MRI Brain Tumor Segmentation

Through

U-Net Convolutional Neural Network

Author: Sanjeev Narasimhan

Date: 23, October 2024

Abstract

This paper presents a study on the segmentation of brain MRI scans to accurately isolate tumor locations using deep learning techniques. To carry out this task, a U-Net convolutional architecture is used to generate a tumor mask that can accurately show the size, shape, and location of the tumor within the cerebrum. The segmentation process involves training and tuning the model on both an input image and an associated ‘ground truth mask’ to allow the model to understand the characteristics to look for. Following the training process, the model’s performance is then evaluated on a separate test dataset. Brain tumors, whether malignant or benign, represent a significant health risk due to their potential to cause increased intracranial pressure and subsequent brain damage. Early detection and classification of these tumors are crucial for determining effective treatment options, thereby enhancing patient outcomes. Through the application of deep learning techniques, this research aims to enhance the diagnostic capabilities from medical imaging by improving the accuracy and efficiency of brain tumor identification in MRI scans.

Introduction

Background

What is a Brain Tumor

Brain tumors are characterized as abnormal growths of cells within the brain or surrounding tissues, and can be classified into two main categories: malignant and benign. Malignant tumors are characterized by their aggressive nature, growing rapidly and often invading surrounding brain tissue. Benign tumors on the other hand, typically grow more slowly and do not have the ability to invade nearby tissues *(Mayo Clinic, 2023)*. Understanding the nature of these tumors, and their respective symptoms/ treatment options is crucial for early detection and effective management to improve patient outcomes.

Malignant Tumors

The growth dynamics of brain tumors play a key role in their clinical management. Malignant tumors are often fast-growing and can grow into irregular shapes. Their ability to invade surrounding tissue makes complete surgical removal especially challenging, leading to higher risks of recurrence and necessitating additional treatments such as radiation therapy and chemotherapy *(Mayo Clinic, 2023).* Symptoms of malignant tumors can include seizures, cognitive impairments, and motor weakness, primarily due to increased intracranial pressure as the tumor grows. This pressure can manifest in headaches, nausea, and vomiting, significantly affecting a patient's overall well-being, as well as increase the risk of more life threatening/ severe complications down the road *(Mayo Clinic, 2023).*

Benign Tumors

In contrast, benign tumors generally present a more favorable prognosis. They tend to be characterized as well-defined masses and do not have the ability to invade surrounding tissues. This inability to spread allows for successful surgical intervention in most cases, however, additional treatment methods such as radiation therapy may be necessary depending on the size/ location of the tumor *(Mayo Clinic, 2023).* While not as severe, symptoms can still arise due to increased intracranial pressure or neurological deficits related to the specific location and size of the tumor.

Effect of Tumor Location & Size

The size and precise location of a tumor are crucial factors influencing the symptoms, risk, and treatment options for patients. Larger tumors exert greater pressure on the brain and surrounding tissue, potentially leading to more pronounced cognitive and functional impairments. Even smaller tumors can cause significant issues if located in critical areas of the brain *(see Figure. 1)*.

Figure 1

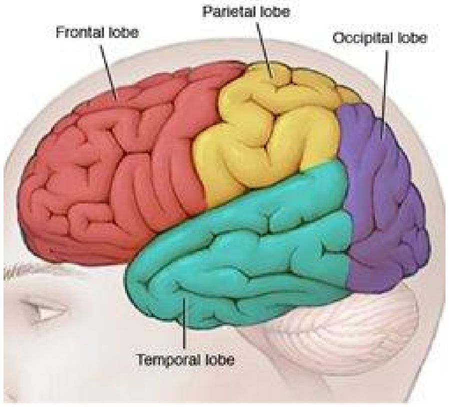


Diagram of Brain Lobes (Mayo Clinic, 2023)

- *Frontal Lobe (Responsible for Thinking and Movement):* The frontal lobe is responsible for higher level functions such as decision-making, planning, and problem-solving. Tumors in the frontal lobe can significantly impact cognitive functions and motor skills in the form of disturbances in balance/ coordination, personality changes (including increased impulsivity, irritability, and forgetfulness), etc.
- *Parietal Lobe (Processes Sensory Information):* The parietal lobe is responsible for processing spatial relationships, as well as sensory information (including touch, temperature, pain, hearing, etc). Tumors in this region can disrupt sensory processing, leading to issues such as poor spatial awareness, difficulty recognizing objects by touch, and alterations in vision or hearing.
- *Occipital Lobe (Responsible for Vision):* The occipital lobe is primarily involved in visual processing and interpretation. Tumors in this area can cause partial or complete vision loss, visual field defects, or visual hallucinations. Patients may have difficulty recognizing familiar faces or objects, leading to disorientation and increased risk of accidents.
- *Temporal Lobe (Involved in Memory and Sensory Processing):* The temporal lobe plays a critical role in memory formation, language comprehension, and sensory processing. Tumors that form in this region can lead to memory problems, such as difficulty forming memories (anterograde amnesia) or recalling past events (retrograde amnesia). Patients may also experience language

deficits which can impact their ability to communicate effectively, as well as experience auditory hallucinations or altered perception of sounds.

Types of Brain Tumors

The most prevalent types of primary brain tumors (tumors that originate in the brain) include gliomas, meningiomas, and pituitary adenomas.

Glioma

A glioma is a tumor that originates from the glial cells in the brain or spinal cord. While glioma cells resemble healthy glial cells, their growth can exert pressure on surrounding brain or spinal cord tissues, leading to various symptoms. Malignant gliomas account for ~70% of newly diagnosed malignant primary brain tumors in adults *(Wen & Kerari, 2008).* Among these, glioblastomas are the most aggressive, with adults experiencing a median survival time of ~15 months.

Meningioma

A meningioma is technically not a "brain tumor" as it is not formed within the brain, but rather in the membranes that surround the brain and spinal cord (meninges). As the growth develops, it can press against adjacent brain tissue, nerves, and blood vessels, with the pressure manifesting into the aforementioned symptoms. While meningiomas are among the most common primary intracranial tumors, < 3% are classified as malignant *(Alruwali & De Jesus, 2020)*.

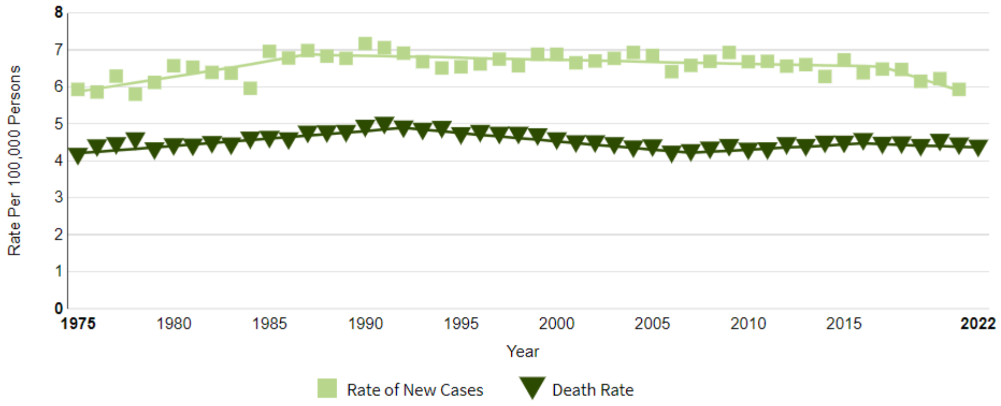
Pituitary

Pituitary tumors develop in the pituitary gland and can disrupt hormone production, affecting various bodily functions/ natural responses. The majority of these tumors are benign, referred to as pituitary adenomas, while only ~0.2% are malignant, and are referred to as pituitary carcinomas *(Yale Medicine, n.d.).*

Research Motivation

Approximately 80,000 new cases of primary brain tumors are diagnosed in the United States annually, and the American Cancer Society estimates that around 25,400 new cases of malignant brain tumors will be diagnosed in 2024 *(Key Statistics for Brain and Spinal Cord Tumors, n.d.)*. While on a slight downward trend, it is still observable that there is little fluctuation in both the rate of new Brain & Other Nervous System Cancer and its associated Death Rate *(see Figure 2)*.

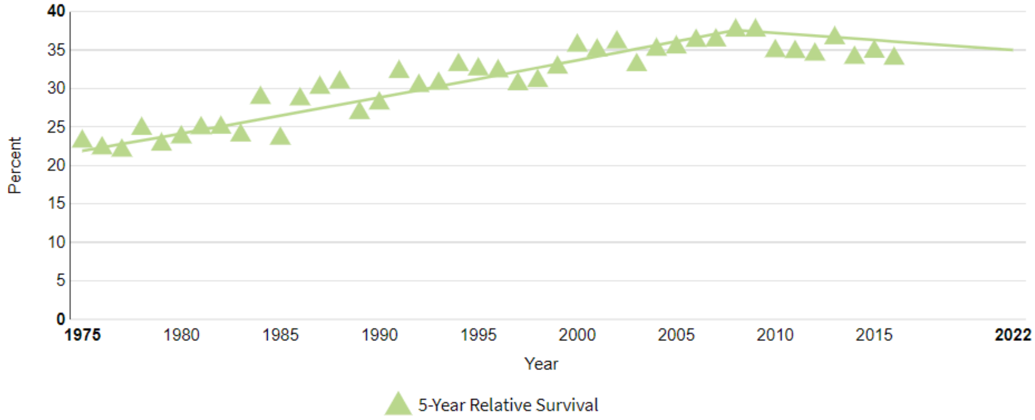
Figure 2



Brain and Other Nervous System Cancer Rates (SEER, 2018)

Furthermore, the five-year relative survival rate for all primary brain tumors is about ~33.4%, though this figure varies significantly based on tumor type and stage at diagnosis *(see Figure 3)*.

Figure 3



Brain and Other Nervous System Cancer 5-Year Survival Rates (SEER, 2018)

To maximize the likelihood of a favorable prognosis, it is essential to initiate the appropriate treatment plan as early as possible. This process begins with a neurological exam to evaluate a patient's reflexes, balance, vision, hearing, etc. Next, the identification of the tumor, typically through MRI imaging (and/ or PET/ CT scan), allows for detailed visualization of brain structures and helps localize the tumor accurately. Following tumor detection, a biopsy is necessary to determine the type of tumor, whether it is malignant, and finally its grade *(What is Cancer Pathology?, 2019)*. Determining these factors are crucial as they significantly influence treatment options and prognosis. Once these factors are established, a tailored treatment strategy can be formulated, which may include surgery, radiation therapy, and/or chemotherapy, depending on the tumor characteristics and the patient’s overall health.

Utilizing image segmentation models, particularly deep learning approaches like U-Net, can significantly enhance the process of tumor diagnosis and treatment planning. These models can generate “tumor masks” from MRI scans, providing precise segmentation to help radiologists better assess the tumor's size, shape, and location, which are vital factors in determining the tumor type and grade. By highlighting tumor regions, these models can reduce the likelihood of human error and variability in interpretation, allowing for more consistent and reliable assessments. Furthermore, segmentation techniques can aid in identifying subtle changes in tumor morphology over time, facilitating early detection of progression or response to treatment. This information can then lead to adjustments in the treatment plan, ensuring that patients receive the most effective interventions as quickly as possible.

Dataset

The images in this dataset (sourced from Kaggle) were obtained from The Cancer Imaging Archive (TCIA), and correspond to 110 patients included in The Cancer Genome Atlas (TCGA) lower grade glioma collection.

The dataset is made up of 7860 .tif files:

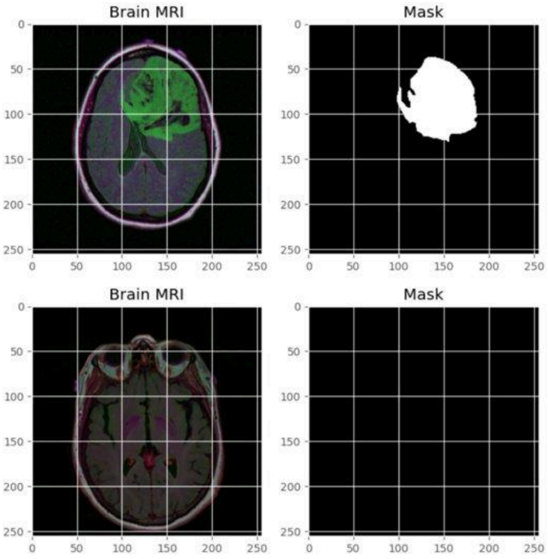
- MRI Brain Scan
- FLAIR Abnormality Segmentation Mask

The dataset was processed using an 80 / 20 Train-Test-Split, and augmentation techniques such as flipping, rotating, and shifting were used to improve model generalization.

The two sets of images *(see Figure 4)* illustrate a scan featuring a tumor (top) and a scan without a tumor (bottom). When a tumor is present in the MRI, an associated "mask" is provided, which delineates the tumor's attributes.

During the training process, the original image (left) is paired with the corresponding ground truth mask (right) to enable the model to learn the characteristics that define a tumor (size, shape, and location). The objective is to train the model such that it can generate a similar mask to the ground truth. In other words, can it accurately identify the size, shape, and location of a tumor based on the MRI scan.

Figure 4



Dataset 1 Images – RGB Color Coded MRI & Ground Truth Mask

Deep Learning Background

What is a U-Net

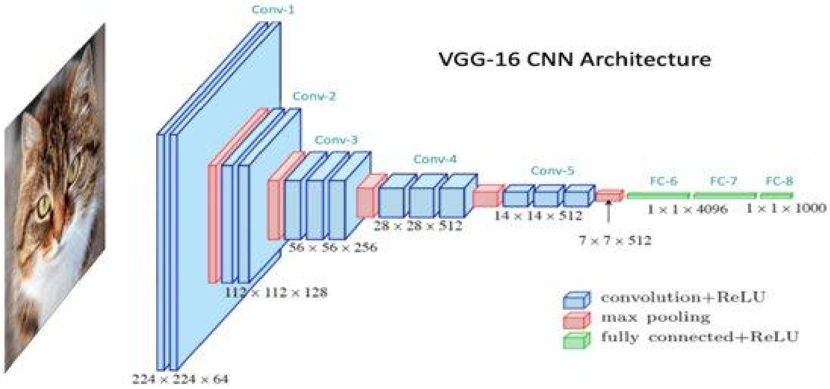
Image processing models have revolutionized the field of medical imaging by enabling more accurate and efficient analysis of complex datasets. Among these models, convolutional neural networks (CNNs) and their advanced variants, such as U-Net, have become essential tools for tasks like image classification and segmentation respectively.

These models leverage hierarchical feature extraction and spatial information to identify and understand the key aspects within medical images, such as tumors in MRI scans. Understanding the differences between a traditional CNN and a U-Net provides insight into their respective capabilities, and best explains why a U-Net was selected for this task of MRI Segmentation.

Traditional Convolutional Neural Network

A Convolutional Neural network (CNN) is an image processing model that extracts an image's features to classify its contents. The VGG-16 model *(see Figure 5)* is a CNN architecture that was proposed by the Visual Geometry Group at the University of Oxford, and is renowned for its simplicity and effectiveness in computer vision tasks.

Figure 5



VGG-16 CNN Architecture Diagram (GeeksforGeeks, 2020)

Aspects of VGG-16 CNN

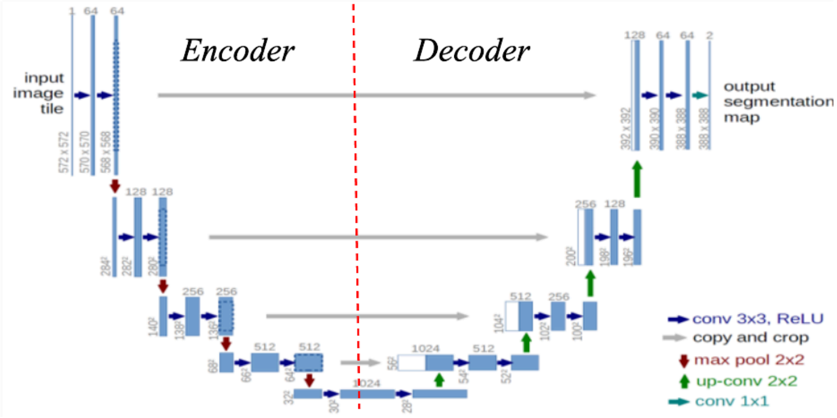
- Convolutional Layers (Blue): Convolutional layers learn to apply various filters to the input image, allowing the network to detect important features such as edges, textures, and patterns

- Pooling Layers (Red): Pooling layers are used to downsample or compress the image, reducing its spatial dimensions while retaining the most critical information
    - Hierarchical Feature Detection: As the network processes the image through successive layers, it learns from increasingly compressed versions, which allow it to better understand the attributes that make up a given image
- Flattening (Green): After the convolutional and pooling layers, the feature map is flattened into a one dimensional vector
    - Fully Connected Layer: The flattened vector is fed into a fully connected layer, which generates the final output → classification score

U-Net CNN

U-Net builds upon the principles of traditional CNNs but introduces several key modifications that enhance its applicability for image segmentation tasks. What makes a U-Net unique is its U-shaped architecture that consists of an encoder network and a decoder network *(see Figure 6)*.

Figure 6



U-Net Architecture Diagram (Zhang et al., 2022)

Aspects of U-Net CNN

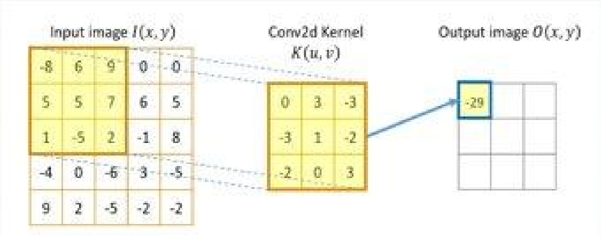
- Encoder – Contraction: Like traditional CNNs, the encoder portion of the U-Net compresses the input image and learns hierarchical features through downsampling. It captures essential features at different scales, making it adept at understanding various aspects of the input image.
- Decoder – Expansion: The U-Net incorporates a decoder that reconstructs the compressed images through upsampling. This process aims to restore the spatial dimensions lost during downsampling, enabling the model to generate detailed segmentation maps.

- Skip Connections (Gray Arrows): One of the defining features of U-Net is the use of skip connections. These connections concatenate feature maps from the encoder with corresponding feature maps from the decoder. By retaining spatial information from earlier layers, the U-Net can produce more accurate segmentations, as it combines the feature information with its precise location.

Architecture Terminology

- *Convolutional Layers:* Convolutional Layers apply a series of filters/ kernels to the input image to extract features (edges, textures, and shapes). This is done by taking the dot product of the matrices as we slide the filter over the input image, thereby creating an output feature map *(see Figure 7)*.

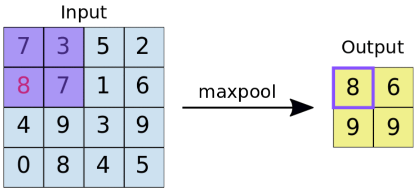
Figure 7



Deep Learning – Convolutional Layer (Vrogue.co, 2024)

- *Max Pooling Layers:* Max pooling layers reduce the spatial dimensions of feature maps by applying a filter to return the maximum value present *(see Figure 8)*.

Figure 8

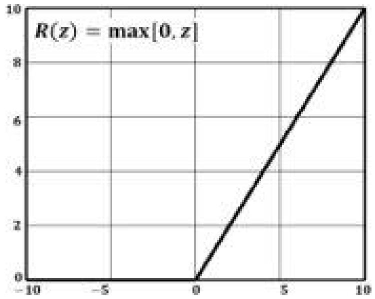


Deep Learning – Max Pooling Layer (Multilayer Perceptrons vs. CNN, 2021)

- *Batch Normalization:* Batch normalization is a technique used to normalize the outputs of a layer during training by adjusting and scaling the activations. This helps stabilize learning, accelerate training, and can improve overall model performance by reducing sensitivity to weight initialization *(GeeksforGeeks, 2024*).

- *Activation Functions:* Activation functions are the means by which a specific layer generates an output, and is a way to introduce non-linearity into the training process.
    - *ReLU Activation:* A ReLU activation is used to create the outputs between layers. It does this by transforming the feature map generated to return all positive values and scale all negative values to 0 *(see Figure 9)*. This function helps prevent issues like vanishing gradients and enables faster convergence.

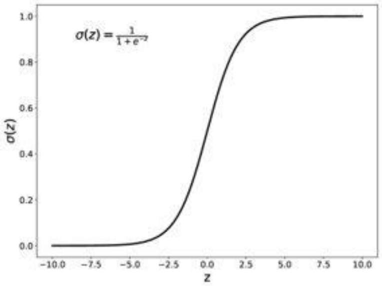
Figure 9



Relu Activation Function (GeeksforGeeks, 2024)

- *Sigmoid Activation*: A Sigmoid activation function is used to create the final output – classifying a pixel in an image as either being a tumor or not. It does this by outputting the map to values between 0 and 1 *(see Figure 10)*.

Figure 10

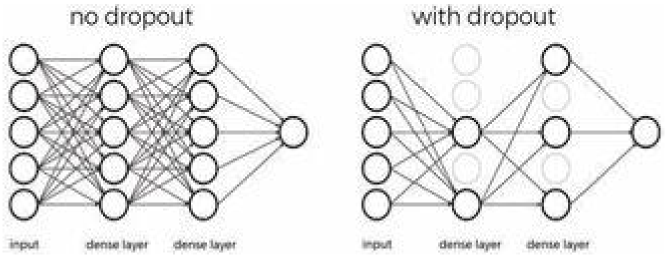


Sigmoid Activation Function (GeeksforGeeks, 2024)

- *Overfitting*: A model is considered to be overfit when it becomes excessively sensitized to the training data, capturing noise and fluctuations rather than the underlying patterns. In deep learning, models with many layers or parameters can become very easily overfit, making it such that it performs poorly when applied to unseen data.

- *Dropout*: Dropout is a regularization technique to prevent overfitting in neural networks by randomly "dropping out" a fraction of neurons or nodes within each hidden layer *(see Figure 11)*. This allows the model to navigate through noise and make more accurate predictions or generalizations with unseen data.

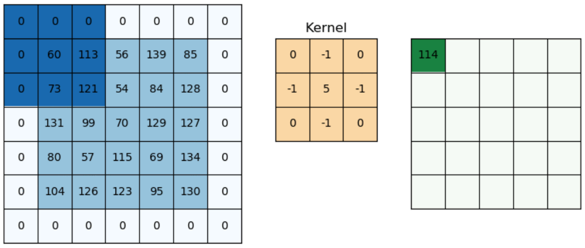
Figure 11



Deep Learning – Dropout Layer (Advanced Layer Types – Introduction to Deep-Learning, 2014)

- *Same Padding:* Same padding ensures that the output feature map has the same spatial dimensions as the input feature map. It does this by “padding” the input image with a border of 0 values, such that when convolution occurs, the output map is of the same size as the actual input map dimensions, thus maintaining spatial information throughout the network *(see Figure 12).*

Figure 12



0	0	0	0	0	0	0
0	60	113	56	139	85	0
0	73	121	54	84	128	0
0	131	99	70	129	127	0
0	80	57	115	69	134	0
0	104	126	123	95	130	0
0	0	0	0	0	0	0

Kernel

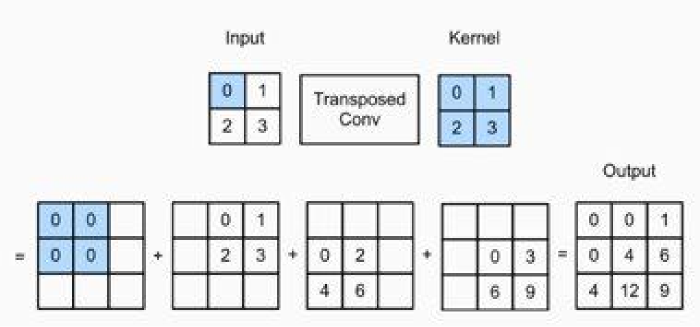
0	-1	0
-1	5	-1
0	-1	0

114				

Deep Learning – Same Padding Layer (Keras Conv2D and Convolutional Layers, 2018)

- *Transposed Convolutional Layers:* Transposed convolutional layers are used to increase the spatial dimensions of feature maps by “reversing” the operations of standard convolutional layers through upsampling. This is done by sliding a filter of greater size than the input area, thus spreading out the features over a larger output space *(see Figure 13)*.

Figure 13

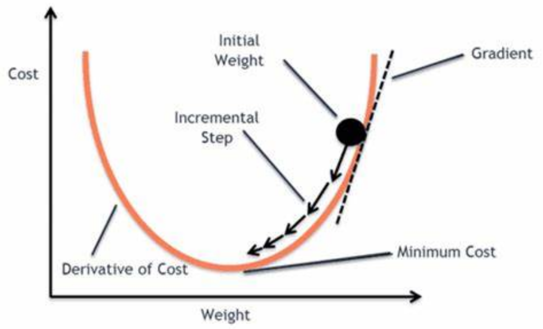


Deep Learning – Transposed Convolutional Layer (Kumar, 2023)

- *Concatenation Layers:* In a U-Net, concatenation layers are the means by which the "skip connections" are created, where the feature maps from the encoder are concatenated with those from the decoder. This preserves spatial information and helps in generating more accurate segmentation masks by integrating the significant features with contextual/ spatial information.

- *Gradient Descent:* Gradient descent is the means by which the model learns to improve by minimizing the loss function of a model. It does this by calculating the gradient (or derivative) of the loss function with respect to the model's parameters, which indicates the direction in which the loss function increases most steeply. Then, the model iteratively adjusts the weights/ parameters of the model in the direction of the negative gradient until the minimum of the cost function is reached *(see Figure 14)*.

Figure 14



Deep Learning – Gradient Descent Diagram (Rosebrock, 2021)

- *Optimizers:* The optimizer's job is to determine which combination of the neural network's weights and biases will give it the best chance to generate accurate predictions *(Efimov, 2023)*. In this, the gradient tells us the direction to move (that minimizes the loss function), and the learning rate tells us how large of a step to take (to reach the minimum loss).

    - *Momentum:* The momentum optimizer uses the momentum (moving average of previous gradients) rather than just the gradient to update weights. In this, if gradients are consistently in the same direction the momentum increases and the step we take can increase in size. This allows for the model to escape being trapped in local minima, as well as smooth out the steps it takes to update weights.
    - *Root Mean Squared Propagation:* The RMSP optimizer allows the learning rate to decay, such that the size of the learning rate gets smaller over time (ensuring we do not boost past the optimal path later in the training process). It does this by dividing the learning rate by an exponentially decaying moving average of the squares of the previous gradients, thus normalizing the updates to the learning rate.
    - *Adaptive Moment Estimation:* The Adam optimizer controls the rate of gradient descent to reduce the oscillation as it nears the global minimum, while still taking large enough steps to converge quickly. It does this by taking the key aspects of Momentum (building up speed) and RMSP (adaptive learning rates for each parameter based on the exponentially decaying sum of squared gradients) to update the learning rates/ weights during the training process.

- *Metrics:*
    - *Binary Cross Entropy:* Binary Cross Entropy measures the difference between the true labels and predicted probabilities, penalizing incorrect predictions → Lower values indicate better model performance.
    - *Precision:* Precision measures how often predictions for the positive class are correct. In the context of this task, it tells us how many predicted tumor pixels are actual tumors → Higher values indicate better model performance & Minimize False Positives.

$$\textit{Precision} = \frac{\textit{True Positives}}{\textit{True Positives} + \textit{False Positives}}$$

    - *Recall:* Recall measures how well the model finds all positive instances in the dataset. In the context of this task, it tells us how many tumor pixels did we correctly identify → Higher values indicate better model performance & Minimize False Negatives.

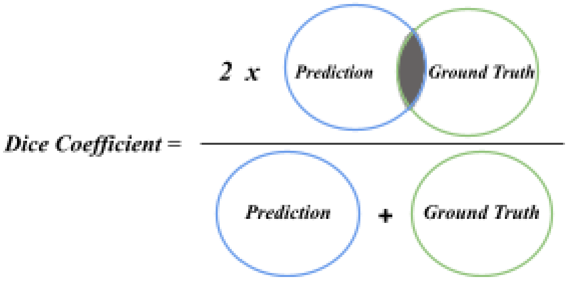
$$\textit{Recall} = \frac{\textit{True Positives}}{\textit{True Positives} + \textit{False Negatives}}$$

- *F1 Score:* The F1 Score is the harmonic mean of precision and recall that balances both metrics and balances the trade-off between precision and recall → Higher values indicate better model performance & Minimize False Positives/ Negatives.

$$\textit{F1 Score} = \frac{2}{\frac{1}{\textit{Precision}} + \frac{1}{\textit{Recall}}}$$

- *Dice Coefficient:* The Dice Coefficient measures the overlap between the predicted and ground truth sets. In the context of this task, it acts as a similarity metric to see how the mask that the model generated compares that the ground truth mask from the dataset → Higher values indicate better model performance.

$$\textit{Dice Coefficient} = \frac{\textit{2 x (Area of Overlap)}}{\textit{Total Area}}$$

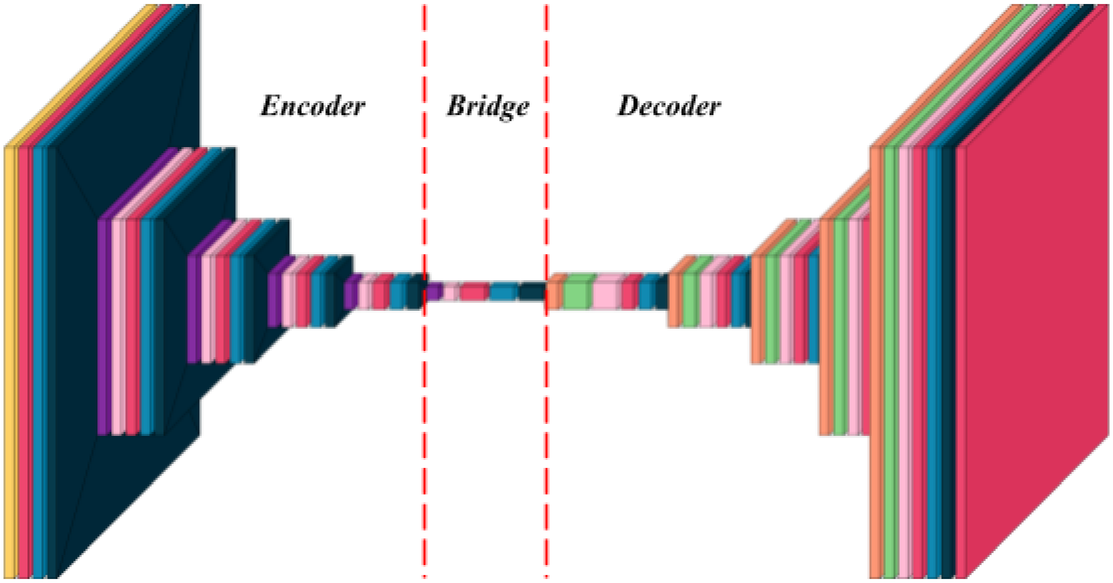


Building Model

Architecture Overview

The structure of the U-Net created for this task is delineated into three distinct components: the encoder, the bridge, and the decoder *(see Figure 15)*.

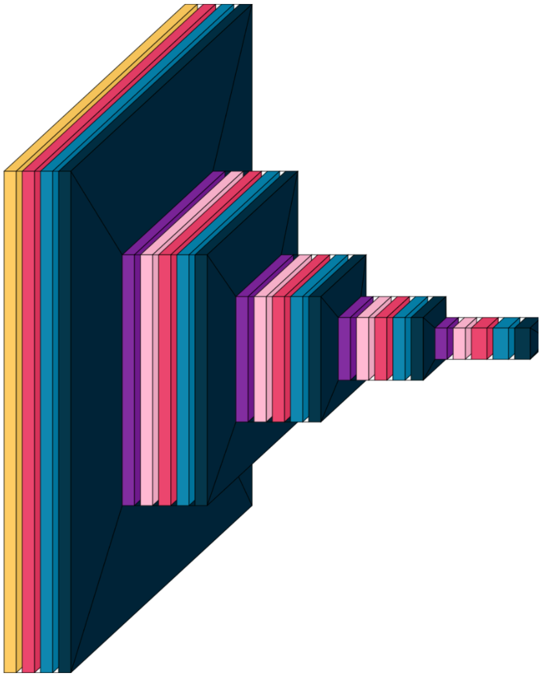
Figure 15



U-Net Architecture Created for MRI Brain Tumor Segmentation

In the encoder, the process begins with an input layer that receives the image. Throughout the network, convolutional and downsampling operations are performed, compressing the spatial dimensions of the feature maps throughout the network until reaching the bridge. The bridge serves as the central segment of the U-Net, where the feature maps are at their smallest spatial dimensions. This section consolidates the most abstract features learned by the encoder through a series of convolutional and compression operations, which allows for the extraction of high-level contextual information. From this, the refined features are sent to the decoder, where the upsampling of feature maps occurs through transposed convolutions, and skip connections are used to map the features to the spatial information.. Finally, the last layer of the decoder enables the generation of pixel-wise classifications, allowing for precise segmentation of the input image.

Encoder Network



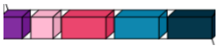
Block 1:

- Yellow → Input Layer
    - Image Width, Height: 224, 224
    - # Channels: 3
- Red → Convolutional Layer
    - # Filters: 16
- Teal → Batch Normalization Layer
- Dark Teal → ReLU Activation Layer

Blocks 2 - 5:

- Purple → Max Pooling Layer
    - Filter Size: 2 x 2
- Pink → Dropout Layer
    - Value = 0.1
- Red → Convolutional Layer
    - # Filters: 32, 64, 128, 256
- Teal → Batch Normalization Layer
- Dark Teal → ReLU Activation Layer

Bridge



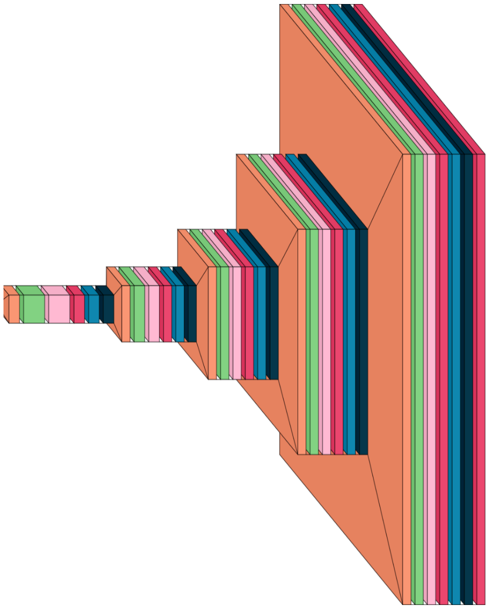
Block 6:

- Purple → Max Pooling Layer
    - Filter Size: 2 x 2
- Pink → Dropout Layer
    - Value = 0.1
- Red → Convolutional Layer
    - # Filters: 512
- Teal → Batch Normalization Layer
- Dark Teal → ReLU Activation Layer

Decoder Network

<u>Blocks 7 - 10:</u>

- Orange → Transposed Convolutional Layer
    - Filter Size: 3 x 3
    - # Filters: 256, 128, 64, 32
    - Stride Filter Size: 2 x 2
    - Padding: Same
- Green → Concatenation Layer
    - Reference: Activation of Mirrored Encoder Block
- Pink → Dropout Layer
    - Value = 0.1
- Red → Convolutional Layer
    - # Filters: 256, 128, 64, 32
- Teal → Batch Normalization Layer
- Dark Teal → ReLU Activation Layer



<u>Block 11:</u>

- Orange → Transposed Convolutional Layer
    - Filter Size: 3 x 3
    - # Filters: 16
- Green → Concatenation Layer
    - Reference: Activation of Mirrored Encoder Block
- Pink → Dropout Layer
    - Value = 0.1
- Red → Convolutional Layer
    - # Filters: 16
- Teal → Batch Normalization Layer
- Dark Teal → ReLU Activation Layer
- Red → Convolutional Layer (Output)
    - Filter Size: 1 x 1
    - Activation Function: Sigmoid
    - Image Width, Height: 224, 224
    - # Channels: 1 (Prediction)

Training the Model

Compiling & Fitting the Model

Compiling Model

- Optimizer: Adam
- Loss: Binary Cross Entropy
- Metrics: Precision, Recall, F1 Score, Dice Coefficient

Fitting Model

- Epochs: 100
- Steps per Epoch: 50
- Callback – Early Stopping: If the validation dice coefficient doesn't improve over the course of 5 epochs, stop the training process and revert weights to previous best score
    - Monitor: Validation Dice Coefficient
    - Patience = 5
    - Mode = Max

Model History

- Epoch 1 – Training Begins
- Epoch 65 – Early Stopping Triggered & Training Stopped
    - Validation Metrics:
        - Binary Cross Entropy: 0.0099
        - Precision: 0.8189
        - Recall: 0.8524
        - F1 Score: 0.8353
        - Dice Coefficient: 0.7423

- Epoch 66 – Model Saved & Training Resumed
- Epoch 81 – Early Stopping Triggered & Training Stopped
    - Validation Metrics:
        - Binary Cross Entropy: 0.0089
        - Precision: 0.8251

- Recall: 0.8926
- F1 Score: 0.8575
- Dice Coefficient: 0.7714

After the first instance of early stopping was triggered, the decision was made to save and resume training to check if the model performance could improve any more. After resuming for another 16 epochs early stopping triggered again, and since the performance didn't improve significantly, the decision was made to end the training process and evaluate the model performance.

Performance Metrics from Evaluating Model

- Binary Cross Entropy: 0.0093
- Precision: 0.8562
- Recall: 0.8435
- F1 Score: 0.8498
- Dice Coefficient: 0.7771

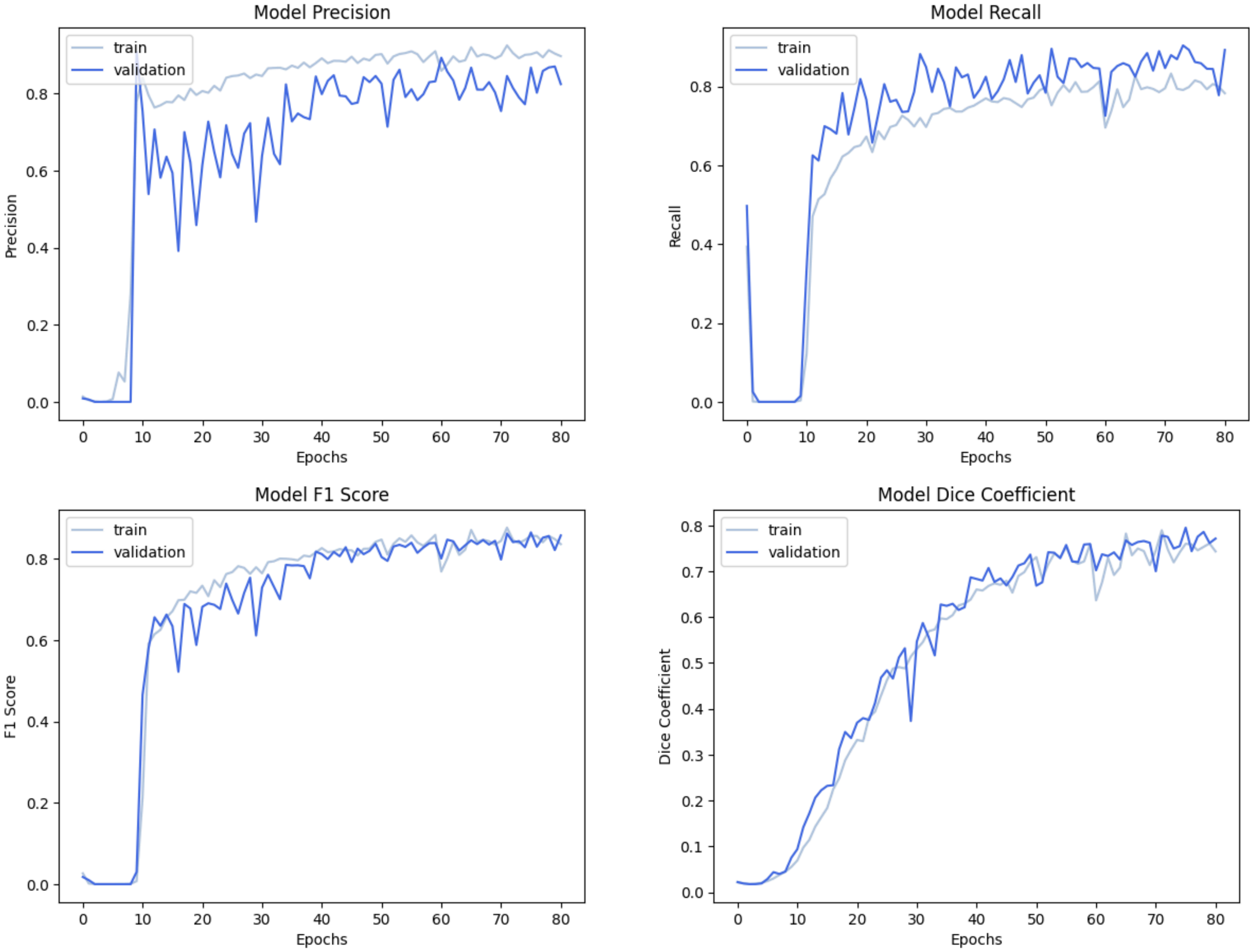
Interpreting Model Performance

The metrics to play closest attention to for this task are the Precision, Recall, F1 Score and the Dice Coefficient.

- Precision: With a test value of 0.8562, this tells us that in a given image, ~85.62% of pixels classified as tumors are actually tumor pixels
- Recall: With a test value of 0.8435, this tells us that in a given image, the model is able to find and classify ~84.35% of all tumor pixels
- F1 Score: With a test value of 0.8498, this tells us that the model's performance is balanced well between correctly classifying tumor pixels and reducing both its False Positive rate and False Negative rate
- Dice Coefficient: With a test value of 0.7771, this tells us that in a given image, there is ~77.71% overlap between the actual tumor and the mask that the model generates.

Gauging Metrics

Figures 16 - 19



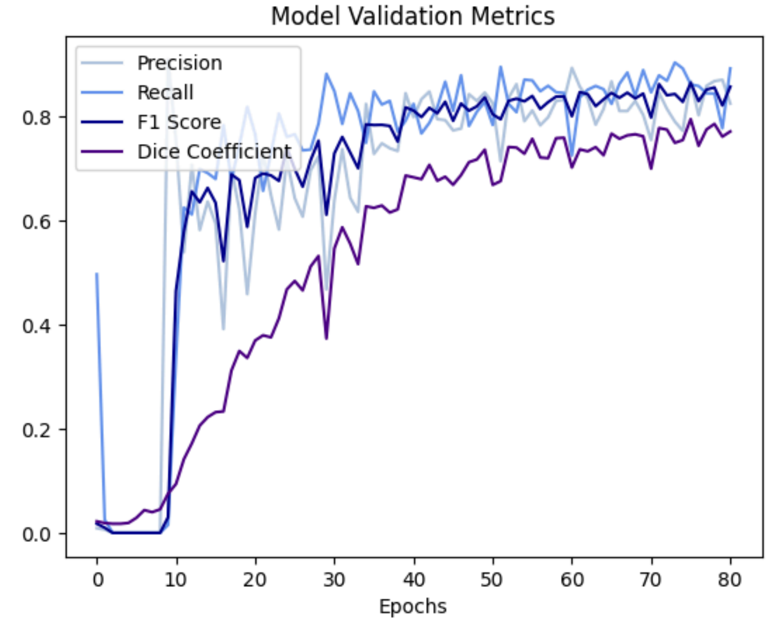
U-Net Model Training & Validation Metrics over Total Training Period

The images presented above *(see Figures 16 - 19)* illustrate the performance of the model throughout the training period, specifically in terms of Precision (upper left), Recall (upper right), F1 Score (lower left), and Dice Coefficient (lower right).

Analysis of the predictive metrics (Precision, Recall, and F1 Score) reveals an early performance spike between epochs 10 and 15, with training and validation metrics closely tracking one another. This indicates that even at an early stage of training, the model effectively classified pixels as "tumor pixels." Although fluctuations were observed, a generally positive trend was maintained.

In contrast, the Dice Coefficient exhibits a slightly different trend. As this metric quantifies the overlap between the predicted and ground truth masks, it serves as a more robust indicator of the model's improvement. The data demonstrates a consistent upward trajectory throughout the training process, characterized by minimal fluctuation in both training and validation metrics.

Figure 20



U-Net Model Validation Metrics Across Total Training Period

A comparison of the trends for all metrics throughout the training process reveals a shared positive trajectory, however, performance begins to plateau around epoch 65, coinciding with the first instance of early stopping *(see Figure 20)*. Even after training resumes, only minimal improvement is observed. This indicates that enhancing the overall performance and capabilities of the model would require modifications to the architecture and processing steps.

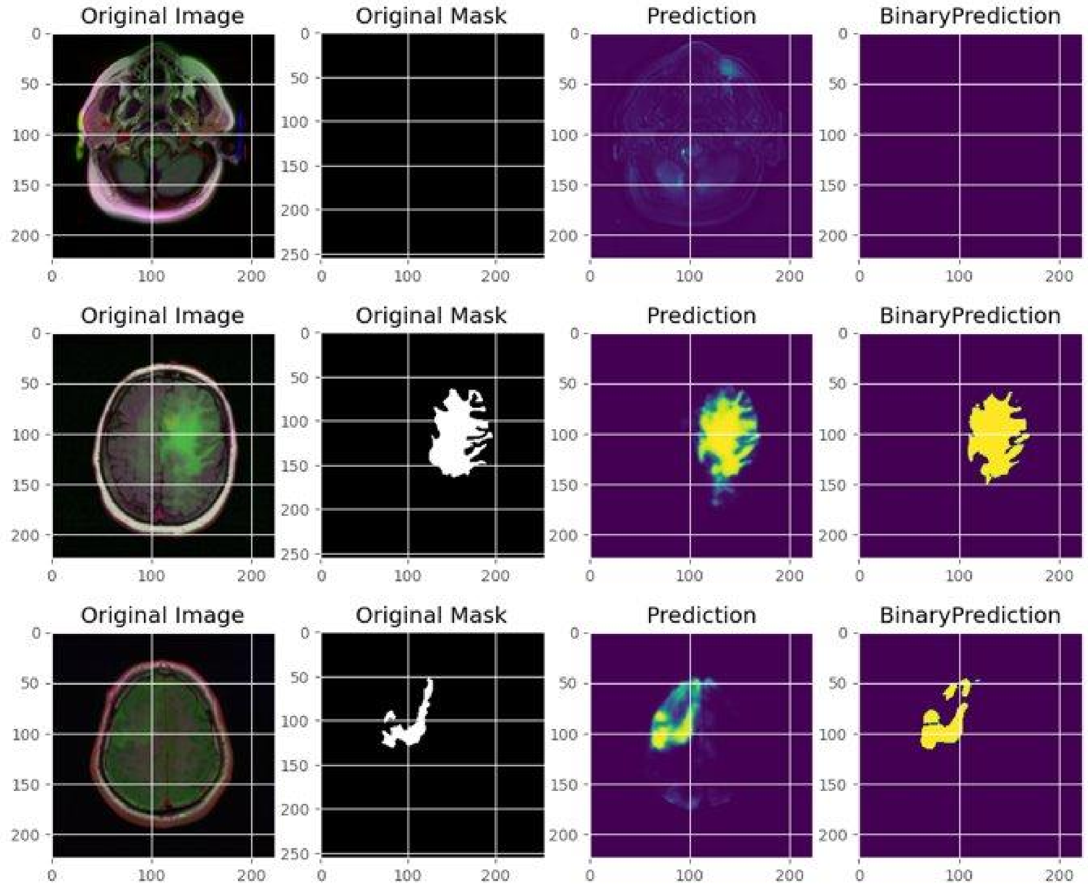
Testing Model Performance

Testing on Original Dataset

With the model fully trained and demonstrating strong metrics, we can randomly select images from the testing dataset (images not used to train the model) to assess the model's overall performance.

Good Performance

Figures 21 - 23



U-Net Model Probabilistic & Binary Mask Generation (Good)

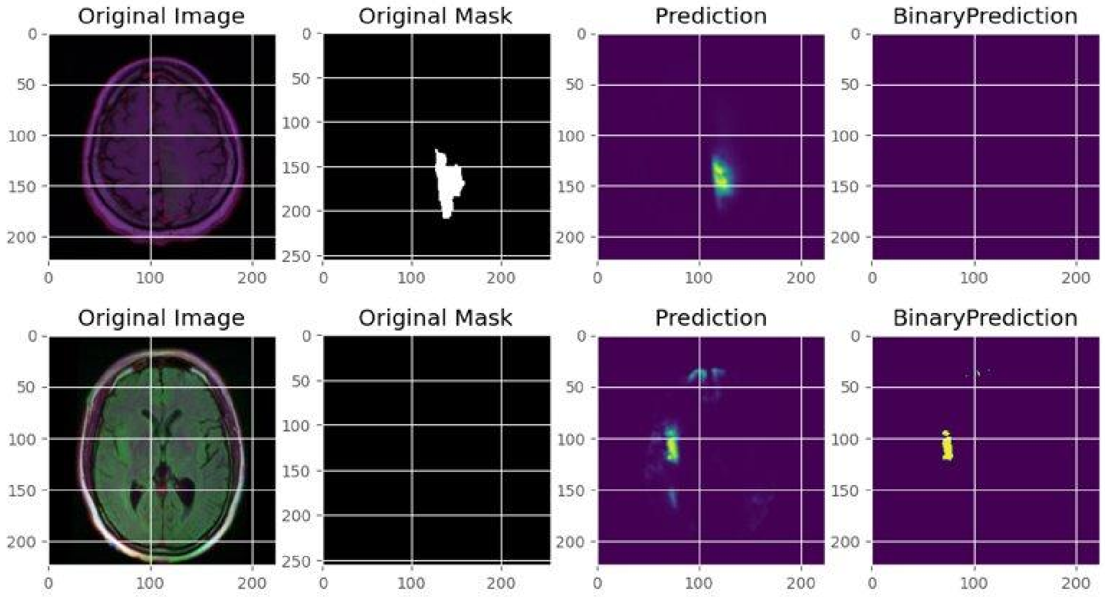
The images above *(see Figures 21 - 23)* illustrate instances in which the model demonstrated particularly strong performance. The two images on the left (Original Image and Original Mask) depict a randomly selected image for processing alongside the ground truth mask, which delineates the precise size, shape, and location of the tumor. The images on the right (Prediction and Binary Prediction) display the outputs generated by the model after processing the Original Image. The image labeled "Prediction" represents a probabilistic output, where brighter colors (approaching yellow) indicate a higher confidence that the corresponding pixel is part of a tumor. This visualization allows users to identify the specific areas the model considered when determining whether a pixel contained tumor tissue, thereby providing a general region for radiologists to examine for potential tumors. The "Binary Prediction" image is generated by applying a threshold of 0.5 to the probabilistic prediction, classifying pixels with values above 0.5 as tumor pixels and those below as background pixels.

The first set of images demonstrates the model's performance in the absence of a tumor. While the model identified some areas of interest in the upper right and lower left regions (shown in pale blue), it confidently classified all pixels in the image as background.

The second and third sets of images reflect the model's performance when a tumor is present. In the second set, the model successfully identifies an abnormally shaped tumor, characterized by a generally circular mass with several "breaks" and "divots". Despite these irregularities, the model accurately recognizes the area as containing a tumor and captures many of its distinguishing features. In the third set, another abnormally shaped tumor is presented, featuring a mass with a protruding "tail" in the upper half. Again, the model performs well, successfully capturing the tumor's general shape, size, and location, although a few characteristics were missed.

Poor Performance

Figures 24 & 25



U-Net Model Probabilistic & Binary Mask Generation (Poor)

Unlike the previous sets of images, the images above illustrate instances in which the model demonstrated poor performance *(see Figures 24 & 25)*.

The first set showcases a false negative, where the model identified a cluster of pixels in the MRI that suggested the presence of a tumor but lacked the confidence to classify them as tumor pixels. The second set highlights a false positive scenario, wherein the model incorrectly detected multiple small clusters of tumor pixels in the absence of any actual tumor in the original image.

Testing on New Dataset

Dataset Overview

The images in this dataset (sourced from Kaggle) are made up of MRI scans classified into 4 classes: Glioma, Meningioma, Pituitary, and No Tumor

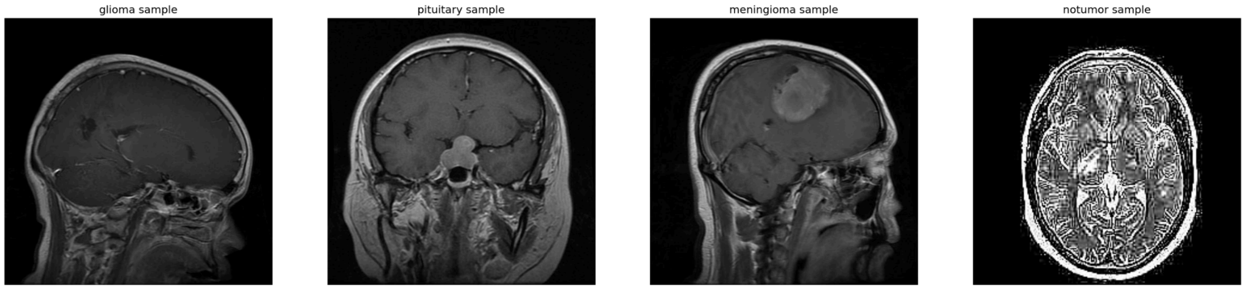
The dataset is made up of 7023.jpg files:

- Grayscale MRI Brain Scan

The only preprocessing done was to standardize the image dimensions as this data was used solely for testing purposes.

The images below *(see Figure 25)* show an instance of a scan for each of the four classes. For the purposes of testing the model performance, we feed in the scans into the trained model to understand how the model performs with files in a different style (grayscale instead of RGB color coded).

Figure 26

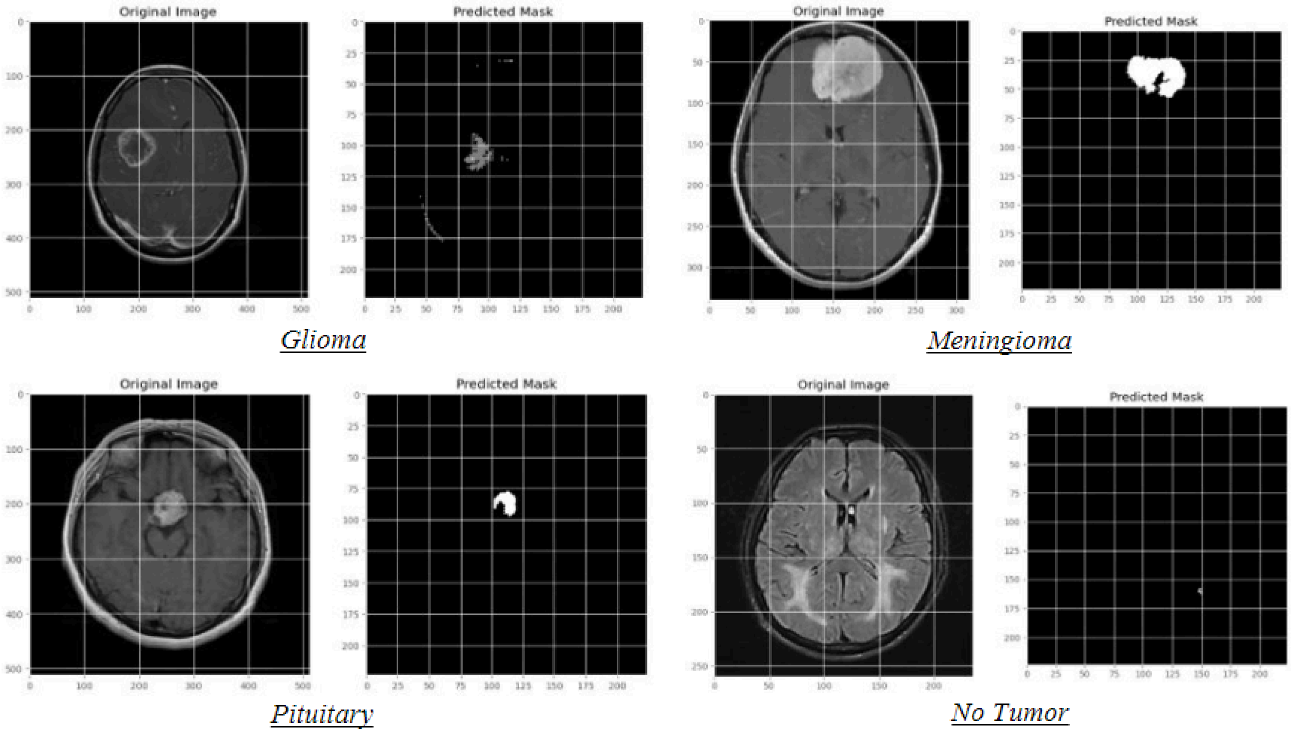


Dataset 2 Images – Labeled Grayscale MRIs

Testing Trained Model

With the model fully trained, we can randomly select images from the dataset to assess the model's overall performance with not only unseen data, but data in a different style. Unlike with the original dataset, these images do not have a ground truth mask, so there is no way to precisely check how accurate the tumor mask generated is. While this makes it challenging to gauge the model's performance, it is still possible to extrapolate the tumor boundaries from the original MRI scan.

Figures 27 - 30



Glioma

Meningioma

Pituitary

No Tumor

U-Net Model Binary Mask Generation on New Data

The images presented above *(see Figures 27 - 30)* demonstrate instances of each labeled class within the dataset, along with the corresponding binary prediction masks generated by the model.

In the top left, an instance of a glioma tumor is shown. While the mask generated by the model effectively captures the tumor's location, it does not accurately reflect its size and shape, likely due to lower confidence in the pixels on the left side of the tumor. Additionally, the model incorrectly classified pixels around the rim of the cerebrum as tumor pixels.

The cases of the meningioma (top right) and pituitary tumor (bottom left) reveal similar performance characteristics. In both instances, the model generates masks that accurately depict the location of the tumors but fail to capture their precise shapes and sizes. Unlike the glioma example however, there are no incorrect classifications of non-tumor pixels in other areas of the image.

Lastly, in the bottom right, an instance of an MRI scan with no tumor is shown. The model largely performs correctly by not identifying a tumor in the majority of the scan, however, a small clustering of pixels is inaccurately classified as tumor tissue in the lower right quadrant (x-axis: ~150, y-axis: ~160).

Gauging Overall Performance

These observations from both the original and new dataset underscore the need for significant enhancements to the model. Although the current model achieves a test F1 Score of 0.8498, the images reveal significant areas of underperformance, demonstrating the model's tendency to incorrectly classify entire clusters of tumor pixels in its binary predictions.

In clinical practice, minimizing false positives and false negatives is essential, particularly when detecting potentially malignant tumors. False positives can place patients in precarious situations, leading to heightened stress and financial burdens from an unnecessary treatment plan. Additionally, they may result in unnecessary diagnostic procedures (such as biopsies or further imaging) which can be invasive and carry their own associated risks, while at the same time not improving the patient’s condition (that they originally came to the hospital for). Conversely, false negatives pose an even greater threat to patient safety, where failing to identify a tumor can jeopardize a patient's life and well being. Especially in the case of malignant tumors, any delays in accurate diagnosis can significantly shorten expected survival times and accelerate the deterioration of a patient’s health.

For radiologists interpreting MRI scans, accurately identifying the presence of a tumor is a critical first step in determining the appropriate treatment plan, and as such, improving the model's performance (particularly in minimizing false positives/ negatives) in tumor detection is imperative to enhance patient outcomes and support clinical decision making.

Conclusion

Utility of Model in Clinical Practice

In clinical practice, technologies such as the model outlined in this paper can provide significant utility across three primary areas:

Reduced Human Error

One of the key applications of AI and machine learning in medicine is the reduction of human error, as the subjectivity and variation in a physician's experience can greatly influence the diagnosis that they make. While it may be relatively straightforward to identify larger tumors in an MRI, smaller or early stage tumors pose a greater challenge, increasing the likelihood that they may be overlooked. Factors such as the parameters set by the MRI technician and the radiologist's expertise introduce additional opportunities for human error in the diagnosis step. Therefore, by employing image processing technologies (like the model outlined in this paper), the variability in human performance can be standardized, mitigating these potential errors.

Time Savings

Another significant benefit of AI and machine learning in medical diagnostics is the time savings they can facilitate. Following an MRI scan, there is often a lengthy wait for diagnosis due to data processing pipelines and the availability of radiologists to make the diagnosis. To bypass these waiting periods, workflows can be established to automatically feed MRI slices into the model immediately after scanning, minimizing delays.

The total processing and diagnosis time can be exacerbated when multiple patients are involved, as a single MRI scan can consist of anywhere from 10 to over 100 slices, depending on the parameters (FOV, Slice Thickness, etc.) set by the technician *(MRI Slice Thickness, n.d.).* For small tumors, radiologists may require a significant amount of time to thoroughly examine each slice, further extending the diagnosis time frame. The proposed model however, can be used to significantly reduce processing and diagnosis time by filtering scans for confirmation based on established confidence thresholds.

For instance, if the model processes a 100 slice MRI scan and identifies tumors in slices 63, 64, and 65 with a confidence threshold of 98%, it can inform the radiologist of a tumor presence with a high level of confidence. Following this, instead of scrutinizing all 100 slices, the radiologist can focus on the identified slices, review the generated mask, and confirm the diagnosis. Conversely, if the confidence is below the threshold, the radiologist can then examine additional slices along with the identified ones to arrive at a conclusion.

Complementary Analysis

The most promising application of these image processing models lies in their capacity for complementary analysis, enhancing and validating diagnoses by cross referencing the doctor's findings with those produced by the model.

For instance, if a 100-slice MRI scan is processed and masks are generated for slices 63, 64, and 65 at a 98% confidence threshold, but the radiologist (on their own) concludes that there is no tumor, this discrepancy can prompt a re-evaluation of the indicated slices. By providing both probabilistic and binary masks, the model can enable the radiologist to visually assess the area where a tumor is suspected, potentially revealing overlooked tumors. This approach not only helps reduce human error but also promotes a data driven diagnostic process, adding an additional layer of validation prior to finalizing the patient's diagnosis.

Takeaways of Model

After comparing the model's performance metrics and visualizing the masks it generates given an MRI scan, it is clear that there is a lot of room for improvement.

The most prominent issue identified pertains to the overall performance in mask generation. While the masks generally indicate the location of the tumor accurately, they fall short in capturing the precise shape and size. This issue is exacerbated by the relatively low confidence threshold of 0.5 used for binary predictions. Under this framework, pixel values greater than 0.5 are classified as tumor pixels, however, even with this low threshold, the model struggled to produce accurate binary masks. For clinical applications, this threshold should be set significantly higher to the upper 90th percentiles to ensure that all model diagnoses are made at a sufficiently high confidence level. To achieve this, the architecture and parameters used to train the model would need to be modified. As observable in the metric charts, the model's performance began to plateau at around epoch 65, which indicates that under the current structure, continuing the training process would result in negligible improvements. To improve the model's performance, modifications to the parameters such as batch size, number of filters, regularization techniques, etc., as well as changes to the model architecture/ number of layers would be necessary. By iteratively improving the structure of the model and comparing performance across each improved version, the model's performance would improve and would be able to function at a high confidence interval.

Additionally, there appears to be challenges related to the model's processing of grayscale images. When comparing the masks generated for the original RGB dataset versus the second grayscale dataset, a decline in the quality of the generated masks is observable, where the model is less effective at capturing the precise shape and size of tumors in the grayscale scans. This suggests that the model may require a different processing approach for grayscale images. During the testing phase with the second dataset, preprocessing was necessary to process the grayscale images into three RGB channels. While this approach facilitated mask generation, it is not the optimal method for processing grayscale MRI scans. Instead, the model should be retrained using grayscale channels directly. By doing so, the model would be able to learn from lower feature (channel) dimensions, enhancing its ability to generalize across both RGB and grayscale images.

Future Steps

In addition to the improvements suggested in the previous section, there are several additional projects that could be considered for future development.

Layered Tumor Classification

While the current model effectively generates masks and identifies the presence of a tumor, this process represents only one aspect of the many considerations used to determine a patient's treatment options and care pathway. A critical factor in this process is the classification of the tumor type. As previously discussed, meningiomas are among the most prevalent intracranial tumors, however, gliomas exhibit the highest malignancy rates and are generally more aggressive. Accurately determining the type of tumor, alongside its size and location, is essential for formulating the most effective treatment plan for patients. Therefore, implementing an image classification model as an additional component of the diagnostic pipeline would be beneficial. In this approach, an MRI scan identified as containing a tumor would be directly fed into the classification model, which would then process the scan to diagnose the specific type of tumor present based on its attributes.

Tumor Image Reconstruction

In the current U-Net model, the network processes MRI scans slice by slice, generating a mask that visualizes the size, shape, and location of the tumor. While this method is effective for identifying the general attributes of the tumor, it operates solely within a two dimensional framework. An interesting next step would be to reconstruct the tumor in a three dimensional space, allowing for a more comprehensive understanding of its characteristics.

Each MRI scan slice maintains a consistent thickness, as determined by the MRI technician or radiologist. By counting the number of consecutive slices that generate a mask, it is possible to obtain a close approximation of the tumor's height. Additionally, by stacking the generated masks and rendering all background pixels transparent, we can extract a clearer visualization of the tumor's location in a three dimensional space. With x and y coordinates derived from each slice, stacking the masks would enable the reconstruction of the z coordinate using the slice thickness.This approach would facilitate the visualization of whether the tumor is merely increasing in size or extending into specific areas or directions, another key consideration in treatment planning.

Final Remarks

In conclusion, this study demonstrates the efficacy of a U-Net convolutional neural network in accurately segmenting MRI brain scans to generate precise tumor masks. By leveraging the attributes of a U-Net architecture we achieved strong segmentation performance, as validated by the evaluation metrics (Dice Coefficient of 0.7771 and F1 Score of 0.8498). These results underscore the potential of deep learning techniques in medical imaging, offering tools that can enhance diagnostic accuracy and reduce the time needed to carry out various tasks.

While this approach yielded promising outcomes, there are considerable limitations/ challenges to acknowledge. To begin with, the model's performance can vary depending on the type of tumor observed and image quality, highlighting the need for further validation across diverse datasets. Future research could incorporate data from multiple other sources during the training process to improve segmentation robustness and generalization capabilities. Additionally, while the model performed well, the quality of the tumor masks generated remains insufficient for clinical deployment, necessitating adjustments to the model's architecture and processing steps to further improve the model's performance.

Ultimately, the findings of this study serve as a proof of concept for advanced machine learning applications in clinical practice, promising not only to assist radiologists but also to contribute to personalized treatment plans that improve the likelihood of positive patient outcomes.

Appendix

Resources

Advanced layer types – Introduction to deep-learning. (2014). Embl-Community.io; The Carpentries Incubator - Introduction to deep-learning. https://grp-bio-it-workshops.embl-community.io/deep-learning-intro/04-advanced-layer-types/index.html

Akkus, Z., Galimzianova, A., Hoogi, A., Rubin, D. L., & Erickson, B. J. (2017). Deep Learning for Brain MRI Segmentation: State of the Art and Future Directions. *Journal of Digital Imaging*, *30*(4), 449–459. https://doi.org/10.1007/s10278-017-9983-4

Alruwaili, A. A., & De Jesus, O. (2020). *Meningioma*. PubMed; StatPearls Publishing. https://www.ncbi.nlm.nih.gov/books/NBK560538/

Bhattacharjee, S., Prakash, D., Kim, C.-H., Kim, H.-C., & Choi, H.-K. (2022). Texture, Morphology, and Statistical Analysis to Differentiate Primary Brain Tumors on Two-Dimensional Magnetic Resonance Imaging Scans Using Artificial Intelligence Techniques. *Healthcare Informatics Research*, *28*(1), 46–57. https://doi.org/10.4258/hir.2022.28.1.46

Brain Tumor Statistics: Key Insights | End Brain Cancer. (2023, March 18). Endbraincancer.org. https://endbraincancer.org/brain-tumor-statistics-ebci/

Cancer of the Brain and Other Nervous System - Cancer Stat Facts. (2018). SEER. https://seer.cancer.gov/statfacts/html/brain.html

Convolution Neural Network Better Understanding. (2024). Vrogue.co. https://www.vrogue.co/post/convolution-neural-network-better-understanding

Efimov, V. (2023, December 30). *Understanding Deep Learning Optimizers: Momentum, AdaGrad, RMSProp & Adam*. Medium. https://towardsdatascience.com/understanding-deep-learning-optimizers-momentum-adagrad-rmsprop-adam-e311e377e9c2

Gavrikov, P. (2022, April 13). *visualkeras for Keras / TensorFlow*. GitHub. https://github.com/paulgavrikov/visualkeras

GeeksforGeeks. (2020, February 26). VGG-16 | CNN model. GeeksforGeeks. https://www.geeksforgeeks.org/vgg-16-cnn-model/

Keras Conv2D and Convolutional Layers. (2018, December 31). PyImageSearch. https://pyimagesearch.com/2018/12/31/keras-conv2d-and-convolutional-layers/

Key Statistics for Brain and Spinal Cord Tumors. (n.d.). Www.cancer.org. https://www.cancer.org/cancer/types/brain-spinal-cord-tumors-adults/about/key-statistics.html

Kumar, A. (2023, March 30). *Transposed Convolution vs Convolution Layer: Examples - Analytics Yogi*. Analytics Yogi. https://vitalflux.com/transposed-convolution-vs-convolution-layer-examples/

Mayo Clinic. (2023, April 21). *Brain tumor - Symptoms and causes*. Mayo Clinic. https://www.mayoclinic.org/diseases-conditions/brain-tumor/symptoms-causes/syc-20350084

Mayo Clinic. (2019). *Glioma - Symptoms and causes*. Mayo Clinic. https://www.mayoclinic.org/diseases-conditions/glioma/symptoms-causes/syc-20350251

Mayo Clinic. (2024, March 29). *Meningioma - Symptoms and causes*. Mayo Clinic. https://www.mayoclinic.org/diseases-conditions/meningioma/symptoms-causes/syc-20355643

Mayo Clinic. (2019). *Pituitary tumors - Symptoms and causes*. Mayo Clinic. https://www.mayoclinic.org/diseases-conditions/pituitary-tumors/symptoms-causes/syc-20350548

MRI Slice Thickness | Slice Thickness in MRI. (n.d.). Mrimaster. https://mrimaster.com/mri-slice-thickness/

Multilayer Perceptrons vs CNN. (2021, January 16). OpenGenus IQ: Computing Expertise & Legacy. https://iq.opengenus.org/multilayer-perceptrons-vs-cnn/

Pituitary Tumors. (n.d.). Yale Medicine. https://www.yalemedicine.org/conditions/pituitary-tumors

Ratan, P. (2020, October 28). *Convolutional Neural Network Made Easy for Data Scientists*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/10/what-is-the-convolutional-neural-network-architecture/

Rosebrock, A. (2021, May 5). *Gradient Descent Algorithms and Variations - PyImageSearch*. PyImageSearch. https://pyimagesearch.com/2021/05/05/gradient-descent-algorithms-and-variations/

Verma, Y. (2021, September 4). *Guide to Different Padding Methods for CNN Models*. AIM. https://analyticsindiamag.com/developers-corner/guide-to-different-padding-methods-for-cnn-models/

Wei, K. (2020, July 29). *Understand Transposed Convolutions - Towards Data Science*. Medium; Towards Data Science. https://towardsdatascience.com/understand-transposed-convolutions-and-build-your-own-transposed-convolution-layer-from-scratch-4f5d97b2967

Wen, P. Y., & Kesari, S. (2008). Malignant Gliomas in Adults. *New England Journal of Medicine*, *359*(5), 492–507. https://doi.org/10.1056/nejmra0708126

What is Batch Normalization In Deep Learning? (2024, May 10). GeeksforGeeks. https://www.geeksforgeeks.org/what-is-batch-normalization-in-deep-learning/

What is cancer pathology? (2019, October 22). Cancer Treatment Centers of America. https://www.cancercenter.com/cancer-types/brain-cancer/diagnosis-and-detection

Zhang, H., Lian, Q., Zhao, J., Wang, Y., Yang, Y., & Feng, S. (2022). RatUNet: residual U-Net based on attention mechanism for image denoising. PeerJ Computer Science, 8, e970. https://doi.org/10.7717/peerj-cs.970

Zou, K. H., Warfield, S. K., Bharatha, A., Tempany, C. M. C., Kaus, M. R., Haker, S. J., Wells, W. M., Jolesz, F. A., & Kikinis, R. (2004). Statistical Validation of Image Segmentation Quality Based on a Spatial Overlap Index. *Academic Radiology*, *11*(2), 178–189. https://doi.org/10.1016/S1076-6332(03)00671-8