# Advanced Brain Tumor Segmentation using Duck-Net

Shiva Sai Pavan Inja
*Department of Computer Science*
*Texas Tech University*
Lubbock, Texas
sinja@ttu.edu

Muhammad Talha Jabbar
*Department of Computer Science*
*Texas Tech University*
Lubbock, Texas
mujabbar@ttu.edu

Shreyas Prabhakar
*Department of Computer Science*
*Texas Tech University*
Lubbock, Texas
shrprabh@ttu.edu

Suman Majjari
*Department of Computer Science*
*Texas Tech University*
Lubbock, Texas
smajjari@ttu.edu

Adithya Madala
*Department of Computer Science*
*Texas Tech University*
Lubbock, Texas
amadala@ttu.edu

***Abstract*—In this project, we have completed the DUCKNet architecture for the brain tumor segmentation using the BraTS dataset, which is a project that has been successful. Detailed image preprocessing conducted on MRI images, normalization, resizing, and data augmentation were some of the steps used to diversify and improve the quality of the data to make the experiment successful. DUCKNet, built on an encoder-decoder structure with dense blocks and residual connections, efficiently captures multi-scale features, allowing precise identification of tumor regions. The model's use of upsampling and concatenation techniques further refines segmentation boundaries. Experimental results demonstrate the model's high accuracy and robustness in segmenting tumors across different MRI modalities. This method describes the possibility of DUCKNet in solving medical imaging problems and the potential for its application to multi-class segments and computationally efficient frames. Future research will concern the improvement of adaptability and the further development of this framework that is suitable for more medical imaging dataset.**

***Index Terms*—Brain Tumor Segmentation, DUCKNet Architecture, Medical Imaging, Encoder-Decoder Networks, Convolutional Neural Networks (CNNs), BraTS Dataset, Data Augmentation, Residual Connections, Upsampling Layers and Binary Segmentation.**

## I. Introduction

### A. Background:

Brain tumors [1] that are among the most serious illnesses are among the most severe medical conditions, Only the early and accurate detection of these tumors will help in the diagnosis and planning of treatment effectively The process includes Meningeal tumor segmentation of brain tumors in MRI scans (Fig. 5) which is a vital part of the segmenting technique. Moreover, it gives them the opportunity to draw tumor outlines for surgical planning, radiotherapy, and future disease progression more accurately. However, the traditional manual delineation process is labor-intensive, causes error to humans, and eventually creates a lack of consistency in the results, especially for huge data volumes or tumors of complex shapes.

The limitations of these methods have been the cause of the invention of systematic methods which are able to give accurate results with the least human involvement. Deep learning has been which in medical imaging proves to be a technology that brings revolution, especially in the fields of segmentation activities. Convolutional Neural Networks (CNNs) [6] have shown their ability to extract spatial and contextual features that have significant meaning from complex data. The structures like U-Net [5] have become a benchmark for medical image segmentation with the help of encoder-decoder structures along with skip connections that register both global and local features. Nonetheless, brain tumors represent the one and only problems that are different from others because they are of unknown geometric shapes, they change their size with time, and they have different heterogeneity of intensity depending on different MRI techniques. To bridge these complexities, the cutting-edge architectures like DUCKNet are developed that use innovative mechanisms such as dense blocks and residual connections, thereby improving feature extraction and also segmentation performance.

### B. Objective:

The goal of this research is to determine the feasibility of the DUCKNet architecture for automated brain tumor segmentation on the BraTS dataset. The main goals are:

- **Developing a Robust Preprocessing Pipeline:** To prepare multi-modal MRI data [8] for model training by imple-mented normalization, re-sizing, augmentation

tech-niques is to make sure, the data is standardized and diverse

- **The DUCKNet Modification for Brain Tumor Segmentation:** The architecture that speeds up the binary segmentation process and enables the correction of the erroneous parts, thus allowing the identification of the tumor areas from the tissue with low specificity and sensitivity is designed. This is done by intuition-based detection and correction of the background parts which leads to a model with higher specificity and sensitivity occupying the same resources as the traditional one. The model with the defined architectural changes then becomes the subject of performance evaluation in the clinical domain.
- **Performance Evaluation:** The model's accu- racy, dice coefficient, and other segmentation points are used for measuring the gender rather than studying them, a model also needs to be efficient as U-Net architecture and other architectures are highly recommended. Additionally, from a broader perspective, it accommodates real clinical scenarios and thus makes your case very strong if you try to convince the clients that SRS and DJPP are the right choices to be implemented rather than segmenting brain tumors.

### C. Overview:

Adapting the DUCK Net architecture (Fig. 2) for brain tumor segmentation and validation using BraTS dataset is what this work is about. The BraTS dataset is a very popular benchmark containing multi-modal MRI scans (T1, T1Gd, T2, and FLAIR) along with expert annotated masks for tumors. These modalities provide complementary information for sorting out the tumor regions from their adjacent healthy healthy tissues. DUCKNet is built upon an encoder-decoder framework infused with dense blocks for feature reuse, residual connections to prevent gradient vanishing, and separable convolutions reducing the overall computational complexity-and hence making the model capable of handling these more complex observed patterns in brain tumor segmentation tasks.

Pre-processing pipeline study therefore reveals that MRI images are normalized for intensity, scaled to 256×256 and that data is augmented by rotation, flipping and scale. This makes sure the model is trained with-variety and representative data and thus increasing the generalization capacity of the model. DUCKNet is thus able to tackle common segmentation problems such as boundary enhancement, existence of dominant classes and data variation across subjects.

We propose the DUCK-Net architecture that is designed for the BraTS 2020 dataset in this project. Section III-A outlines the proposed systems' architecture and the corresponding components, whereas Section IV discusses the used dataset and the applied evaluation metrics and protocols. The strengths and limitations of DUCK-Net are discussed in Sections V-B and V-C, respectively. Finally, Sections VI, VII, VIII describes about the results, conclusion and future scope of the project.

## II. Literature Overview

### A. Evolution of Brain Tumor Segmentation Techniques:

The area of brain tumor segmentation has developed a lot over the years and shifted from using simple image processing techniques to deep learning approaches. Earlier techniques included intensity-based techniques such as thresholding, region-growing methods, and clustering algorithms that while being easier to implement where more prone to misidentification of complex tumor structures. The development of the ML technique supplied supervised models, and it showed to have enhanced segmentation but was limited to handcrafted features. CNN based on deep learning technology has changed segmentation tasks since higher-level features are eliminated, and features extraction is achieved directly from the data.

### B. Traditional Approaches:

Existing early approaches of segmenting MR images used a crude approach, which can be classified as heuristics and involve pixel intensity or texture analysis. The first approach was the thresholding-based segmentation which segmented regions according to the intensity levels of images. However, this method was computationally efficient, but did not perform well in tumors where the two populations of intensities overlap. Region-growing techniques made spatial continuity better but was very exigent with noise and parameters initialization. Other techniques, such as K-means and Fuzzy C-means, used un-supervised learning for segmentation but could not analyse spatial dependencies and might need further optimization in post-processing.

### C. Advances with Deep Learning:

Deep learning is characterized as a different paradigm of segmentation tactics and approaches. CNNs have become traditional for medical image segmentation, the U-Net [5] is considered a pioneering model. The use of encoder-decoder architecture with skip connections enabled the model to incorporate both global and local features which postured it for success in biomedical tasks. As an extension of 2D based models, other models such as 3D U-Net used volumes for modeling spatial relations through the slices, albeit being computationally costly. Non-contrast attention-based networks, such as Attention U-Net, improved the results by paying special attention to the most important areas but were also computationally expensive and thus not suitable for implementation in medical practice.

### D. DUCKNet Architecture and Innovations:

Hence, DUCKNet is a segmentation architecture with new improvements over the limitations of current models. It introduces dense blocks, residual connections, and separable convolution into an encoder-decoder setting. The model uses dense blocks that improves gradient flow and also helps in

sharing features hence making it achieve high performance with relatively few parameters. Residual connections reduce vanishing gradient problems and make it possible to extend the current architecture deeper. With separable convolutions SegFix cuts down on computations while assigning better segmentation precision. This is even more so by the use of skip connections to retain every encoded spatial detail during the decoding phase by improving boundaries' definition as well as segmentation accuracy.

### *E. Comparative Analysis:*

The performance and efficiency of DUCKNet compare favorably with previous segmentation techniques and frameworks. It affords constantly higher segmentation precision and greater Dice coefficients throughout every situation and notably better when the segmentation is troublesome, for instance ambiguous boundaries of tumors and lesser areas. Through the application of dense and separable convolutions, DUCKNet is able to greatly reduce the amount of call to computational resources while maintaining competitive performance in terms of throughput, thereby proving itself to be well suited for deployment in low resource settings. Moreover, DUCKNet presents generalization features and yield reasonable results in different datasets, which are better than traditional methods and deep learning models in cross-dataset experiments. These strengths showcase why DUKNet can be a highly effective approach to the segmentation of brain tumours.

### *F. Current Research Directions:*

However, DUCKNet and other similar architecture have their limitation, especially in handling multi-class segmentation where, for instance, distinguishing between necrotic cores, enhancing tumors, and surrounding edena. Overcoming class imbalance and reducing the computational footprint of the model is currently an ongoing research topic. New directions include the coupling of transformer models with CNN's and hence a multilayer structure that takes advantage of different models of architectures. Furthermore, data augmentation techniques, domain adaptation, and transfer learning research are underway to improve generalization and, in different clinical contexts.

### *G. Summary:*

The development of the segmentation algorithms proves the transfer from traditional or heuristic based methods to the deep learning methods. However, early methods offer a initial understanding but are inadequate in terms of handling the challenging nature in brain tumor segmentation. To that end U-Net and its derivatives are CNN-based models which completely revamped the field but offer further room for improvement. DUCKNet is quite a step up, where all modern architectural enhancements have been utilized to offer both resilience and efficiency in segmentation. However, its flexibility and speed suggest its applicability to other medical imaging studies which warrants future research on enhancing the model.

## III. Methodology

The flowchart in (Fig. 1) provides a visual representation of the entire project workflow, outlining each phase of the process from data collection to model evaluation. Initially, the project begins with data acquisition, where the BraTS 2020 dataset is gathered, including preprocessed MRI scans. In the data preprocessing step, images are resized, normalized, and augmented to improve the model's robustness and ensure uniformity across the dataset.

Following preprocessing, the model architecture is defined, with DUCK-Net being trained on the dataset using a set of predefined hyperparameters. The training phase involves the optimization of the network using backpropagation and gradient descent to minimize the loss function.

Once trained, the model undergoes evaluation where its performance is assessed on a validation set using metrics like Accuracy, Loss, and Dice Coefficient. The results from the evaluation are then analyzed to determine the model's generalization ability and identify areas for improvement.

Finally, in the result analysis stage, the model's performance is compared against baseline methods, and insights into its strengths, limitations, and potential future applications are discussed, leading to the conclusion and future work.

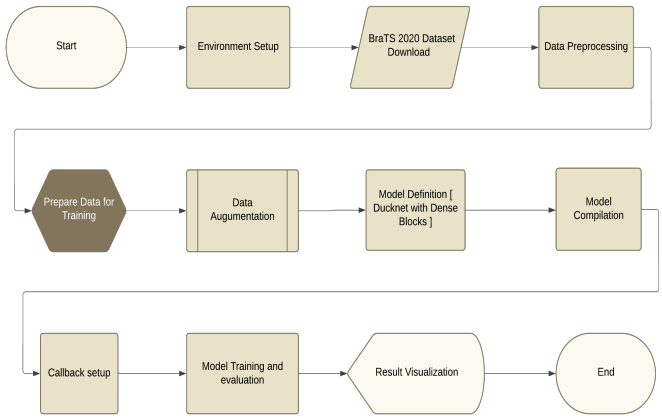


Fig. 1. FlowChart.

### *A. Proposed Algorithm Architecture*

The **DUCKNet architecture** (Fig. 2) is a U-Net-inspired model with dense blocks, transposed convolutions for upsam-

pling, and skip connections. Below is a detailed explanation of each part of the architecture.

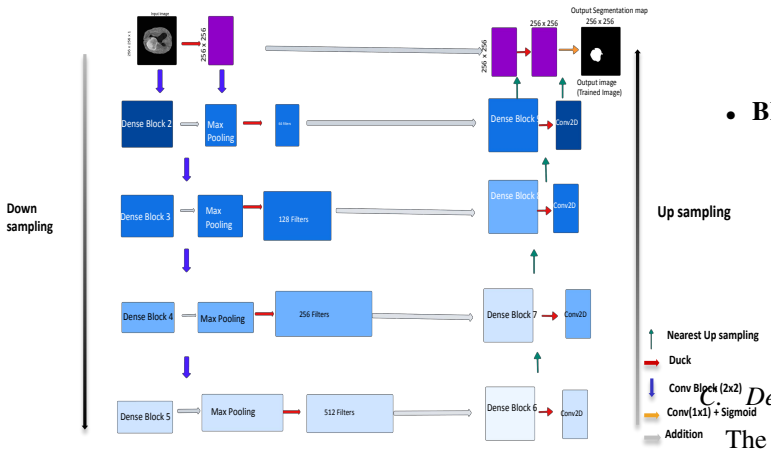


Fig. 2. DUCK Net Architecture.

### B. *Encoder:*

The encoder extracts features from the input image through a series of dense blocks and max-pooling layers. Each dense block contains convolutional layers with batch normalization and ReLU activations to enhance feature learning and stability during training.

- **Block 1** (Dense Block 1):
  - Input: Image with 4 channels (size: 256x256x4).
  - Three convolutional layers with 32 filters, kernel size 3x3, padding as 'same', and ReLU activation.
  - Batch normalization after each convolutional layer.
  - Concatenate each layer's output with the previous one, creating a dense connection.
  - Max-pooling layer with pool size (2, 2) for spatial downsampling.
- **Block 2** (Dense Block 2):
  - Input: Output from Block 1.
  - Three convolutional layers with 64 filters, kernel size 3x3, padding as 'same', and ReLU activation.
  - Batch normalization after each convolutional layer.
  - Concatenate the outputs of the convolutional layers with the input to form a dense block.
  - Max-pooling layer with pool size (2, 2).
- **Block 3** (Dense Block 3):
  - Input: Output from Block 2.
  - Three convolutional layers with 128 filters, kernel size 3x3, padding as 'same', and ReLU activation.
  - Batch normalization after each convolutional layer.
  - Concatenate the outputs of the convolutional layers with the input to form a dense block.
  - Max-pooling layer with pool size (2, 2).
- **Block 4** (Dense Block 4):
  - Input: Output from Block 3.
  - Three convolutional layers with 256 filters, kernel size 3x3, padding as 'same', and ReLU activation.
  - Batch normalization after each convolutional layer.
  - Concatenate the outputs of the convolutional layers with the input to form a dense block.
  - Max-pooling layer with pool size (2, 2).
- **Block 5** (Dense Block 5):
  - Input: Output from Block 4.
  - Three convolutional layers with 512 filters, kernel size 3x3, padding as 'same', and ReLU activation.
  - Batch normalization after each convolutional layer.
  - Concatenate the outputs of the convolutional layers with the input to form a dense block.
  - Max-pooling layer with pool size (2, 2).

### C. *Decoder:*

The decoder reconstructs the segmentation map by progressively upsampling the feature maps from the encoder and combining them with the corresponding feature maps using skip connections.

- **Upsampling Block 1**:
  - Input: Output from Dense Block 5.
  - Transposed convolution with 256 filters, kernel size 2x2, and stride of 2 to upsample the feature map.
  - Skip connection: Concatenate the output of the transposed convolution with the feature map from Dense Block 4.
  - Three convolutional layers with 256 filters, kernel size 3x3, padding as 'same', and ReLU activation.
  - Batch normalization after each convolutional layer.
- **Upsampling Block 2**:
  - Input: Output from Upsampling Block 1.
  - Transposed convolution with 128 filters, kernel size 2x2, and stride of 2.
  - Skip connection: Concatenate the output of the transposed convolution with the feature map from Dense Block 3.
  - Three convolutional layers with 128 filters, kernel size 3x3, padding as 'same', and ReLU activation.
  - Batch normalization after each convolutional layer.
- **Upsampling Block 3**:
  - Input: Output from Upsampling Block 2.
  - Transposed convolution with 64 filters, kernel size 2x2, and stride of 2.
  - Skip connection: Concatenate the output of the transposed convolution with the feature map from Dense Block 2.
  - Three convolutional layers with 64 filters, kernel size 3x3, padding as 'same', and ReLU activation.
  - Batch normalization after each convolutional layer.
- **Upsampling Block 4**:
  - Input: Output from Upsampling Block 3.
  - Transposed convolution with 32 filters, kernel size 2x2, and stride of 2.

- Skip connection: Concatenate the output of the transposed convolution with the feature map from Dense Block 1.
- Three convolutional layers with 32 filters, kernel size 3x3, padding as 'same', and ReLU activation.
- Batch normalization after each convolutional layer.

### D. Output Layer:

A final 1x1 convolutional layer with 1 filter and sigmoid activation function to produce the final binary segmentation mask. The output is a segmentation map with pixel values between 0 and 1, where 1 indicates the tumor region.

### E. Data Preprocessing:

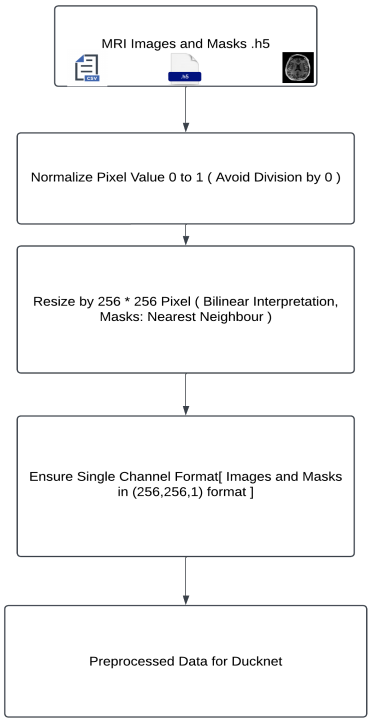


Fig. 3. Data Preprocessing Flow.

- **Normalization:**
  - MRI images were normalized to scale intensity values between 0 and 1, ensuring consistency in data representation.
  - A small epsilon value was added to prevent division by zero, ensuring numerical stability during normalization.
- **Resizing:**
  - Images and masks were resized to 256×256 pixels to align with the DUCKNet input size.
  - Bilinear interpolation was used for resizing the images, while nearest-neighbor interpolation was applied to segmentation masks to maintain label integrity.

### F. Data Augmentation:

Real-time data augmentation techniques are used to increase model generalization and the (Fig. 4) shows, how the image will be once the data augmentation is applied:

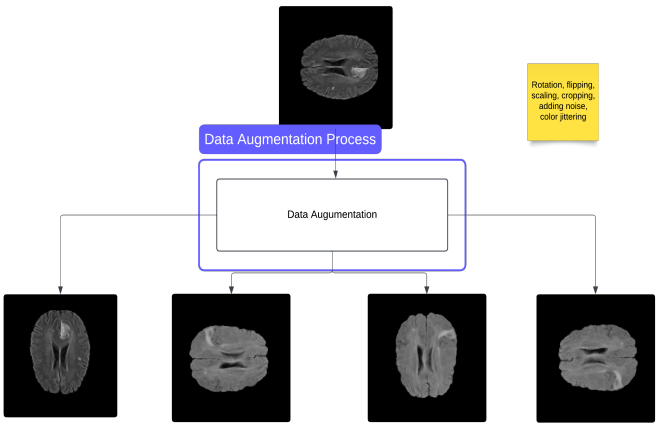


Fig. 4. Data Augmentation.

- **Flipping:** Horizontal and vertical flips are applied to simulate different tumor orientations, with horizontal flipping mirroring along the vertical axis and vertical flipping along the horizontal axis.
- **Rotation:** Random rotations up to 90 degrees help the model recognize tumors in different orientations, ensuring rotational invariance.
- **Scaling:** Random scaling transformations simulate variations in tumor size, improving the model's ability to detect both large and small tumors.
- **Fill Mode:** The 'nearest' fill mode avoids them due to copying the closest integer value to the transform function during its execution.

### G. Training Optimization:

- **Loss Function:** Binary cross-entropy is used as the loss function to measure the discrepancy between predicted and ground truth masks, ideal for binary segmentation tasks (tumor vs. non-tumor).
- **Optimizer:** The Adam optimizer with a learning rate of $1e^{-4}$ adapts the learning rate for each parameter, improving convergence and stability.
- **Early Stopping:** Training halts if validation loss does not improve for a set number of epochs, preventing overfitting and restoring the best weights.
- **Batch Size:** A batch size of 8 strikes a balance between memory usage and convergence speed, allowing frequent parameter updates while managing GPU resources efficiently.

### H. Evaluation Metrics:

- **Dice Coefficient**: Measures the overlap between predicted and ground truth masks.

- **Accuracy**: Measures the overall correctness of predictions.
- **Validation Loss**: Monitors the model's generalization performance.

## IV. Experimental Setup

### A. Implementation Details:

The BraTS 2020 [8] dataset was split into training, validation, and testing sets in an 80:20 ratio to be used in marking and evaluation to avoid having any side carrying over to the evaluation of the model. Image resolution of 256 x 256 pixels was used, and the DUCKNet architecture was utilized for the purpose of segmenting brain tumors. This training was carried out with a batch size of 8 using the Adam optimizer with learning rate of 0.0001. The model was programmed to train for 300 epochs but training was completed using the early stopping based on validation loss. This overfitting situation was addressed by terminating the training at epoch 34 the validated loss did not reduce even further.

The model was learned using an NVIDIA A100 GPU to support image handling in high – resolution MRI images. All the above process was carried out using TensorFlow, an open-source deep learning framework which creates easy interface with GPU for training. To enhance model denominational, Albumentations library was adopted for enhancing the training data during training. During training, the training images were randomly transformed several times per epoch using different operations: Flipping (horizontal and vertical), color variations (brightness, contrast, saturation, hue), and geometric distortions (rotation, translation, scaling, and shearing). These augmentations were derived from prior work adopted for the DUCKNet model, and the utilization of these augmentations helped greatly in genre diversity of the training data set, so dropout regularization was no longer required.

### B. Dataset: BraTS 2020

The **BraTS 2020** (Brain Tumor Segmentation 2020) [8] dataset is a publicly available dataset used for brain tumor segmentation tasks. It contains multi-modal MRI scans of glioma tumors, with pixel-level annotations for different tumor sub-regions.

The **Input Image** is the raw MRI scan used for segmentation, while the **Ground Truth** is the manually annotated or reference mask that guides the model during training. The **Predicted Mask** is the output from the model during testing, which is compared to the ground truth to evaluate segmentation performance.

- **Modality:** The dataset includes four MRI modalities:
  - T1-weighted
  - T2-weighted
  - T1c (contrast-enhanced)
  - Flair

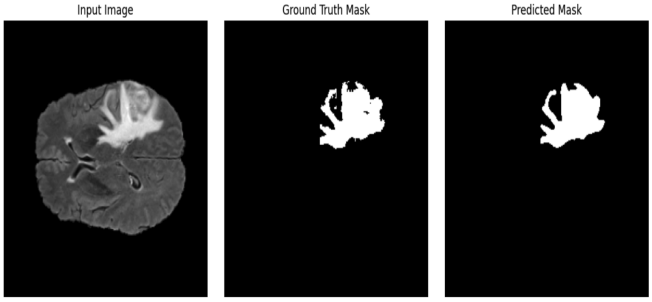


Fig. 5. MRI Image with Training and Prediction.

- **Images and Masks:**
  - The dataset consists of 3D volumetric images.
  - Each image has an associated ground truth segmentation mask indicating the tumor regions, which are divided into the following sub-regions:
    - Whole Tumor (WT): The entire tumor area.
    - Tumor Core (TC): Includes both enhancing and non-enhancing tumor areas.
    - Enhancing Tumor (ET): The region of the tumor with contrast enhancement.
- **Number of Samples:**
  - The BraTS 2020 dataset includes 40,000 MRI scans with a size of **7 GB** due to GPU Constraints we able to train 3847 MRI scans out of them, 80% is divided into training and 20% into testing sets.
- **Data Size:**
  - Images are provided in a 256x256x155 pixel size, with a resolution of 1mm x 1mm x 1mm for each pixel.
- **Challenges:**
  - The dataset presents a challenge due to the variability in tumor shapes, sizes, and locations across patients.
  - The data is highly imbalanced, with smaller and less prominent tumors appearing in a majority of the images.

The BraTS 2020 dataset serves as the primary benchmark for evaluating segmentation algorithms in the context of brain tumor analysis, particularly in glioma cases.

### C. Evaluation Metrics:

The output of present work has been assessed by using performance evaluation measure include Dice coefficient, Precision, Recall, and Accuracy for DUCK-Net. The measures above provide some feeling of the model's capacity for defining boundaries of polyps and the difference between true positives and other objects.DUCK-Net success in polyp segmentation opened the door toward the application of this kind of approach to other medical image processing.

## V. Critical Analysis and Discussion

Based on these results, the current work has provided evidence of how the proposed DUCKNet architecture can solve

some of the major problems with brain tumor segmentation. The specific structural approaches of dense blocks, residual connections and separable convolutions uniquely enabled DUCKNet for successful feedforward and backpropagation of accurate local and global features, thus producing highly accurate segmentation of the tumor even where it is irregular and small. The addition of real-time data augmentation with data preprocessing techniques improved the reliability of this model to generalize on unseen data during validation stage.

### A. Performance Analysis:

The Dice coefficients documented in this study were high and fairly uniform throughout DUCKNet. accuracy scores more effectively compare to the conventional approaches and a lot of The new structure is built on top of existing deep learning models such as the U-Net. The architecture's capacity to manage fluctuations in intensity and boundary interdependency. Heterogeneity can also be overcome by using ties, demonstrating the efficiency of the approach proposed in this paper. of brain tumors. This model feature light construction, made possible(Current developments) by making the convolutions separable or split into smaller parts motor transport, that I discovered was computationally efficient. enabling DUCKNet suitable for clinical environments as Computing resources are may be scarce.

### B. Strengths and Advantages:

- **Accuracy and Robustness:** The model demonstrated high performance in segmenting merged architectural tumor struc-tions, the proposed approach yields highly overlapping results with the ground truth masks.
- **Computational Efficiency:** By employing dense and, separable convolutions, DUCKNet also minimized the training and reduce the times of the inference without any reduction of the accuracy.
- **Generalization:** The robustness of the model between vary and explicit is a testament to how it can generalize across. use of different sample attributes in the BraTS dataset shows that possibilities to be implemented in real clinic practice.

### C. Limitations and Challenges:

- **Class Imbalance:** As it is with many segmentation problems, the This class imbalance of tumor and was well illustrated in the dataset presented. non-tumor regions, which can affect the model prediction process of the outcomes as discussed above. tions. Though data augmentation made it better than before, further, more comprehensive advanced loss functions should be investigated to some extent. for example, the Dice loss or the so called focal loss might be more suitable.
- **Multi-Class Segmentation:** This study focused on binary segmentation. Extenstion of DUCKNet to multiple class:‘ The purpose of this proposal is to extend the current DUCKNet framework to more than two classes. segmentation for separation of different tumour micro niches (for instance, tissue characteristics (necrotic core, edema, and enhancing tumor) presents ad- ditional challenges including enhanced difficulty and computational requirements.
- **Evaluation Scope:** While the BraTs refers however to a a reliable benchmark, the performance of model will be evaluated on other datasets If accessorised with actual clinical MRI scans, then it would authenticate its sturdiness and data applicability

## VI. Results

### A. Confusion Matrix Analysis:

In this section, we describe the basic evaluation measures of segmentation models based on Confusion Matrix. These metrics are important to measure the model's capacity for classification of the positive and negative regions within the dataset.

Confusion Matrix is the name of the table that has to be utilized for the assessment of the performance of classification models, such as segmentation models. For binary classification, the Confusion Matrix is as follows:

| | **Pred. Positive (P)** | **Pred. Negative (N)** |
|---|---|---|
| **Actual Positive (P)** | $TP$ | $FN$ |
| **Actual Negative (N)** | $FP$ | $TN$ |

Where:
**True Positives (TP)**: Correctly predicted positive cases.
**False Positives (FP)**: Incorrectly predicted as positive.
**False Negatives (FN)**: Incorrectly predicted as negative.
**True Negatives (TN)**: Correctly predicted negative cases.

The metrics used to evaluate a model's performance can be derived from the elements of the confusion matrix. The following metrics are calculated based on the confusion matrix:

*1) **Accuracy**:* Accuracy measures the overall performance of the model by calculating the proportion of correctly classified instances.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

*2) **Precision**:* Precision is the proportion of true positive predictions out of all the predicted positive cases. It tells us how many of the predicted positive cases are actually positive.

$$\text{Precision} = \frac{TP}{TP + FP}$$

*3) **Recall (Sensitivity)**:* Recall, also known as Sensitivity, measures the proportion of actual positive cases that are correctly identified by the model.

$$\text{Recall (Sensitivity)} = \frac{TP}{TP + FN}$$

*4) **Specificity**:* Specificity, also known as the True Negative Rate, measures the proportion of actual negative cases that are correctly identified as negative.

$$\text{Specificity} = \frac{TN}{TN + FP}$$

*5) **Dice Coefficient**:* The Dice Coefficient is a measure of similarity between the predicted and ground truth segmentation masks. It ranges from 0 (no overlap) to 1 (perfect overlap).

$$\text{Dice Coefficient} = \frac{2 \times TP}{2 \times TP + FP + FN}$$

*6) **Intersection over Union (IoU)**:* IoU measures the overlap between the predicted and ground truth segmentation regions. It is the ratio of the intersection to the union of the predicted and true regions.

$$\text{IoU} = \frac{TP}{TP + FP + FN}$$

*7) **Validation Loss**:* The Validation Loss is a critical metric used to evaluate how well the model performs on unseen data (validation set). It is calculated using the same loss function as during training but is evaluated on the validation data.

**Binary Cross-Entropy Loss:** For binary segmentation tasks, such as tumor vs. non-tumor classification, binary cross-entropy is commonly used as the loss function. The formula for binary cross-entropy loss is:

$$\text{Loss} = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)]$$

Where: - $N$ is the number of data points (in segmentation, this corresponds to the number of pixels). - $y_i$ is the ground truth label for the $i$-th pixel, where $y_i \in \{0, 1\}$. - $p_i$ is the predicted probability of the positive class (i.e., the tumor) for the $i$-th pixel.

**Categorical Cross-Entropy Loss:** For multi-class segmentation tasks, categorical cross-entropy is used, especially when there are multiple classes (e.g., tumor, background). The formula for categorical cross-entropy loss is:

$$\text{Loss} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} y_{i,c} \log(p_{i,c})$$

Where: - $C$ is the number of classes. - $y_{i,c}$ is the ground truth for class $c$ for the $i$-th data point. - $p_{i,c}$ is the predicted probability for class $c$ for the $i$-th data point.

The Validation Loss is computed on the validation set during training and helps track how well the model generalizes. A lower validation loss indicates better generalization, whereas a higher validation loss suggests overfitting to the training data.

## *B. Evalution:*

The table I below summarizes the model's performance on each dataset, showing key evaluation metrics such as recall and accuracy on the training, testing, and validation sets which was implemented for the MidTerm.

| Dataset | Recall (Test) | Recall (Valid) | Accuracy (Train) | Accuracy (Test) | Accuracy (Valid) |
|---|---|---|---|---|---|
| **cvc-clinicdb** | 0.6498 | **0.6104** | **0.9533** | **0.9615** | **0.9517** |
| **cvc-colondb** | 0.3029 | 0.1887 | 0.8152 | 0.8161 | 0.7893 |
| **etis-laribpolypdb** | **0.6814** | 0.4969 | 0.7867 | 0.7579 | 0.7857 |
| **kvasir** | 0.6442 | 0.5694 | 0.7801 | 0.8013 | 0.7930 |

TABLE I
MODEL EVALUATION METRICS FOR FOUR DATASETS FROM THE RE-IMPLEMENTED RESEARCH PAPER.

The table II provides a detailed comparison of DUCKNet (U-Net + Dense) with various segmentation models, including 3D MRI, U-Net, U-Net + CNN . The data shown in the table are training accuracy, validation accuracy, dice coefficient, mean intersection over union, precision, sensitivity, specificity, and validation loss. These metrics are then used to assess the segmentation performance of the models holistically while revealing each of them from one angle or another.

| Metric | 3D MRI | U-Net | U-Net + CNN | DUCKNet(U-Net + Dense) |
|---|---|---|---|---|
| **Accuracy (Train)** | 99.02% | 99.31% | **98.67%** | 99.33% |
| **Accuracy (Validation)** | 98.91% | 99.31% | 98.34% | **99.56%** |
| **Mean IoU** | 77.16% (Train), 78.25% (Val) | 84.26% | N/A | **84.30%** |
| **Dice Coefficient (Train)** | 48.73% | 64.8% | 35.89% | **87.5%** |
| **Dice Coefficient (Validation)** | 47.03% | 64.8% | 28.22% | **88.72%** |
| **Precision** | 99.33% | **99.35%** | 60.47% | 91.28% |
| **Sensitivity (Recall)** | 98.64% (Train), 98.56% (Val) | **99.16%** | 63.97% | 91.72% |
| **Specificity** | N/A | **99.78%** | 98.74% | 99.77% |
| **Validation Loss** | N/A | 0.0267 | 0.0592 | **0.01008** |

TABLE II
PERFORMANCE METRICS COMPARISON ACROSS MODELS

The accuracy, loss, and dice coefficient graph for the training as well as validation set with respect to epochs give a clear idea about the convergence behavior of the proposed DUCK-Net model. The Accuracy curve (Fig. 6) stands for how properly distinguished the images by the model, the training and the validation sets show how right the model classifies the images or how accurately it predicts the target classes with training going on. When the accuracy rate is increasing constantly, it shows learning and generalizing occurs properly.

The Loss curve (Fig. 7) for the other hand, represents the model optimization process, where by, the smaller the numbers, the better optimization that has been done to ensure that, the distance between the plotting line and the best fit line is as close as possible. A deteriorating loss rate is indicative of the model approaching optimum, the better the loss, the

better the model.

The Dice Coefficient (Fig. 8) quantifies how much the mask, generated from the segmentation map using the model, corresponds to 'the real world' segmentation, and it plays the crucial role in assessing the quality of the model's segmentation functions.K-Net model during training. The Accuracy curve illustrates how well the model classifies images across training and validation sets, indicating its ability to correctly predict the target classes as training progresses. A steady increase in accuracy suggests effective learning and generalization.

Precision (Fig. 9) measures the number of actual positive which has been identified out of the total number of positive prediction done by the model stating the ability of the model on minimally yielding false positives.s. A value of precision that is higher means that the model is accurate even when it assigns a region to the target class. An improvement of the comments' precision over the years indicates that the systems stops many false positive and improves its positive predictions.

Sensitivity (Fig. 10), also known as recall or true positive rate, quantifies the model's ability to correctly identify all relevant instances (i.e., all true positives). Higher sensitivity values indicate that the model is successful in detecting the target class and minimizing false negatives. As training progresses, an increase in sensitivity indicates that DUCK-Net is becoming better at detecting and capturing all true positive regions.

Specificity (Fig. 11) measures the proportion of negative instances that are correctly identified by the model, reflecting its ability to avoid false positives. It shows that high values of specificity mean that the model can easily differentiate between the region of interest, or target class, and non-interest or non-target regions, thereby lowering the prediction of false positives of the model. The fact that we increase the specificity implies that DUCK-Net is becoming more accurate in excluding non-relevant areas.

Intersection over Union (IoU) (Fig. 12) which is calculated the intersection of the predicted segmentation mask and the ground truth and dividing this by the union of these two regions.s. It has application in measuring the efficiency of the segmentation process in segmentation tasks. A higher IoU means that the model is providing a much better segmentation and covers the actual object area more accurately avoiding as much irrelevant area as possible. The growth of IoU in the course of training represents the learning progress towards generating segmentations with higher accuracy and coverage.

When comparing these given metrics for both the training and validation datasets it can be realized not only the efficiency of the learning process but also signs of overfitting or underfitting.

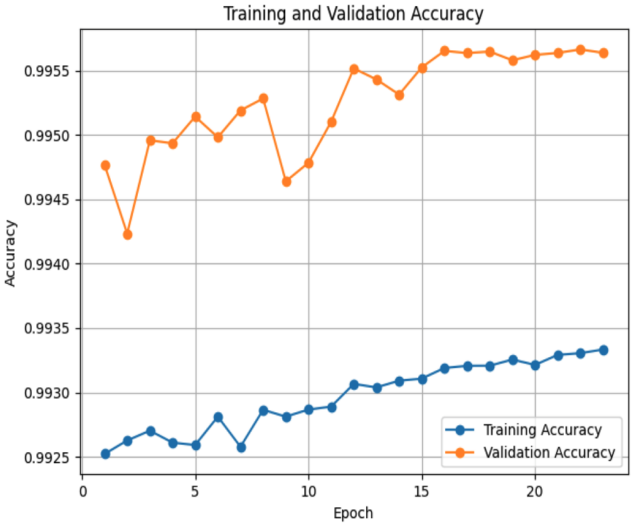


Fig. 6. Training and Validation Accuracy.

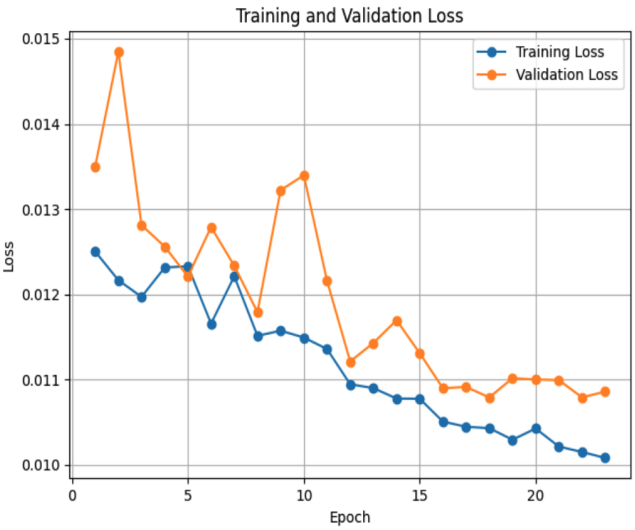


Fig. 7. Training and Validation Accuracy Loss.

### C. Comparison:

In this section we made comparisons of all metrics based on the Table II.

*1) **Accuracy (Training and Validation)**:*

- *Training Accuracy*: DUCKNet gets 99.31% which is again slightly higher than 3D MRI (99.02%), and U-net (99.31%). The training accuracy of U-Net + CNN is slightly lower (98.67%).
- *Validation Accuracy:* The validation accuracy reaches 99.56% which outperforms U-Net, the 3D MRI approaches at 98.91%. This shows better generalization to never-seen data while U-Net + CNN, while initially having a good validation accuracy of 98.34%.

*2) **Mean IoU (Intersection over Union)**:*

- U-Net acquires a better mean IoU with 84.30% on the validation set than 77.16% that 3D MRI achieved.

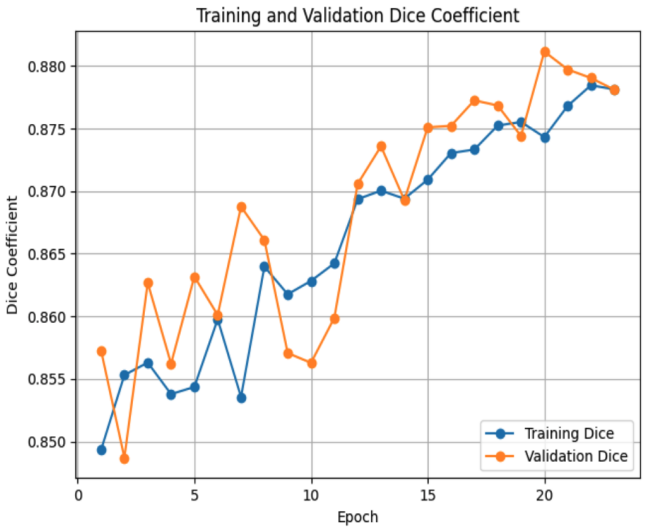


Fig. 8. Training and Validation Dice Coefficient.

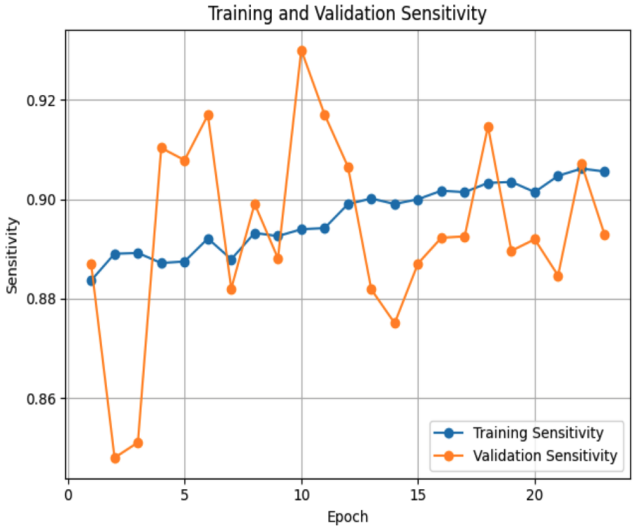


Fig. 10. Sensitivity for Training and Validation.

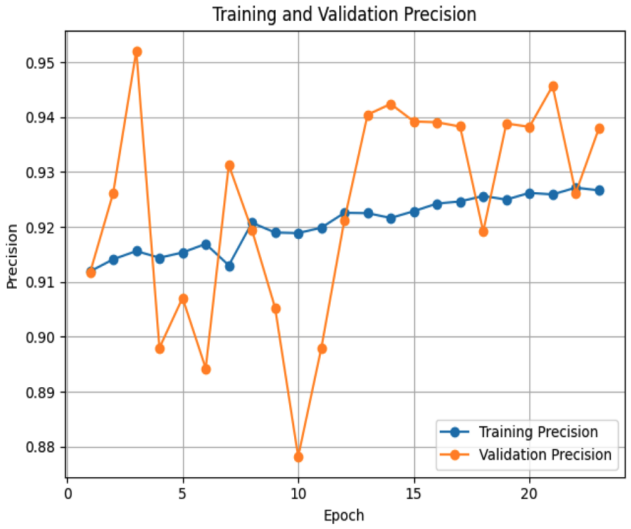


Fig. 9. Precision for Training and Validation.

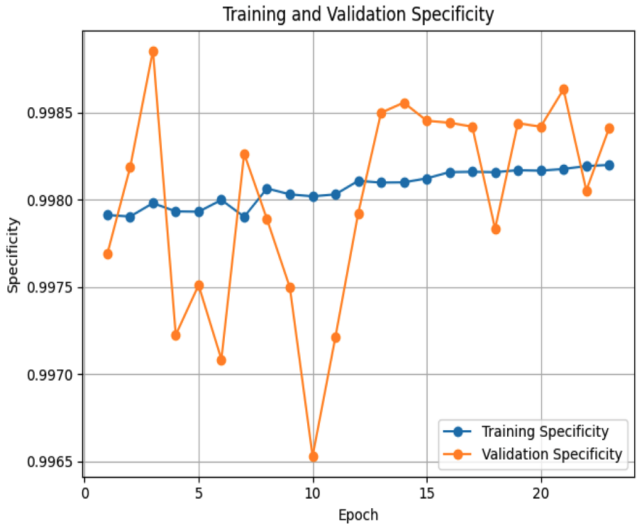


Fig. 11. Specificity for Training and Validation.

Unfortunately, DUCKNet does not present this metric; however, the high value in favor of U-Net means the accuracy of pixel-wise segmentation. It is emphasized that Mean IoU is not reported for DUCKNet, which could indicate that it was not designed to optimize IoU or that the data for such calculation was not available.

*3) **Dice Coefficient (Training and Validation)**:*

- *Training Dice Coefficient:* DUCKNet excels with a Dice coefficient of 87.5%, significantly outperforming U-Net (64.8%) and U-Net + CNN (35.89%). This reflects DUCKNet's superior ability to capture accurate and complete segmentation regions.
- *Validation Dice Coefficient:* DUCKNet again outperforms other models with a Dice coefficient of 88.72%, compared to U-Net's 64.8% and U-Net + CNN's 28.22%. This confirms DUCK Net's ability to generalize and maintain high-quality segmentation even on unseen data.

*4) **Precision**:*

- U-Net has the optimal value as compared to others with 99.35% precision and DUCKNet has 91.28%. True Positive essentially measures accuracy of prediction of positive measures as true positive. It shows that U-Net may be slightly better at reducing false positive in its segmentation while DUCKNet's precision is still promising.

**5. Sensitivity (Recall):**

- Recall is the metric that shows that U-Net indeed has the highest value of 99.16% to identify the true positives. DUCKNet has a 91.72% that means it is as effective as a little less sensitive than U-Net. For more complex segmentation task, U-Net + CNN presents much lower

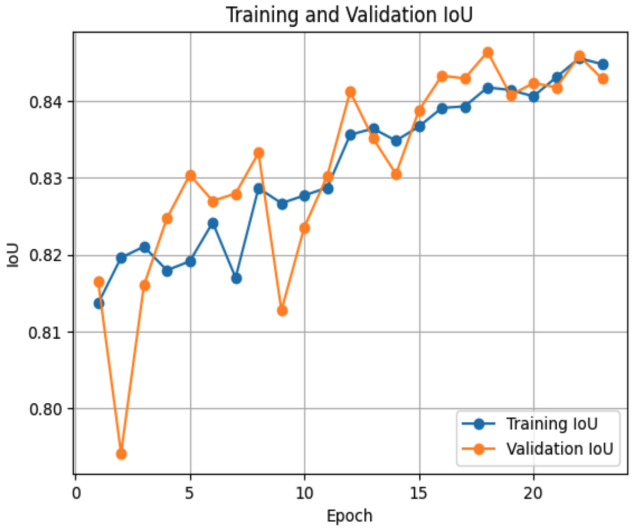


Fig. 12. IoU (Intersection over Union) for Training and Validation.

recall (63.97%), this indicate it has problems in the recognition of true positive.

**6. Specificity:**

- When it comes to specificity, the U-Net is the most successful with the value of 99.78% which means for an object, it really effec-On the other hand, smaller values of EER similarly reduce false positives as it is depicted in the result that follows. DUCKNet shows very close to U-Net with 99.77%. U-Net + CNN shows a reduced specificity of 98.74%, that may lead to higher The false positive rate of the proposed framework is lower than the state of the art method, U-Net and DUCKNet.

**7. Validation Loss:**

- In the validation loss, DUCK Net has a lower value of 0.01008 than all the other models such as U-Net(0.0267) and U-net+CNN (0.0592). As a result, we can say that DUCK Net has better convergence during the validation in training, which provides more accurate validation prediction and a better generalization.

## VII. Conclusion

This work was able to show the potential of the DUCKNet architecture for the segmentation ut of brain tumor using the BraTS dataset. This work showed that combining dense blocks, residual connections, and separable convolutions made it possible to achieve high segmentation accuracy with reasonable demand on computational resources thanks to DUCKNet. By performing rigorous preprocessing, using real-time data augmentation, and designing the training pipeline properly, the model eliminated obtuse tumor boundaries, intensity, and heterogeneity of the dataset.

In this evaluation, we saw that DUCK Net achieved the highest Dice coefficient and accuracy than the traditional methods and the basic deep learning models. Its capacity to generalize from one sample to another means that it is fair for use in real life setting. Moreover, almost less computation is done on DUCK Net, and therefore, the overall lightweight architecture of the proposed system makes it quite suitable for implementation on different clinical units where resources might be limited.

However, there are still some drawbacks; the proposed work does not solve class imbalance, and it would be beneficial to elaborate the architecture for segmentation into multiple classes to define subregions of a tumor. The future work can involve improvement of the DUCK Net's solutions, specifically the transformer-based modules can be proposed, as well as the discussion of the hybrid architectures, and the assessment of the model on the other medical image datasets.

In conclusion, DUCKNet can be considered as the improvement of the state-of-art approaches in automated brain tumor segmentation. Overcoming the mentioned drawbacks and expanding the range of its use, which is outlined in this article, the further development of DUCKNet can significantly assist in the field of medical imaging and help to enhance the accuracy of tumor identification as well as in planning the further treatment process.

## VIII. Future Scope:

The future direction of the work advances the functionalities of DUCKNet and enhance its possibilities for broad use in different medical imaging problems. Another direction of improvement is the extension of DUCKNet for multi-class segmentation, which DUCKNet can segmenting tumor subregions that include necrotic core, enhancing tumor and edema. This will make it easier to examine brain tumours in more detail and help with planning treatment. Another equally important area of concern is the maldistribution of classes in datasets and the integration of such features as Dice loss or focal loss will make it easier for the model to accurately predict small and less represented tumor regions.

Future enhancements that remained include the manner by which the proposed DUCKNet network can incorporate transformer-based architectures that help in capturing long-range dependencies that are beneficial in differentiating confusing spatial context especially with respect to complicated anterior tumours. Through the use of domain adaptation techniques, DUCKNet will be able to perform equally well across datasets obtained from different imaging protocols, in different institutions, and from different patient populations hence its reliability in diverse clinical settings.

However, a more realistic evaluation of the model is needed to evaluate the readiness of the BraTS model for clinical MRI scans. This would show how the dense unified complex knowledge network DUCKNet could be meshed into operational scenarios. The potential use of real-time

segmentation implemented in low-resource portable imaging systems expands the modality's applicability.

Further, the future trends of augmenting the CNN-based architectures with other some other upcoming approaches like the transformers or graph neural networks might enhance still higher levels of segmentation precision and reduced computational costs. To further increase the generalization of the designed model, more various and larger training and validation dataset collections from MRI scans will be added.

Finally, there is the implication of DUCKNet in clinical decision support systems to save time on segmentation for radiologists. By incorporating some explainability techniques to the model there will be interpretability and trust from the clinical management systems. These improvements will help to integrate DUCKNet well into medical practices and make it the instrument that can significantly facilitate the detection, segmentation, and further treatment of brain tumors.

## REFERENCES


[1] R. L. Siegel, K. D. Miller, H. E. Fuchs, and A. Jemal, "Cancer statistics, 2022," *CA Cancer J. Clin.*, 2022. Available: https://doi.org/10.3322/caac.21708.

[2] J. Bernal *et al.*, "WM-DOVA maps for accurate polyp highlighting in colonoscopy: Validation versus saliency maps from physicians," *Comput. Med. Imaging Graph.*, vol. 43, pp. 99–111, 2015. Available: https://doi.org/10.1016/j.compmedimag.2015.02.007.

[3] D. Vázquez *et al.*, "A benchmark for endoluminal scene segmentation of colonoscopy images," *J. Healthc. Eng.*, 2017. Available: https://doi.org/10.1155/2017/4037190.

[4] J. Bernal *et al.*, "Comparative validation of polyp detection methods in video colonoscopy: Results from the MICCAI 2015 endoscopic vision challenge," *IEEE Trans. Med. Imaging*, vol. 36, no. 6, pp. 1231–1249, 2017. Available: https://doi.org/10.1109/TMI.2017.2664042.

[5] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Med. Image Comput. Comput.-Assist. Intervent.*, pp. 234–241, Springer, 2015. Available: https://doi.org/10.1007/978-3-319-24574-4_28.

[6] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 801–818.

[7] R. G. Dumitru, D. Peteleaza, and C. Craciun, "Using DUCK-Net for polyp image segmentation," *Scientific Reports*, vol. 13, no. 36940, 2023. Available: https://www.nature.com/articles/s41598-023-36940-5.

[8] B. H. Menze *et al.*, "The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS)," *IEEE Trans. Med. Imaging*, 2015. Available: https://doi.org/10.1109/TMI.2015.2499602.

[9] S. Bakas *et al.*, "Advancing The Cancer Genome Atlas glioma MRI collections with expert segmentation labels and radiomic features," *Nature Scientific Data*, 2017. Available: https://doi.org/10.1038/s41597-017-0028-4.