decoder to use both high-level and low-level features.

UNet has achieved state-of-the-art results on various medical image segmentation tasks, including brain tumor segmentation, cell segmentation, and retinal vessel segmentation.

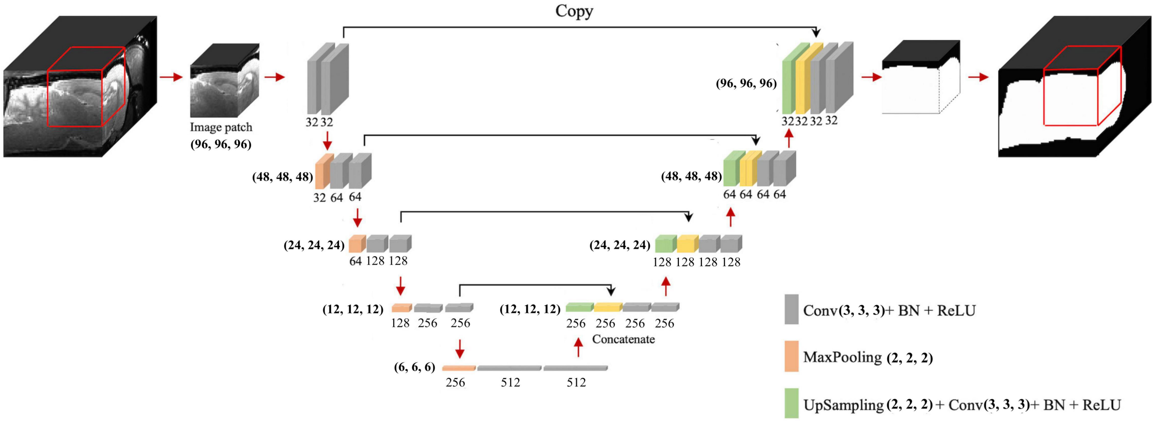


Figure 3.1: UNet Architecture (for 3d Scans)

3.2 Results

After the UNet architecture was trained for 3D MRI scans, a thorough analysis was conducted to evaluate the performance of the model with varying learning rates. The obtained results were carefully examined and compared to identify the best possible learning rate for the UNet architecture. This process involved analyzing the Validation Dice Score of the model.

Throughout the evaluation, it was found that the performance of the UNet architecture varied significantly depending on the learning rate used during training. Specifically, certain learning rates resulted in higher dice score than others.

By analyzing and comparing the results obtained from each learning rate, I was able to identify the most appropriate learning rate for the UNet architecture in the context of 3D MRI scans from NYUMets dataset

Best Validation Dice Score			
Learning Rate			
Combination	1e-3	1e-4	1e-5
(16, 32, 64)	0.3340	0.3401	0.2016
(16, 32, 64, 128)	0.3326	0.2996	0.1987
(64, 128, 256)	0.3610	**0.3761**	0.3170
(64, 128, 256, 512)	0.3487	0.3553	0.2165

Table 3.1: UNet Architecture Results (for 3d Scans)

Upon analyzing the results obtained after training the UNet architecture for 3D MRI scans using different learning rates, it can be observed that for our particular set up the learning rate of 1e-4 tends to consistently outperform the other two learning rates for all the combinations tested. This observation is significant as it suggests that the choice of learning rate plays a crucial role in the performance of the UNet architecture for 3D MRI scans.

The finding highlights the importance of hyperparameter tuning, particularly the learning rate, for achieving optimal performance in deep learning models. It is also important to note that while a particular learning rate may perform well for a specific dataset and setup, it may not generalize well to other datasets. Therefore, it is essential to tune hyperparameters for each new dataset or problem.

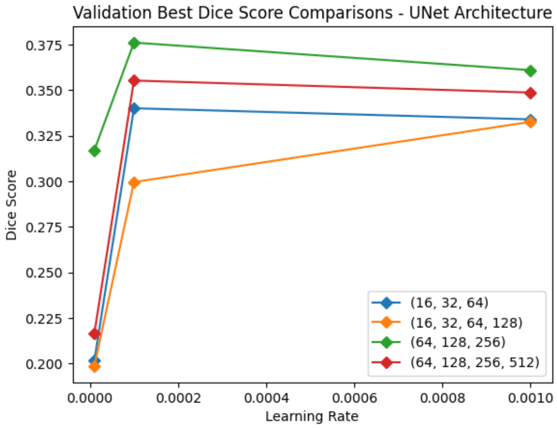


Figure 3.2: UNet Architecture Performance across Learning Rates

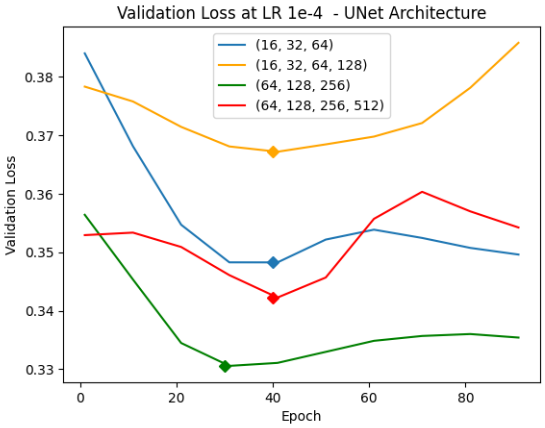


Figure 3.3: UNet Validation Loss at LR 1e-4

Based on the experimentation results for our particular set up, it has been seen that the most effective configuration for the UNet model involves the use of a 3-layered network consisting of 64, 128, and 256 features, with a learning rate of 1e-4. It has been found that when using 4-layered networks, overfitting tends to occur, which leads to a reduction in performance. Therefore, the 3-layered network is the optimal configuration for our experimentation setup.

In addition to this, the validation loss curve has been analyzed to further evaluate the performance of different UNet combinations. It has been observed that the combination of 64, 128, and 256 features has the least validation loss, which indicates that it is the most effective configuration for the UNet model based on Dice Score for our experimentation setup.

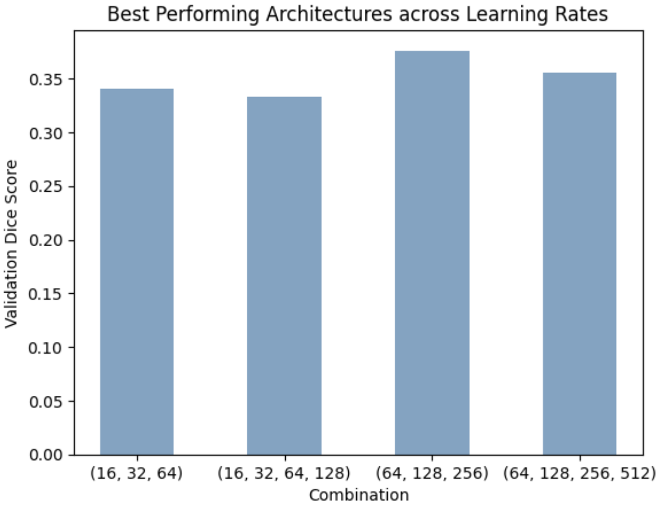


Figure 3.4: UNet Architecture Performance on Validation Dice Score

In conclusion, based on both experimentation and analysis of the validation loss curve, it can be observed that the most optimal UNet configuration involves the use of a 3-layered network consisting of 64, 128, and 256 features, with a learning rate of 1e-4 for our experimentation setup (in terms of training, validation data set, and choice of the number of channels per layer and filter size). This configuration provides the best performance and minimizes the risk of overfitting, leading to a more accurate and reliable model.

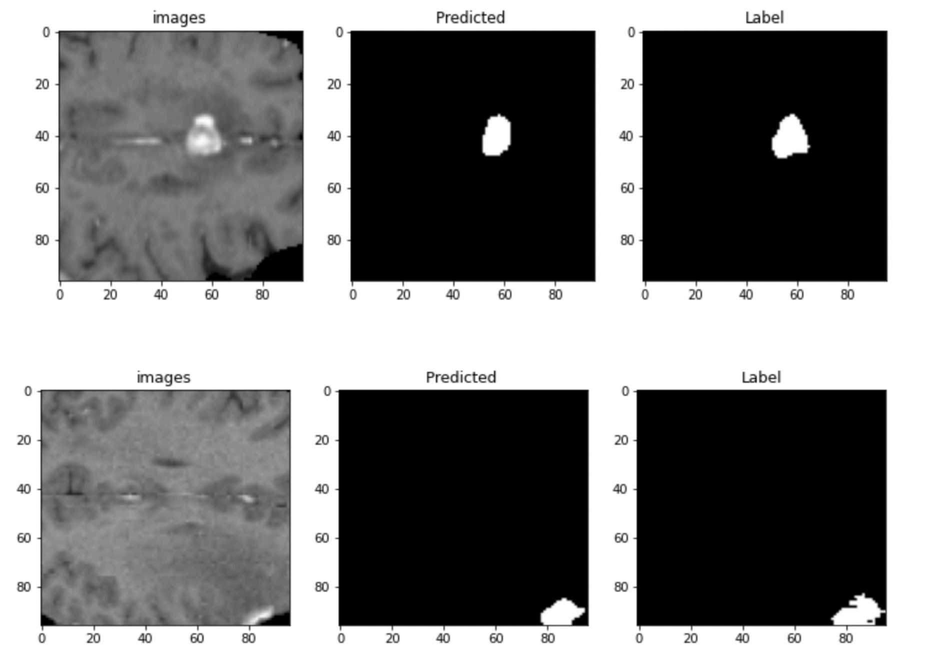


Figure 3.5: UNet Architecture Performance Visualization

4. LSTM Based Segmentation Through Time (Stt) - UNet Architecture

4.1 Architecture Background

The UNet architecture has been a popular choice for segmentation tasks in medical imaging due to its ability to accurately segment objects of interest. However, the UNet architecture has limitations when it comes to handling temporal dependencies in a sequence of images. This is because the architecture does not take into account the sequentiality of the input images, which can result in suboptimal performance in segmentation tasks where temporal dependencies are important.

To address this limitation, a recurrent network can be incorporated into the architecture to capture temporal dependencies. One such architecture that has been proposed is an LSTM-based stt - UNet architecture. In this architecture, an LSTM is added to the bottleneck layer to capture the temporal dependencies of the input image sequence. By doing so, the LSTM-based stt - UNet architecture is better able to understand the sequentiality of the input images, leading to improved performance in segmentation tasks that require consideration of temporal dependencies.

The addition of an LSTM layer in the bottleneck of the UNet architecture allows the model to learn how to track the progression of objects or regions of interest in a sequence of images, which can be particularly useful in applications such as medical imaging where time-series data is common. By incorporating a recurrent network, the model can effectively capture the temporal dependencies between consecutive images, allowing for more accurate segmentation of objects or regions of interest.

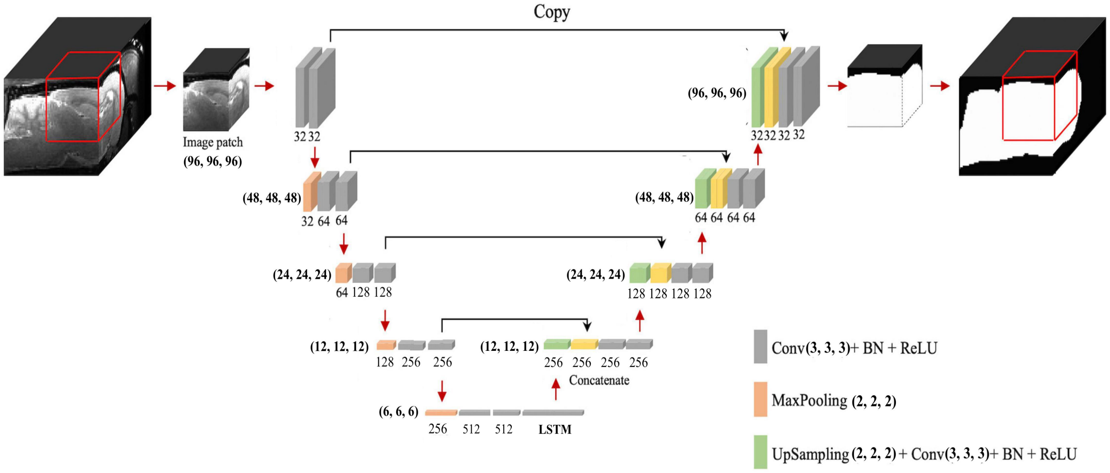


Figure 4.1: LSTM Based Stt - UNet Architecture

The use of an LSTM-based stt - UNet architecture is a promising approach for segmentation tasks that require consideration of temporal dependencies. The incorporation of a recurrent network allows the architecture to capture the sequentiality of input images, resulting in improved performance and accuracy in segmentation tasks. As such, this architecture may be particularly useful in medical imaging applications where time-series data is common.

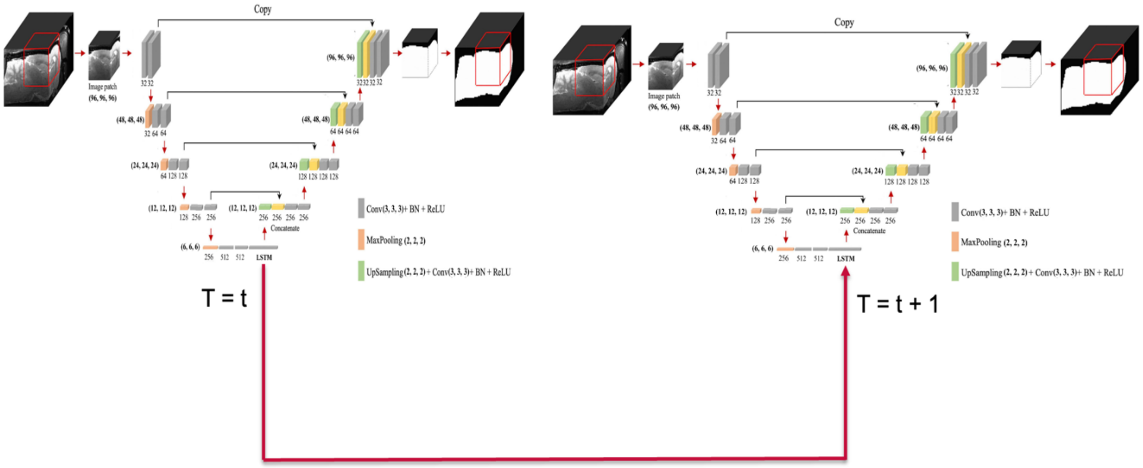


Figure 4.2: Detailed Description of LSTM Based Stt - UNet Architecture

4.2 Results

After conducting a range of trials to evaluate the performance of the LSTM-based stt-UNet architecture, the optimal combination for achieving the best segmentation results is 64, 128, 256, and 512 features for our experimentational setup. This combination was found to be the best performing among all the configurations tested, in terms of validation Dice Score for our setup.

Best Validation Dice Score			
Learning Rate			
Combination	1e-3	1e-4	1e-5
(16, 32, 64, 128)	0.3206	0.4053	0.3321
(32, 64, 128, 256)	0.3013	0.4296	0.3563
(64, 128, 256, 512)	0.2722	**0.4363**	0.3470
(128, 256, 512, 1024)	0.2964	0.4284	0.3918

Table 4.1: LSTM based stt - UNet Architecture Results (for 3d Scans)

Furthermore, the validation loss curve of the LSTM-based stt-UNet architecture clearly showed that the combination of 64, 128, 256 and 512 features had the least validation loss for our setup. This is a strong indication that the architecture has the ability to capture temporal dependencies in the data and produce accurate segmentations.

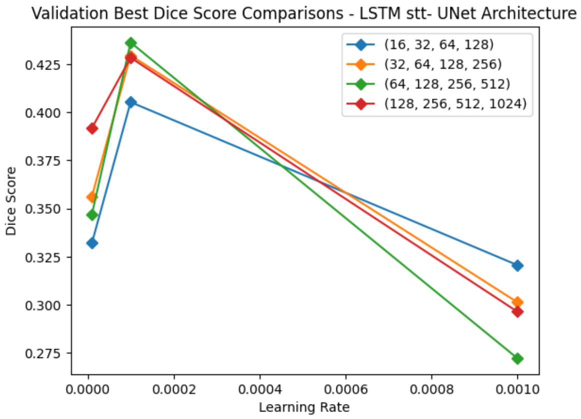


Figure 4.3: LSTM based UNet Architecture Performance

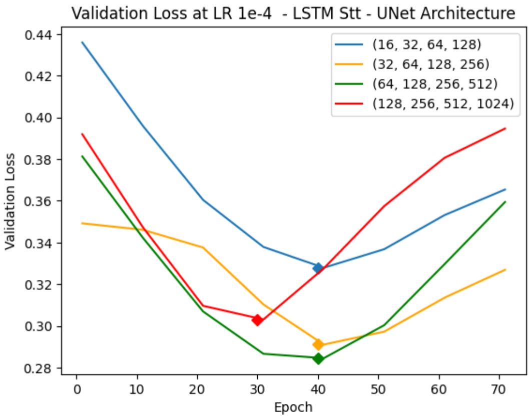


Figure 4.4: Validation Loss at LR 1e-4

It is worth noting that using a learning rate of 1e-4 was optimal for achieving the best segmentation results for our experimentational setup. This learning rate allowed the model to converge quickly and produce accurate segmentations, without overfitting the data. Overall, these findings suggest that the LSTM-based stt-UNet architecture with the 64, 128, 256 and 512 features configuration and a learning rate of 1e-4 is an effective approach for performing segmentation tasks that require consideration of temporal dependencies in our setup.

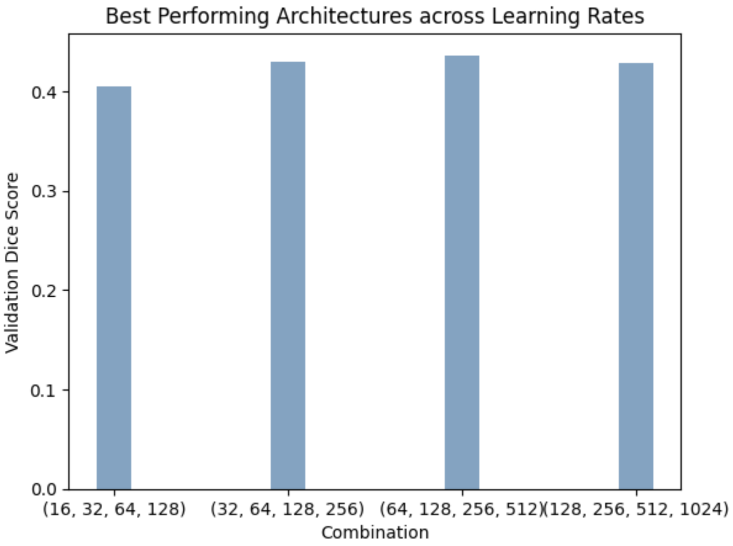


Figure 4.5: LSTM Based stt - UNet Architecture Performance on Validation Dice Score

In conclusion, based on both experimentation and analysis of the validation loss curve, it can be observed that the most optimal LSTM based stt - UNet configuration involves a network consisting of 64, 128, 256 and 512 features, with a learning rate of 1e-4 for our experimentation setup (in terms of training, validation data set, and choice of the number of channels per layer and filter size). This configuration provides the best performance and minimizes the risk of overfitting, leading to a more accurate and reliable model.

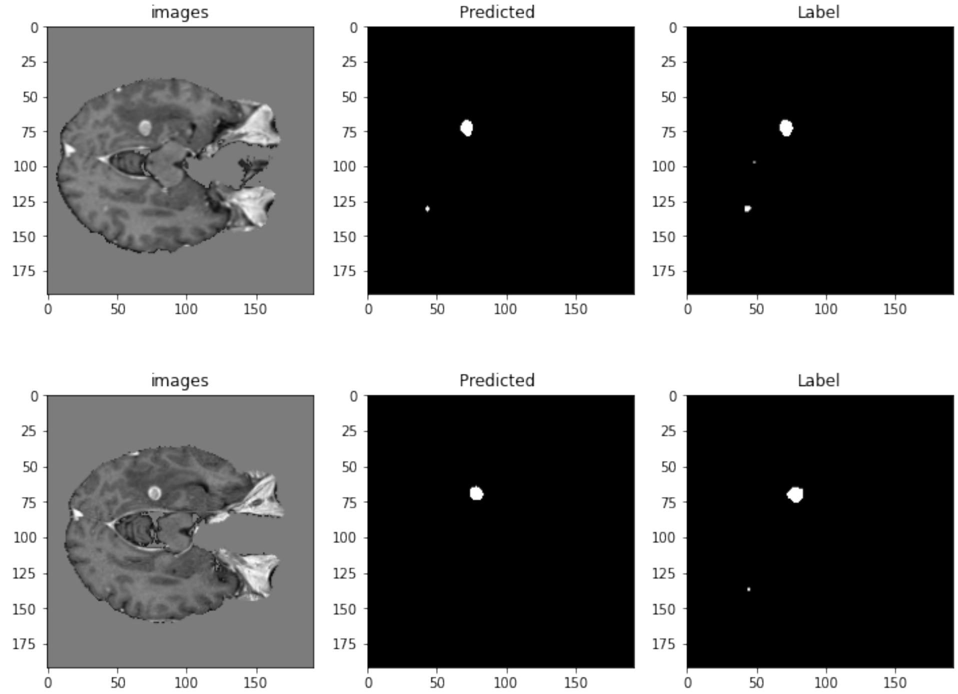


Figure 4.6: LSTM based stt - UNet Architecture Performance Visualization

5. Enhancing UNet Architecture with a Spatial Transformer

5.1 Architecture Background

Another promising modification involves the use of transformer encodings in the bottleneck layer.

Transformers are a type of neural network architecture that have shown great promise in various natural language processing tasks and have recently been adapted for use in image processing tasks as well. In this implementation, the transformer used in the UNet architecture is a spatial transformer, which operates between each voxel of the image.

The use of transformer encodings in the UNet architecture is intended to capture long-range dependencies in the data, which may be difficult to capture using the standard convolutional layers in the UNet architecture. By incorporating transformer encodings into the bottleneck layer of the UNet architecture, it may be possible to achieve more accurate and robust segmentation results.

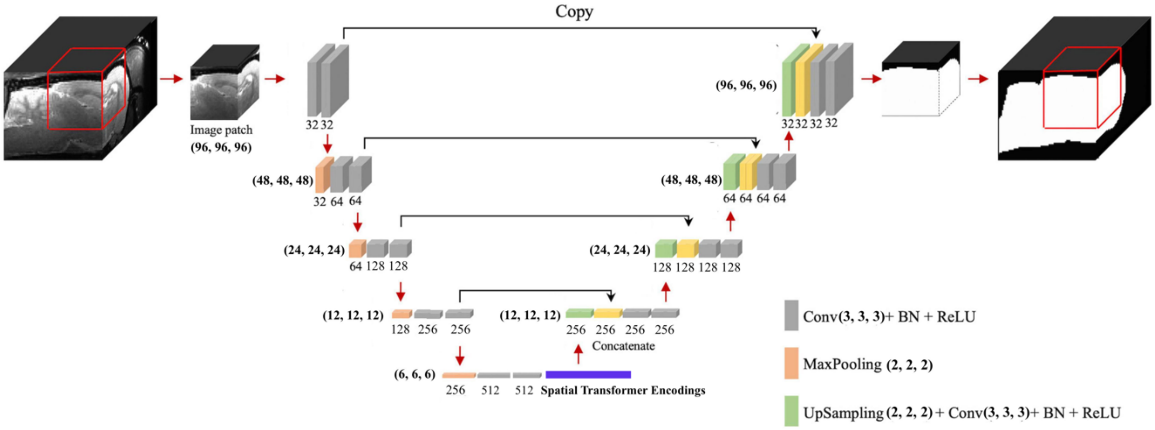


Figure 5.1: Spatial Transformer incorporated UNet Architecture

In the spatial transformer, each voxel in the input volume at a time represents a token. Self-attention between tokens occurs in the bottleneck layer, where a Transformer Encoder is applied to capture spatial relationships and dependencies among the tokens.

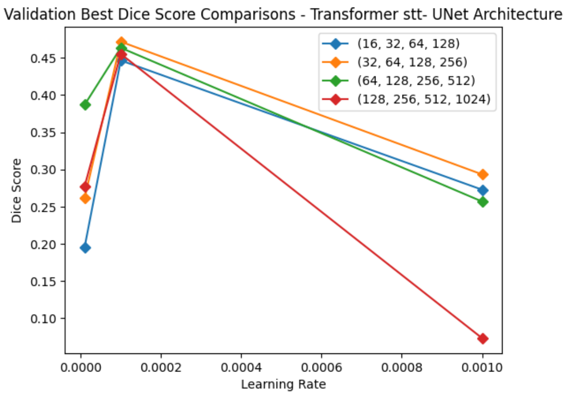
5.2 Results

After conducting a range of trials to evaluate the performance of the Spatial Transformer-incorporated UNet architecture, the optimal combination for achieving the best segmentation results is 32, 64, 128, and 256 features on our experimentation setup. This combination was found to be the best performing among all the configurations tested, in terms of validation Dice Score.

Best Validation Dice Score			
Learning Rate			
Combination	1e-3	1e-4	1e-5
(16, 32, 64, 128)	0.2727	0.4466	0.1955
(32, 64, 128, 256)	0.2933	**0.4718**	0.2611
(64, 128, 256, 512)	0.2568	0.4636	0.3869
(128, 256, 512, 1024)	0.073	0.4553	0.2774

Table 5.1: Spatial Transformer incorporated UNet Architecture Results (for 3d Scans)

Furthermore, the validation loss curve demonstrated that this combination also had the least validation loss, indicating that the model was not overfitting the training data and was able to generalize well to new data on our setup.



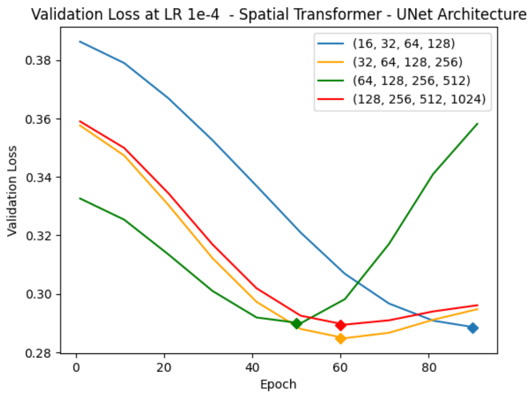


Figure 5.2: Spatial Transformer incorporated UNet Architecture Performance

Figure 5.3: Validation Loss at LR 1e-4

It is worth noting that using a learning rate of 1e-4 was optimal for achieving the best segmentation results on our experimentational setup. This learning rate allowed the model to converge quickly and produce accurate segmentations, without overfitting the data. Overall, these findings suggest that the Spatial Transformer incorporated UNet architecture with the 32, 64, 128, and 256 features configuration and a learning rate of 1e-4 on our setup (in terms of your training, validation data set, and your choice of the number of channels per layer and filter size) is an effective approach for performing segmentation tasks as per our setup.

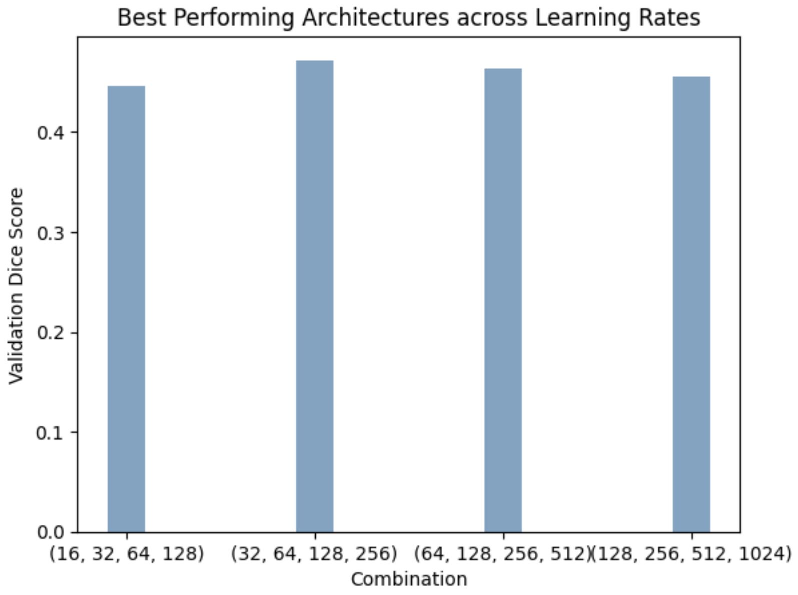


Figure 5.4: Spatial Transformer incorporated UNet Architecture Performance on Validation Dice Score

This suggests that incorporating a spatial transformer into the UNet architecture can improve its ability to capture complex spatial relationships in the image, leading to better segmentation performance.

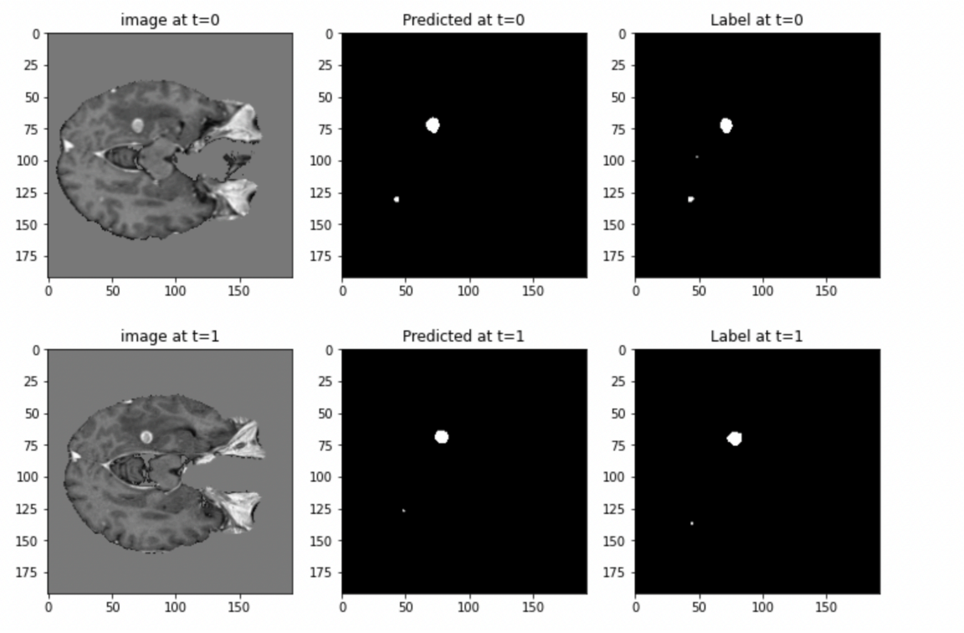


Figure 5.5: Spatial Transformer incorporated UNet Architecture Performance Visualization

6. Temporal Transformer Based Segmentation Through Time (Stt) - UNet Architecture

6.1 Architecture Background

Introducing a temporal transformer is an alternative method to implement transformer encoding that takes into account the temporal aspect of the data. This method involves applying the transformer architecture to a sequence of data points over time.

Compared to the traditional transformer architecture that processes inputs in a parallel manner, the temporal transformer encodes temporal dependencies between the input sequence, allowing it to capture dynamic changes in the data over time.

Each token represents a voxel at a specific position and time step, including the input channel features associated with it. The attention mechanism considers both the current voxel's features within the current time step and attends to the features of other time steps, enabling the model to capture both temporal dependencies and contextual representations within each time step.

The inclusion of temporal information has been shown to improve performance and accuracy compared to traditional encoding methods that do not consider temporal dependencies.

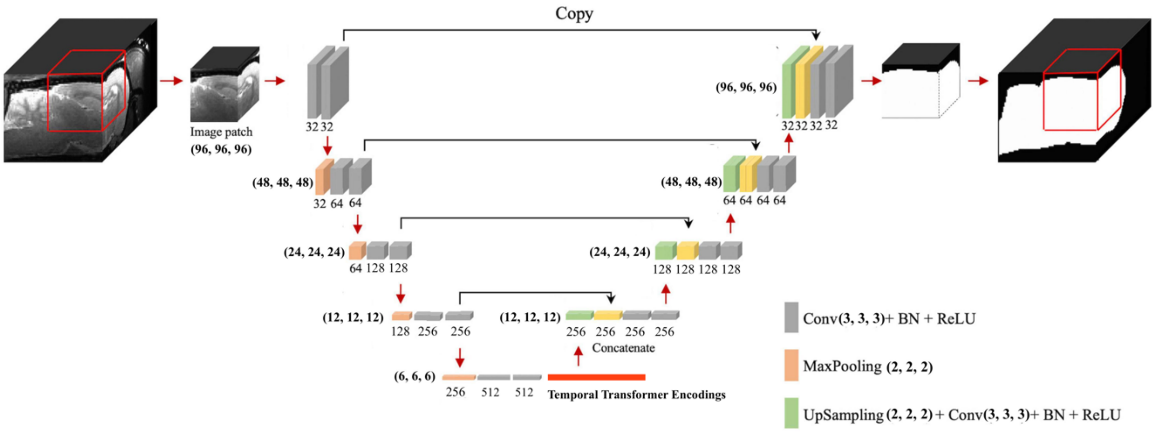


Figure 6.1: Temporal Transformer Based Stt - UNet Architecture

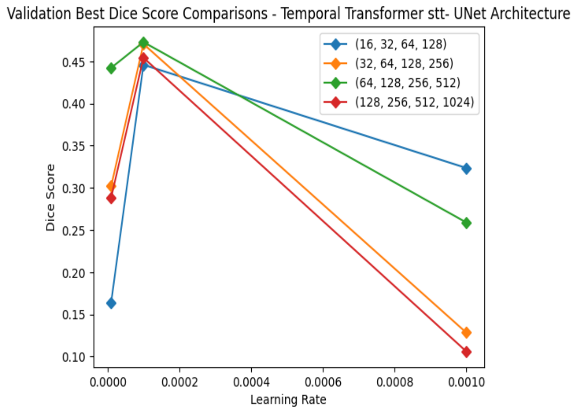
6.2 Results

After conducting a range of trials to evaluate the performance of the Temporal Transformer-based stt-UNet architecture, the optimal combination for achieving the best segmentation results is 64, 128, 256 and 512 features on our experimentational setup. This combination was found to be the best performing among all the configurations tested, in terms of validation Dice Score.

Best Validation Dice Score			
Learning Rate			
Combination	1e-3	1e-4	1e-5
(16, 32, 64, 128)	0.3235	0.4459	0.1641
(32, 64, 128, 256)	0.1283	0.4705	0.3020
(64, 128, 256, 512)	0.2589	**0.4728**	0.4422
(128, 256, 512, 1024)	0.106	0.4542	0.2884

Table 6.1: Temporal Transformer based stt - UNet Architecture Results (for 3d Scans)

Furthermore, the validation loss curve demonstrated that this combination also had the least validation loss, indicating that the model was not overfitting the training data and was able to generalize well to new data.



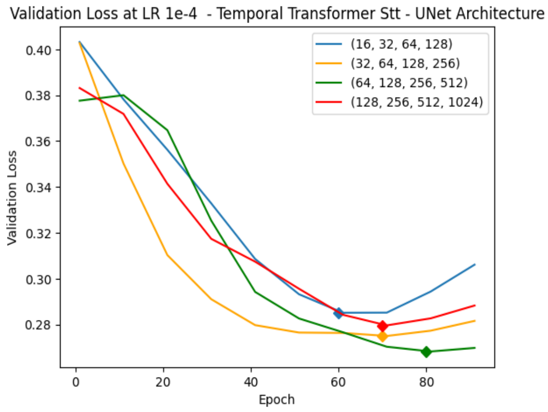


Figure 6.2: Temporal Transformer based UNet Architecture Performance

Figure 6.3: Validation Loss at LR 1e-4

It is worth noting that using a learning rate of 1e-4 was optimal for achieving the best segmentation results on our setup. This learning rate allowed the model to converge quickly and produce accurate segmentations, without overfitting the data. Overall, these findings suggest that the Temporal Transformer-based stt-UNet architecture with the 64, 128, 256 and 512 features configuration and a learning rate of 1e-4 on our setup (in terms of your training, validation data set, and your choice of the number of channels per layer and filter size) is an effective approach for performing segmentation tasks that require consideration of temporal dependencies.

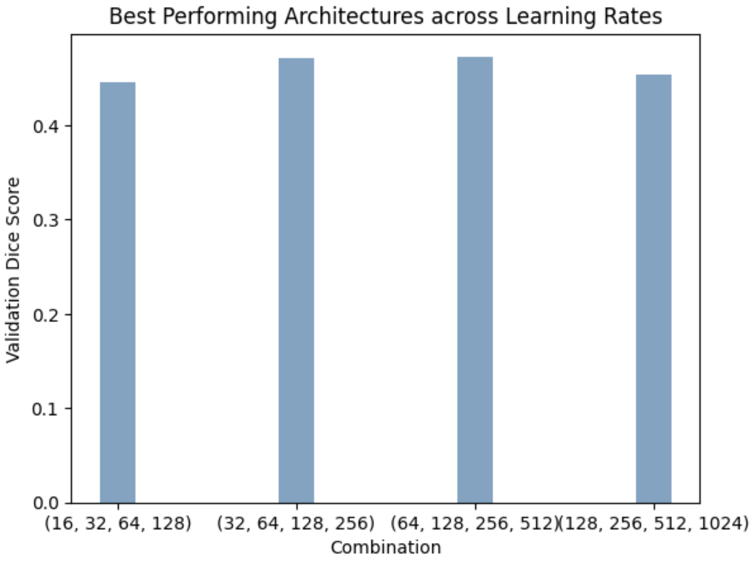


Figure 6.4: Temporal Transformer Based stt - UNet Architecture Performance on Validation Dice Score

This suggests that incorporating a temporal transformer into the UNet architecture can improve its ability to capture complex temporal relationships in the image, leading to better segmentation performance.

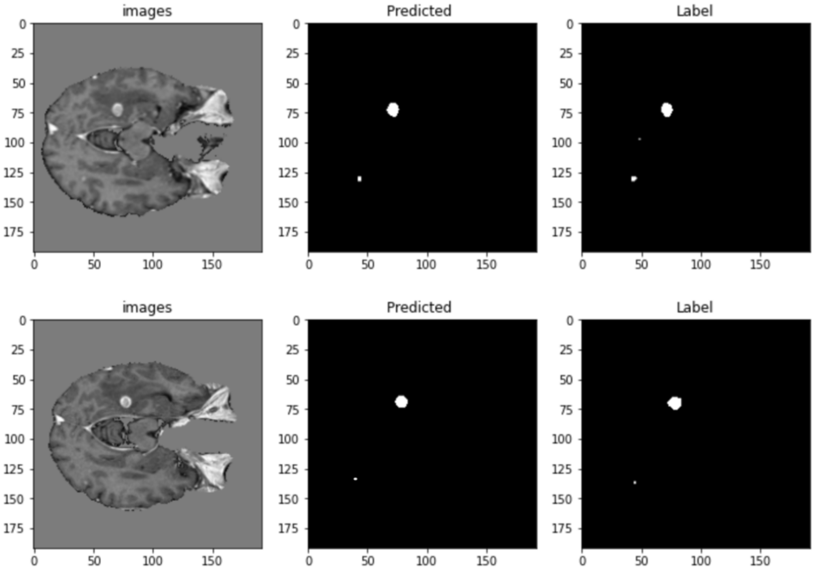


Figure 6.5: Temporal Transformer Based stt - UNet Architecture Performance Visualization

7. Performance Comparisons

7.1 Performance Comparisons on Validation and Test Dice Score

In the study, the performance of four different architectures was compared for their effectiveness in temporal image segmentation: a temporal transformer-based stt-UNet architecture, a spatial transformer incorporated UNet architecture, an LSTM-based stt-UNet architecture, and a regular UNet architecture on our experimentation setup. After training and testing these models, the validation and test dice scores were used as a metric to evaluate their performance.

Combination	Validation Dice Score
UNet	0.3761
LSTM - based Stt - UNet	0.4363
Spatial Transformer incorporated UNet	0.4718
Temporal Transformer based Stt- UNet	**0.4728**

Table 7.1: Performance Evaluation on Validation Dice Score

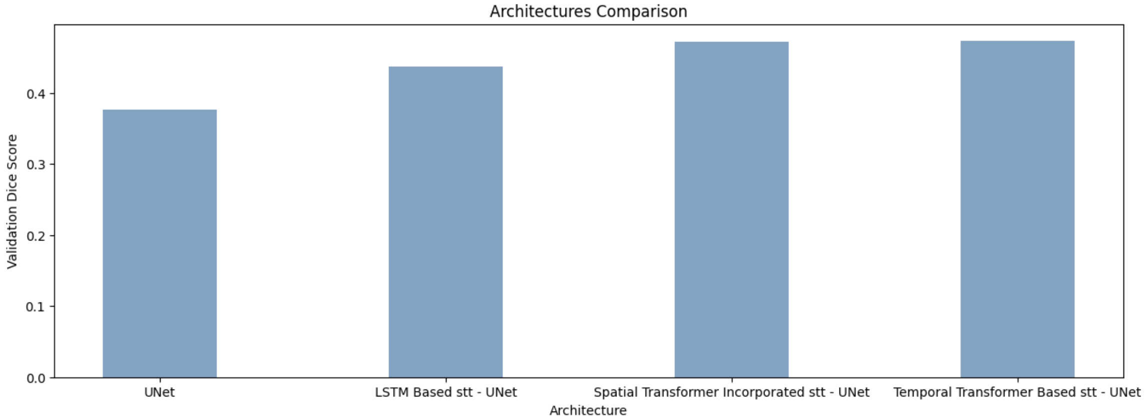


Figure 7.1: Performance Comparison on Validation Dice Score

Combination	Test Dice Score
UNet	0.3655
LSTM - based Stt - UNet	0.4413
Spatial Transformer incorporated UNet	0.4745
Temporal Transformer based Stt - UNet	**0.4781**

Table 7.2: Performance Evaluation on Test Dice Score

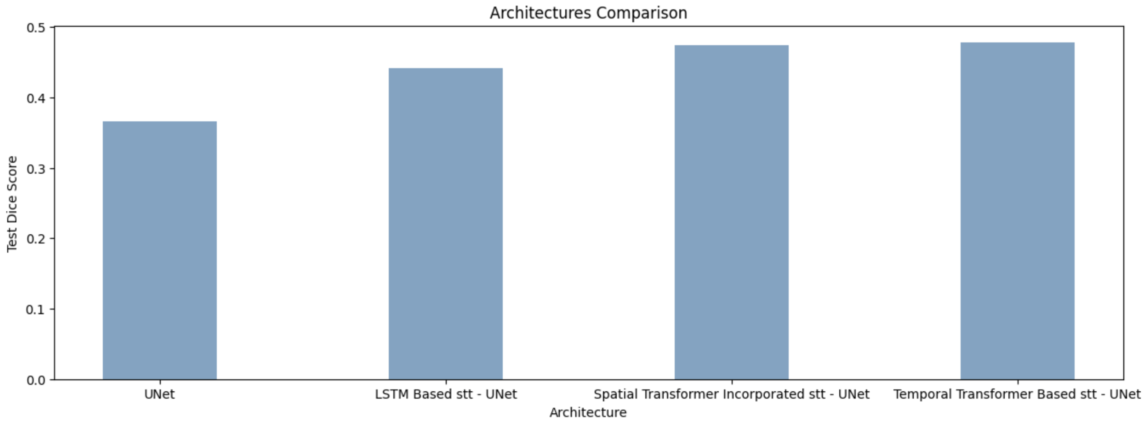


Figure 7.2: Performance Evaluation on Test Dice Score

The results showed that the transformer incorporated UNet architecture outperformed the other architectures in terms of the dice score on our experimentational setup (in terms of your training, validation data set, and your choice of the number of channels per layer and filter size). This indicates that incorporating the transformer encoding at the bottleneck layer improved the model's ability to capture spatial or temporal dependencies and resulted in better segmentation results.

To go in detail, when comparing the performance of incorporating a temporal transformer versus a spatial transformer, the former was found to have a slight advantage in improving overall performance.

On the other hand, the LSTM-based stt-unet architecture had a lower validation dice score than the transformer incorporated UNet architecture, indicating that the use of LSTMs to capture temporal dependencies was not as effective as the transformer encoding.

Lastly, the regular UNet architecture yielded the poorest performance in the task of segmentation through time, further highlighting the need for models that can effectively comprehend sequentiality.

7.2 Evaluating other metrics

In addition to the validation dice score, the performance of different architectures was also evaluated on other metrics. While these metrics serve as valuable indicators, it is important to understand that oftentimes, these metrics offer only a partial view and may not capture the full complexity and nuances of algorithmic performance. A more comprehensive evaluation and comparison is still the Dice Score metric.

Combination	Tumor Vol	Tumor Count Agg	Small Tumor Count Agg	fbeta	Best dice
UNet	2192.97	4.243	1.176	0.3727	0.3761
LSTM based Stt - UNet	2198.277	4.9892	1.08	0.5919	0.4363
Spatial Transformer incorporated UNet	2202.064	5.78	1.03	0.6107	0.4718
Temporal Transformer based Stt - UNet	2246.24	5.914	1.431	0.6472	0.4728

Table 7.3: Performance Evaluation on other metrics

8. Multi-Layer Recurrence Comparisons

8.1 Overview

In the previous research on the UNet architecture, it was found that incorporating a transformer encoding at the bottleneck layer can lead to improved performance on temporal datasets. However, it was also important to investigate whether the performance could be further improved by introducing recurrence at not only the bottleneck layer but also to other layers as well.

The experiment involved adding transformer encodings to each skip connection layer of the UNet architecture and measuring the validation dice score as a metric of performance. The results showed that adding transformer encodings at each layer can affect the performance significantly for the validation dice score, indicating that the model was either better able to or not able to capture complex temporal dependencies in the data.

It is worth noting that this change in performance was subject to computational limitations, and it may not be feasible to add recurrence at each layer for all datasets due to constraints on computational resources. Nonetheless, these results highlight the importance of considering the incorporation of recurrence in the architecture design when working with temporal datasets.

Overall, this research suggests that incorporating certain transformer encodings at each layer in the UNet architecture can lead to improved performance on temporal datasets, and further investigation in this area may yield even better results in the future.

8.2 Multi-Layer Encodings with spatial transformer encodings

In order to compare the performances of different architectures, three specific architectures were taken into consideration. The first architecture incorporated a spatial transformer encoding only at the bottleneck layer of the UNet architecture. In the second architecture, the spatial transformer encoding was introduced not only at the bottleneck layer but also at the skip connection of the fourth layer. Finally, in the third architecture, the spatial transformer encoding was introduced at the skip connections of the third layer, fourth layer, and bottleneck layer. These three different architectures were then evaluated on the basis of their performance on the given dataset to determine which architecture provided the best results.

In this work, a self-attention mechanism to capture the relationships within the final features of the left branch is utilized. This self-attention operation is performed exclusively on the features within the left branch, just before they undergo the process of concatenation.

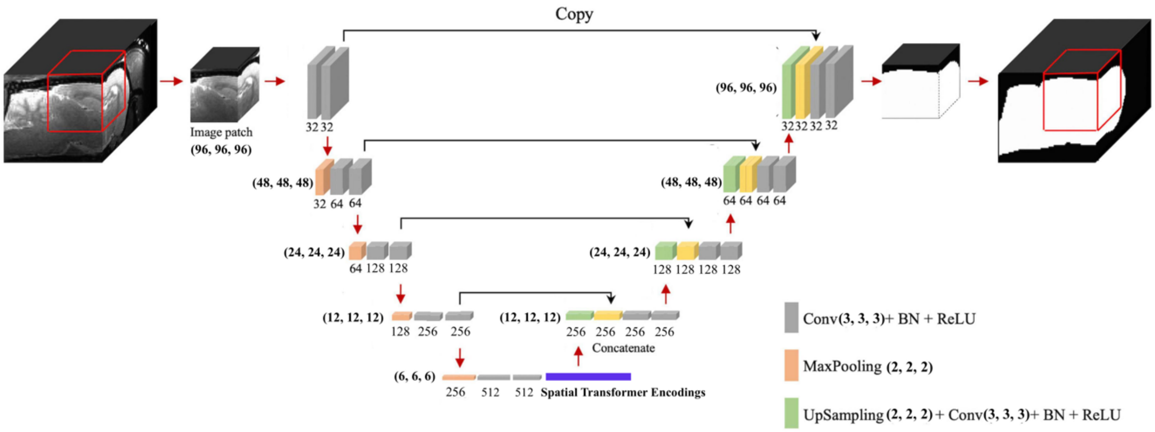


Figure 8.1: Multi-Layer encodings at only bottleneck layer with Spatial Transformer Encodings

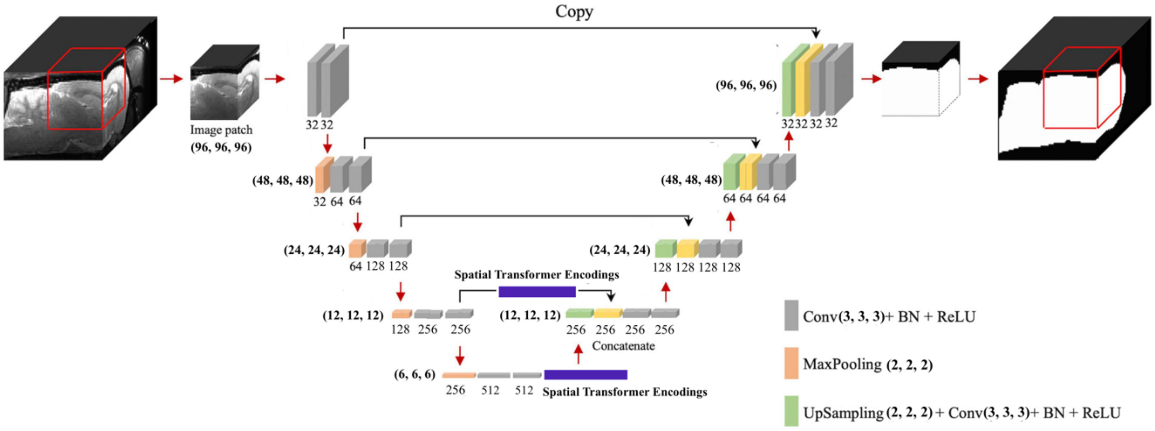


Figure 8.2: Multi-Layer encodings at 4th layer and bottleneck layer with Spatial Transformer Encodings

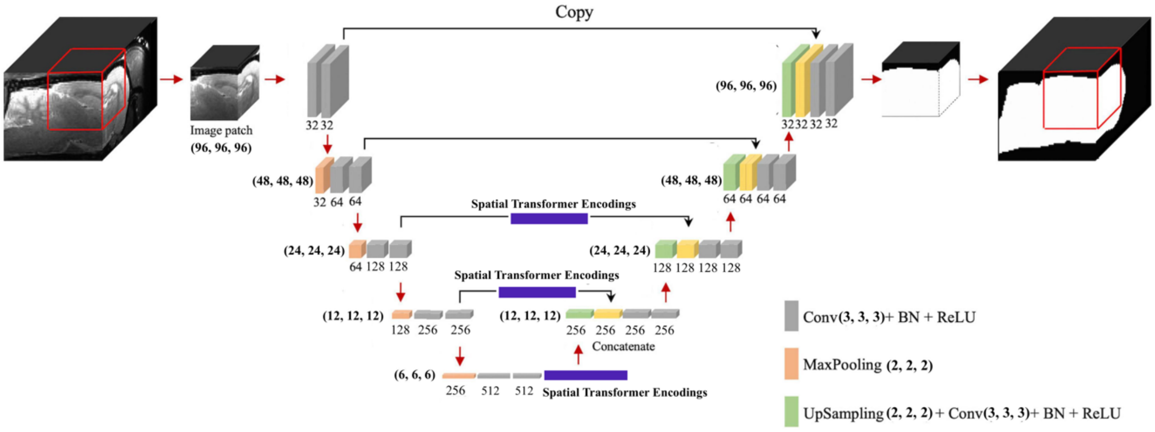


Figure 8.3: Multi-Layer encodings at 3rd layer, 4th layer and bottleneck layer with Spatial Transformer Encodings

8.3 Results for Multi-Layer Encodings with spatial transformer encodings

Given the computational and memory limitations that was encountered, we were compelled to develop the simplest architecture possible. However, it is worth noting that this simplicity actually works in our favor when it comes to integrating spatial transformer encodings at each layer. Surprisingly, despite the straightforwardness of the network, the addition of these encodings does not lead to overfitting issues. In fact, it allows us to introduce more complexity into the network without compromising its overall performance. This outcome provides us with the opportunity to enhance the network's capabilities and tackle more intricate tasks while still operating within the constraints imposed by computational and memory limitations.

Every spatial transformer encoding in our architecture is comprised of a compact yet powerful structure. It encompasses two layers, featuring two heads each, and encompasses a feed-forward vector with a dimensionality of 128 and the UNet architecture has 16, 32, 64, and 128 features.The results of the study suggest that adding spatial transformer encodings on multiple layers can lead to a significant improvement in the validation dice scores.

The study found that the third architecture, which had spatial transformer encoding at multiple layers, performed the best in terms of validation dice scores.

Combination	Validation Dice Score
Bottleneck	0.3618
4th Layer + Bottleneck	0.3753
3rd Layer + 4th Layer + Bottleneck	**0.4002**

Table 8.1: Multi-Layer Encodings with Spatial Transformer Results

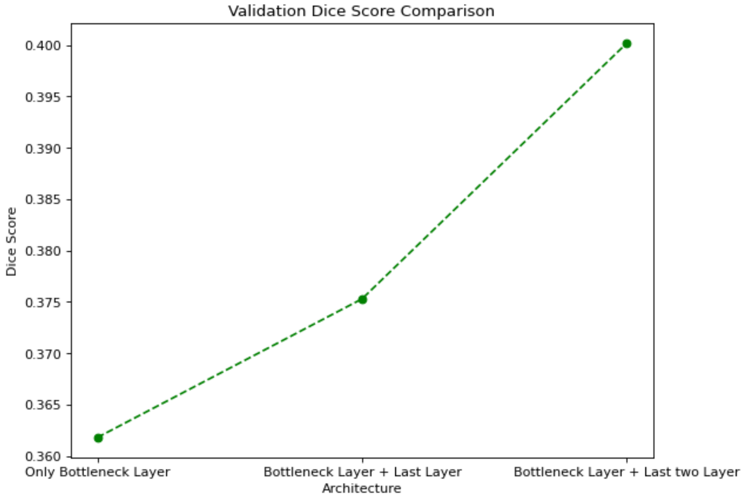


Figure 8.4: Multi-Layer Encodings with Spatial Transformer Results

8.4 Multi-Layer Recurrence with Temporal Transformer Encodings

In order to investigate the potential benefits of temporal transformer encodings in the context of multi-layered recurrence UNet architectures, four different network configurations were compared. The first architecture utilized a temporal transformer solely at the bottleneck layer. The second architecture incorporated a temporal transformer at both the 4th layer and bottleneck layer. The third network introduced temporal encodings at the 3rd layer, 4th layer, and bottleneck layer. The fourth network configuration included temporal transformers at the 2nd layer, 3rd layer, 4th layer, and bottleneck layer. By comparing the performances of these five different models, the impact of multi-layer temporal transformer encodings on the overall performance of the network was assessed.

In this work, an attention mechanism not only considers the current time step but also takes into account the information from both previous and future time steps within the final features of the left branch. This operation is performed exclusively on the features within the left branch, just before they undergo the process of concatenation.

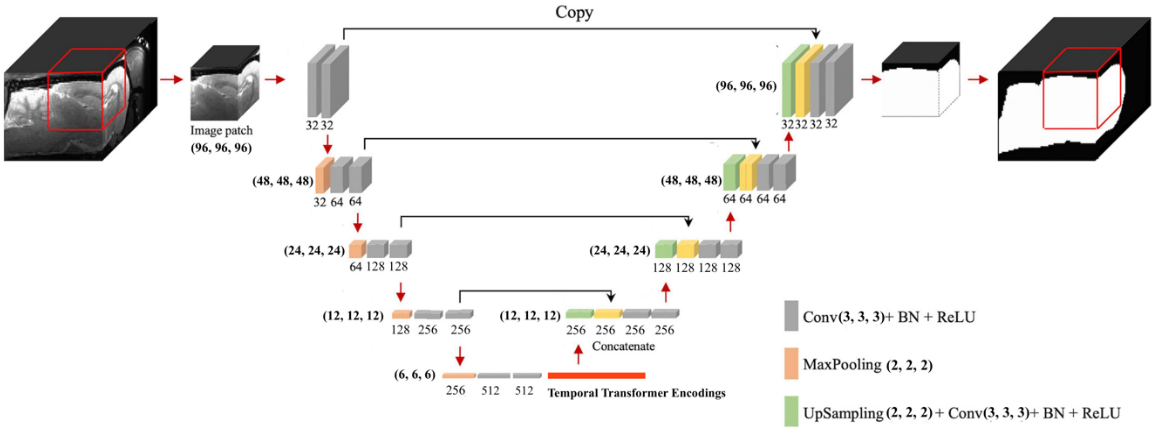


Figure 8.5: Multi-Layer Recurrence at only bottleneck layer with Temporal Transformer Encodings

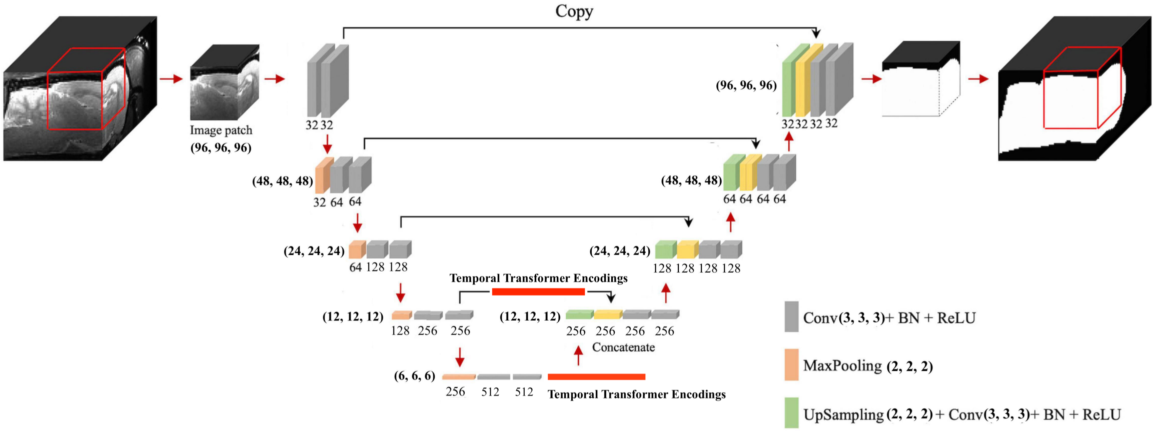


Figure 8.6: Multi-Layer Recurrence at 4th layer and bottleneck layer with Temporal Transformer Encodings

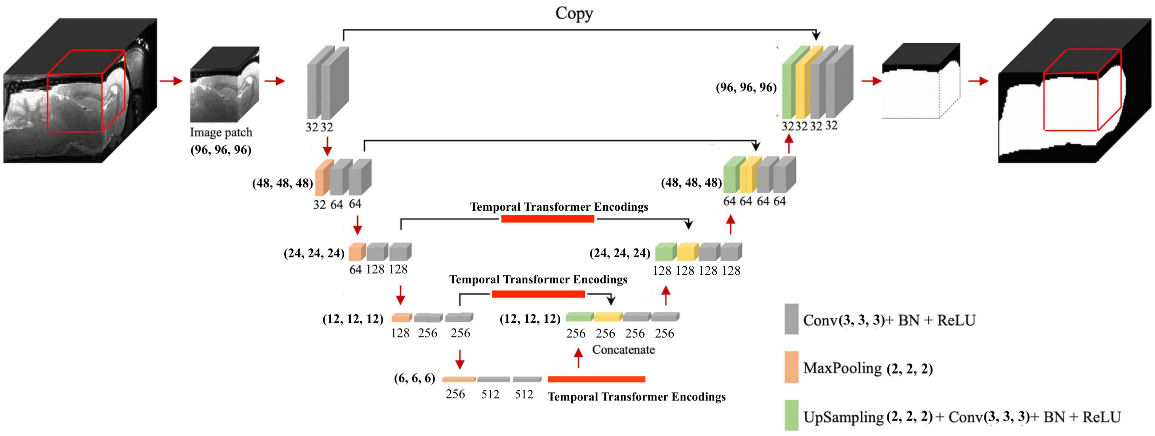


Figure 8.7: Multi-Layer Recurrence at 3rd layer, 4th layer and bottleneck layer with Temporal Transformer Encodings

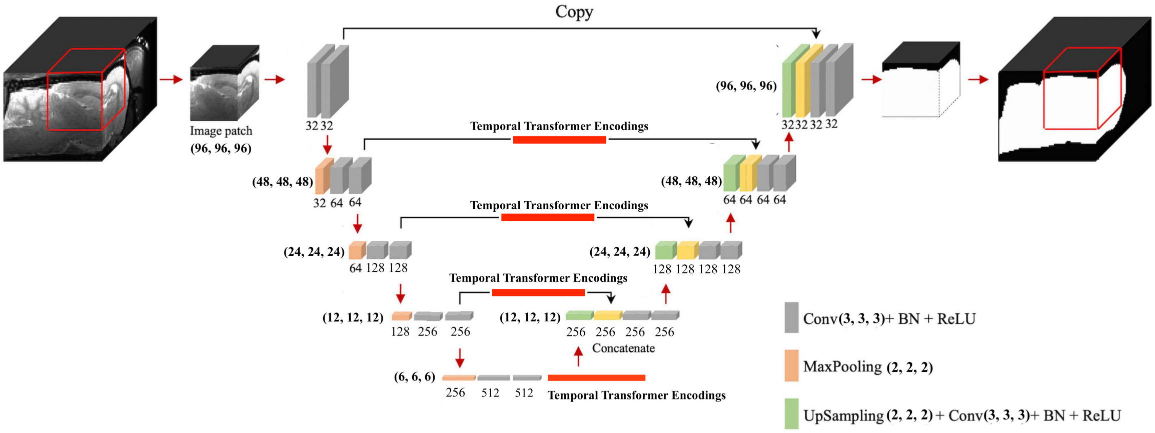


Figure 8.8: Multi-Layer Recurrence at 2nd layer, 3rd layer, 4th layer and bottleneck layer with Temporal Transformer Encodings

8.5 Results for Multi-Layer Recurrence with Temporal Transformer Encodings

Every temporal transformer encoding in our architecture is comprised of a compact yet powerful structure. It encompasses four layers, featuring size heads each, and encompasses a feed-forward vector with a dimensionality of 512 and the UNet architecture has 16, 32, 64, and 128 features. Note that this encoding has different complexity than spatial transformer encodings, temporal transformer encodings are more complex than spatial transformer encodings which were presented earlier, and the computation and memory allowed a more complex architecture for temporal transformer encodings. It is possible that if compared on the same complexity, both spatial and temporal transformer encodings may show similar results when added to different layers.

The results of the study suggest that adding recurrence on multiple layers can lead to a decrease in the validation dice scores most likely due to overfitting as the network becomes more and more complex. The study found that the first architecture, which had temporal transformer encoding at bottleneck layer, performed the best in terms of validation dice scores.

Overall, these findings highlight the importance of considering the impact of different architectural modifications on the performance of the model and the potential benefits of incorporating or removing recurrence in the design of deep learning models.

Combination	Validation Dice Score
Bottleneck	**0.4103**
4th Layer + Bottleneck	0.4098
3rd Layer + 4th Layer + Bottleneck	0.4083
2nd Layer + 3rd Layer + 4th Layer + Bottleneck	0.3761

Table 8.2: Multi-Layer Recurrence Temporal Transformer Results

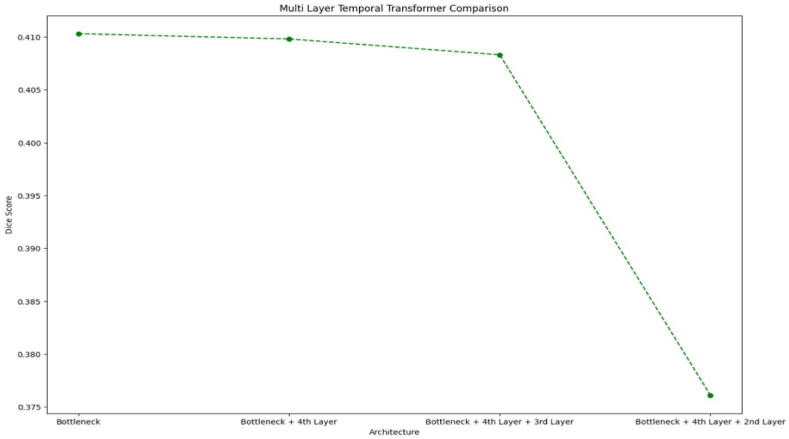


Figure 8.9: Multi-Layer Recurrence Temporal Transformer Results

9. Conclusion

9.1 Summary of Results

The analysis of segmentation through time poses a unique challenge, as it requires the architecture to comprehend the sequentiality of the images to provide an accurate segmentation analysis. The study conducted revealed that the regular UNet architecture yielded the poorest performance in this task.

To address this issue, an LSTM-based UNet architecture was designed and tested, which outperformed the regular UNet architecture when temporal images were utilized for segmentation analysis. However, the transformer-based stt-UNet architecture exhibited better performance than the LSTM-based stt-UNet architecture. This indicates that incorporating a transformer-based model into the architecture design can significantly improve the performance of the model. An even detailed work revealed that a temporal transformer was a better choice than a spatial transformer when considering temporal scans.

Furthermore, it was investigated that introducing recurrence at each layer of the architecture design can improve or decrease the performance of the same combination of the architecture depending on the transformer introduced or the complexity of the architecture. This highlights the importance of incorporating or reducing sequentiality and recurrence in the architecture design to achieve better performance in segmentation through time analysis.

In conclusion, the study emphasizes the need for an architecture that can comprehend sequentiality and effectively analyze segmentation through time. The performance analysis of various architectures revealed that incorporating a transformer-based model can lead to significant improvements in performance. Therefore, incorporating these elements in the architecture design can lead to better performance in segmentation through time analysis.

9.2 Future Works

As advancements in medical imaging technologies continue to expand, so too do the opportunities for leveraging this data to improve patient outcomes. One potential future direction in medical imaging research is to utilize all existing previous MRIs to predict a patient's immediate next stage of cancer. This approach could help inform treatment decisions and improve patient outcomes by enabling doctors to anticipate the progression of the disease and act accordingly.

Another possible avenue of research involves using previous MRIs to predict the next N stages of cancer spread. By analyzing a patient's imaging data over time, medical professionals could more accurately predict how the disease is likely to progress, allowing for more tailored and effective treatment plans to be developed. This approach could significantly improve patient prognosis by enabling earlier intervention and potentially reducing the risk of metastasis.

Furthermore, additional metadata such as drugs taken, and other relevant factors could be considered in a prescription-based analysis. This could provide further insights into effective treatment options for patients, enabling medical professionals to develop personalized treatment plans that are tailored to the specific needs of each patient.

In conclusion, these potential future works have significant implications for the diagnosis, treatment, and management of cancer, and warrant further investigation. By utilizing medical imaging data and incorporating additional metadata, researchers could develop more accurate and effective strategies for combating cancer, ultimately improving patient outcomes and quality of life.

Appendix

1. For regular UNet architecture the following architecture was followed.
   1.1 Kernel Size = (3, 3, 3)
   1.2 Input channels = 1
   1.3 Output channels = 2
   1.4 Strides = (2, 2, 2)
   1.5 Spatial Dims = 3
   1.6 Optimizer used was Adam Optimizer

2. For LSTM stt - UNet architecture the following architecture was followed.
   2.1 Kernel Size = (3, 3, 3)
   2.2 Input channels = 1
   2.3 Output channels = 2
   2.4 Spatial Dims = 3
   2.5 Optimizer used was Adam Optimizer

3. For Spatial Transformer Encoded UNet architecture the following architecture was followed.
   3.1 Kernel Size = (3, 3, 3)
   3.2 Input channels = 1
   3.3 Output channels = 2
   3.4 Spatial Dims = 3
   3.5 Number of heads = 8
   3.6 Number of layers = 6
   3.7 Dim Feed Forward = 512
   3.8 TransformerEncoderLayer had batch_first = False and input to TransformerEncoder was [depth * height * width, seq_len, input_channels]
   3.9 Optimizer used was Adam Optimizer

4. For Temporal Transformer Encoded UNet architecture the following architecture was followed.
   4.1 Kernel Size = (3, 3, 3)
   4.2 Input channels = 1
   4.3 Output channels = 2
   4.4 Spatial Dims = 3
   4.5 Number of heads = 8
   4.6 Number of layers = 6
   4.7 Dim Feed Forward = 512
   4.8 TransformerEncoderLayer had batch_first = True and input to TransformerEncoder was [depth * height * width, seq_len, input_channels]
   4.9 Optimizer used was Adam Optimizer

5. For Multi-Layer Spatial Transformer Encoded UNet architecture the following architecture was followed.
   5.1 Kernel Size = (3, 3, 3)

5.2 Input channels = 1
5.3 Output channels = 2
5.4 Spatial Dims = 3
5.5 Number of heads = 2
5.6 Number of layers = 2
5.7 Dim Feed Forward = 128
5.8 UNet architecture = (16, 32, 64, 128)
5.9 Optimizer used was Adam Optimizer
5.10 TransformerEncoderLayer had batch_first = False and input to TransformerEncoder was [depth * height * width, seq_len, input_channels]

6. For Multi-Layer Temporal Transformer Encoded UNet architecture the following architecture was followed.

6.1 Kernel Size = (3, 3, 3)
6.2 Input channels = 1
6.3 Output channels = 2
6.4 Spatial Dims = 3
6.5 Number of heads = 6
6.6 Number of layers = 4
6.7 Dim Feed Forward = 512
6.8 UNet architecture = (16, 32, 64, 128)
6.9 Optimizer used was Adam Optimizer
6.10 TransformerEncoderLayer had batch_first = False and input to TransformerEncoder was [depth * height * width, seq_len, input_channels]

Bibliography

[1] **Olaf Ronneberger, Philipp Fischer and Thomas Brox**, "U-Net: Convolutional Networks for Biomedical Image Segmentation", 18th International Conference Munich, Germany, 18 November 2015

[2] **Mayo Clinic**, Source: https://www.mayoclinic.org/diseases-conditions/brain-metastases/symptoms-causes/syc-20350136, 25 October 2022.

[3] **Antonio Di Ieva, Carlo Russo, Sidong Liu, Anne Jian, Michael Y. Bai, Yi Qian and John S. Magnussen**, "Application of deep learning for automatic segmentation of brain tumors on magnetic resonance imaging: a heuristic approach in the clinical scenario", In Springer Nature 2021.

[4] **Elekta**, Source: https://gammaknife.com/what-is-gamma-knife, 2019

[5] **NYUMets**, Source: https://nyumets.org 2022

[6] **Spyridon Bakas, Mauricio Reyes, Andras Jakab and Stefan Bauer**, "Identifying the Best Machine Learning Algorithms for Brain Tumor Segmentation, Progression Assessment, and Overall Survival Prediction in the BRATS Challenge", In arXiv:1811.02629, 23 Apr 2019.

[7] **Li-Ming Hsu, Shuai Wang, Lindsay Walton, Tzu-Wen Winnie Wang, Sung-Ho Lee and Yen-Yu Ian Shih**, "3D U-Net Improves Automatic Brain Extraction for Isotropic Rat Brain Magnetic Resonance Imaging Data", In Frontiers in Neuroscience, 16 December 2021.

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346 USA