

Tetra-ConvBiNet: MRI-based brain tumor classification using distributed tetra head attention-enabled convolutional bidirectional network

Aasha Mahesh Chavan* and
Vanita Mane

Department of Computer
Engineering, Ramrao Adik Institute
of Technology, Pad. Dr. D. Y.
Patil Vidyapeeth, Navi Mumbai,
Maharashtra 400706, India

*Email: chavanaasha13@gmail.com

Received for publication
August 22, 2025.

Abstract

Brain tumor (BT) involves the persistent proliferation of abnormal brain cells, often resulting in mortality. Therefore, early detection and classification are required for timely diagnosis and treatment planning. However, previous methodologies rely on analysis of manual imaging and which is a time-consuming process and leads to errors. Therefore, a sonar energy optimized distributed tetra head attention-based convolutional bidirectional network (SEnO-DTCBiNet) framework is proposed to detect and classify BTs through magnetic resonance imaging (MRI) images. The architecture leverages the precise segmentation through optimized fused attention enabled distributed W-Net based segmentation (OFA-based DW-Net) that includes tetra-head attention and SEnO algorithm to fine-tune segmented features and hyperparameters of the model. Moreover, the comprehensive multistructural feature extraction process extracts the inherent tumor characteristics, thereby enhancing the capability of the model. For the BraTS 2020 dataset, the SEnO-DTCBiNet model achieved an accuracy of 97.18%, a precision of 97.31%, a recall of 97.89%, and an F1-score of 97.26%.

Keywords

Brain tumor, magnetic resonance imaging, healthcare, deep learning, attention mechanism

Abbreviations

2D-CNN-CAE: 2D Convolutional Neural Network and convolutional auto-encoder; BA-Bat algorithm; BAM: Bottleneck Attention Mechanism; BA-MCBM: Bat algorithm enabled Multi-head Convolutional Bidirectional Memory; BiLSTM: Bidirectional Long Short-Term Memory; CHBA: Chronological Honey Badger Algorithm; CJHBA: Chronological Jaya Honey Badger Algorithm-based Deep Residual Network; DEA: Dolphin echolocation algorithm; DW: Distributed W-Net; EDN-DRN: Ensemble Deep Neural-Deep Residual Network; EDN-SVM: Ensemble Deep Neural Support Vector Machine; GD-Gadolinium; GLCM: Gray-Level Co-occurrence Matrix; LIME: Local Interpretable Model-agnostic Explanations; LSTM: Long Short-Term Memory; MR: Magnetic Resonance; OFA: Optimized Fused Attention; PDCNN: Parallel Deep Convolutional Neural Network; ReLU: Rectified Linear Unit; ROC: Receiver Operating Characteristic; SEnO: Sonar Energy Optimization; SPO: Sonar Prey Optimization; ULSAM: Ultra Lightweight Sub-Space Attention Module.

I. Introduction

The brain, a vital organ in the human body, is responsible for managing emotions, thoughts, behaviors, and sensory processing. Its cerebral cortex contains billions of neurons that facilitate signal transmission. The *abnormal growth of brain* cells in the human body causes tumors, which result in brain tumor (BT) [1]. BT is one of the most severe ailments that occurs in the body, which leads to a decrease in the normal life expectancy of humans. The tumors developed from the brain tissue spread aggressively to every part [2]. The abnormalities that develop in the human body may lead to death. One of the most dangerous abnormalities produced in the brain tissue causes BT, and it is categorized into two types. They are primary and secondary tumors [3, 4]. When a tumor forms inside the brain without any symptoms, it is called a primary tumor, which develops in the brain tissue itself. Another one is metastatic BT, which is formed in another part and spreads to the brain eventually [5]. The most widespread BT are pituitary adenomas, meningiomas, and gliomas. Several endeavors have been taken to diagnose the BT in an efficient way to automatically classify or detect the disease [6].

Magnetic resonance imaging (MRI) is one of the primary diagnostic tools used for identifying BT. However, due to the intricate structure of the brain, achieving precise diagnosis remains complex, demanding, and time-intensive [7]. BTs sometimes cause various symptoms in the body, based on the structure and position of tumors present, which include seizures, balance issues, unusual behavior, confusion, changes in vision, and memory issues [8]. To monitor, treat, analyze, and diagnose the human body, miscellaneous medical techniques are utilized, such as MRI, computerized tomography (CT), X-rays, and ultrasound imaging (UI). Among these medical professionals, most preferred MRI techniques because of their nonionizing radiation and noninvasive methods. Moreover, the MRI technique easily identifies the blood circulation in the veins and detects the cells [9, 10]. Many machine learning (ML) and deep learning (DL) algorithms use brain MRI images for detecting BTs using the datasets [11]. Because of the nonionizing radiation produced in the cells, an MRI scan can produce various image features that can be captured and used to recognize the tumor cells. The diagnosis of an MRI tumor has many modalities for separately classifying and segmenting [12]. The early-stage diagnosis and detection of cancer cells becomes critical, and it is a challenging task because

of the tumor's formation, shape, and dimensions. ML techniques are time-consuming, so advanced DL techniques are preferred. [3].

Numerous methods have been established to identify the BTs, with various techniques, such as ML and DL. One of the most widely used methods is convolutional neural network (CNN) because it can accurately classify and recognize images. For the identification of cellular structures and molecular identification, image classification, prognosis of diseases, and tissue segmentation, ML is preferred [13]. The method of analyzing, organizing, and collecting the medical images has become digitalized. Even in cutting-edge methods, the explication of medical-related images requires more time and mitigates the accuracy of classification [5]. The CNN approach used in the classification of multi-grade BTs using MRI first segments the images, then augments the images, and then pretraining is given to correctly classify the BT [14]. The tumors can be either small or big, which makes the model harder to analyze the BTs. So, a balanced dataset is needed to accurately classify the brain images and detect the type of disease. But some datasets used in the research are imbalanced, and they affect the performance of the model [15, 16]. The NeeuroNet19 model accurately classifies the four types of tumor cells, and it faces some challenges, such as in tumor sizes and the position of the tumor cells. Artificial neural network with the integration of DL and ML has made a remarkable success in classifying the tumor cells and enhancing the overall performance [2].

To address these challenges, this study proposes a novel framework named sonar energy optimized distributed tetra head attention-based convolutional bidirectional network (SEnO-DTCBiNet), which aims to classify BT effectively, ensuring early identification and timely diagnosis, ultimately reducing BT-related fatalities. The MRI images are allowed for the classification model, where the precise region of interest (ROI) extraction is carried out using a binary thresholding technique to ensure only the most relevant features for analysis. The segmentation process allows for precise segmentation of tumor regions, which is crucial for reliable diagnosis. Moreover, the multi-structural matrix-based statistical deep flow features (MSMSD) allows for capturing diverse rich information, which includes structural patterns, texture, and statistical properties, for classifying more discriminative features. The SEnO-DTCBiNet architecture utilizes the extracted features to differentiate between tumor categories. The core components and strengths of this approach are detailed in the following sections.

- **Optimized fused attention enabled distributed W-net based segmentation (OFA-based DW-net):** Integrating tetra-head attention and the SEnO algorithm within the OFA-based DW-Net segmentation process improves accurate tumor region identification. This design combines several attention modules, such as BAM for enhanced feature learning, squeeze and excitation (SE) for channel-specific attention, Ultra Lightweight Sub Space Attention Module ULSAM for focusing on essential structures, and Criss-Cross (CC) Attention for dynamic contextual information capture. Additionally, the fine-tuning of W-Net through SEnO allows to attain precise outcomes in segmentation.
- **SEnO-DTCBiNet:** The SEnO-DTCBiNet combines the distributed tetra head attention (DTHA) and the deep BiLSTM layers to capture temporal dependencies and enhance contextual understanding of tumor regions, leading to increased reliability and accuracy in BT classification. Additionally, the SEnO algorithm obtained by hybridization of Bat, dolphin echolocation, and the Sparrow Search characteristics effectively fine-tunes the model parameters, resulting in higher efficiency and robustness.
- **The structure of this paper is outlined as follows:** Section II reviews related literature, Section III aligns the approach with the system model, Section IV details the methodology for tumor classification, Section V presents the outcomes, and Section V concludes the study.

II. Literature Survey

This section explains the literature review part, describing the traditional approaches used in the context. Ozkaraca et al. [1] proposed a Dense-CNN approach for classifying BT using MRI data. The model incorporates existing DL networks, such as VGG16 and DenseNet, to process MRI images more effectively. While this approach boosts classification performance, it requires considerable processing time. Anantharajan et al. [11] developed a BT detection method based on DL and ML techniques. After acquiring the MRI images, they undergo preprocessing, segmentation, and feature extraction using various techniques. Ultimately, a DL model is employed to detect abnormal brain tissues. The result found that the suggested model established good sensitivity, accuracy, and specificity in classifying the normal

and abnormal tissues. But it faces some concerns in classifying the image, as it considers only the gray-scale images.

Deepa et al. [4] suggested a Chronological Jaya Honey Badger Algorithm-based Deep Residual Network (CJHBA-DRN) for categorizing the BTs based on an optimization method. However, DeepMRSeg is used for segmenting the images, and for training the CHBA algorithm is introduced. After that, features are extracted and then augmented by the CJHBA algorithm to categorize the BT using the dataset. Even though this method performs better in detecting BT, it has some challenges. This method considers only the selected features, which decreases the performance of classification. Muezzinoglu et al. [7] introduced a PatchResNet-based model to classify BTs using MRI images. In the developed method, two feature extractors are introduced in addition to the three feature selectors. The image classification method of the model was improved, and the accuracy rate for classifying BT was also enhanced by the Iterative hard majority voting (IHMV) technique. The main drawback occurred in this model was that the datasets had to be improved, and along with this, the absence of a better optimization method reduced the classification accuracy.

Rahman and Islam [13] suggested a DL model to detect BT identification through the datasets. Here, the images are first resized and deployed with a grayscale transformation in the model, and then augmented to reduce the complexity issues. Moreover, this PDCNN method extracts both local features as well as global features to avoid the over-fitting problem with the normalization method. This method has some limitations in handling the dataset, which has to be enlarged to identify the tumors in 3D pictures. Haque et al. [16] suggested a DL method-based NeuroNet19 method to detect BTs. VGG19 with the cascading method was utilized to extract features of global and local features in the image. An explainable artificial intelligence (AI) is employed to increase the model's accountability, and LIME, which is used to improve the classification accuracy and mitigate false negative images. Therefore, this method shows an outstanding performance both in accuracy and promptness. The outcoming barriers that occurred in this model are, it does not detect binary class tumors, and the training can be improved with enlarged datasets.

Babu Vimala et al. [2] introduced a DL model to detect and categorize the BTs by Efficient Net. The

Grad-CAM is utilized in the model to highlight the affected areas and classify the tumorous cells in the brain. The fine-tuned EfficientNetB2 accurately explains the generalization and efficiency of the model through testing phases and achieves better performance. But the flaws involved in this method need more time for training, which suppresses the speed of classification. Islam et al. [6] proposed a DL model to diagnose BT cells using MRI images. Here, four architectures are utilized to classify BTs, which include Mobile Net, InceptionV3, DenseNet121, and VGG19. The BT cells are first classified through a fine-tuning method, and a comparison is made with the above approaches. From that, the Mobile Net achieves better accuracy by classifying BT cells. Even though this method enhances classification accuracy, it is selected only for a few features.

a. Challenges

The limitations that occurred during the classification of BT in earlier approaches are discussed below.

- In the suggested model [1], the classification accuracy for detecting the BT is improved, but the processing time required is longer.
- The EDN-SVM classifier method [11] considers only the gray-scale images to accurately detect the normal and abnormal tissues present in the brain cells, which limits its applicability across diverse types.
- In [6], the PDCNN method captures both high- and low-level image features, but expanding the dataset is necessary for improved BT cell classification accuracy.
- The EfficientNetB2 method shows better performance with good efficiency, but the training time required is more, which diminishes the speed of detection [2].
- In the developed method [6], only the selected features are utilized by the model, so the classification process depends on the selected region and suppresses the other tissues in the MRI image.

b. Problem definition

BT causes disability, dreadful illness, and even death to people at an early stage. The DL model with advanced techniques in the medical field enables to diagnosis of BT at an early stage to save human life. But still, the researchers face many challenges

in detecting the BT and diagnosing it, because of its complex nature of the brain. The main barriers in the existing models involved lower accuracy, overfitting problems, low classification performance was low, and higher time complexity. Therefore, to detect the disease accurately with more precision and effectively enhance the tumor grading performance, a SEnO-DTCBi Net model is proposed. The input images collected from the datasets is denoted as:

$$D = \{(s_1, y), (s_2, y), \dots, (s_i, y), \dots, (s_m, y)\} \quad (1)$$

where D represents the dataset, s_i implies the input samples of i^{th} image, s_m implies the total amount of input samples, m denotes the total number of samples, and y expresses the tumor types for the input samples. The samples are passed through the ROI extraction stage, in which specific areas are isolated and forwarded to the segmentation module. This step, executed using OFA-based DW-Net, enhances model efficiency and reduces computational load. The segmented samples are combined with the fused attention mechanism and capture the multiple-scale features. This provides an enhanced feature refinement as well as the localization effect. Furthermore, the segmentation model extracts the segmented regions from the samples and categorizes the different BT cells from the scanned image samples, and addresses the variations correspondingly. Therefore, from the segmented output G , the segmented results are accurately obtained. Furthermore, the extracted feature map Q is given as input to the proposed model for tumor classification. Following feature extraction, the resulting feature maps serve as input to the tumor classification model. The SEnO-DTCBiNet architecture is assessed using the categorical cross-entropy (CCE) loss function throughout the training process, which is mathematically defined as:

$$Loss_{fun} = - \sum_{a=1}^n b_a \log(U_a) \quad (2)$$

where the predicted probability is denoted as U_a , the actual label as b_a , and n indicates the total number of classes. Additionally, the real class labels from the BT dataset [17] are defined as:

$$b_a = \begin{cases} 1, meningioma \\ 2, glioma \\ 3, pituitary tumor \end{cases} \quad (3)$$

The actual labels of the Brats2020 dataset [18] and Brats2018 dataset [19] are specified as:

$$b_a = \begin{cases} 1, \text{Necrotic and non – enhancing tumor} \\ 2, \text{GD enhancing tumor} \\ 3, \text{peritumoral edema} \end{cases} \quad (4)$$

III. System Model

The research implemented for an effective BT classification system includes the initial process of collecting diverse MRI images through CT scan machines from laboratories or hospitals. These images are aggregated in BraTS 2020 [18], BraTS 2018 [19], the BT dataset [17], and the real-time dataset and are further allowed for the ROI extraction process to ensure that only the most relevant and important features are considered for analysis. Then, a series of features is captured to understand the complex structures of tumor regions more precisely, leading to more discriminative features for classification. These features are then fused into the proposed classification system to initiate the training phase. Test features are then applied to the trained model to obtain the tumor grading for differentiating the types of BT for timely and early diagnosis.

IV. Classification of BT Using SEnO-DTCBiNet

This search introduces an approach named the SEnO-DTCBiNet framework, designed for BT detection and classification using MRI scans. Initially, the collected BT images undergo a ROI extraction stage, where significant features are identified and passed on to the OFA-driven DW-Net architecture. Through this model, a SEnO optimization is utilized with the integration of bat optimization, Dolphin Optimization, and Sparrow Search optimization to train the model and enhance the model's effectiveness. Consequently, the segmentation phase is carried out on a w-net model with four fused attention mechanisms for the accurate segmentation of images. Following this, the segmented output proceeds to a dedicated feature extraction unit, where vital attributes are drawn out using a MSMSD model. The MSMSD model includes three different, distinct features, such as GLCM-derived features, statistical deep flow-based features, and hybrid 3D Structural pattern features. Once these MSMSD features are obtained, they are given to the SEnO-DTCBiNet model to classify BT such as Necrotic and nonenhancing tumors, GD-enhancing tumors, and peritumoral edema. A schematic

representation of the SEnO-DTCBiNet architecture is shown in Figure 1.

a. Input MRI image collection

The input images are collected from the Brats2020 dataset [18], the BT dataset [17], and the Brats2018 dataset [19]. Let us assume the dataset is D , and it is given by,

$$D = \{s_i\}; 1 \leq i \leq m \quad (5)$$

where s_i denotes the i^{th} input image sample with a dimension of $[240, 240, 1]$, and the total number of samples present in an input image is denoted as m .

b. Thresholding-based ROI extraction

The collected input image s_i is fed to an ROI extraction phase, which extracts the selected ROI from the images for further processing through a binary thresholding mechanism. The original image is changed into a grayscale image with pixel values ranging from $[0, 255]$. The threshold value T_r is calculated by taking the average value of the grayscale image. For each pixel s_i , the intensity is compared with the threshold value T_r . During comparison, when the pixel intensity value is higher than T_r , then the binary value is denoted as "1," and that is taken as the interested regions, and if the pixel intensity value is less than T_r , then the binary value will be represented as "0," and those regions will be eliminated. So, to obtain a clear image, a binary mask is applied that retains the required region. The major purpose of the mechanism is to reduce errors and to enhance classification accuracy. In the ROI extraction phase, the output dimension of the images varies for each slice, and hence, the resized images are formed with a size $[512, 512, 1]$ that helps the segmentation process to be carried out more smoothly. Thus, the resultant output H is obtained from the ROI extraction, which is further transferred to the segmentation phase.

c. Optimized fused attention-based distributed W-Net for segmentation

Segmentation is employed to enhance the precision of BT classification and support more accurate diagnosis. By segmenting the images, the detection process becomes more efficient, and the overall analysis of the medical images is significantly streamlined. Hence, the

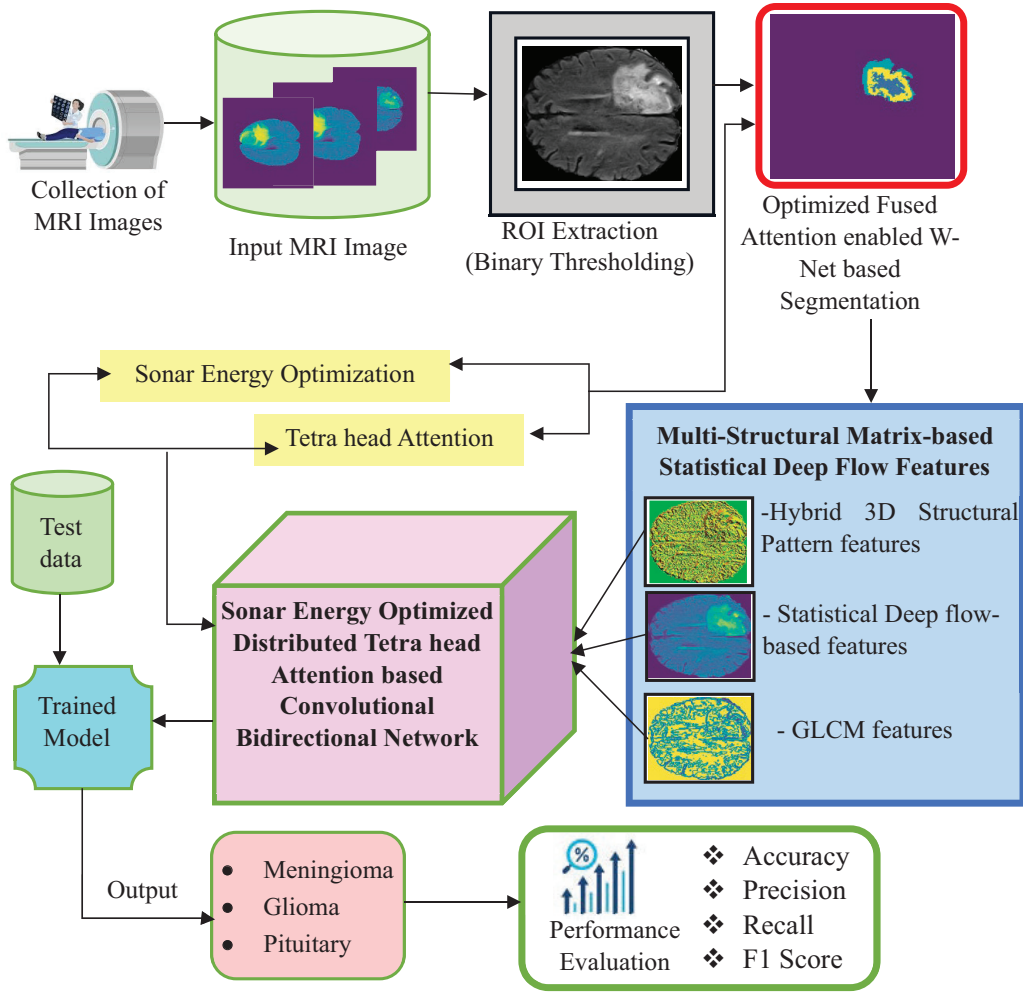


Figure 1: Block diagram of the SEnO-DTCBiNet framework. SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network.

segmentation process is performed by an OFA-based DW-Net model for identifying the accurate position of the BT. The image obtained from the ROI extraction step serves as the input for the OFA-based DW-Net method [20]. In general, the W-Net structure is considered, where four attention mechanisms are fused, namely, BAM, SE, ULSAM, and CC attention, in a distributed manner to derive the OFA-based DW-Net segmentation model. Additionally, the model's segmentation component is fine-tuned through an optimization algorithm that operates at the fifth layer of the framework, with detailed insights provided in Section 4.6.

The OFA-based DW-Net segmentation depends on the W-Net model, which reconstructs the input image and predicts the output map without degrading the information. It is composed of two parts encoder and a decoder, with 47 convolutional layers with 18 modules. In the encoder path, the down-sampling

process is carried out with the fused Attention mechanisms of BAM, SE, ULSAM, and CC. These fused mechanisms are arranged in a distributed manner, and following this, the up-sampling phase is employed. The context is captured through the utilization of a contacting path, which is presented as Z_{enc} encoder, the expansive path that is useful for precise localization is known as the decoder Z_{dec} . The encoder part of the segmentation model takes the input as H having the dimension of dimension $(N, 512, 512, 1)$. Moreover, the encoder module is composed of 2×2 convolutional layers followed by a dropout layer and a max pooling layer, such that the dimension at each stage of the down-sampling layer in the encoder phase increases. On the contrary, the dimension is reduced at each layer of the up-sampling layer that lies in the encoder module. The OFA-based DW-Net architecture is depicted in Figure 2.

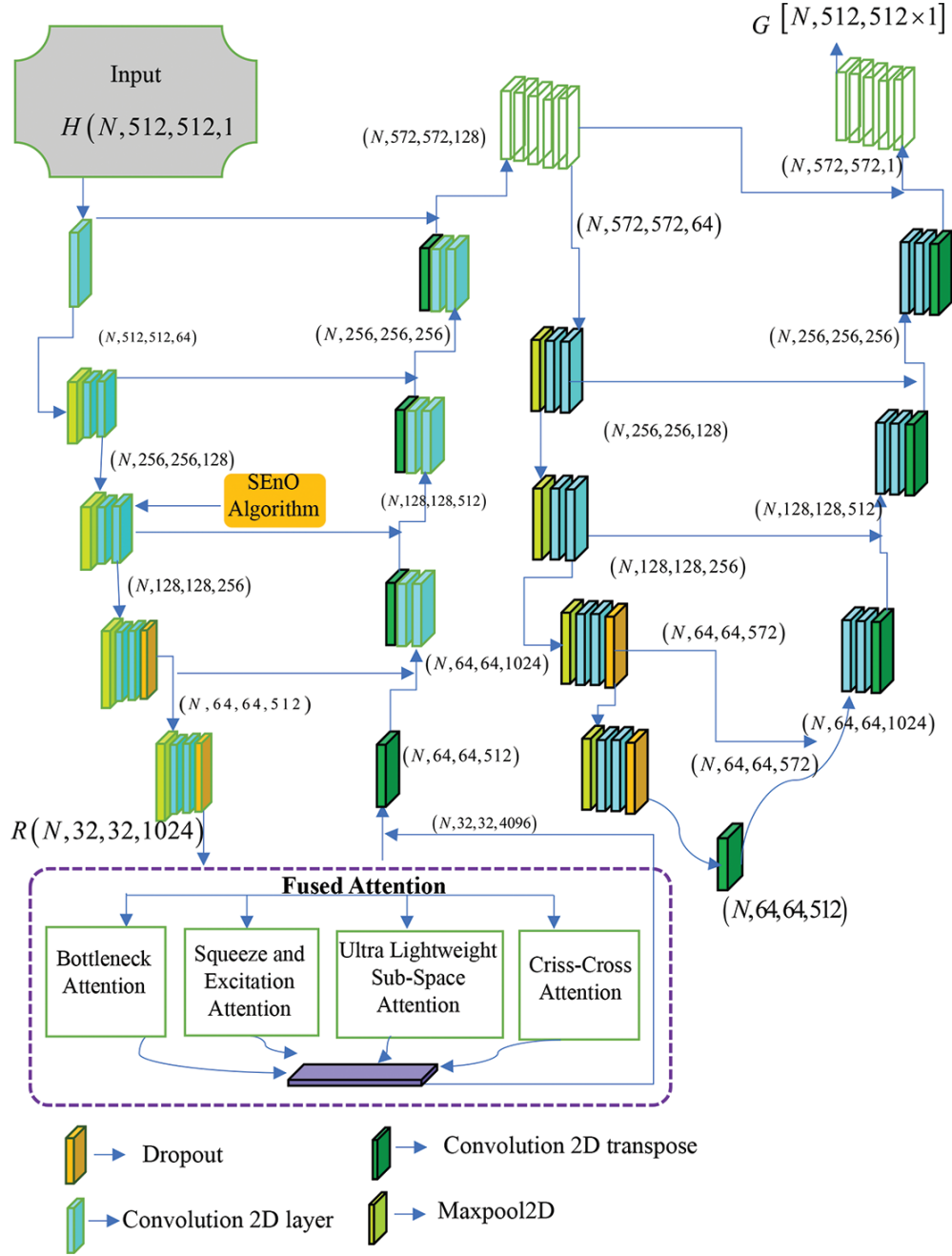


Figure 2: OFA-based DW-Net model architecture.

The resultant output obtained at the final dropout layer of the down $(N, 64, 64, 512)$ -sampling process at the encoder side is represented as R , having the dimension of $[N, 32, 32, 1024]$, which is further passed to the fused attention framework in a distributed manner, and the concatenated output derived through the attention model has the size of $[N, 32, 32, 4096]$, respectively. In the up-sampling process, with the

transposed $H(N, 512, 512, 1)$ 2D convolutional layer, the feature maps are expanded and combined with the equivalent sizes of the feature map in the down-sampling process by a layer called the skip layers. The skip layers are utilized in up-sampled feature maps for improving the region localization. A SoftMax layer is applied after the final convolutional layer in the encoder path to minimize the normalized cut (Ncut) value.

The Ncut loss function is used to eliminate the segmentation noise and smooth the layer. Therefore, the Ncut value is calculated by a mathematical expression as,

$$Ncut_j(N) = \sum_{j=1}^J \frac{cut(E_j, N - E_j)}{asso(E_j, N)} \quad (6)$$

where N represents the all-pixel groups, and E_j describes the group of pixels in j^{th} the image. Furthermore, the OFA-based DW-Net model is ready to minimize reconstruction loss for implementing a contracted path and consists of numerous collections of information about the original image. Thus, the reconstruction loss is expressed by a mathematical equation of,

$$\psi_{reconst} = \|s_i - Z_{dec}(Z_{enc}(s_i; B_{enc}); B_{dec})\|_2^2 \quad (7)$$

where the parameters of the encoder and decoder are represented by B_{enc} , and B_{dec} , respectively, and the original input image is depicted as s_i . Hence, the pixel set values in the selected region are maximized and acquire the regenerated image at the same time, to boost the segmented precision of the image. The

main advantage of using the OFA-based DW-Net method is that it enhances the accurate performance of the image segmentation and captures fine details of the features. Along with this computational cost is mitigated with better generalization capability, robustness, and good flexibility. So, these fused attention mechanisms are utilized in this research to provide better performance in disease classification. Based on the segmentation process, nontumor cells are removed, and G represents the segmented tumor cell. Hence, the segmented image's size is $[N, 512, 512, 1]$, which is further used for the feature extraction process. The brief explanation of BAM, SE, ULSAM, and CC fused attention mechanisms is elaborated below.

c.i Bottleneck attention mechanism

BAM [21] is widely used with the convolutional network model because of its simple and efficient size. The major purpose of this attention is to increase the representation capability of the model by considering spatial and channel attention. The Channel attention and the spatial attention are the two different branches exist in the channels of the feature map. The input feature is taken as $R \in \mathbb{R}^{c \times h \times w}$, and a 3D attention map for BAM is noted as $A(R) \in \mathbb{R}^{c \times h \times w}$. Figure 3 represents the structure of BAM.

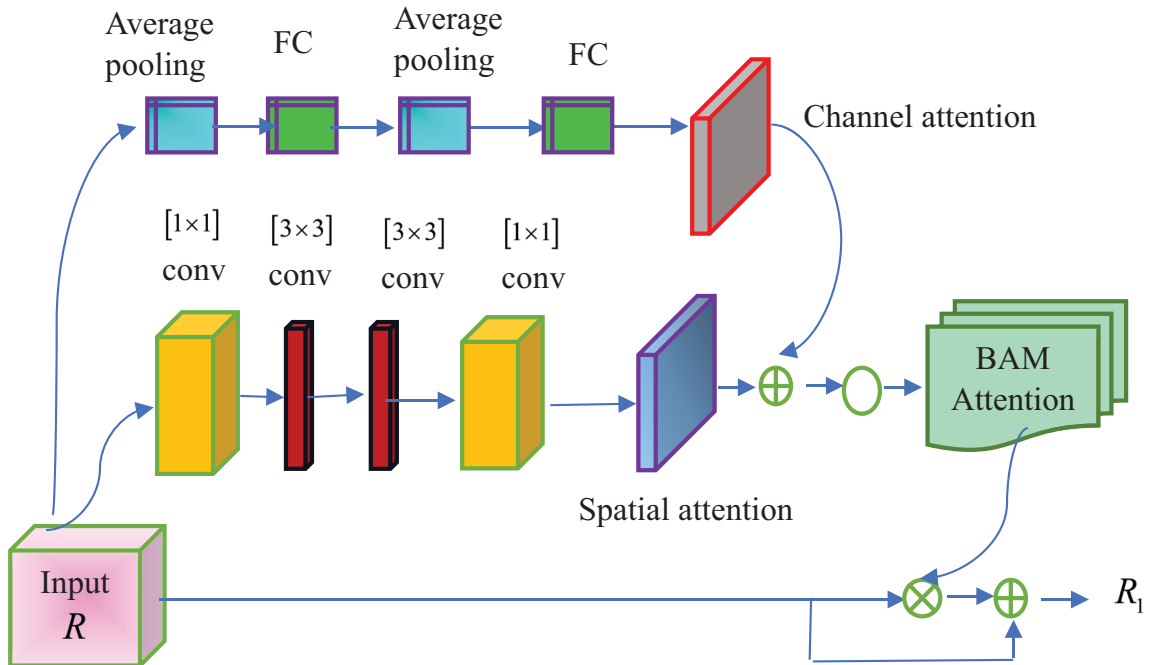


Figure 3: Bottleneck attention mechanism.

In the channel attention branch, to calculate a feature map, average pooling layer is taken, which encodes the information present in each channel. For determining the attention, a multilayer perceptron is used in a hidden layer, and a batch normalization layer is also used for adjusting scale parameters. Hence, the channel attention is mathematically expressed as:

$$A_c(R) = Ba(MI(Ap(R))) \quad (8)$$

where $Ap(R)$ implies average pooling, MI is the multilayer perceptron, Ba denotes Batch normalization, and $A_c(R)$ denotes channel attention.

Spatial attention is employed to filter out features distributed across different spatial positions. This module utilizes two key hyperparameters: the dilation rate, which helps in capturing contextual information, and the reduction ratio, which manages channel capacity and controls computational overhead. The feature is first reduced using (1×1) convolution, and then (3×3) dilated convolutions are used, and finally, the reduced features are again convoluted using (1×1) batch normalization in the spatial branch output for scale adjustment. Therefore, spatial attention is represented as,

$$A_s(R) = Ba(f_3^{1 \times 1}(f_2^{3 \times 3}(f_1^{3 \times 3}(f_0^{1 \times 1}(R)))))) \quad (9)$$

where f represents a convolution operation, Ba implying the batch normalization, $A_s(R)$ and denotes the spatial attention. To obtain an efficient module, channel attention is added with the spatial attention along with the sigmoid function, and then the attention map is calculated as:

$$A(R) = \sigma(A_c(R) + A_s(R)) \quad (10)$$

The sigmoid function is denoted by σ , spatial attention is expressed by $A_s(R)$, and channel attention is denoted by $A_c(R)$. Thus, the BAM output R_1 is calculated by adding the original input feature with the attention module, and it is mathematically expressed as,

$$R_1 = R + R \otimes A(R) \quad (11)$$

where $\otimes$ express the element-wise multiplication. Thus, the BAM model effectively enhances the model performance and mitigates the overhead and computational complexity problems.

c.ii SE attention

The working principle of SE attention [22] depends on the convolution network model, and the spatial

dependencies are figured out by global average pooling. The SE method in image segmentation provides more pixel-wise information about the spatial blocks, which is more accurate. Assume the input of SE as $R = \{r_1, r_2, \dots, r_n\}$ with the number of channels, such that $r_k \in \mathbb{R}^{h \times w}$, where spatial height and spatial width are indicated as h and w , respectively. The working mechanism of SE is based on two different frameworks, namely the spatial framework and channel framework. Hence, the spatial model employs global average pooling as a squeeze operation. This generates a vector by summarizing the spatial information of each feature map q , and it is represented by:

$$q = \frac{1}{h \times w} r_k(i, j) \quad (12)$$

Furthermore, the vector q is processed through two fully connected (FC) layers with the ReLU operator. Finally, the resultant factor of the FC layer is given to the sigmoid layer to generate the output $\sigma(\hat{q})$.

$$\hat{R}_{cSE} = [\sigma(\hat{q}_1)r_1, \sigma(\hat{q}_2)r_2, \dots, \sigma(\hat{q}_n)r_n] \quad (13)$$

The channel model performs channel squeeze operations using a convolutional layer, its output is denoted as c , and it can be represented by $c = R \times w_t$, where w_t denotes the weight factor, and c implies the output of the projection sensor. Each projection tensor is further fed to the sigmoid layer to retrieve the output as $\sigma(c)$, and the channel model output is represented by:

$$\hat{R}_{sSE} = [\sigma(c_1)r_1, \sigma(c_2)r_2, \dots, \sigma(c_n)r_n] \quad (14)$$

Also, the irrelevant features are reduced, and important spatial locations are delivered by the recalibration. The spatial, channel, squeeze, and excitation components are integrated into a single block by concatenating the outputs from both channels.

$$R_2 = \text{con}(\widehat{R}_{cSE}, \widehat{R}_{sSE}) \quad (15)$$

This combined network of the feature map provides both channel-wise and spatial information, segments the important features, reduces the complexity problems, and the resultant output will be denoted as R_2 . Figure 4 represents the SE structure.

c.iii Ultralightweight subspace attention mechanism

The structure of ULSAM [23] is designed by considering a convolution and pooling layer such that

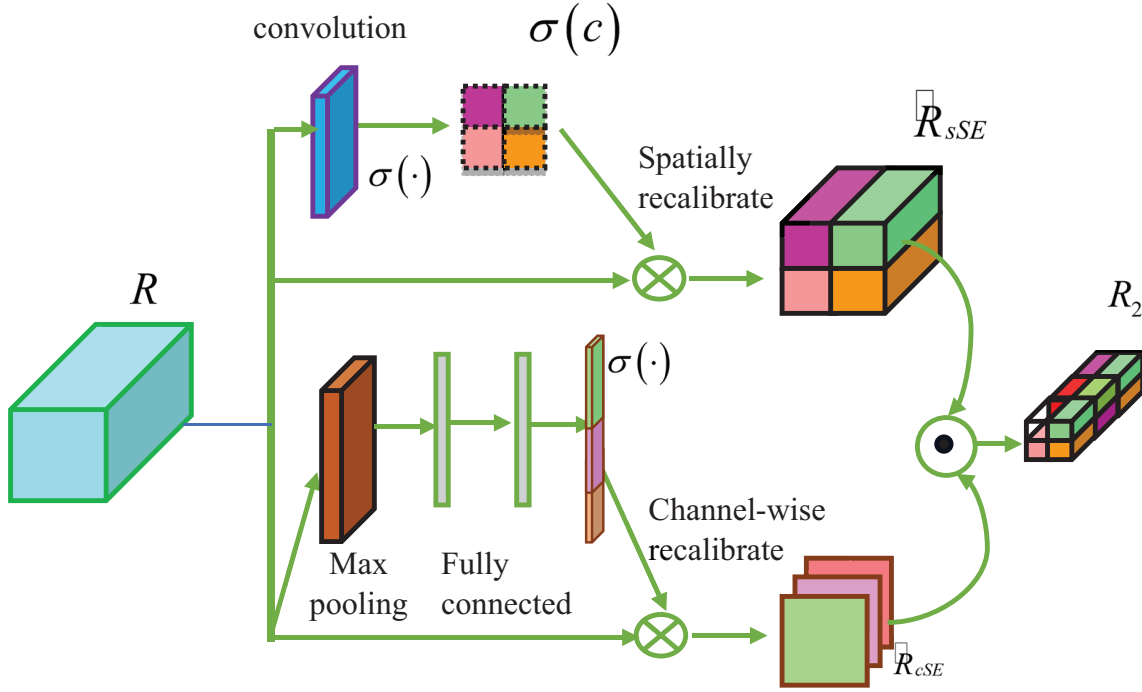


Figure 4: SE attention. SE, squeeze and excitation.

it learns an attention map in each feature subspace, which minimizes the channel and spatial redundancy. Since the attention map was learned by the model, it enhances the image segmentation and efficiently learns the cross-channel feature map interdependencies. Let us consider a feature map $R \in \mathbb{R}^{b \times h \times w}$ from the dropout layer, b be the number of input channels, and h w represents the dimension of the spatial map. The major goal is to capture features of cross-channel interdependencies effectively in the feature map. The feature map R is divided into d groups, and each group has a feature map $D [R_1, R_2, \dots, R_n, \dots, R_d]$. In ULSAM, the features are extracted through a max pooling layer, a depth-wise convolution layer, and a point-wise convolution layer with a feature map R_n . The structure of ULSAM is depicted in Figure 5.

An attention map $Attn_n$ captures each group's nonlinear dependencies to capture the cross-channel information in a feature map. To verify weighing attention, a gating function is utilized with the SoftMax activation function and refines the feature map. The output R_3 obtained in the feature map is obtained by joining the feature maps of every group.

$$Attn_n = \text{softmax}(PC(MP(DC(R)))) \quad (16)$$

$$\hat{R}_n = (Attn_n \otimes R_n) \oplus R_n \quad (17)$$

$$R_3 = \text{concat}(\hat{R}_1, \hat{R}_2, \dots, \hat{R}_n, \dots, \hat{R}_d) \quad (18)$$

where *softmax* denotes the SoftMax activation, *PC* signifies point-wise convolution, *MP* represents a max pooling layer, *DC* shows depth-wise convolution, $Attn_n$ is the attention map in each group, the feature map at each layer is shown as $\hat{R}_n$, defines the refined feature map, and $\oplus$ implies the element-wise addition.

c.iv CC attention

The CC attention [24] model is mainly used to capture the contextual information from the images in an optimal way. Because of its lightweight computation and more memory capability to define the local feature representations. This attention mechanism collects the vertical and horizontal contextual information to improve the pixel-wise explanation. The structure of CC attention is presented in Figure 6.

Consider a local feature map $R \in \mathbb{R}^{c \times w \times h}$, which is fed into convolution layers to generate the feature maps as, J , L , and X . $\{J, L\} \in \mathbb{R}^{c' \times w \times h}$, c' denotes the number of channels that exist. An attention map

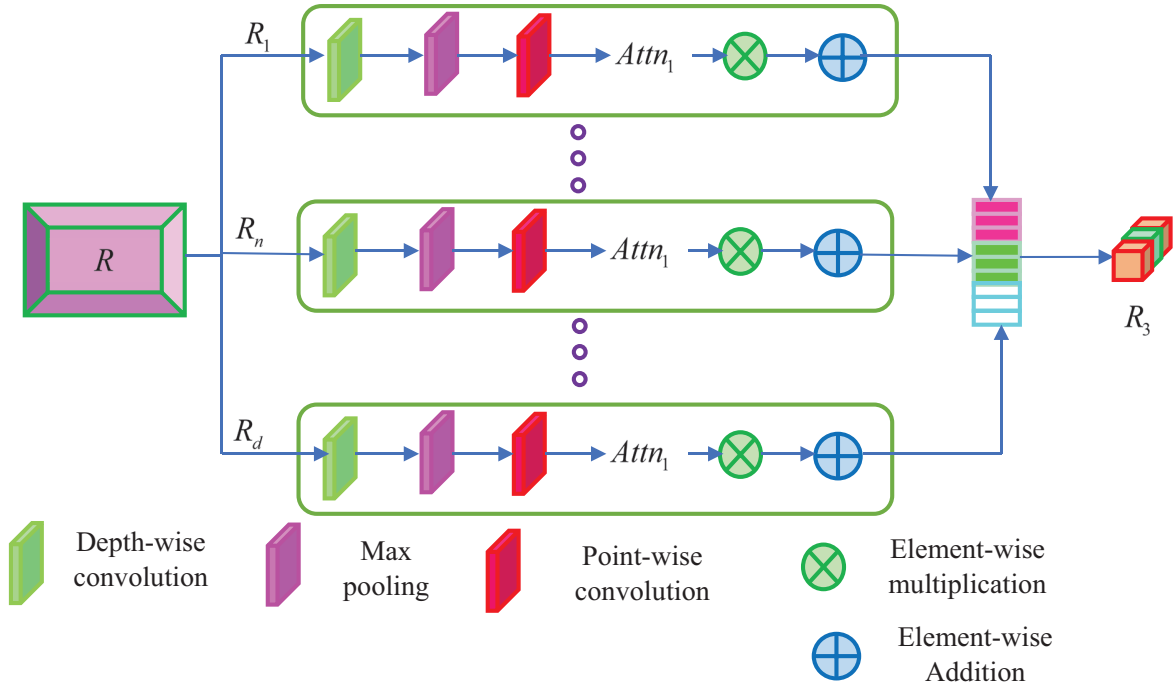


Figure 5: Structure of ULSAM.

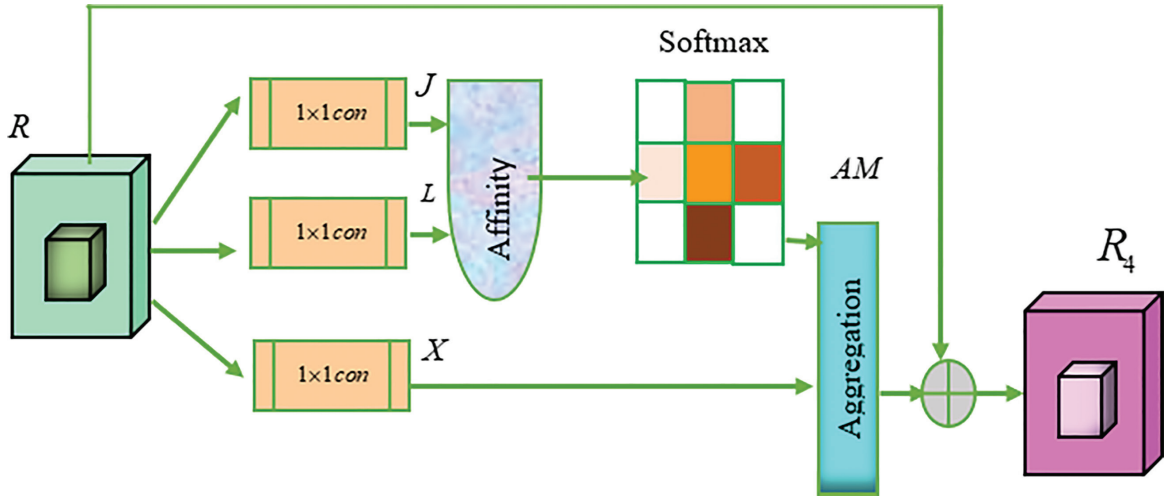


Figure 6: Structural diagram of CC. CC, criss-cross.

is generated after acquiring the feature maps J, L through an affinity operation. At each position τ in the spatial dimension, feature vectors generated from J and L are indicated as J_τ and L_τ , respectively. Hence, the affinity operation is expressed as $A_\tau = J_\tau L_\tau$, which refers to the degree of correlation among feature vectors J_τ and L_τ . Furthermore, a SoftMax operation

is applied to A_τ to compute the attention map AM . Moreover, a convolution layer with $[1 \times 1]$ a filter is applied over the input feature to create X for feature adaptation process, and the respective feature vector is denoted as X_τ .

$$R_4 = \sum AM_\tau X_\tau + R \quad (19)$$

where AM_r is the attention map's output, and R_4 represents the output of the CC attention model. Thus, the CC attention captures the contextual information pixel-wise. Therefore, the total fused attention mechanism output is concatenated, and the resultant output Q is passed as an input to the feature extraction phase.

$$Q = \{R_1 \| R_2 \| R_3 \| R_4\} \quad (20)$$

d. Extraction of MSMSD

The MSMSD-based feature extraction strategy incorporates multiple feature types, including statistical deep flow features, GLCM-based features, and 3D structural pattern features. This approach helps minimize computational complexity and enhances the model's accuracy.

Hybrid structural pattern features encompass a range of descriptors, including local binary pattern (LBP), local ternary pattern (LTP), and local gradient pattern (LGP).

The effective features from the images are acquired using the statistical approach known as LBP [25]. Therefore, it considers only the local features of the ROI extracted images, and it is mathematically expressed as:

$$LBP(Q) = \sum \alpha (P_z - P_x) 2^z \quad (21)$$

where $LBP(Q)$ denotes the features of LBP with the dimension $[N, 120, 120, 1]$, P_x represents the center pixel values, and P_z implies the z^{th} neighbor pixels. The LTP is a three-valued texture feature in determining the pixel values α denotes the constant. The extension of LBP is called the LTP and assigns the values from -1 to 1 . Therefore, the expression of LTP is denoted by:

$$LTP(Q) = \begin{cases} 1 & \text{if } P_z - P_x > \eta \\ 0 & \text{if } |P_z - P_x| \leq \eta \\ -1 & \text{if } P_z - P_x < -\eta \end{cases} \quad (22)$$

where the features of LTP with the dimension $[N, 120, 120, 1]$ is represented by $LTP(Q)$, and η is defined as the threshold value. LTP feature extraction method is utilized because of its classification accuracy and for extracting the texture features of an image. The LGP uses the adjacent pixel gradient in the quantified pixel values. The LGP method [26] enhances the interpretability of the model when

extracting the features for classification. The expression for LGP is mathematically denoted by:

$$LGP(Q) = \sum \alpha (v_n - \bar{v}) 2^n \quad (23)$$

where $LGP(Q)$ implies the features of LGP with the dimension $[N, 120, 120, 1]$, $\bar{v}$ denotes the mean gradient, and v_n expresses the central pixels and the adjacent pixels' average value. Therefore, the hybrid 3D structural pattern-based features are represented by concatenating all the extracted features.

$$\beta = \{LGP(Q) \| LBP(Q) \| LTP(Q)\} \quad (24)$$

β represents the extracted hybrid 3D structural pattern-based features with dimension $[N \times 120 \times 120 \times 3]$.

Statistical deep flow-based features are used to isolate the ROI from the extracted image, allowing the generation of a deep flow feature map. The statistical features [27] improve the performance of classification accuracy and make the model more reliable. The statistical deep flow feature uses the "cv2.calcOpticalFlowPyrLK" package to acquire the deep flow feature, and it is mathematically denoted as ϕ . As a result, the statistical features derived include mean, standard deviation, variance, skewness, and kurtosis. Table 1 demonstrates the characteristics obtained from the deep flow feature representation.

Eq. (25) demonstrates the statistical deep-flow-based features with the dimensionality $[N, 120, 120, 5]$.

$$\theta = \{T_1 \| T_2 \| T_3 \| T_4 \| T_5\} \quad (25)$$

GLCM [28] method is preferred because of its accuracy, and the computing time for extracting the images is shorter. It consists of entropy, energy, contrast, homogeneity, and dissimilarity. In Table 2, the features of GLCM are tabulated below.

The equation below represents the extracted GLCM features:

$$\mu = \{E_1 \| E_2 \| E_3 \| E_4 \| E_5\} \quad (26)$$

where μ denotes the GLCM features by containing a dimension of $[N, 120, 120, 5]$. Thus, the MSMSD-based features are obtained by concatenating all the features, and it is established as:

$$U = \{\beta \| \theta \| \mu\} \quad (27)$$

Table 1: Features based on statistical deep flow

Features	Description	Mathematical notation	Output size
Mean	It is the ratio between the total intensity of all pixels and the total number of pixels within the deep flow feature image.	$T_1 = \frac{1}{r} \sum_{i=1}^r \varphi_i$ where φ_i is the feature and r the total images.	$[N, 120, 120, 1]$
Kurtosis	Kurtosis is defined as the shape of the selected images taken for the statistical measurement.	$T_2 = \frac{\sum (T_1 - \bar{T}_1)^4}{r(T_3)}$ T_2 is the kurtosis	$[N, 120, 120, 1]$
Standard deviation	The standard deviation is defined as the square root of the variance and represents the average deviation of each pixel intensity from the mean.	$T_3 = \sqrt{\frac{1}{r} \sum_{i=1}^r (\varphi_i - T_1)^2}$ standard deviation is denoted as T_3	$[N, 120, 120, 1]$
Skew	Skewness is defined as a measure of symmetry of the image.	$T_4 = \frac{1}{r} \sum_{i=1}^r \left[\frac{\varphi_i - T_1}{T_3} \right]$ T_4 is the skew	$[N, 120, 120, 1]$
Variance	Variance is defined as the square of the standard deviation.	$T_5 = (T_3)^2 = \frac{1}{r} \sum_{i=1}^r (\varphi_i - T_1)^2$ T_5 is the variance	$[N, 120, 120, 1]$

Table 2: Features based on GLCM

Features	Overview	Formula	Dimensions of outputs
Homogeneity	Homogeneity calculates the similarity of the texture in the distributed gray-level object pairs.	$E_3 = \sum_{kl} \frac{M_{kl}}{1 + k - l }$ homogeneity feature is represented as E_3 .	$[N, 120, 120, 1]$
Energy	Energy is used to calculate the uniformity of an image.	$E_1 = \sum_{kl} (M_{kl})^2$ E_1 It is the energy feature, and GLCM of the image Q is denoted as M_{kl} .	$[N, 120, 120, 1]$
Entropy	Entropy reproduces the complexity of an image present in the GLCM features.	$E_4 = \sum_{kl} M_{kl} \log_2 (M_{kl})$ E_4 Represents the entropy.	$[N, 120, 120, 1]$
Contrast	Contrast calculates the local variation amounts present in the image.	$E_5 = \sum_{kl} M(k - l)^2$ E_5 Depicts the contrast.	$[N, 120, 120, 1]$
Dissimilarity	It measures the gaps between the mean variances and ROI in the gray-scale image	$E_2 = \sum_{kl} M_{kl} k - l $ E_2 Denotes the dissimilarity feature of GLCM.	$[N, 120, 120, 1]$

ROI, region of interest.

The extracted MSMSD-based features are specified as U with dimension $[N, 120, 120, 5]$. These are transferred to the distributed tetra head attention-based convolutional bidirectional network (DTCBiNet) model to classify the BTs.

e. DTCBiNet for the classification of BT

The DTCBiNet model is the integration of a convolutional and BiLSTM network, and it is used for the classification of BT. The DTCBiNet model comprises different layers, such as an input layer, a convolutional layer, a Leaky ReLU layer, a max pooling layer, a Dropout layer, a Reshape layer, an attention mechanism layer, a concatenation layer, a BiLSTM layer, and a Dense layer, respectively. This DTCBiNet model helps to extract the complex patterns in the feature and boosts the performance of the model with better classification accuracy. Along with this, the DTCBiNet model extracts both the spatial and the temporal relationships of the features and enhances the model's performance. The introduction of an attention mechanism in the model mitigates redundancy factors and effectively maximizes the learning process.

In this DTCBiNet model, the initially extracted image U is passed as input to the model, with $[N, 120, 120, 13]$ dimension, the image fed to the convolutional 2D layer. This layer extracts more features, and it acts as the main component in the DTCBiNet model. In the first convolutional layer, low-level features are first extracted, and high-level features are transferred to further convolutional layers. The mathematical equation of a convolutional layer is represented by,

$$J_1 = \sum L \otimes fi + bi \quad (28)$$

where the feature map is indicated by J , the bias is denoted by bi , and the convolutional filter is depicted by fi . The feature map J output is passed to the max pooling layer, and if the feature map gained more output from the convolutional layer, then it cannot be applied to the training directly. This causes more issues in overfitting problems and leads to more consumption of time, so down-sampling is used to minimize the parameters to reduce the size of the max pooling to produce an effective outcome. The size is reduced by the dimension of $[N, 60, 60, 16]$, after the maxpooling convolutional operation, and it is transferred to a second convolutional 2D network to extract the features, and it is represented by J_2 . The output from another convolutional layer is passed to

an activation function, which is called Leaky ReLU. This activation function is utilized because of its low computation time and simplicity. The Leaky ReLU is represented by,

$$LRL(J_2) = \max\{0, J_2\} + \nu \times \min\{0, J_2\} \quad (29)$$

where $LRL(J_2)$ is the Leaky ReLU activation function, ν implies the parameter of the Leaky ReLU function, and has a dimension of $[N, 60, 60, 64]$. Next, output from the Leaky ReLU is transferred to a next max pooling layer, followed by a Dropout layer with a dimension of $[N, 30, 30, 64]$. Consequently, the output from the dropout layer with size $[N, 30, 30, 64]$ gets reshaped to the size of $[N, 900, 64]$, and it can be depicted as d , which is transferred to the multi-head attention mechanisms. This MA mechanism is a combination of various attention mechanisms and gains a corresponding input feature. The combination of BAM, SE, ULSAM, and the CC mechanism is obtained by concatenating the features of all mechanisms, and they are elaborated in Section 4.3 with a detailed explanation and diagram. And these concatenated mechanisms produce an output with a dimension of $[N, 900, 256]$. The BiLSTM consists of an input layer, a bidirectional layer, and an output layer, which boosts the model's performance and captures both the bidirectional features of the model. The bidirectional layer included both forward and backward layers, in which the forward layer extracts forward features from the input and the backward layer extracts backward features. Then, both layers are concatenated to produce the output. The mathematical expression for representing these BiLSTM layers is:

$$\vec{fl} = \sigma \left[wts \left(V, \overline{w_{x-1}} \right) + bi \right] \quad (30)$$

$$\overline{bf} = \sigma \left[wts \left(V, \overline{w_{x-1}} \right) + bi \right] \quad (31)$$

where $\vec{fl}, \overline{bf}$ denotes the forward and backward layer outputs, σ implies the sigmoid function, wts and bi denotes the model's weight and bias, $\overline{w_{x-1}}, \overline{w_{x-1}}$ denotes the forward hidden and the backward hidden layer, and the output is represented as:

$$K = \left[\vec{fl} \oplus \overline{bf} \right] \quad (32)$$

where BiLSTM layer output is represented as K , and $\oplus$ denotes the addition operation, and then the output is transferred to the next BiLSTM layer, with

the size of $[N, 128]$ is passed to a dense layer. The dense layer has a dimension $[N \times 64]$ and is then transferred to the next dense layer with a reduced dimension $[N, 32]$. Finally dense layer produces the output with the size of $[N, 3]$. Finally, output is categorized into three labels as glioma, pituitary tumor, and meningioma, and the output is mathematically expressed as U_a . The biases and weights of the DTCBiNet model were optimized by the SEnO algorithm, and it is discussed in Section 4.6 section, the

optimization is applied after the result of the BiLSTM layer. The DTCBiNet model architecture is depicted in Figure 7.

f. Sonar energy optimization algorithm

The SEnO algorithm is obtained by integrating the bat optimization [29], the echolocation-based Dolphin optimization [30] algorithm, and the Sparrow Search Algorithm (SSA) [31]. The SEnO algorithm is employed

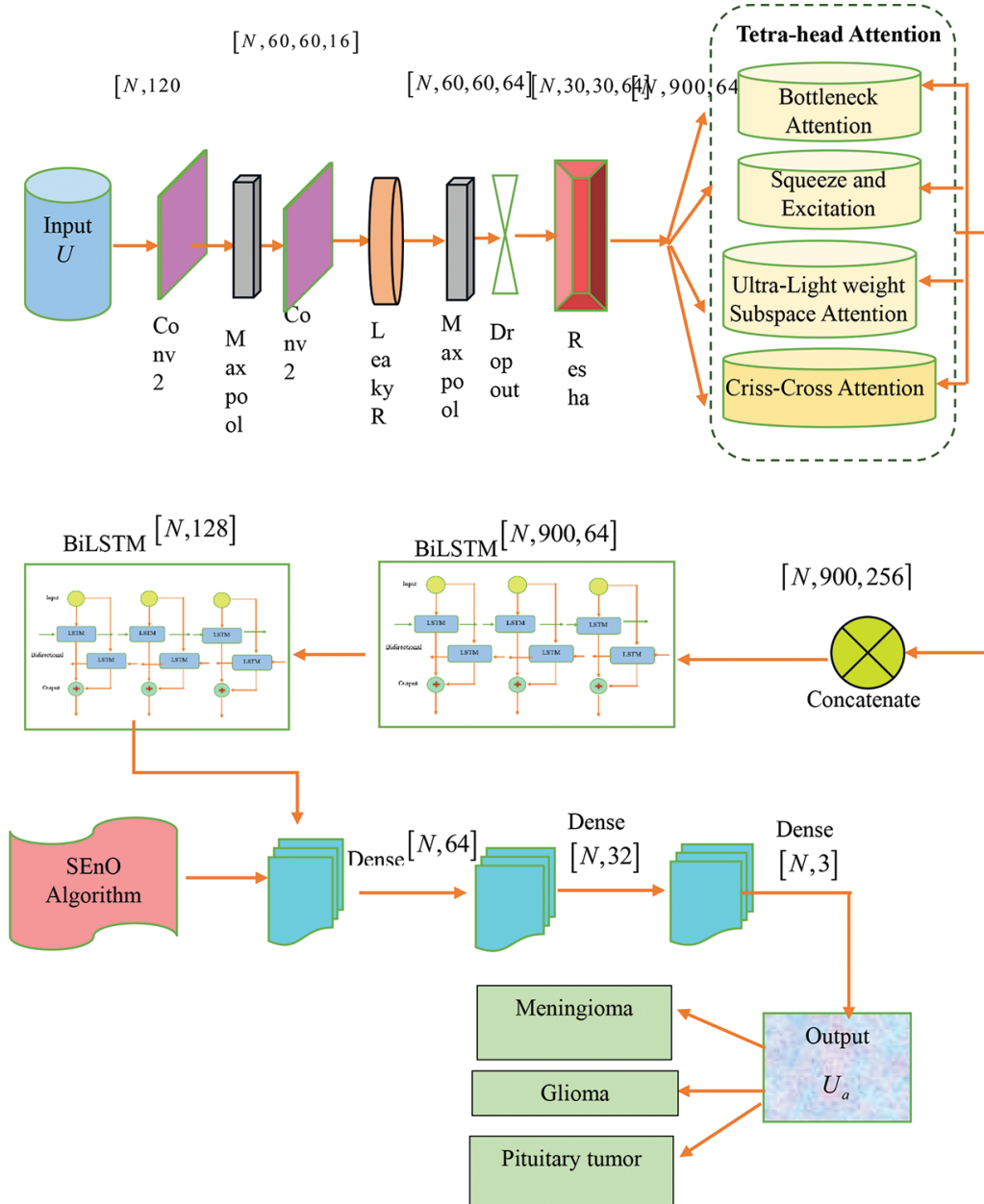


Figure 7: Architecture of DTCBiNet. DTCBiNet, distributed tetra head attention-based convolutional bidirectional network.

to optimize the bias and weight of the DTCBiNet model by applying the optimization at the end of the BiLSTM layer. On the contrary, for training the segmentation approach, on the fifth layer, the optimization is integrated to improve the performance of the model.

f.i Motivation

For finding prey and to navigate by listening to the echoes in the surroundings, the bat uses the echolocation behavior. In dark situations, it avoids the hindrance problem. Likewise, the dolphin utilizes echolocation by generating the sound, which is in the form of clicks, to locate the obstacles and its prey. The SEnO algorithm rapidly improves the model performance and reduces the overfitting challenges. The development of the SEnO algorithm in the model executes with better escaping, grouping, and alarming strategies and enhances the overall model mechanisms.

a) Initialization

In the initialization phase, first consider the population with a number of solutions h , and the total number of solutions t . Algorithm's initial population is mathematically expressed as:

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_h \end{bmatrix} = \begin{bmatrix} F_{1,1} \cdots F_{1,q} \cdots F_{1,t} \\ \vdots \\ F_{i,1} \cdots F_{i,q} \cdots F_{i,t} \\ \vdots \\ F_{h,1} \cdots F_{h,q} \cdots F_{h,t} \end{bmatrix}_{h \times t} \quad (33)$$

where $F_{i,q}$ implies the location of the i^{th} solution in the q^{th} dimension.

$$F_{i,q} = Up_{i,q} + e_1 (Lw_{i,q} - Up_{i,q}) \quad \begin{cases} i = 1, 2, \dots, h \\ q = 1, 2, \dots, t \end{cases} \quad (34)$$

where the number of dimensions is denoted as t , the number of solutions is expressed as i in the search space, h refers total number of solutions, and $F_{i,q}$ denotes the first position of the i^{th} solution in q^{th} the dimension. e_1 Refers to a random number within the interval of (0,1). The q^{th} search space upper bound is denoted as $Up_{i,q}$ and the lower bound is represented as $Lw_{i,q}$.

b) Calculation of fitness function

Fitness function of the SEnO algorithm is computed depending on the accuracy measure, such that the model's performance is observed by attaining maximum accuracy, which is represented by:

$$Fitn = \begin{bmatrix} f_1 \\ \vdots \\ f_i \\ \vdots \\ f_h \end{bmatrix}_{h \times 1} = \begin{bmatrix} f(F_1) \\ \vdots \\ f(F_i) \\ \vdots \\ f(F_h) \end{bmatrix}_{h \times 1} \quad (35)$$

where $Fitn$ denotes the vector obtained from the objective function, and f_i represents the objective value obtained from the i^{th} solution.

c) Enhancement parameter consideration

The SEnO algorithm moves toward the optimal solution for obtaining a better solution with two decision factors, and it can be expressed as:

$$M_i^{(t+1)} = \kappa M_i^{(t)} \quad (36)$$

$$V_i^{(t+1)} = V_i^{(0)} [1 - \exp(-\lambda t)] \quad (37)$$

where $M_i^{(t+1)}$ $M_i^{(t)}$ represents the learning rate factor

at $(t+1)^{th}$ t^{th} iterations, V_i^{t+1} shows pulse emission

at $(t+1)^{th}$ iteration, $V_i^{(0)}$ represents the first pulse emission, κ, λ are the parameters that are used to improve the decision, and they fall in the range of $[0 < \kappa < 1], [0 < \lambda < 1]$.

Case (i): Exploration Phase: When $(Ra < V_i^t)$ Whenever the random variable Ra ranges among $[0, 1]$, this criterion occurs, the exploration phase will be implemented.

Subcase (i): Wide search based on Velocity: If $(S < 0.5)$, when the random values fall in the middle of $[0, 1]$ and < 0.5 , this case is considered. The wide region searching capability is improved to the extended global search and making the solution wider to find the best optimal solution. This uses a velocity factor to speed up the exploration phase to find the best regions correspondingly. Therefore, the characteristics of a positional update can be expressed as:

$$F_i^{(t+1)} = ve_i^{(t+1)} + F_i^{(t)} \quad (38)$$

where the previous iteration solution's position is denoted as $F_i^{(t+1)}$, the current iteration solution position is denoted as F_i^t and the wide search parameter is denoted by:

$$F_{ig}^t = F_{i,q}^t \exp\left(\frac{-i}{\kappa \cdot t \max}\right) \quad (39)$$

where κ is the constant parameter, and the velocity parameter can be expressed as:

$$ve_i^{(t+1)} = ve_i^{(t)} + \left(F_i^{(t+1)} - F_{best}^{(t)} \right) \cdot fr \quad (40)$$

where fr represents the frequency parameter, and it can be expressed as:

$$fr = fr_{min} + (fr_{max} - fr_{min}) \cdot \rho \quad (41)$$

where $\rho \in (0, 1)$ it is a random vector which is taken from the uniform distribution. fr_{min}, fr_{max} is the minimum and maximum frequency of pulse emissions.

Subcase (ii): Random Search: If $Ra \geq 0.5$ Here, search variables fall among the values (0, 1), where it is higher than or the same value as 0.5, then this case will be implemented. The updated position will be mathematically expressed as:

$$F_i^{(t+1)} = F_{rand}^{(t)} - Dis \cdot w \quad (42)$$

where Dis refers to the distance between the target and solution, and w denotes the iteration period. Therefore, the iterations and the distance are represented as:

$$Dis = \left| 2\Phi \cdot F_{rand}^{(t)} - F_i^{(t)} \right| \quad (43)$$

$$w = 2 \left(1 - \frac{y}{y_{max}} \right) \quad (44)$$

where the random integer with the value of [0,1] is expressed as Φ , and the maximum number of iterations and the current iteration are represented as y, y_{max} . Furthermore, the updated new position equation is denoted by:

$$F_i^{t+1} = \kappa \left[\frac{1}{Re} * (Re - |z|) Fit(F_i) + F_i^{(t)} \right] + (1 - \kappa) \left[F_{rand}^{(t)} \cdot Dis w \right] \quad (45)$$

where κ act as a hybridization factor, which is in the range of (0, 1), Re is the alternative index. By the above Eq. (43), the positions are updated, and it traverses the current search space, which improves the chance of getting Persistent local optima are reduced.

explores the new search space, and the chances of getting stuck in the local optima are minimized.

Case (ii)-Exploitation Phase: If $(Pa \geq F_i^t)$, this phase will be introduced where the randomly cre-

ated value is superior to or equal to pulse emissions. The updation of the solution depends on the average loudness and the randomness of the solution. The slower convergence rate and the exploitation phase issues are overcome by generating a new solution and which is represented by:

$$F_i^{(t+1)} = \partial \left[F_i^{(t)} + \varpi \cdot Avg^{(t)} \right] + (1 - \partial) \left[\left| F_{opt}^{(t)} - F_i^{(t)} \right| \cdot \cos(2\pi R_5) + F_{opt}^{(t)} \right] \quad (46)$$

where $Avg^{(t)}$ implies the average of all learning rate parameters, ∂ denotes the hybridization parameter in the range of [0,1], ϖ denotes the random integer in between the value of [-1, 1], $F_{opt}^{(t)}$ is the current iteration optimal solution, and R_5 denotes the arbitrary integer with the value of -1 and 1. Eq. (44) can be updated through the quick movement by considering the random variable that is greater than the threshold value, which enables the solution to quickly reach the optimal region of the exploitation phase. Therefore, the updated solution can be mathematically represented by,

$$F_i^{(t+1)} = F_i^t + A \cdot B + \partial \left[F_i^t + \varpi Avg^{(t)} \right] + (1 - \partial) \left[\left| F_{opt}^{(t)} - F_i^{(t)} \right| \cdot \cos(2\pi R_5) + F_{opt}^{(t)} \right] \quad (47)$$

where A denotes a random number that obeys the normal distribution, and B represents a matrix, and each element inside is 1. Therefore, the introduction of the quick movement factor within the exploitation phase enables the solution to find a better optimal region. Hence, the SEnO algorithm finds the best solution by searching and exploring the location. Thus, the computational complexity and the best solution delivered within the minimum time by the algorithm.

d) Termination

The SEnO algorithm halts execution once a predefined condition is met, indicating convergence, and it then returns the most optimal solution. The flowchart outlining the SEnO algorithm's steps is shown in Figure 8.

V. Results and Discussion

This section describes the results obtained by the SEnO-DTCBiNet model, the evaluation metrics utilized to evaluate the SEnO-DTCBiNet model, the dataset description, and the experimental setup.

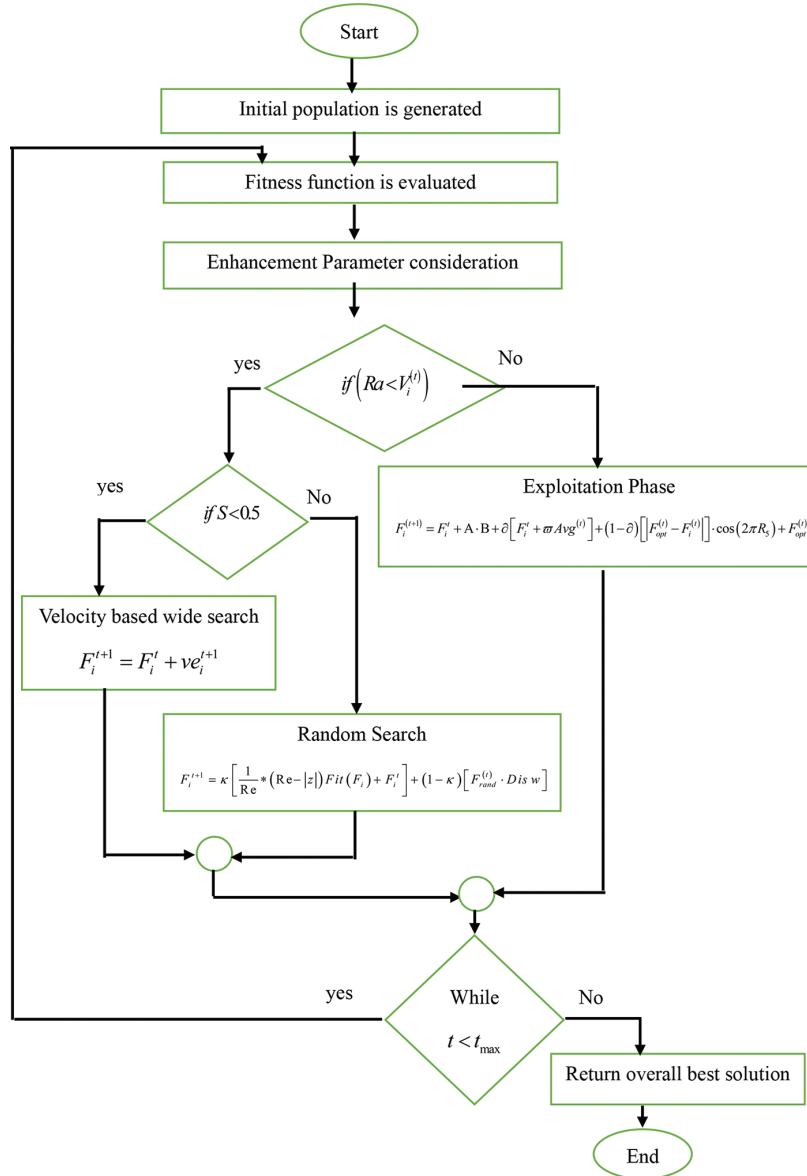


Figure 8: The flowchart of the SEnO algorithm.

a. Experimental setup

The SEnO-DTCBiNet framework was developed using Python 3.7 on a Windows 11 platform. The system setup includes a 13th-generation Intel Core i7-13770K processor, 16 GB of RAM, and an Nvidia GeForce RTX 3080 Ti GPU with 12 GB of dedicated memory. Development was carried out using the PyCharm Community Edition, with storage handled by a 128 GB ROM. The hyperparameters used by the SEnO-DTCBiNet model for classification are displayed in Table 3.

b. Dataset description

Datasets included in the research are described below:

BT dataset [17]: It consists of three different kinds of BT with 3064 images. Moreover, the collected images are split into four subgroups, and they are stored in 4.zip files, respectively.

BraTS 2020 dataset [18]: It comprises MR images from 369 patients, and four types of MR images were gathered from each patient for the training set. There are four kinds of images present in this dataset.

Table 3: Hyperparameters of the SEnO-DTCBiNet model

Hyperparameters	Values
Kernel size	(3 × 3)
Pooling	(2,2)
Convolution 2D layers	2
Activation function	ReLU
Learning rate	0.02
Dropout rate	0.5
No. of BiLSTM layers	2
LSTM Units	64
Loss function	Categorical-cross entropy
Optimizer	Adam
Number of epochs	100
Pooling	MaxPooling2D
Metrics	Accuracy
Padding	Same
Stride size	2

SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network.

Moreover, the validation set consists of MR images of 125 patients, and these images are collected from the various 19 institutes with several clinical protocols.

BraTS 2018 dataset [19], the training set comprises 285 patients; among them, 210 patients have high-grade glioma and 75 patients have low-grade glioma. Likewise, the validation set includes 66 patients with unknown-grade BTs. The dataset visualization is illustrated in Figure 9.

c. Evaluation metrics

Precision: It measures the percentage of the SEnO-DTCBiNet model's all positive classifications, and it is denoted as:

$$pre = \frac{te_{ps}}{te_{ps} + fl_{ps}} \quad (48)$$

where *pre* indicates the precision measure, the true positive is represented as te_{ps} , and the false positive is denoted as fl_{ps} .

Recall: It measures the percentage of all positives, which are classified correctly by the SEnO-DTCBiNet model as positive, and it is represented as:

$$re = \frac{te_{ps}}{te_{ps} + fl_{ng}} \quad (49)$$

where fl_{ng} is the false negative, and *re* indicates the recall measure.

Accuracy: It measures the SEnO-DTCBiNet model's overall correctness in classifying data, and it is represented as:

$$acc = \frac{te_{ps} + te_{ng}}{te_{ps} + te_{ng} + fl_{ps} + fl_{ng}} \quad (50)$$

where accuracy is denoted as *acc*, and the true negative is presented as te_{ng} .

Fi-score: It estimates the performance of the SEnO-DTCBiNet model by the harmonic mean of recall and precision, which is denoted as:

$$F1 = 2 \times \frac{pre \times re}{pre + re} \quad (51)$$

where *F1*-measure is signified as *F1*.

d. Experimental results

The image results of the SEnO-DTCBiNet model across the BraTS 2018 dataset, BraTS 2020 dataset, and the BT dataset are represented in Figures 10–12, respectively.

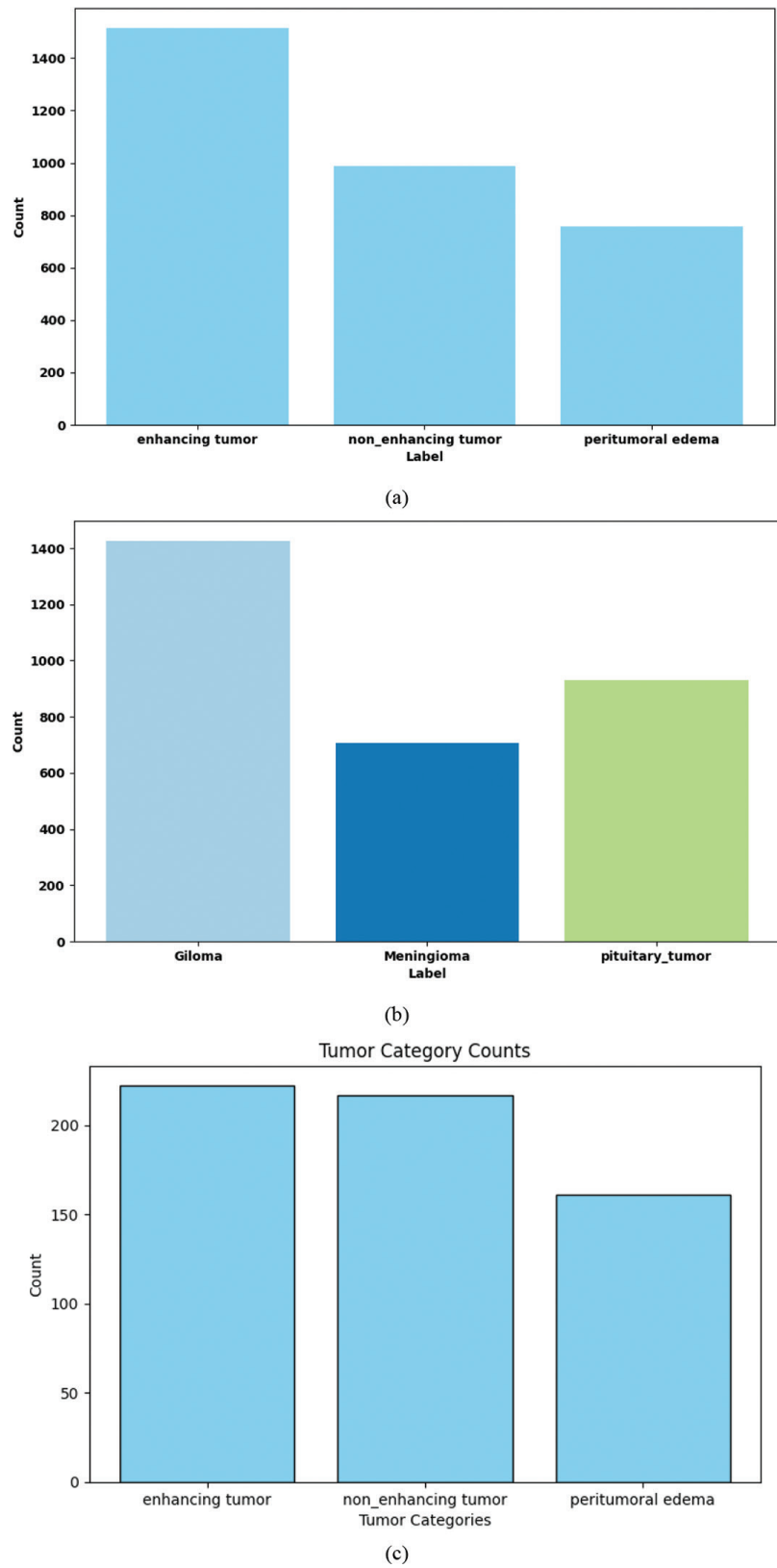
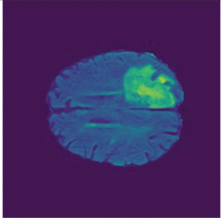
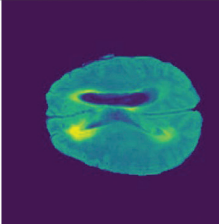
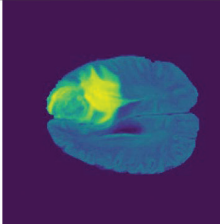
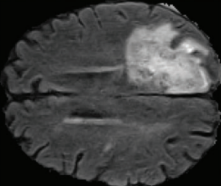
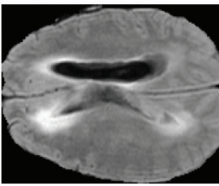
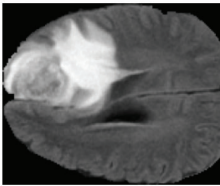
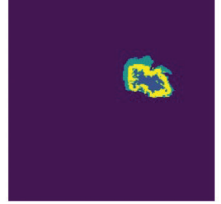
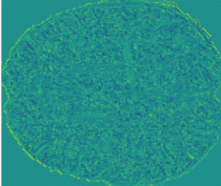
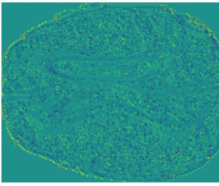
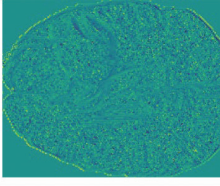
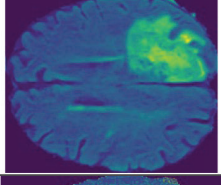
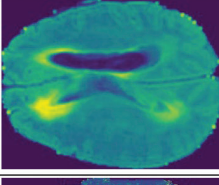
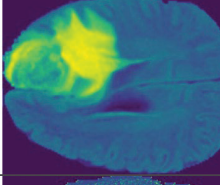
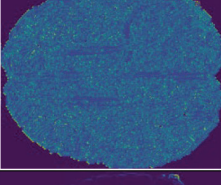
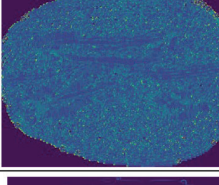
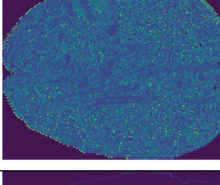
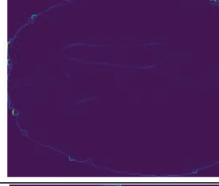
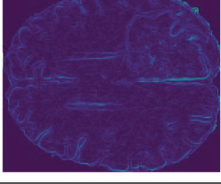
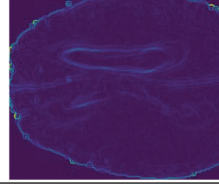
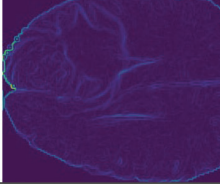


Figure 9: Dataset visualization. (A) BraTS 2020 dataset, (B) BT dataset, and (C) BraTS 2018 dataset. BT, brain tumor.

			Non-Enhancing Tumor	Enhancing Tumor	Peritumoral Edema
Input Image					
ROI extracted Image					
Segmented Image					
Extracted Features	Statistical Features	Skew			
		Mean			
		Kurtosis			
		Variance			
		Standard Deviation			

(Continued)

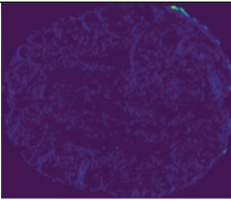
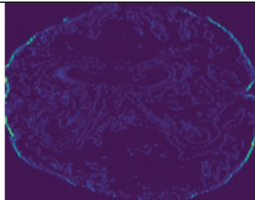
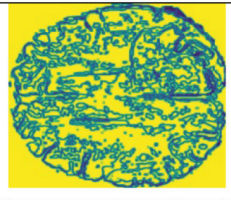
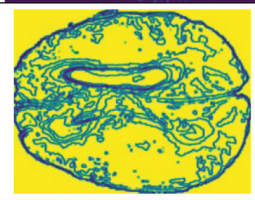
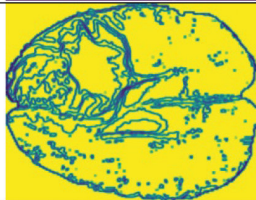
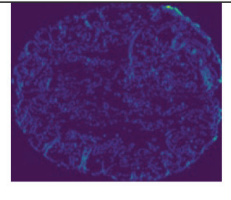
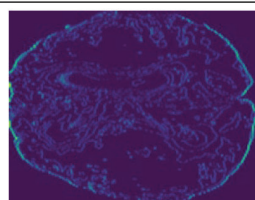
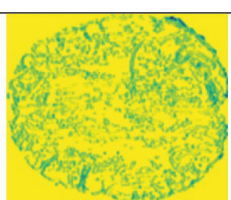
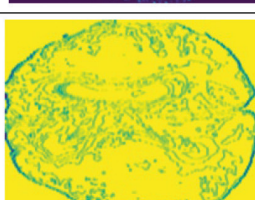
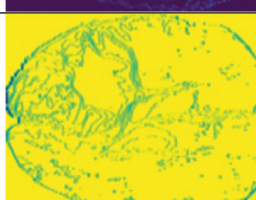
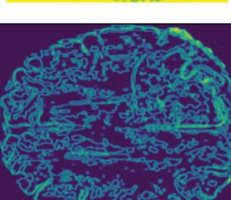
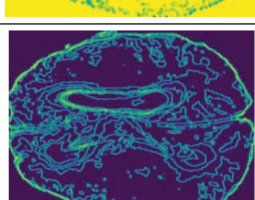
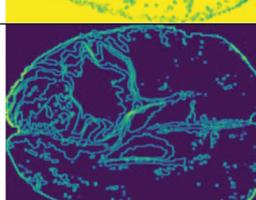
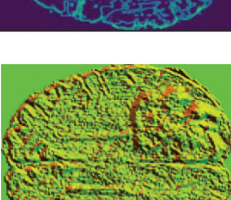
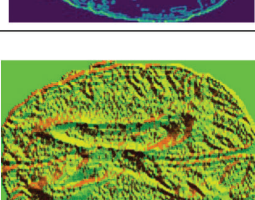
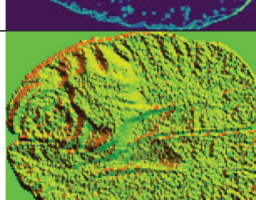
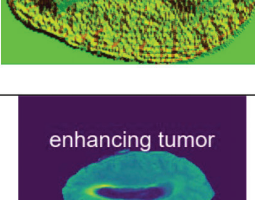
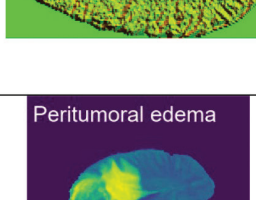
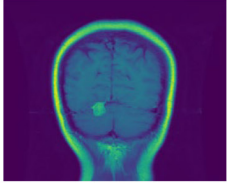
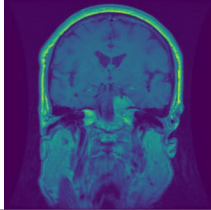
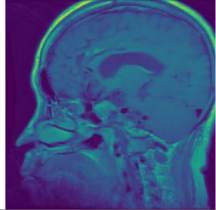
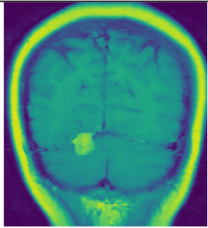
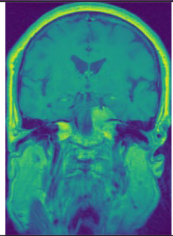
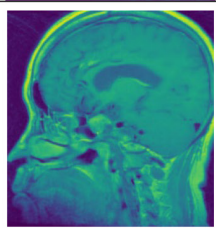
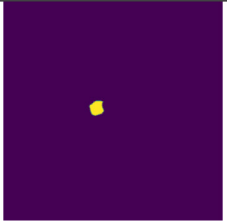
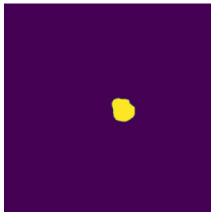
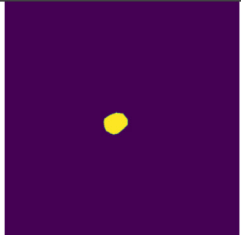
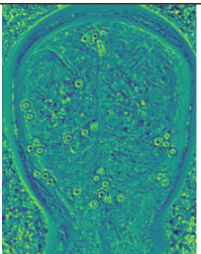
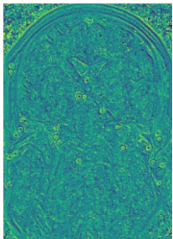
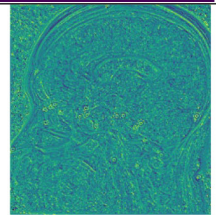
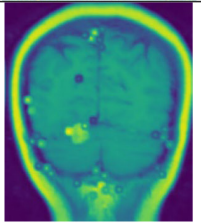
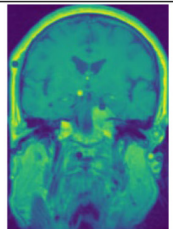
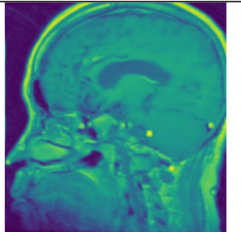
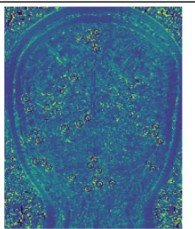
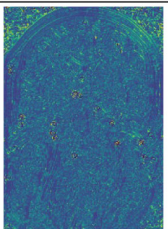
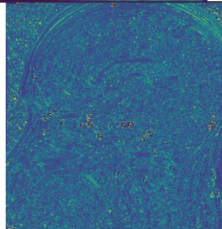
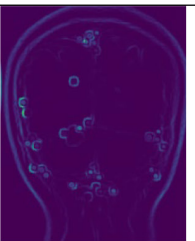
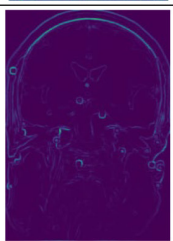
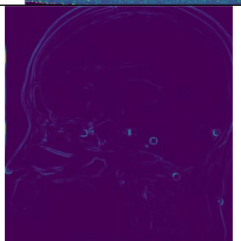
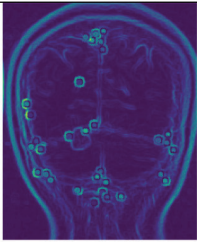
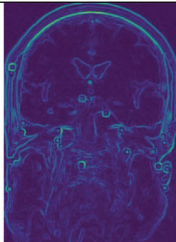
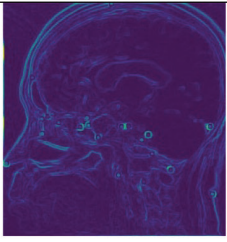
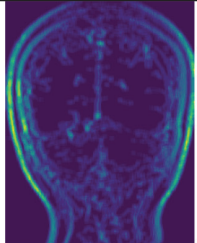
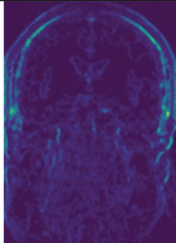
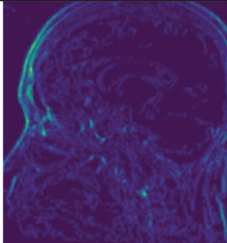
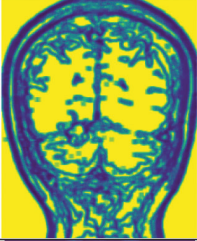
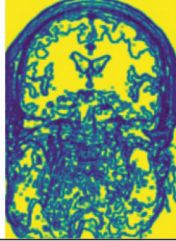
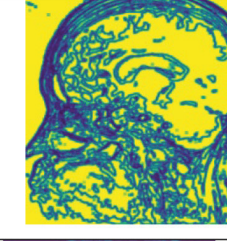
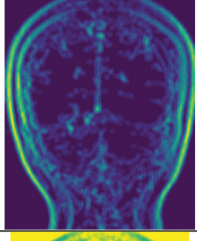
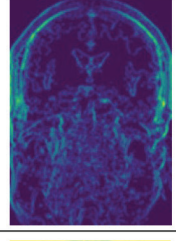
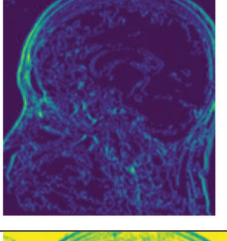
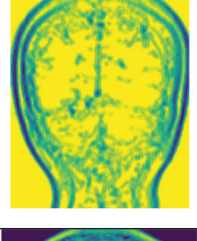
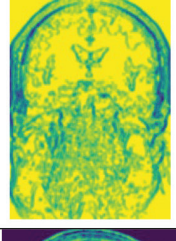
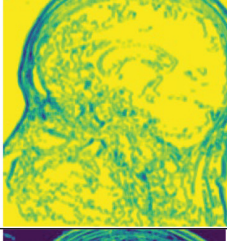
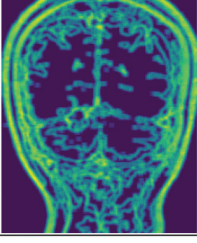
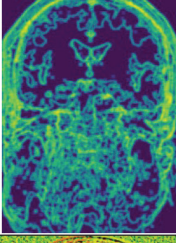
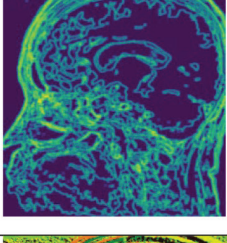
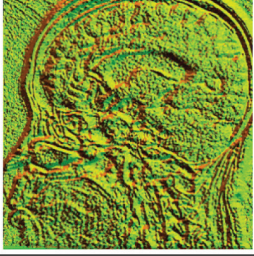
GLCM	Contrast			
	Energy			
	Dissimilarity			
	Homogeneity			
	Entropy			
	Hybrid 3d Pattern			
Output Image		non enhancing tumor 	enhancing tumor 	Peritumoral edema 

Figure 10: Image results based on the BraTS 2020 dataset. ROI, region of interest.

			Glioma	Pituitary Tumor	Meningioma
Input Image					
ROI extracted Image					
Segmented Image					
Extracted Features	Statistical Features	Skew			
		Mean			
		Kurtosis			
		Variance			

(Continued)

GLCM	Standard Deviation			
	Contrast			
	Energy			
	Dissimilarity			
	Homogeneity			
	Entropy			
	Hybrid 3d Pattern			

(Continued)

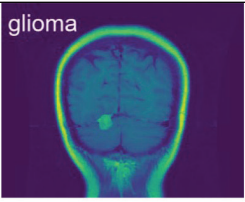
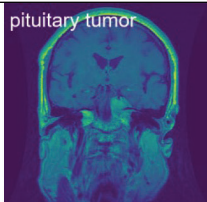
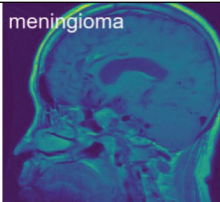
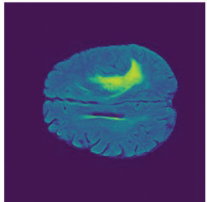
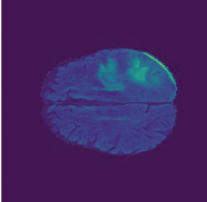
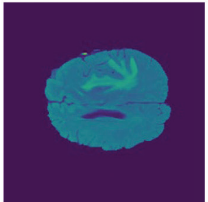
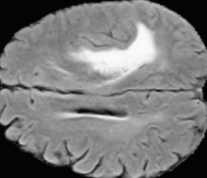
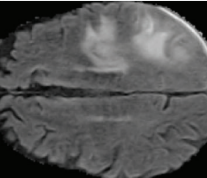
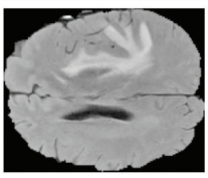
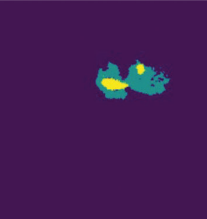
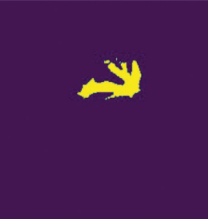
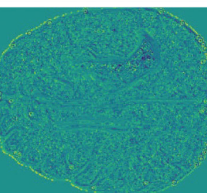
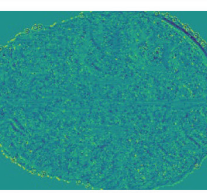
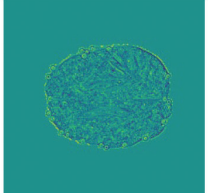
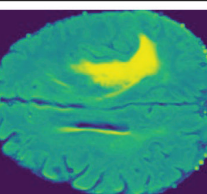
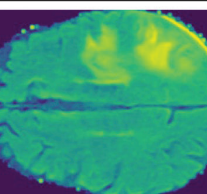
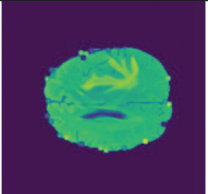
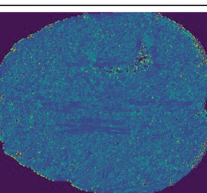
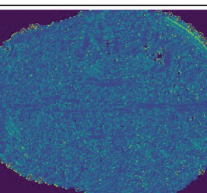
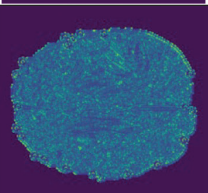
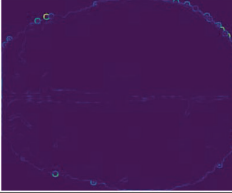
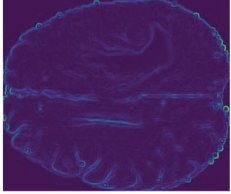
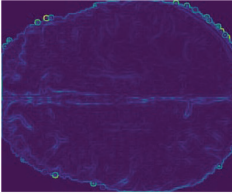
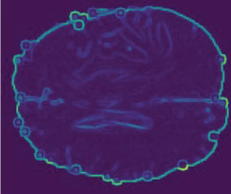
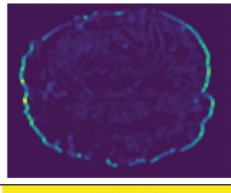
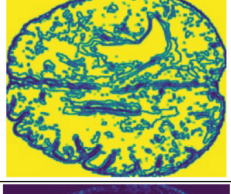
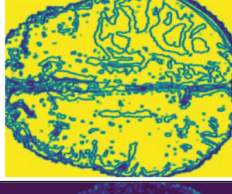
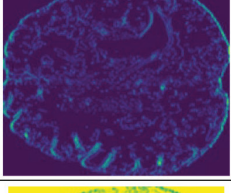
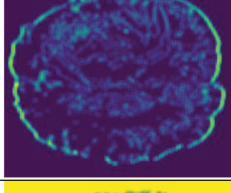
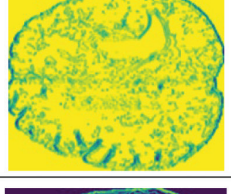
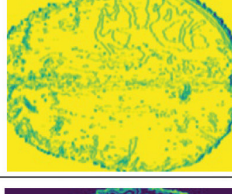
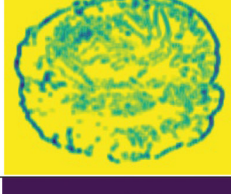
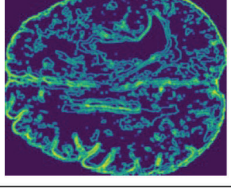
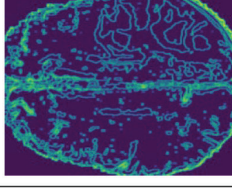
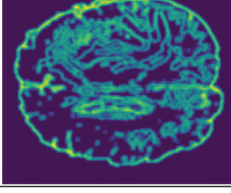
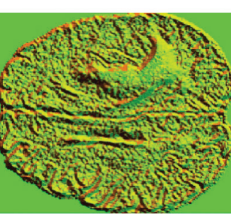
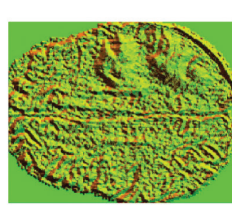
Output Image	glioma	pituitary tumor	meningioma
			

Figure 11: BT dataset-based image results. BT, brain tumor; ROI, region of interest.

			Non-Enhancing Tumor	Enhancing Tumor	Peritumoral Edema
Input Image					
ROI extracted Image					
Segmented Image					
Extracted Features	Statistical Features	Skew			
		Mean			
		Kurtosis			

(Continued)

		Variance			
		Standard Deviation			
	GLCM	Contrast			
		Energy			
		Dissimilarity			
		Homogeneity			
		Entropy			
		Hybrid 3d Pattern			

(Continued)

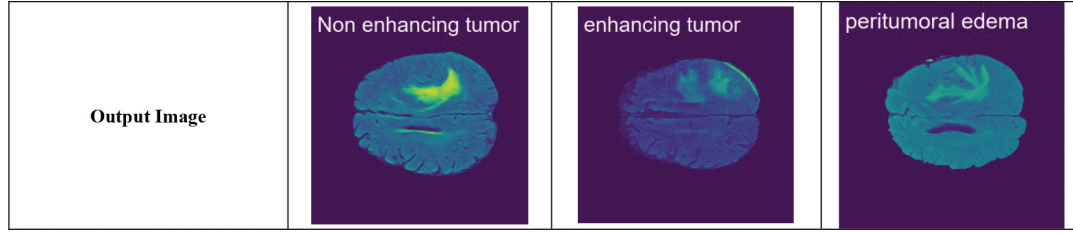


Figure 12: BraTS 2018 dataset-based image results. ROI, region of interest.

e. Performance analysis

The model performance across Brats2020, BT dataset, and the Brats2018 for different amounts of training data and epochs is demonstrated in this section.

e.i BraTS 2020 dataset-based performance analysis

Figure 13 represents the SEnO-DTCBiNet model performance on the BraTS 2020 dataset, assessed using training data. The precision value of the SEnO-DTCBiNet model at epoch 100 with the training percentage of 50%, 70% and 90% is 94.21%, 95.87%, and 97.30%, respectively. The SEnO-DTCBiNet model's recall value at epoch 60 is 94.45% and at epoch 100 is 97.88%. Likewise, the accuracy and F1-score value achieved by the SEnO-DTCBiNet model at epochs 20 to 100 with 90% of training data are 88.59%, 90.00%, 92.49%, 93.59%, 97.17% and 87.06%, 87.36%, 88.03%, 88.27% and 97.25%, respectively.

e.ii BT dataset-based performance analysis

Figure 14 represents the proposed SEnO-DTCBiNet model's effectiveness utilizing the BT dataset. Precision value obtained by the SEnO-DTCBiNet model at epochs 100, 60, and 20 for 90% of the training data is 96.86%, 92.58%, and 91.10%, respectively. Similarly, the recall of the SEnO-DTCBiNet model at epoch 40 is 92.31% and then it increases to 93.84% at epoch 80, and lastly it reaches 98.72% at epoch 100, respectively. The SEnO-DTCBiNet model's accuracy at epochs 20–100 is 92.53%, 93.15%, 94.33%, 95.50%, and 98.48%. The SEnO-DTCBiNet model achieved an F1-score of 96.54%, 97.78%, and 97.78% at epoch 100, corresponding to training percentages of 60%, 80%, and 90%, respectively.

e.iii BraTS 2018 dataset-based performance analysis

The SEnO-DTCBiNet model performance is evaluated under varying training percentages is presented in Figure 15. The precision of the SEnO-DTCBiNet model at epochs 20–100 is 87.45%, 91.38%, 93.50%, 93.75%, and 94.16% for the training percentage of 90%. Similarly, recall and accuracy values obtained by the SEnO-DTCBiNet model at epochs 60 and 100 with 90% of training data are 90.46%, 97.42% and 92.29% and 97.76%, respectively. The SEnO-DTCBiNet model F1-score at epoch 20 is 86.18% and then it rises to 89.83% at epoch 60, and finally it reaches 95.76%, respectively.

f. Comparative methods

The comparative approaches are CJHBA-DRN [4], EDN-SVM [11], 2D-CNN-CAE [15], PatchResNet [7], Bat-Algorithm-based multihead convolutional bidirectional memory (BA-MCBM) [29], Dolphin swarm algorithm-based multihead bidirectional convolutional memory (DE-MCBM) [30], multihead sonar prey optimized convolutional bidirectional memory (SPO-MCBM), and SSA-based distributed tetra head attention-based convolutional bidirectional network (SSA-DTCBiNet) [31].

f.i Comparative analysis on the BraTS 2020 dataset in terms of training percentage

The SEnO-DTCBiNet model performance against several traditional approaches on the BraTS 2020 dataset is presented in Figure 16. The precision of the SEnO-DTCBiNet model is 97.30%, which shows that it is higher than the EDN-SVM by 12.31%, 2D-CNN-CAE by 8.16%, CJHBA-DRN

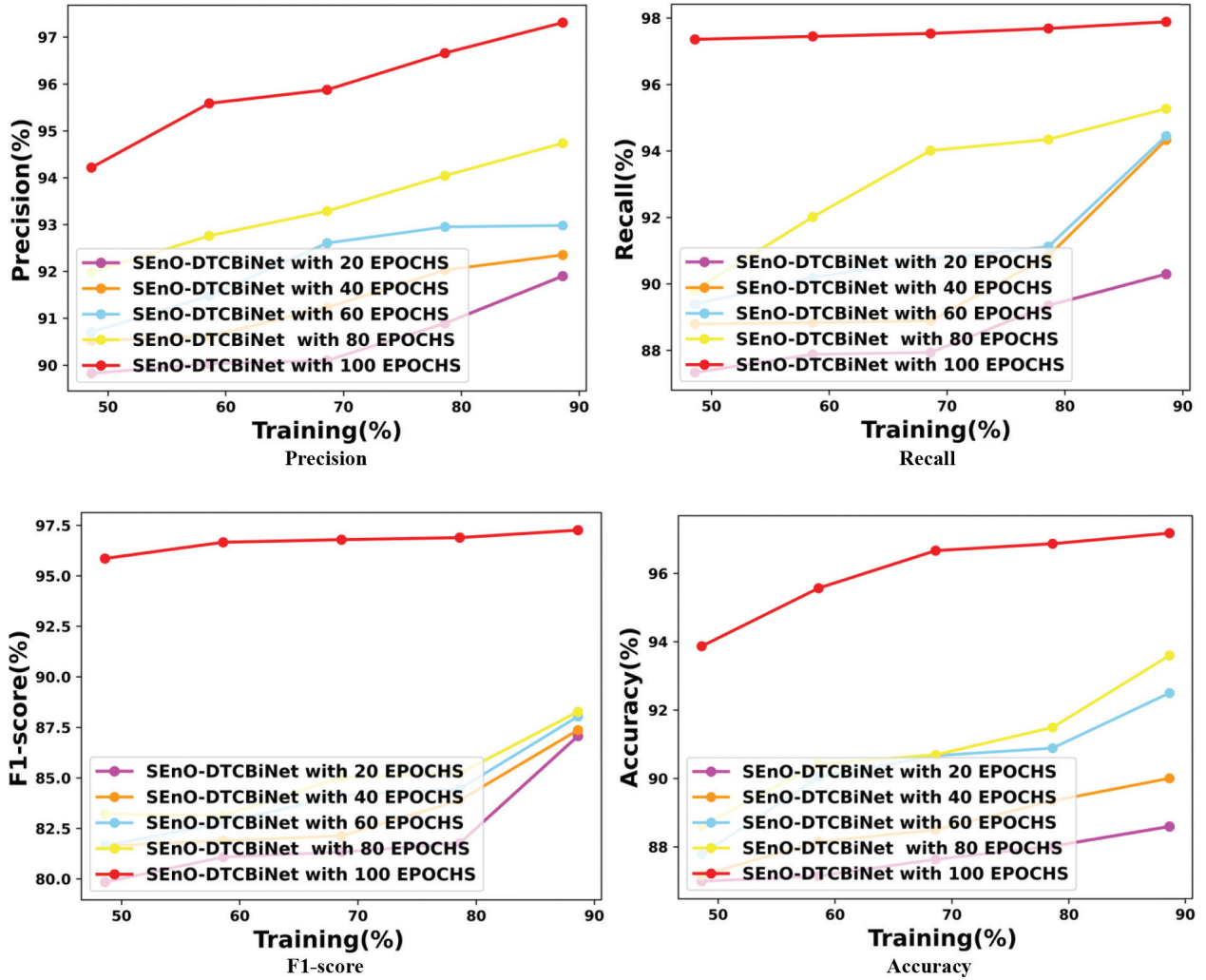


Figure 13: SEnO-DTCBiNet model performance on the BraTS 2020 dataset. SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network.

by 10.25%, BA-MCBM by 7.28%, Dense-CNN by 14.27%, PatchResNet by 9.66%, DE-MCBM by 6.36%, SSA-DTCBiNet by 6.82%, and SPO-MCBM by 1.86% for 90% of the training data. The SEnO-DTCBiNet model achieved a recall value of 97.88%, which indicates an enhancement by 5.06%, 2.45%, 4.61%, 1.85%, 6.11%, 4.24%, 1.68%, 1.76%, and 0.11 for the EDN-SVM, 2D-CNN-CAE, CJHBA-DRN, BA-MCBM, Dense-CNN, PatchResNet, DE-MCBM, SSA-DTCBiNet, and SPO-MCBM, respectively. The accuracy of the SEnO-DTCBiNet model is 97.17%, and it is improved by 10.05% with EDN-SVM, 6.63% with 2D-CNN-CAE, 8.59% with CJHBA-DRN, 6.10% with BA-MCBM, 11.29% with Dense-CNN, 7.49%

with PatchResNet, 3.59% with DE-MCBM, 4.85% with SSA-DTCBiNet, and 0.65 with SPO-MCBM for 90% of the training data. Likewise, the SEnO-DTCBiNet model attained an F1-score of 97.25%, which is an improved value over other comparative approaches.

f.ii K-fold cross-validation using the BraTS 2020 dataset

The SEnO-DTCBiNet model performance against the other conventional models based on K-fold I using the BraTS dataset is shown in Figure 17. K-fold cross-validation is performed for KF-4 to

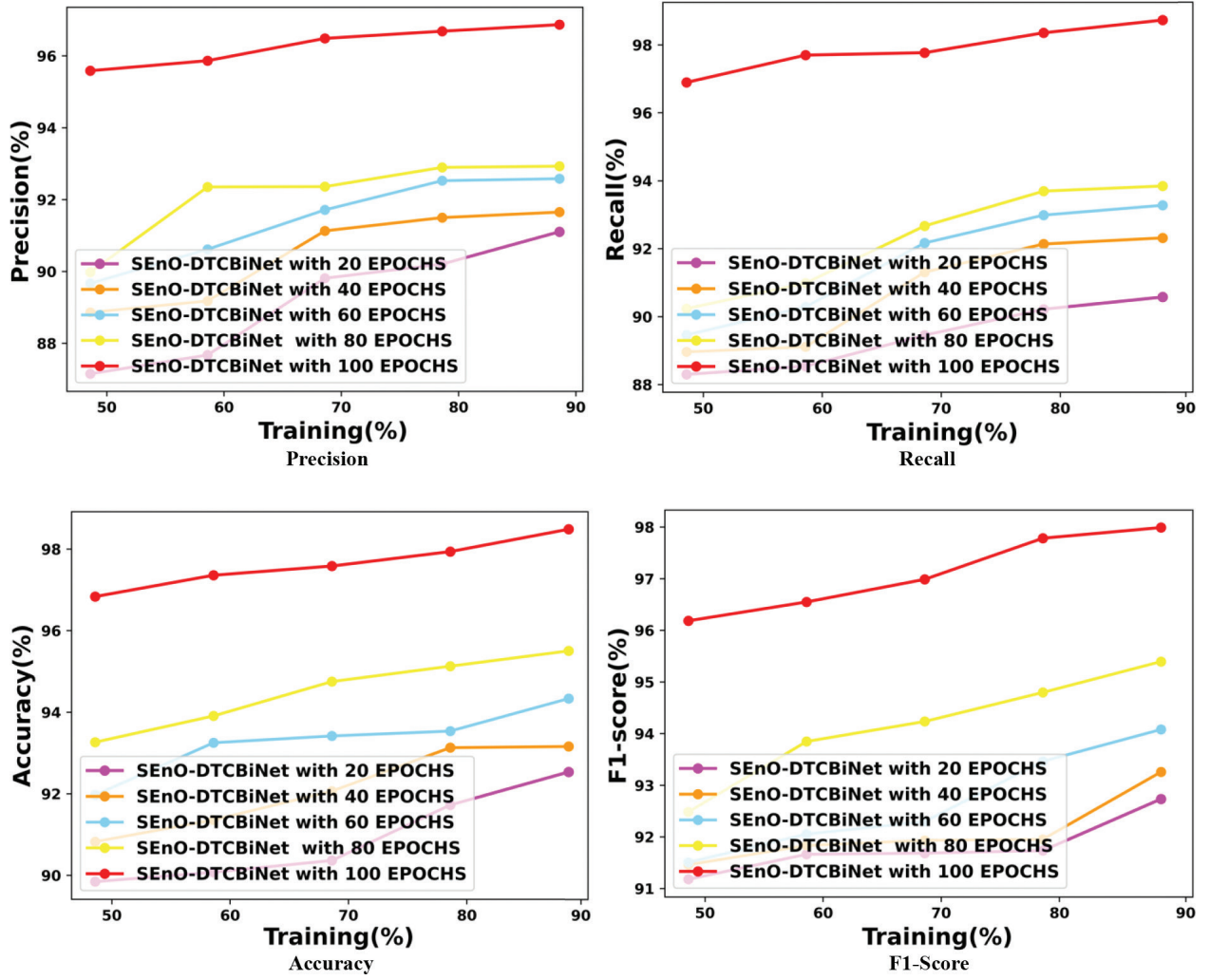


Figure 14: Performance of the SEnO-DTCBiNet model using the BT dataset. BT, brain tumor. SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network.

KF-10 against the baseline models, with the proposed SEnO-DTCBiNet. The SEnO-DTCBiNet model achieved 96.27% of accuracy, which is an improved value than Dense-CNN and EDN-SVM, by 8.92% and 6.31% at a fold of 10. Likewise, in the precision, the proposed model attains a higher value of 95.53%, showing improvement over 5.99% by CJHBA-DRN and 3.44% by PatchResNet. The recall value achieved by the SEnO-DTCBiNet framework is 97.74% showing an enhancement of 3.69% over 2D-CNN-CAE and 2.52% over BA-MCBM, respectively. The F1-score of the SEnO-DTCBiNet model is 96.62%, which is greater than DE-MCBM by 1.18%, SSA-DTCBiNet by 0.32%.

f.iii Training percentage-based comparative analysis using the BT dataset

The comparative results of the SEnO-DTCBiNet model against the baseline model based concerning training Percentage on the BT dataset is presented in Figure 18. The SEnO-DTCBiNet model obtained a precision of 96.86%, which is also higher than other comparative EDN-SVM, 2D-CNN-CAE, CJHBA-DRN, BA-MCBM, Dense-CNN, PatchResNet, DE-MCBM, SSA-DTCBiNet, and SPO-MCBM approaches. The recall of the SEnO-DTCBiNet model is 98.72%, and it is improved over 6.29% with EDN-SVM, 4.03% with 2D-CNN-CAE, 5.83% with CJHBA-DRN, 2.93% with BA-MCBM, 7.06% with Dense-CNN, 5.65%

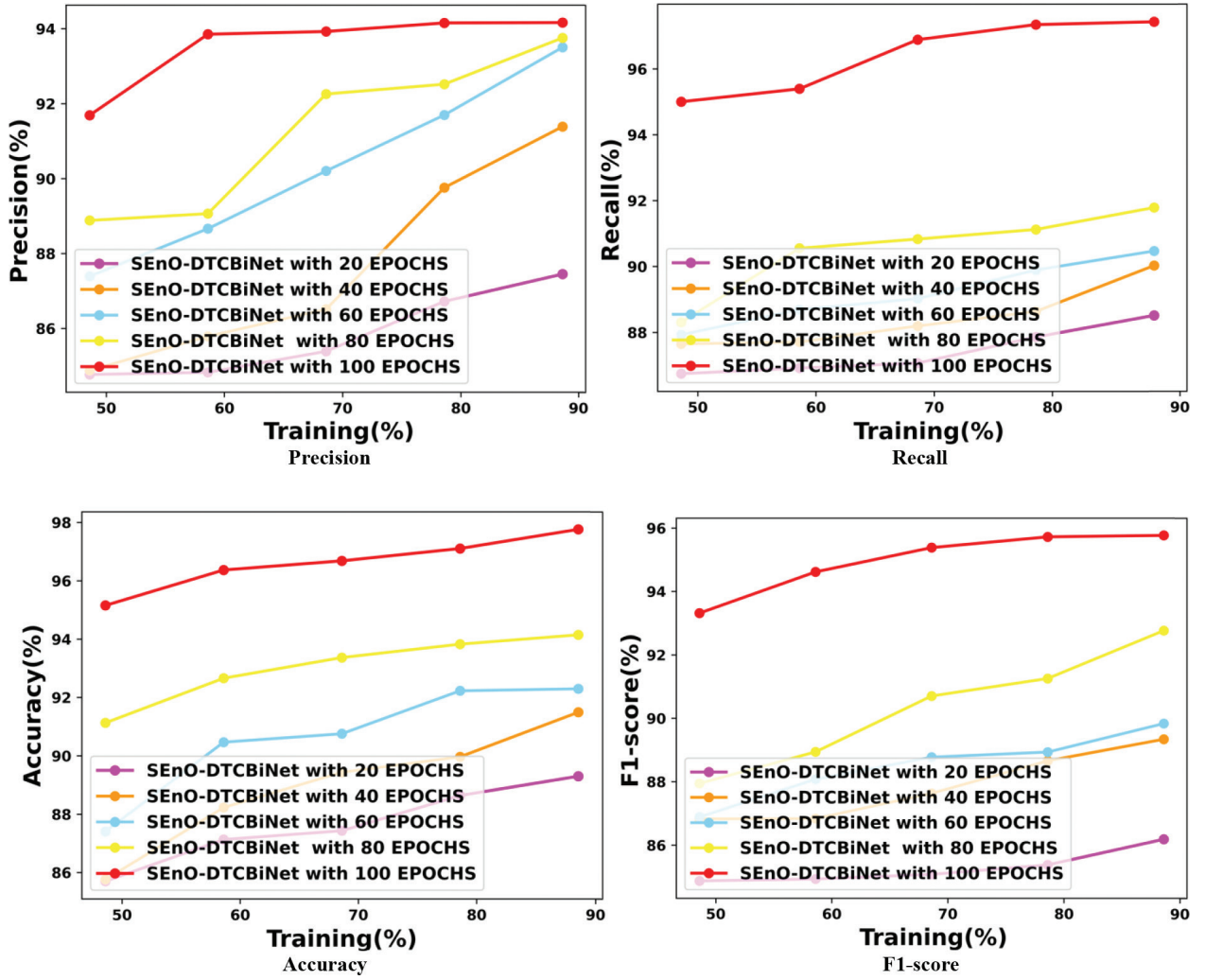


Figure 15: Performance of SEnO-DTCBiNet model using the BraTS 2018 dataset. SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network.

with PatchResNet, 2.61% with DE-MCBM, 1.40% by SSA-DTCBiNet, and 0.20 by SPO-MCBM. The SEnO-DTCBiNet model achieved an accuracy of 98.48%, which shows that it is higher than EDN-SVM by 9.57%, 2D-CNN-CAE by 6.80%, CJHBA-DRN by 7.46%, BA-MCBM by 6.76%, Dense-CNN by 10.06%, PatchResNet by 7.24%, DE-MCBM by 4.23%, SSA-DTCBiNet by 2.40%, and SPO-MCBM by 0.57 for 90% of the training data. The F1-score of SEnO-DTCBiNet model is 97.985, which displays the enhancement by 8.62%, 5.72%, 8.40%, 4.96%, 9.89%, 7.08%, 2.78%, 1.58%, 0.38 for the EDN-SVM,

2D-CNN-CAE, CJHBA-DRN, BA-MCBM, Dense-CNN, PatchResNet, DE-MCBM, SSA-DTCBiNet, and SPO-MCBM, respectively.

f.iv K-fold cross-validation using BT dataset

The proposed SEnO-DTCBiNet model's performance against the other traditional models, based on the K-fold using the BT dataset, is depicted in Figure 19. The proposed SEnO-DTCBiNet model achieved 96.23% of accuracy, showing improved

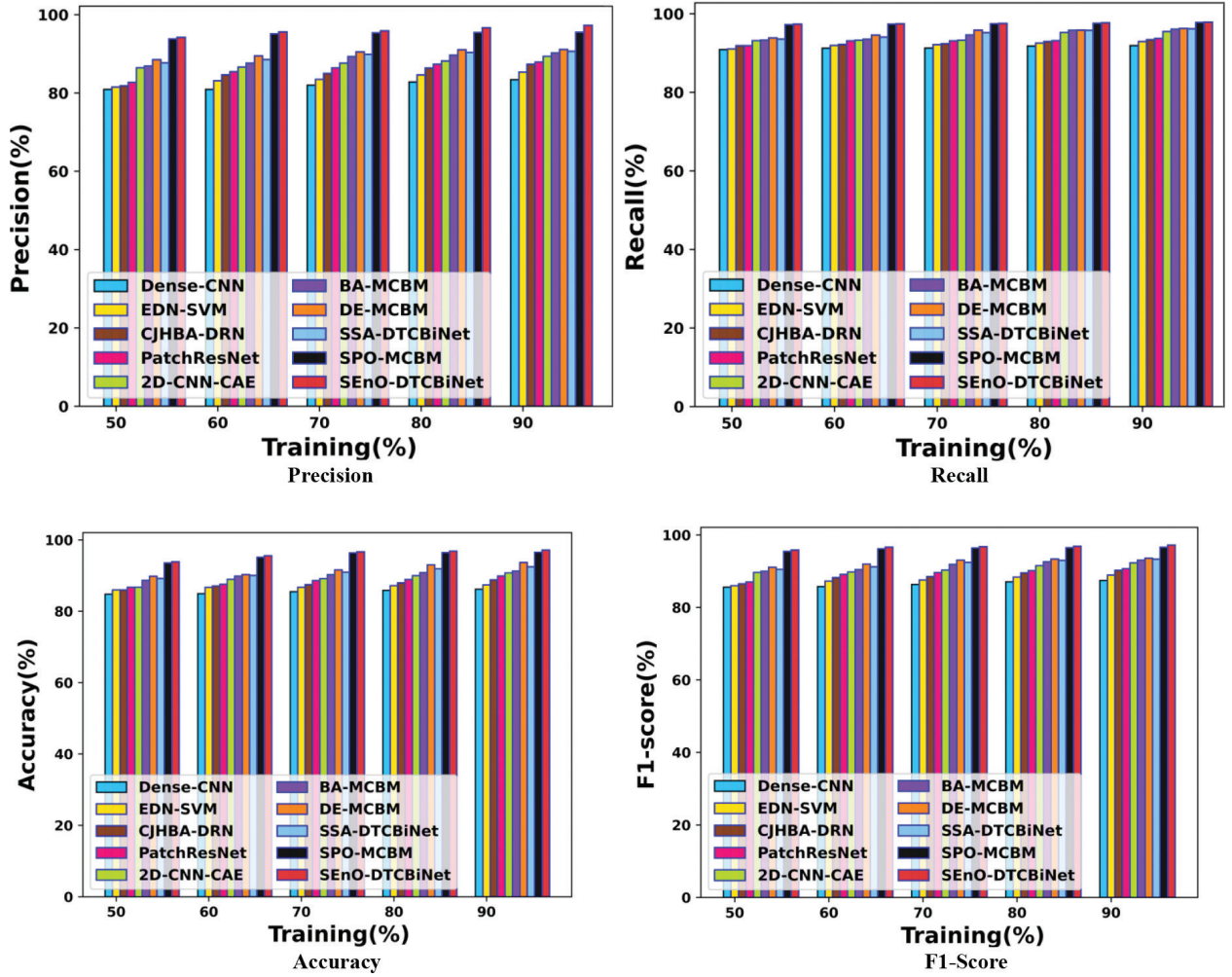


Figure 16: SEnO-DTCBiNet model performance comparison on the BraTS 2020 dataset. SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network.

performance over Dense-CNN by 7.18% and EDN-SVM by 4.92%. Additionally, its precision reached a high value of 95.49%, outperforming CJHBA-DRN by 4.65% and PatchResNet by 4.31%. The SEnO-DTCBiNet framework also demonstrated a superior recall value of 97.71% representing an enhancement of 0.66% and 0.33% over 2D-CNN-CAE and BA-MCBM. Furthermore, the model's F1-score reached 0.81% at fold-10, exceeding DE-MCBM by 1.06% and SSA-DTCBiNet by 0.81%. The proposed SEnO-DTCBiNet framework significantly advances BT segmentation by achieving improved performance. This makes it a more effective tool for analysis compared to the conventional methods.

f.v Comparative analysis based on training percentage using BraTS 2018 dataset

Figure 20 presents the performance comparison of the proposed SEnO-DTCBiNet model which attained of 94.16%, which shows a development by 4.47%, 1.48%, 4.32%, 1.15%, 5.12%, 2.66%, 1.08%, 0.70, and 0.28 for the EDN-SVM, 2D-CNN-CAE, CJHBA-DRN, BA-MCBM, Dense-CNN, PatchResNet, DE-MCBM, SSA-DTCBiNet, and SPO-MCBM. The SEnO-DTCBiNet model recall and accuracy with 90% of training data are 97.42% and 97.76% respectively. The F1-score of SEnO-DTCBiNet model is 95.76%, which is greater than

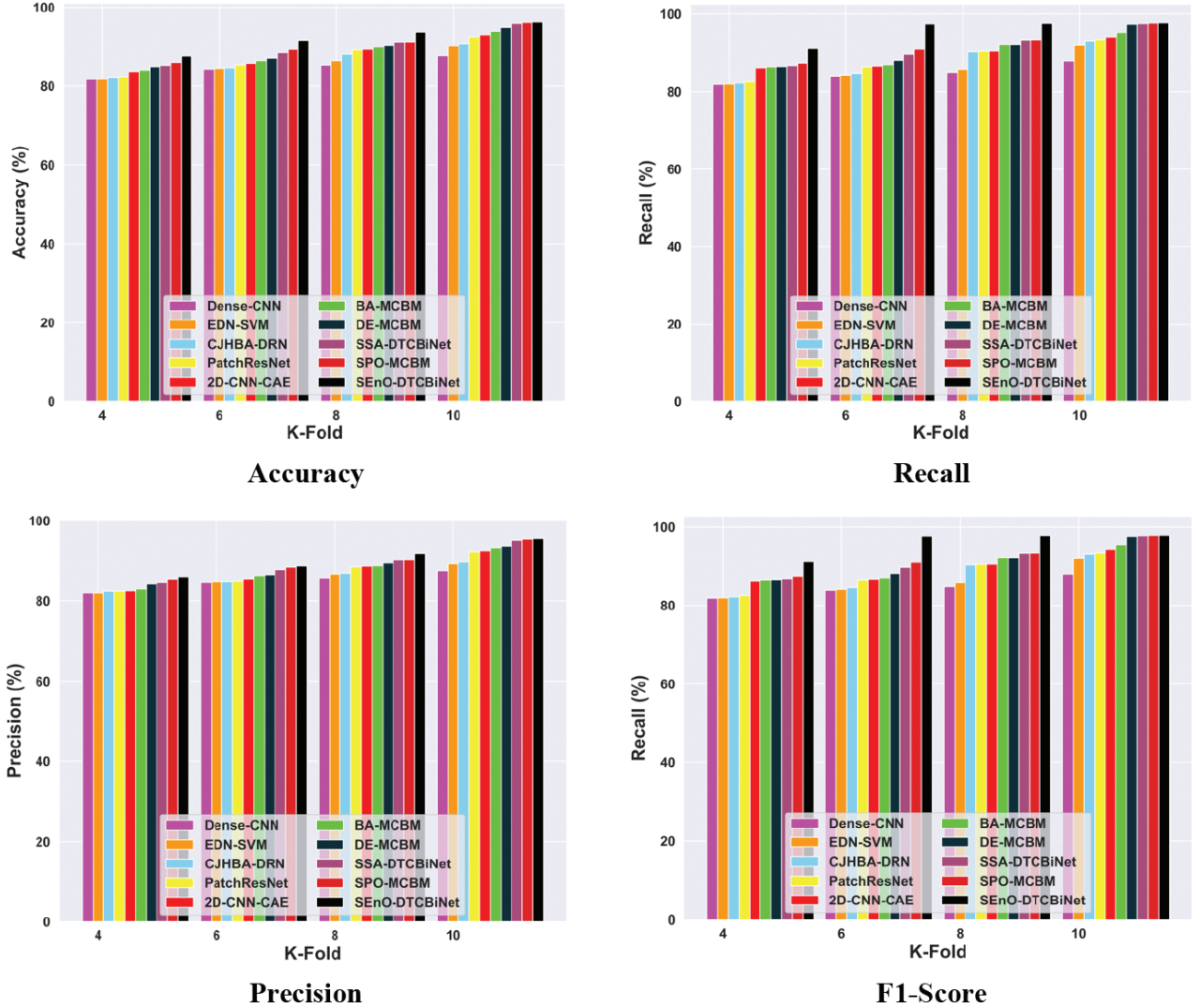


Figure 17: K-fold analysis on the BraTS 2020 dataset.

the EDN-SVM by 5.73%, 2D-CNN-CAE by 3.00%, CJHBA-DRN by 4.57%, BA-MCBM by 2.29%, Dense-CNN by 6.29%, PatchResNet by 3.68%, DE-MCBM by 1.90%, SSA-DTCBiNet by 1.35%, and SPO-MCBM by 0.34.

f.vi K-fold cross-validation using the BraTS 2018 dataset

The model evaluation on the K-fold cross-validation against the baseline models across the BraTS 218 dataset is depicted in Figure 21. Contrasting it with conventional approaches. With a K-fold 10, the SEnO-DTCBiNet model achieved an accuracy of

95.51%, exceeding that of Dense-CNN and EDN-SVM by 6.70% and 5.96% respectively. The precision also registered a higher value of 95.37%, showing an improvement of 3.68% over CJHBA-DRN and 2.45% over PatchResNet. The SEnO-DTCBiNet framework's recall reached 95.79%, outperforming 2D-CNN-CAE by 3.77% and BA-MCBM by 2.75%. Finally, at fold-10, the model secured an F21-score of 96.62%, which is 1.18% greater than DE-MCBM and 0.32% greater than SSA-DTCBiNet. By consistently beating multiple baseline models across different K-folds, the SEnO-DTCBiNet model demonstrates superior performance and robustness.

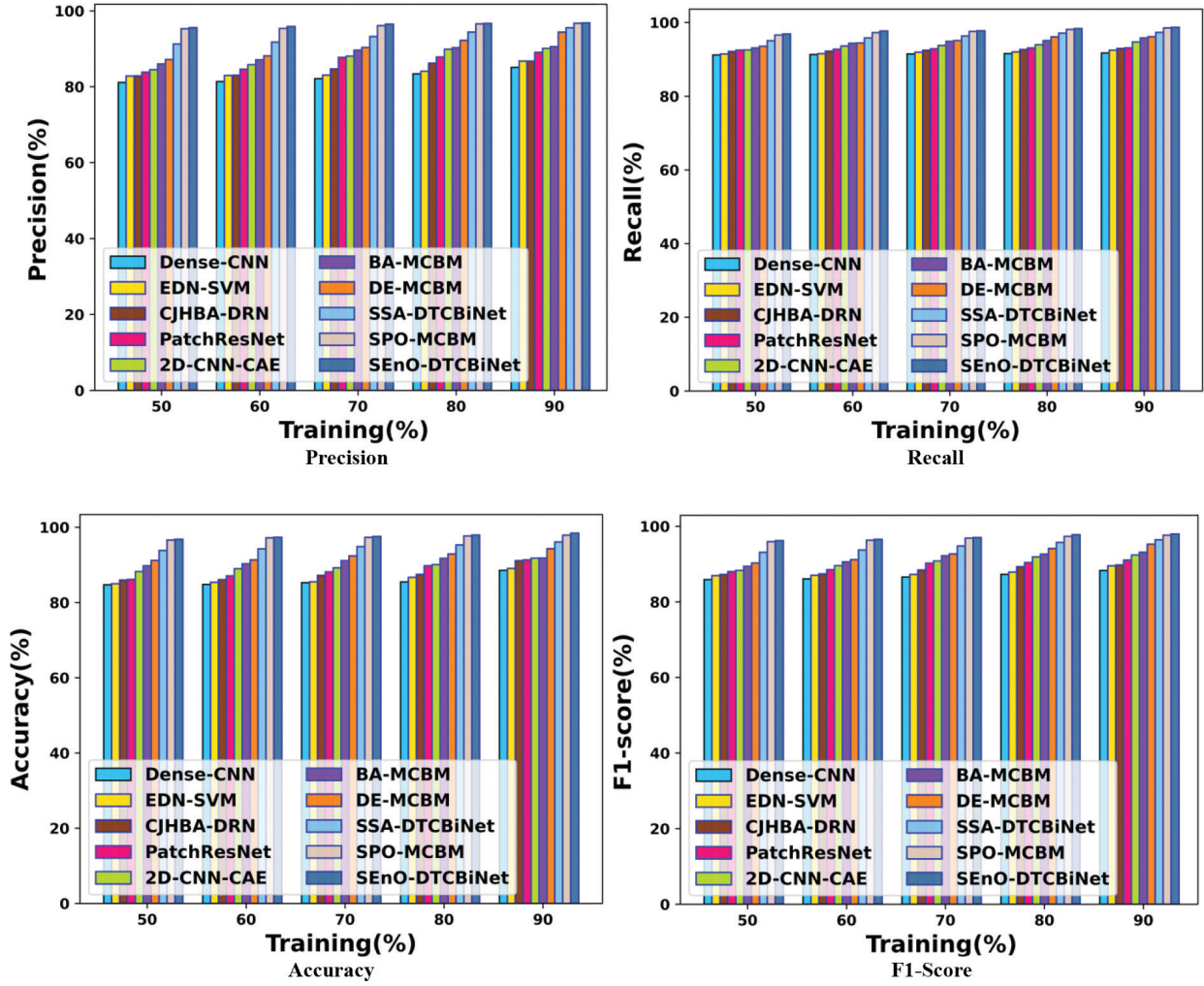


Figure 18: Comparison of the SEnO-DTCBiNet model based on the training percentage using the BT dataset. BT, brain tumor. SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network.

f.vii Comparative analysis based on training percentage using the real-time dataset

The SEnO-DTCBiNet model performance evaluation across the other baseline models using the real-time dataset is presented in Figure 22. The SEnO-DTCBiNet model achieved 96.35% accuracy, performing better than the Dense-CNN by 5.40%, EDN-SVM by 4.46%, and CJHBA-DRN by 4.06% when trained on 90% of the data. The model attains a precision rate of 95.98%, showing improvements over other models such as 2D-CNN-CAE, BA-MCBM, and PatchResNet by 2.10%, 1.81% and 2.82% respectively. The SEnO-DTCBiNet model achieved a recall value of

97.10%, which is 1.25% better than DE-MCBM and 0.21% better than SSA-DTCBiNet. Additionally, the F1-score of the model is 96.54%, with a gain over SPO-MCBM of 0.38%. Overall, these results indicate that the SEnO-DTCBiNet model offers an effective performance enhancement, particularly in the context of BT classification, when compared to conventional models.

f.viii K-fold cross validation using the real-time dataset

Figure 23 represents the SEnO-DTCBiNet model outcomes across the traditional methods based on the training Percentage of the real-time dataset. the SEnO-DTCBiNet model achieved an

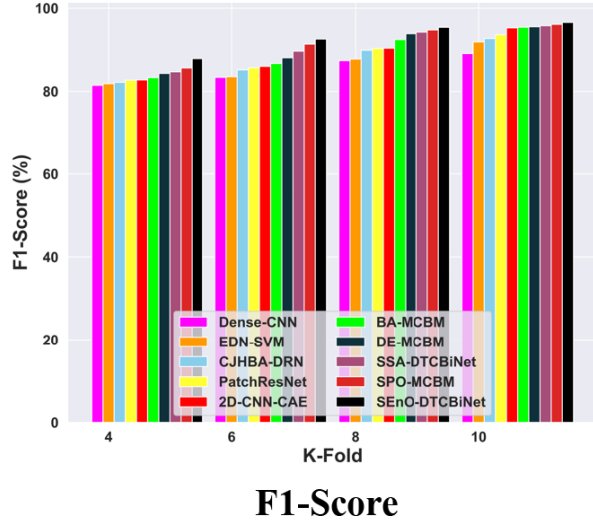
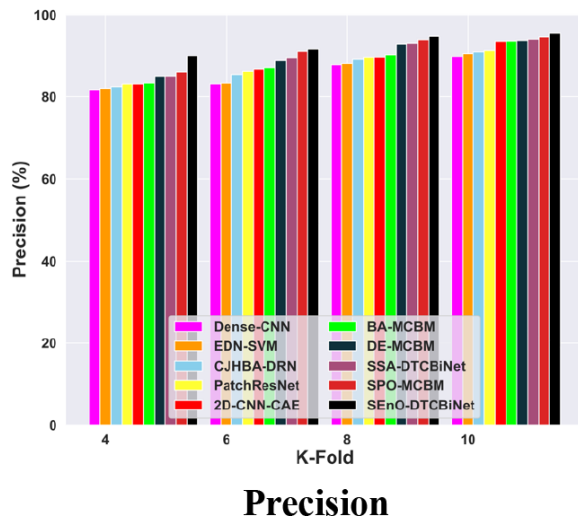
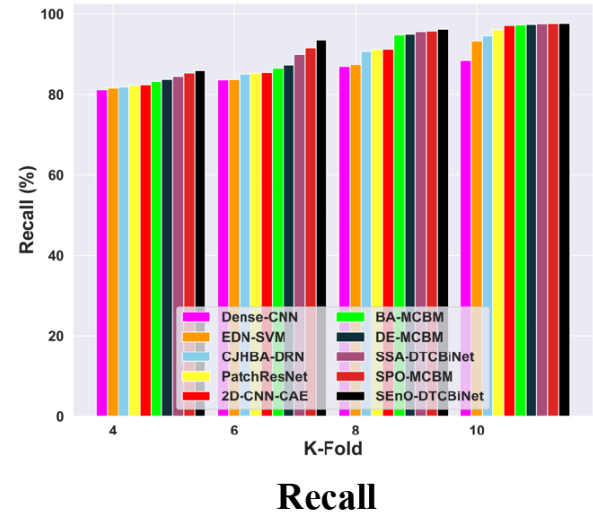
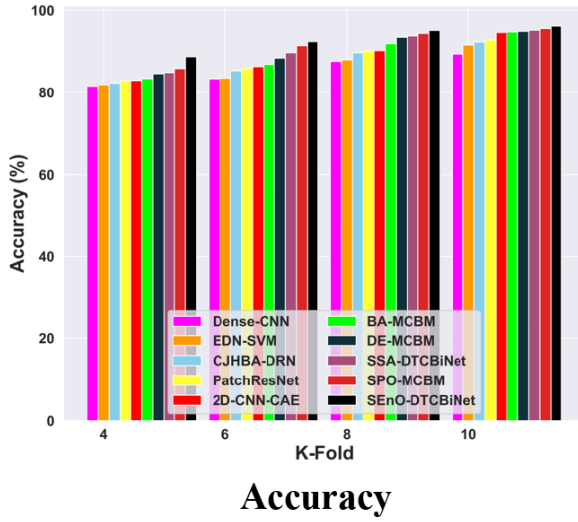


Figure 19: K-fold analysis on the BT dataset. BT, brain tumor.

accuracy of 96.47%, surpassing Dense-CNN by 8.78%, and EDN-SVM by 6.23%, on K-fold 10. Furthermore, the model demonstrated a high precision of 95.78%, outperforming approaches such as 2D-CNN-CAE, BA-MCBM, and PatchResNet by 2.38%, 2.33% and 3.52%. In recall, the SEnO-DTCBiNet reached 97.87%, which is 0.54% higher than DE-MCBM and 0.43% above SSA-DTCBiNet. Its F1-score also shows a higher value, 96.81% presenting an improvement over SPO-MCBM by 0.80%. These results highlight the SEnO-DTCBiNet framework as a significant advancement in performance, particularly in BT classification compared to other conventional models.

g. ROC analysis

Figure 24 portrays a Receiver Operating Curve of the SEnO-DTCBiNet framework against the other comparative models, including CJHBA-DRN, BA-MCBM, 2D-CNN-CAE, Dense-CNN, EDN-SVM, PatchResNet, DE-MCBM, SSA-DTCBiNet, and SPO-MCBM methods based on three datasets. Based on this ROC analysis, it indicates that the SEnO-DTCBiNet model performs significantly to the other comparative methods.

h. Time complexity analysis

The Time Complexity of the SEnO-DTCBiNet model against other existing methods using the BraTS2019,

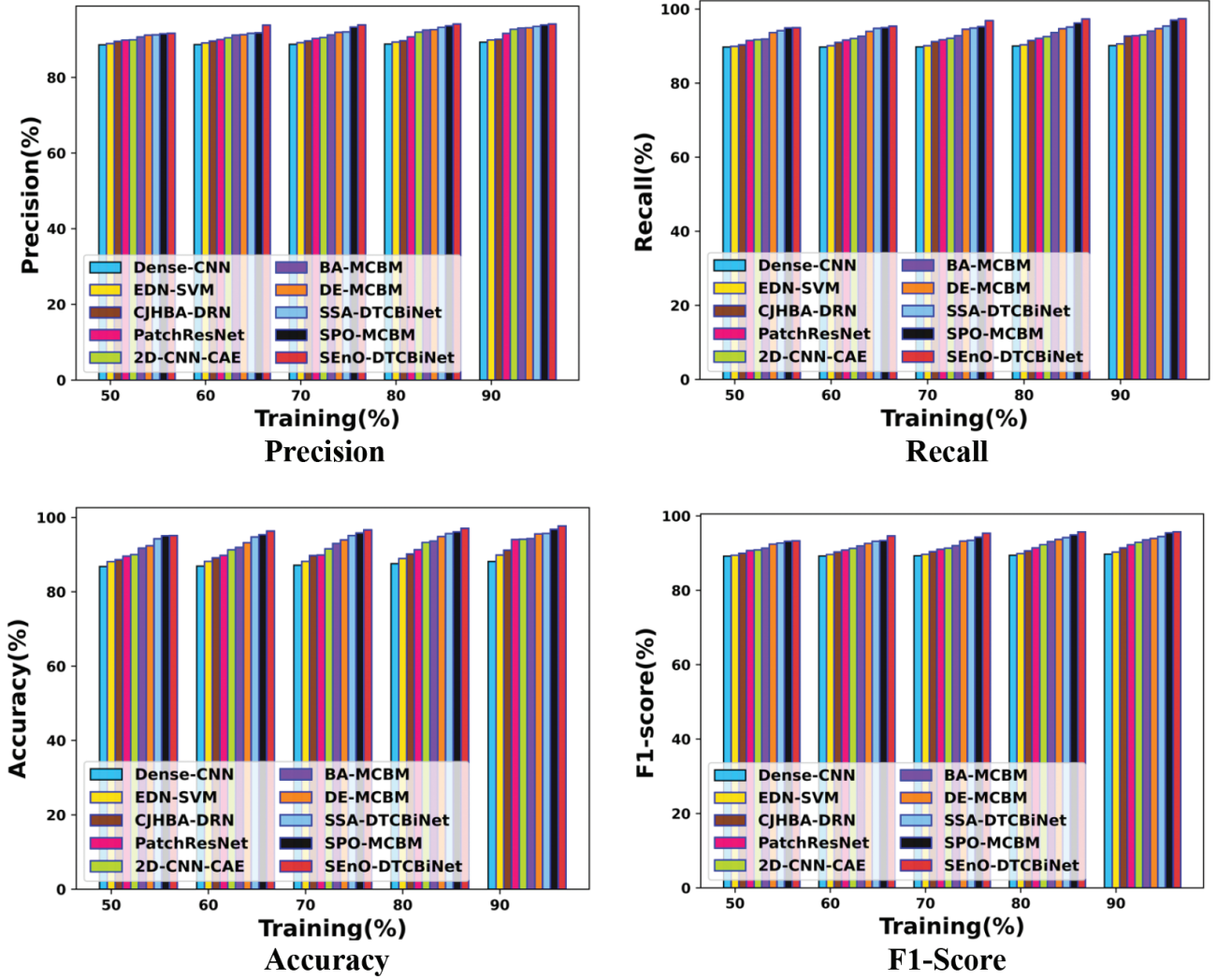


Figure 20: Comparison of the SEnO-DTCBiNet model based on training percentage using the BraTS 2018 dataset.

BT, and the BraTS2020 datasets is depicted in Figure 25. The analysis reveals that the SEnO-DTCBiNet model requires less time for tumor classification as the number of iterations increases, outperforming the comparative approaches.

i. Convergence analysis

The convergence analysis of the SEnO optimization compared to approaches such as RMsProp, Adam, BA, SSA, DEA, and SPO is depicted in Figure 26. The analysis indicates that SEnO's loss value decreases more effectively than its counterparts as the number of epochs increases. At epoch 50, the SSA and DEA methods reach a loss of 6.73×10^{-07} , 3.13×10^{-06} , whereas the proposed optimization

reaches a lower loss of 1.00×10^{-06} , respectively. Similarly, at epoch 99, the proposed optimization remains the lower loss value of 1.00×10^{-06} , whereas the existing approaches, RMsProp, Adam, and BA reach a loss of 0.630, 0.002, and 0.001. These findings therefore demonstrate the effectiveness of the SEnO optimization in achieving an optimal solution for classifying the BT.

j. Computational complexity analysis in terms of floating-point operations per second (FLOPS)

Figure 27 presents the computational complexity of the proposed brain tumor detection model and various other methods, using FLOPS as the single arithmetic

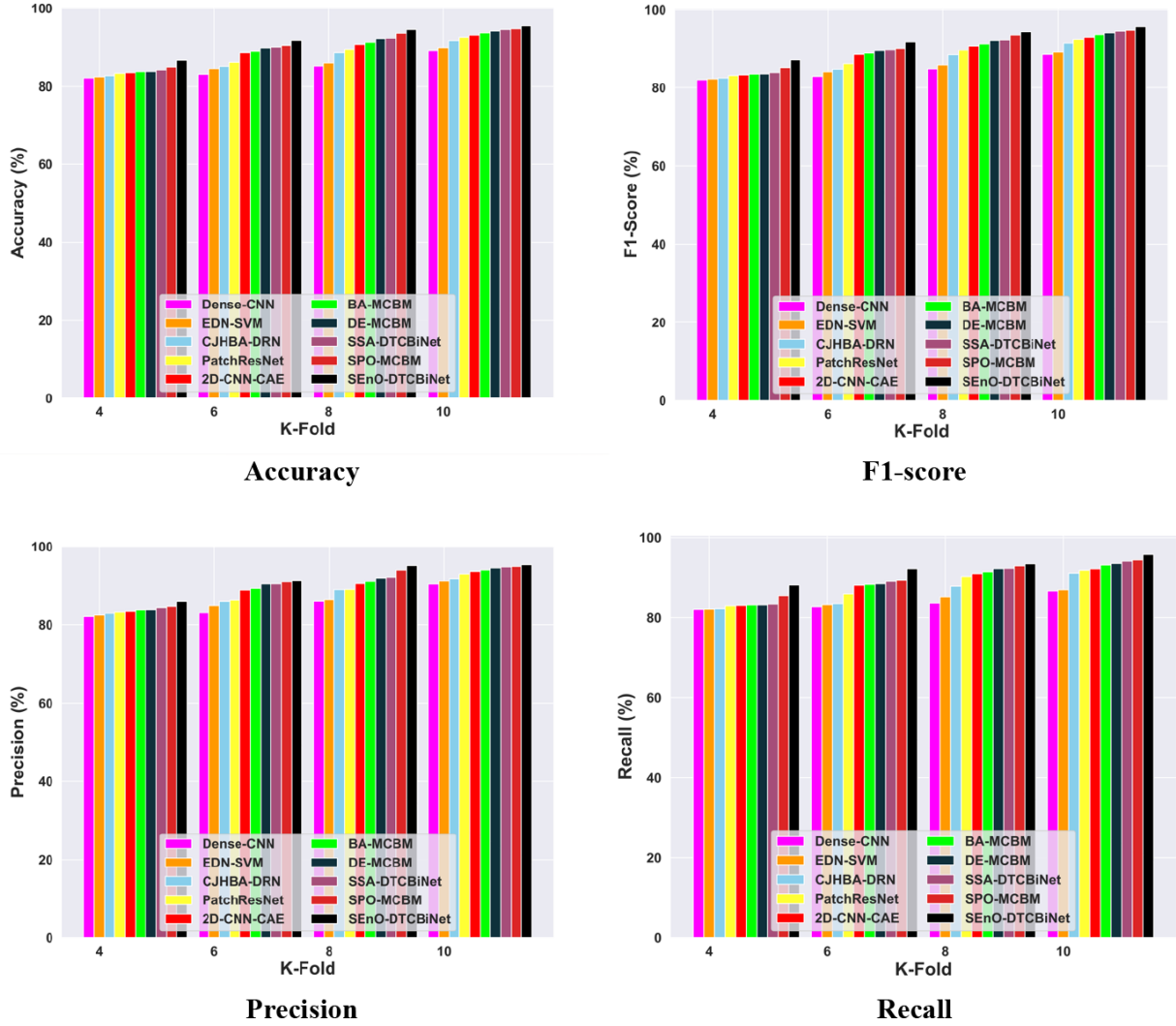


Figure 21: K-fold based on the BraTS 2018 dataset.

operation, such as addition or multiplication, performed on two floating-point numbers. At epoch 50, existing methods exhibit high FLOPS requirements, such as the Dense-CNN report 1.59×10^{08} and EDN-SVM report 1.45×10^{08} , whereas the proposed model utilizes fewer FLOPs of 1.42×10^{08} . Similarly, at epoch 99, the SSA-DTCBiNet, SPO-MCBM uses fewer FLOPs of 2.08×10^{08} ; in comparison, the proposed SEnO-DTCBiNet model significantly reduces the computational complexity to 2.05×10^{08} FLOPS through the SEnO algorithm, which significantly configures the model to enhance classification performance.

k. Ablation study for model components

As described in Figure 28, ablation is performed to evaluate the individual contributions of various architectural components in the proposed model. Starting with a BiLSTM, the model achieved 93.68% accuracy, effectively capturing temporal dependencies. Incorporating the SEnO optimizer with CNN shows improved performance to 93.96%, showing the benefit of optimized weight updates in the convolutional network. The CBiNet model, individually reached 94.64%, while the combination of SEnO + BiLSTM yielded 95.24%, demonstrating enhanced training stability. Adding the Tetra Head

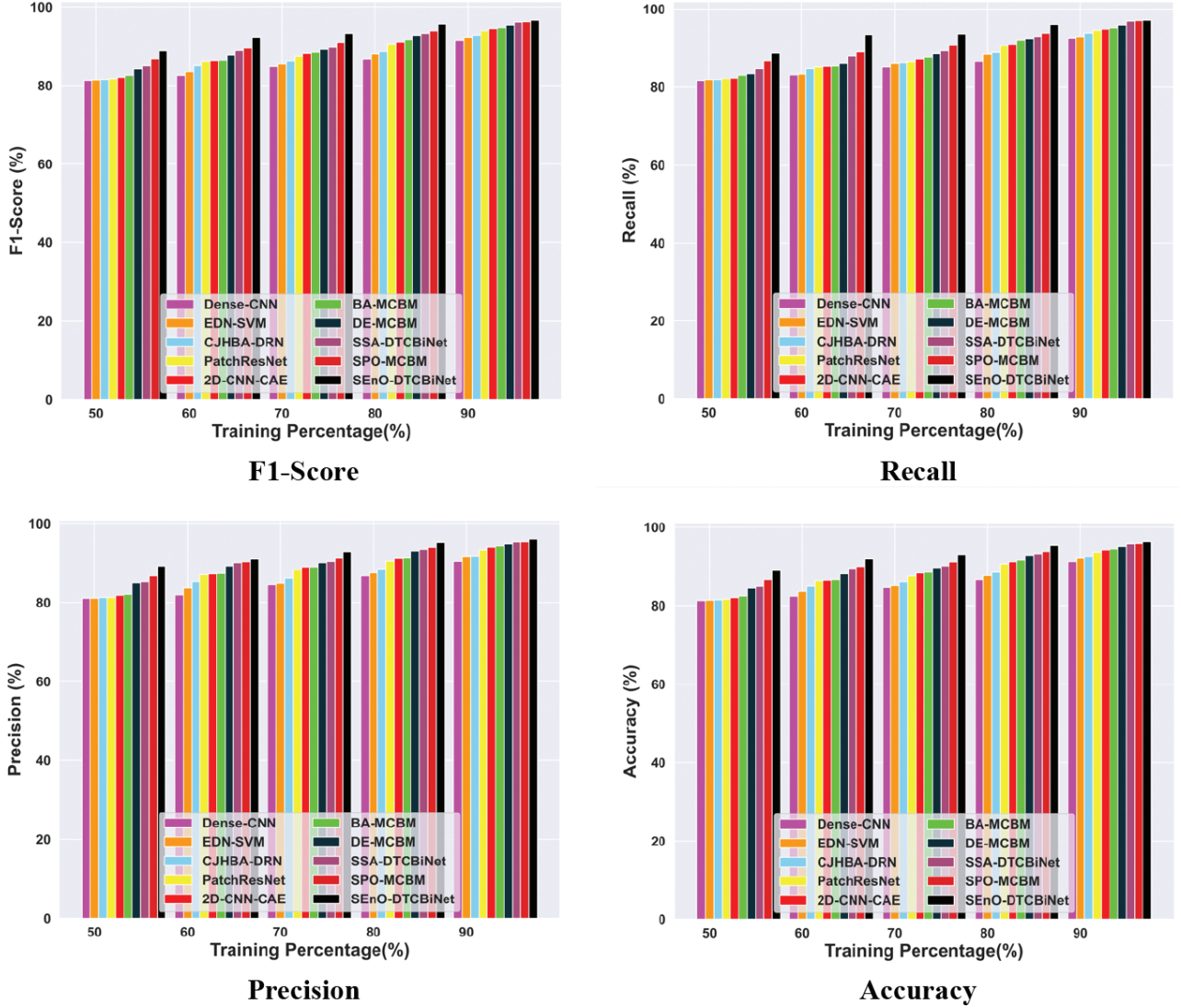


Figure 22: Comparison of the SEnO-DTCBiNet model on the real-time dataset concerning training percentage. SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network.

Attention to CBiNet further increases the performance to 95.74%, and SEnO + CBiNet reached 96.35% by unifying optimization and hybrid architecture. Finally, the complete SEnO-DTCBiNet model achieved an accuracy of 97.18%, validating the effectiveness of combining CNN, BiLSTM, SEnO optimization, and multihead attention in one unified framework. These results confirm that each component contributes to a layered improvement in classification, with the fully integrated model delivering the most robust performance.

I. Ablation study for the attention mechanism

An ablation study focused on the impact of individual attention mechanisms and their combinations is depicted in Figure 29. When used independently, BAM, SE, ULSAM, and CC achieved an accuracy of 87.57%, 85.68%, 91.55%, and 92.54% respectively, each excelling in different aspects, such as channel recalibration and spatial attention. Combining BAM + SE improved accuracy to 95.55%, and

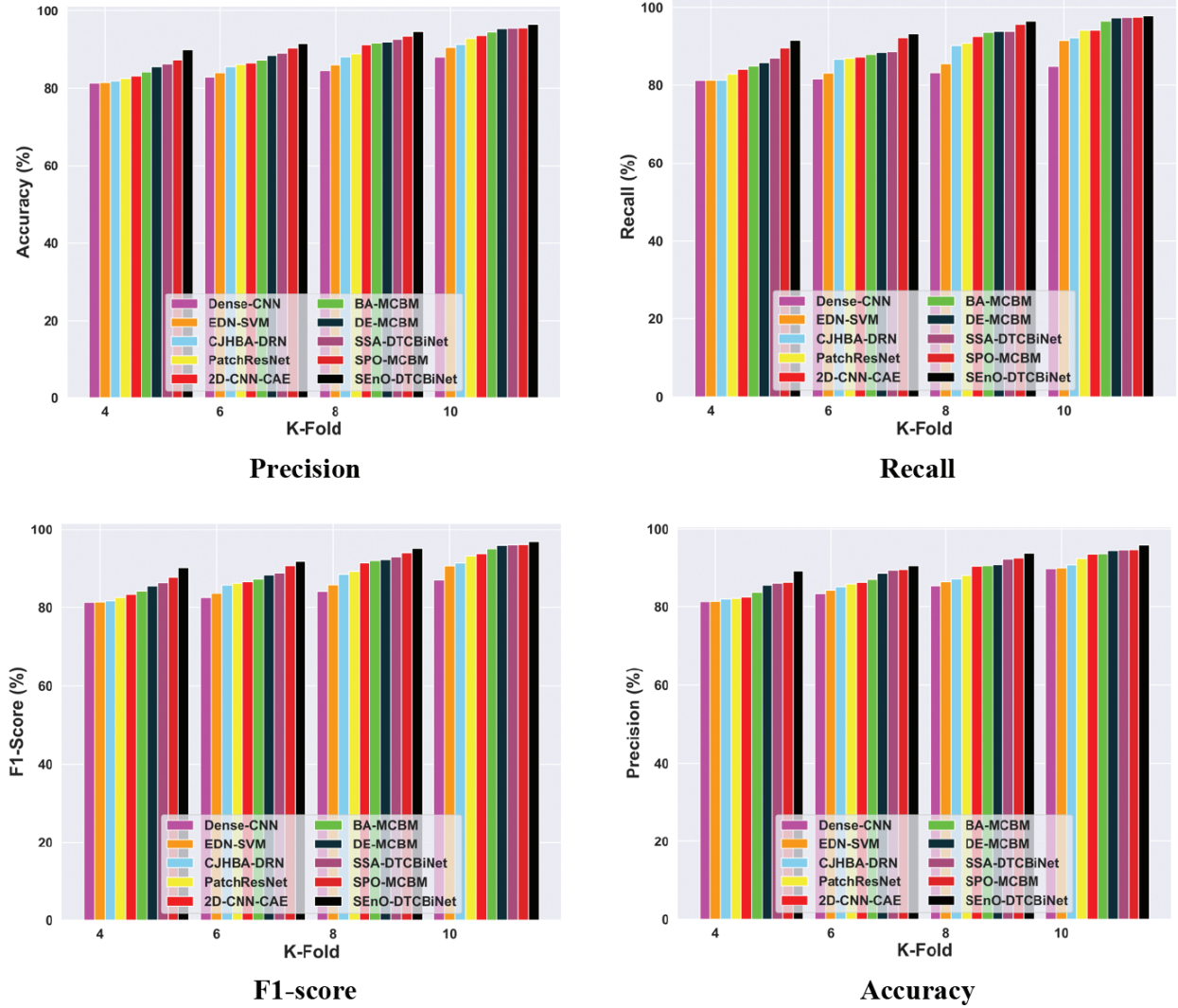


Figure 23: K-fold analysis on the real-time dataset.

CC + BAM + SE achieved 94.55%, reflecting the complementary effect of combining spatial and channel attention. The trio of ULSAM + SE + CC achieved 96.46% of accuracy, showcasing the synergy between cross-channel, lightweight, and contextual attention mechanisms. Ultimately, the full DTHA configuration combining BAM, SE, ULSAM, and CC achieved the peak accuracy of 97.18%, proving that fusing diverse attention modules significantly boosts feature representation. These attention mechanisms, when integrated into the proposed model, enhance fine-grained tumor localization and classification, making SEnO-DTCBiNet a highly effective tool in classifying the BT.

m. Confusion matrix

The SEnO-DTCBiNet model confusion matrix, across the BraTS 2020, BT, and the BraTS 2018 datasets is represented in Figure 30. This matrix serves to validate the model's performance by comparing its classification against the actual labels.

n. Statistical analysis

The statistical Analysis for the proposed SEnO-DTCBiNet model across the BraTS 2020 is presented in Table 4, across the BT is presented in Table 5, for

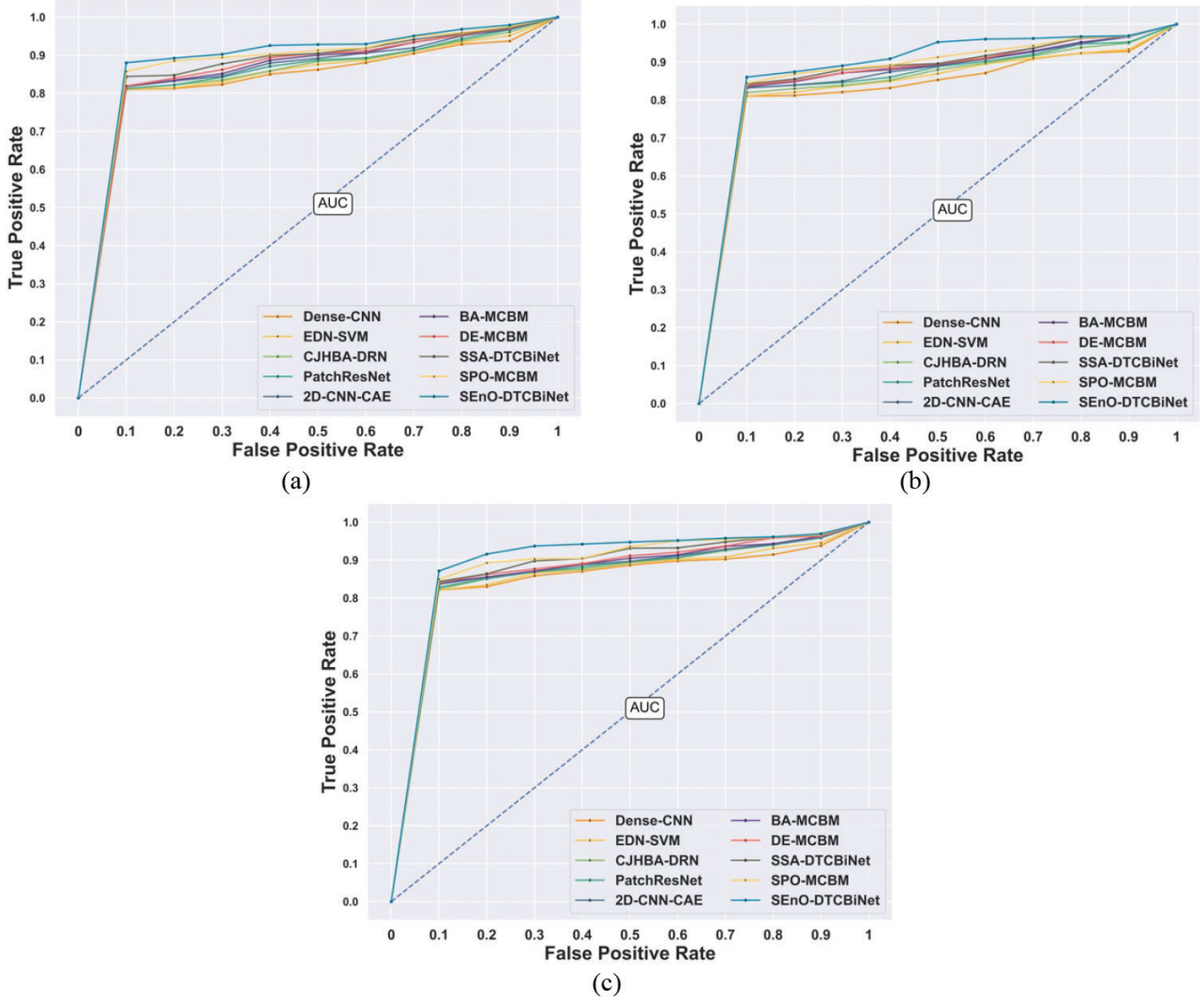


Figure 24: ROC analysis. **(A)** BraTS 2020 dataset, **(B)** BT dataset, and **(C)** BraTS 2018 dataset. BT, brain tumor.

the BraTS 2018 is shown in Table 6, and the real-time dataset is presented in Table 7.

o. *T*-test analysis

A *T*-test is performed to compare the means of different methods. A *p*-value of <0.05 indicates a statistically significant difference between the results of the compared models. Table 8 represents the *T*-test analysis on the BraTS 2020 dataset, Table 9 presents the *T*-test analysis on the BT dataset, Table 10 depicts the BraTS 2018 dataset based on the *T*-test analysis, and Table 11 represents the *T*-test analysis results on the real-time dataset. Thus, the results ensure that the model enhanced BT classification

performance across the baseline models under different datasets.

p. Comparative discussion

The increase in mortality rate due to BT has led to the development of an effective system that detects and classifies the type of BT based on their origin. However, conventional approaches used in the context provide promising outcomes; their performance is hindered by increased complexities and abnormal performance. The Dense-CNN [1], due to its sophisticated architecture, leads to increased processing time and inherited feature detection capability. While the EDN-SVM [11] improvised by utilizing

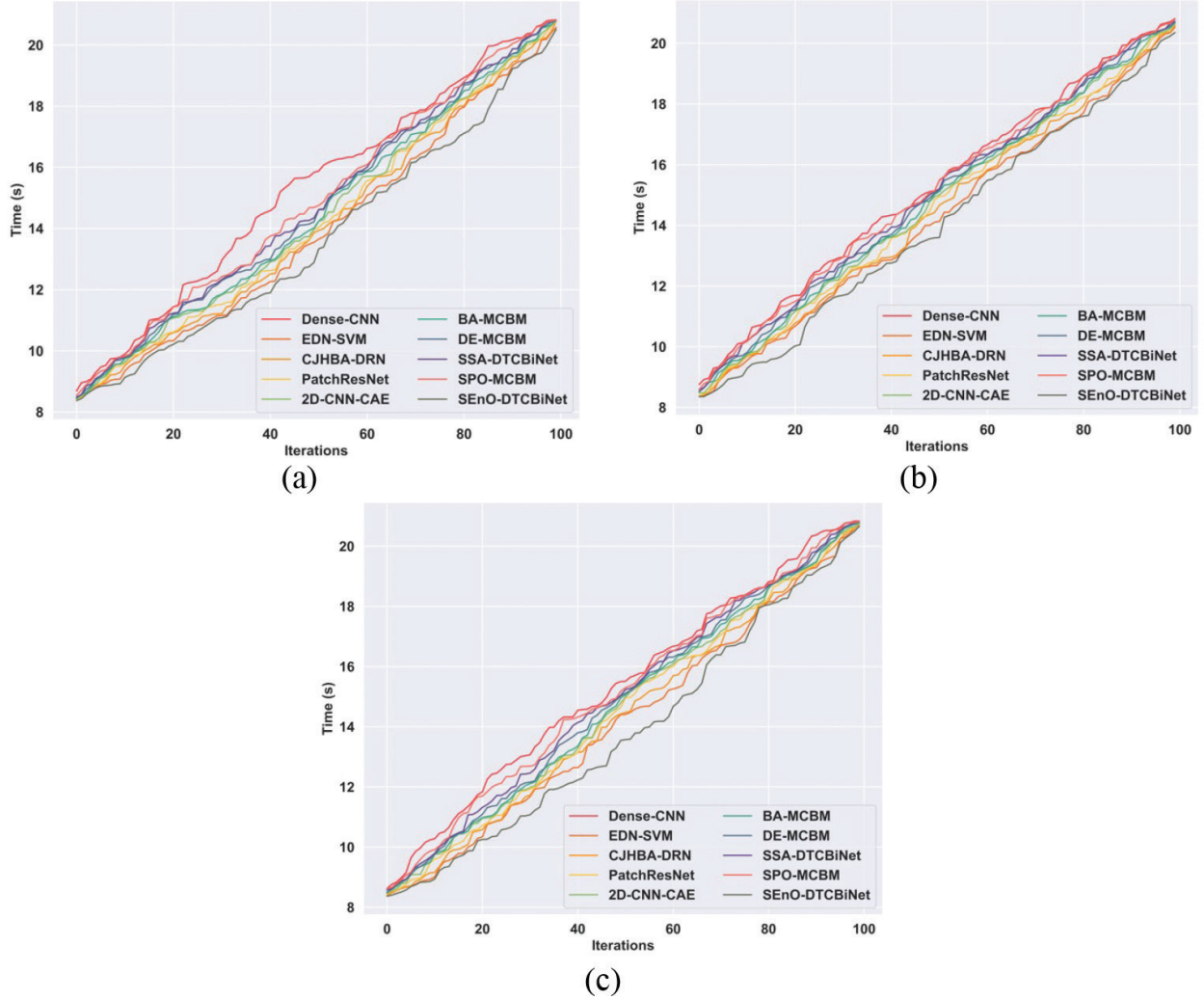


Figure 25: Time complexity analysis. **(A)** BraTS 2020 dataset, **(B)** BT dataset, and **(C)** BraTS 2018 dataset. BT, brain tumor.

a machine-learning architecture leveraging its simple structure, the inability to handle structural patterns caused overfitting and computational inconsistencies. The CJHBA-DRN [4] limited its application to recognizing diverse features that are essential for discriminating among the complex brain structures, thus leading to misclassification or diminished performance. The PatchResNet [7] involved different layered patches along with the pretrained models to detect BT; however, the approach lacked optimization of the learning parameters, which led to reduced classification accuracy. The 2D-CNN-CAE [15] leads to overfitting and underfitting issues due to the single-domain data for classification. The BA-MCBM [29] and DE-MCBM [30], due to the individual

characteristics of optimization, lead to premature convergence or getting stuck in local optima. The above-mentioned drawbacks are effectively tackled by the proposed SEnO-DTCBiNet by combining the OFA-based DW-Net for precise segmentation of tumor regions, deep BiLSTM layers for contextual understanding of complex structures. Additionally, the SEnO algorithm is used for hyperparameter tuning. These strategies allow the proposed SEnO-DTCBiNet to obtain improved results in classification. The comparison performance of the SEnO-DTCBiNet with other approaches based on Training Percentage and K-fold under the BraTS 2018, BraTS 2020 dataset, BT dataset, and real-time dataset is depicted in Tables 12 and 13.

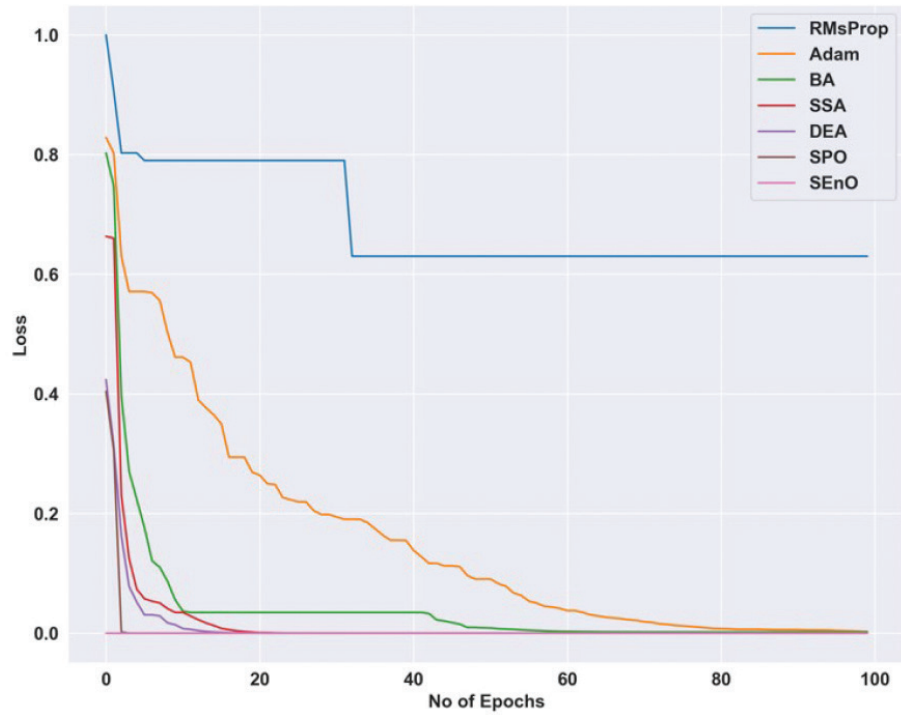


Figure 26: Convergence analysis. SSA, Sparrow Search Algorithm.

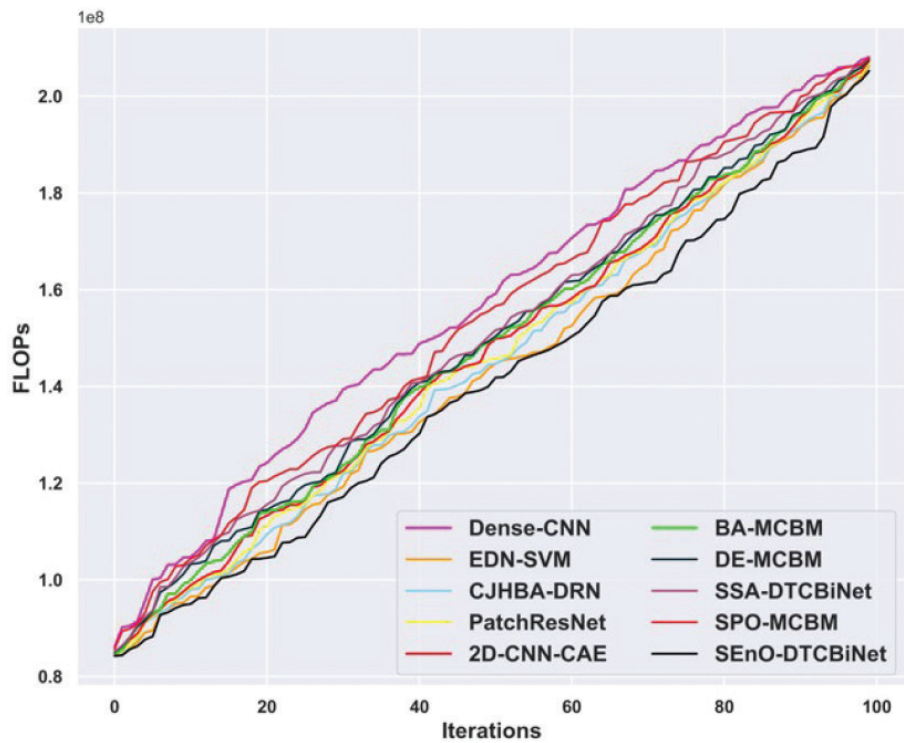


Figure 27: Computational complexity analysis in terms of FLOPs. FLOPs, floating-point operations per second.

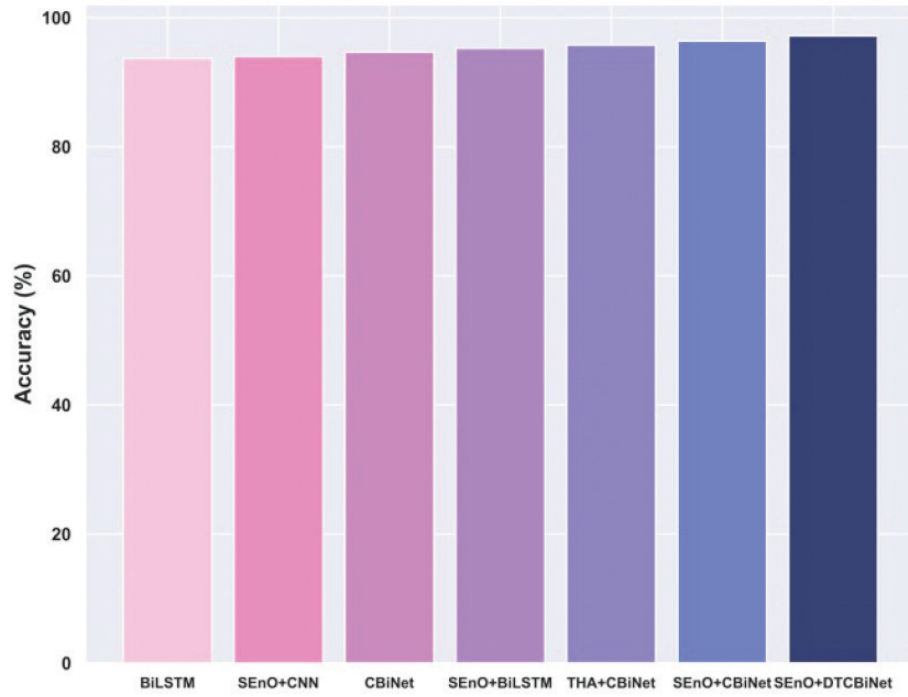


Figure 28: Ablation study for SEnO-DTCBiNet model components. SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network.

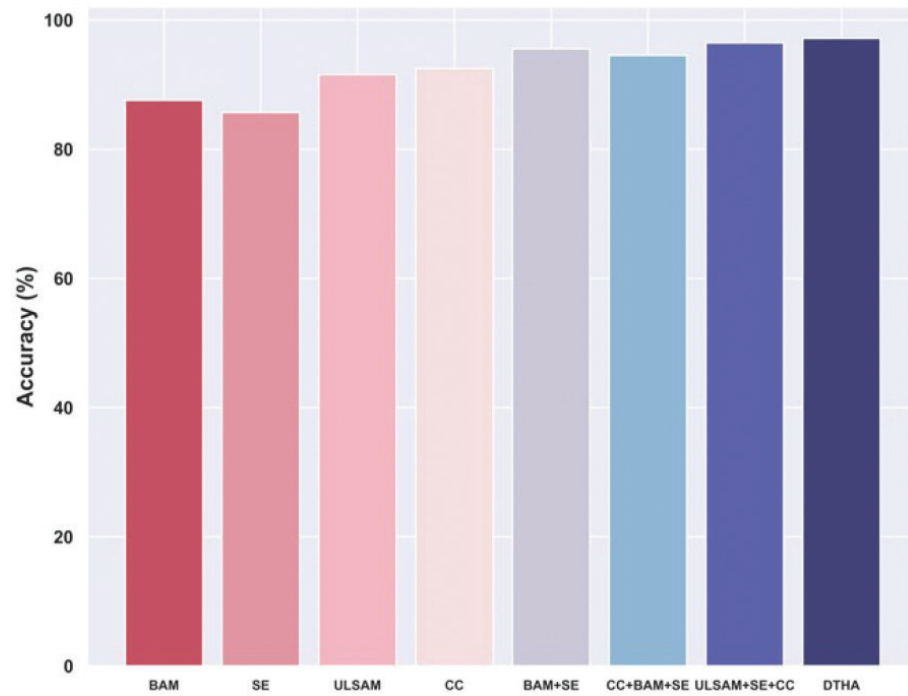
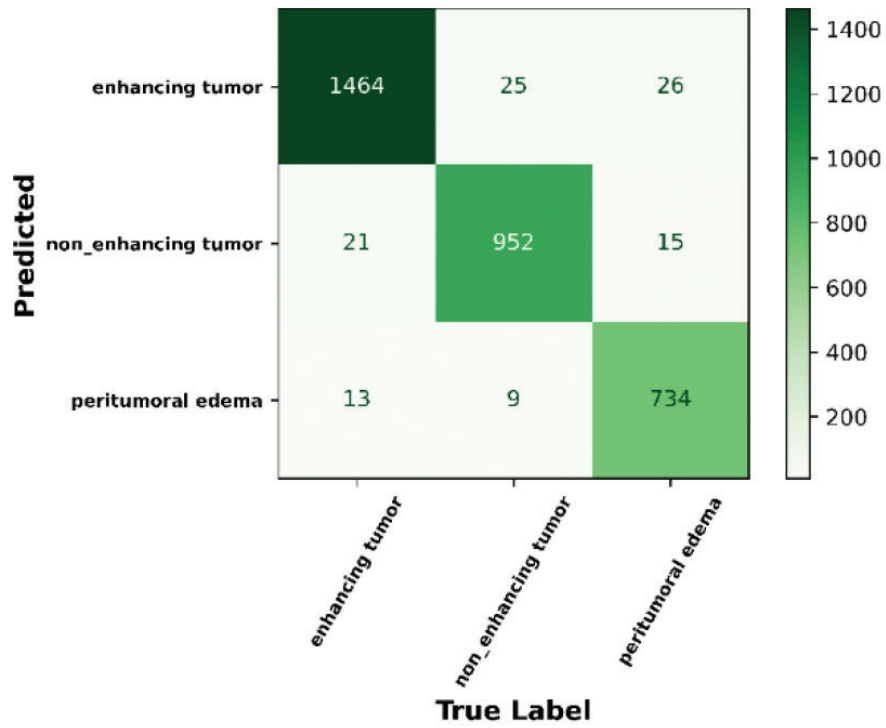
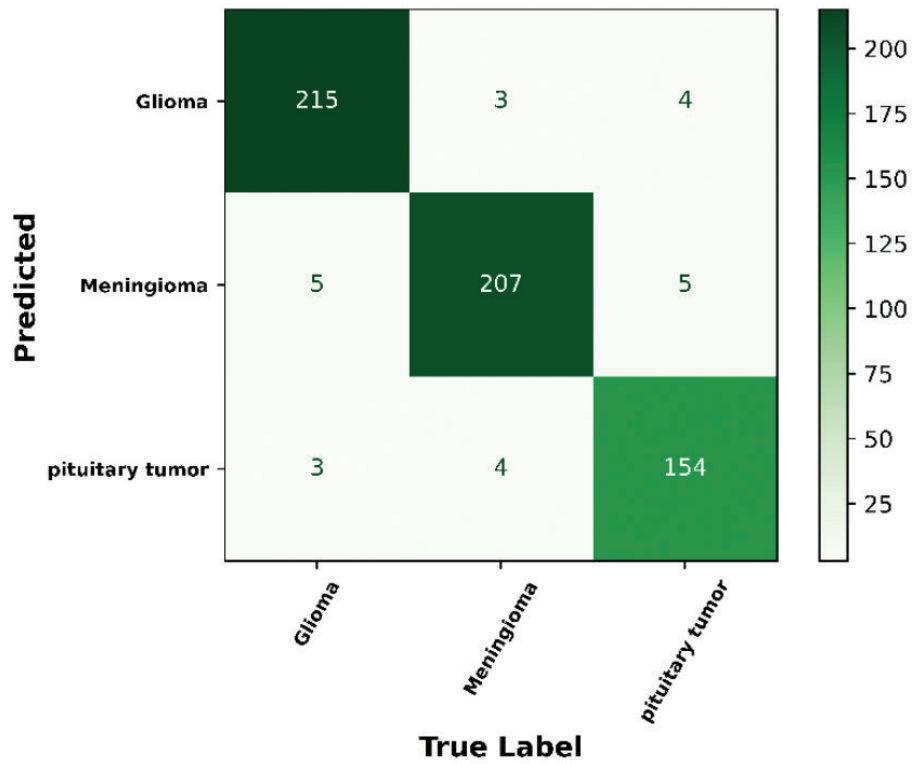


Figure 29: Ablation study for attention mechanisms. CC, criss-cross; DTHA, distributed tetra head attention; SE, squeeze and excitation.



(a)



(b)

(Continued)

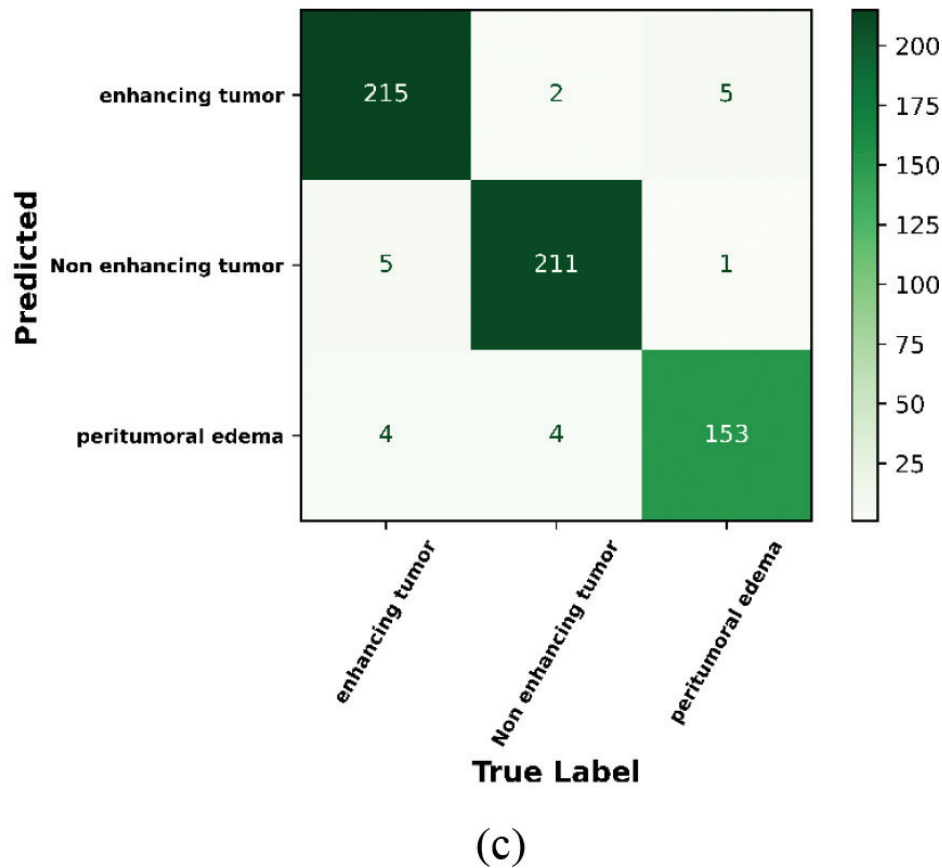


Figure 30: Confusion matrix.

VI. Conclusion

The SEnO-DTCBiNet framework introduced for BT classification leverages the sophisticated strategies of segmentation and DL techniques, thus ensuring early and timely diagnosis. The framework incorporates OFA-based DW-Net for segmenting the complex brain structures by leveraging the tetra-head attention to discriminate the abstractive features and the SEnO optimization for enhancing the feature representation. Moreover, the deep architecture and the BiLSTM networks enable extracting long-range relationships and the structural patterns more effectively. Additionally, the SEnO algorithm used for tuning the parameters of the model enhances the model performance by finding the optimal solution. The MSMSD for the feature extraction process captures comprehensive tumor characteristics, enhancing classification performance. The integration of these strategies allows SEnO-DTCBiNet to obtain effective outcomes in differentiating tumor types and an automated solution for diagnosis, ensuring early intervention

and treatments. During validation of the approach using the BraTS 2020 dataset, the proposed SEnO-DTCBiNet showed superior performance by acquiring an accuracy of 97.18%, precision of 97.31%, recall of 97.89%, and F1 score of 97.26% reflecting its proficiency against traditional approaches. The SEnO-DTCBiNet framework is limited by its opacity and lack of inherent interpretability, as well as the framework's multicomponent complexity also risks overfitting. Future work will focus on developing and integrating explainable AI techniques that provide clear visual justifications for their decisions, fostering trust and transparency for medical professionals. The future work of the model would focus on integrating feature selection strategies, integrating explainable AI techniques that provide transparency for medical professionals, and the SEnO-DTCBiNet framework could be extended to integrate multimodal images, such as MRI and CT scans, by developing a robust image fusion strategy. This would unlock richer information from different imaging techniques, thereby improving diagnostic accuracy and robustness.

Table 4: BraTS 2020 dataset-based statistical analysis

Methods/metrics		PatchRes Net	BA- MCBM	Dense- CNN	CJHBA- DRN	EDN- DRN	2D- CNN-CAE	SPO- MCBM	SSA- DTCBiNet	DE- MCBM	SEnO- DTCBiNet
Accuracy	Best	92.58	93.91	87.68	90.85	90.19	93.05	94.89	95.92	96.16	96.27
	Mean	87.40	88.59	84.81	86.49	85.78	87.96	89.33	90.24	90.70	92.34
	Variance	14.65	13.72	4.27	10.55	9.03	12.63	14.05	15.28	13.50	10.02
	Standard deviation	3.83	3.70	2.07	3.25	3.01	3.55	3.75	3.91	3.67	3.16
Precision	Best	92.24	93.22	87.57	89.81	89.33	92.50	93.65	95.11	95.39	95.53
	Mean	87.02	87.80	84.92	85.98	85.69	87.28	88.50	89.46	89.89	90.51
	Variance	13.94	14.22	4.09	7.56	7.31	14.04	12.45	14.85	13.31	12.81
	Standard deviation	3.73	3.77	2.02	2.75	2.70	3.75	3.53	3.85	3.65	3.58
Recall	Best	93.26	95.28	87.89	92.94	91.92	94.14	97.38	97.55	97.69	97.75
	Mean	88.17	90.17	84.59	87.50	85.94	89.34	91.00	91.78	92.31	95.99
	Variance	16.34	13.51	4.76	18.41	13.76	10.43	17.68	16.26	13.95	7.93
	Standard deviation	4.04	3.68	2.18	4.29	3.71	3.23	4.20	4.03	3.74	2.82
F1-score	Best	92.75	94.24	87.73	91.35	90.61	93.31	95.48	96.31	96.53	96.63
	Mean	87.59	88.96	84.76	86.73	85.81	88.29	89.73	90.61	91.09	93.16
	Variance	15.03	13.60	4.37	12.19	10.01	12.07	14.83	15.50	13.60	9.17
	Standard deviation	3.88	3.69	2.09	3.49	3.16	3.47	3.85	3.94	3.69	3.03

BA-MCBM, Bat-Algorithm-based multithread convolutional bidirectional memory; CJHBA-DRN, Chronological Java Honey Badger Algorithm-based Deep Residual Network; CNN, convolutional neural network; DE-MCBM, Dolphin swarm algorithm-based multithread bidirectional convolutional memory; SPO-MCBM, Sonar Prey Optimized Convolutional Bidirectional Memory; SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network; SSA-DTCBiNet, Sparrow Search Algorithm-based distributed tetra head attention-based convolutional bidirectional network.

Table 5: BT dataset-based statistical analysis

Methods/metrics	SSA-DTCBiNet		Dense-CNN		BA-MCBMM		EDN-SVM		2D CNN-CAE		CJHBA-DRN		Patch ResNet		DE-MCBM		SPO-MCBM		SEnO-DTCBiNet	
	Best		Best		Best		Best		Best		Best		Best		Best		Best		Best	
Accuracy	Best	95.24	89.32	94.84	91.49	94.71	92.19	92.91	94.71	92.19	92.19	92.91	92.91	92.91	94.97	95.63	95.63	95.63	96.24	96.24
	Mean	90.89	85.44	89.21	86.21	88.51	87.35	87.94	88.51	87.35	87.35	87.94	87.94	87.94	90.36	91.80	91.80	91.80	93.10	93.10
	Variance	16.56	9.66	19.58	14.07	19.31	14.63	14.81	19.31	14.63	14.63	14.81	14.81	14.81	17.29	14.61	14.61	14.61	8.64	8.64
	Standard deviation	4.07	3.11	4.43	3.75	4.39	3.82	3.85	4.39	3.82	3.82	3.85	3.85	3.85	4.16	3.82	3.82	3.82	2.94	2.94
Precision	Best	94.05	89.73	93.59	90.62	93.53	91.05	91.37	93.53	91.05	91.05	91.37	91.37	91.37	93.73	94.60	94.60	94.60	95.49	95.49
	Mean	90.41	85.64	88.61	86.06	88.26	87.03	87.59	88.26	87.03	87.03	87.59	87.59	87.59	90.11	91.44	91.44	91.44	93.02	93.02
	Variance	12.50	10.48	14.22	11.83	14.31	10.83	9.82	14.31	10.83	10.83	9.82	9.82	9.82	12.12	11.31	11.31	11.31	4.85	4.85
	Standard deviation	3.54	3.24	3.77	3.44	3.78	3.29	3.13	3.78	3.29	3.29	3.13	3.13	3.13	3.48	3.36	3.36	3.36	2.20	2.20
Recall	Best	97.63	88.49	97.34	93.23	97.07	94.49	96.00	97.07	94.49	94.49	96.00	96.00	96.00	97.47	97.69	97.69	97.69	97.72	97.72
	Mean	91.86	85.05	90.41	86.49	89.00	88.00	88.64	89.00	88.00	88.00	88.64	88.64	88.64	90.86	92.53	92.53	92.53	93.27	93.27
	Variance	26.47	8.24	33.81	19.70	31.94	24.11	28.37	31.94	24.11	24.11	28.37	28.37	28.37	30.91	22.62	22.62	22.62	20.79	20.79
	Standard deviation	5.14	2.87	5.82	4.44	5.65	4.91	5.33	5.65	4.91	4.91	5.33	5.33	5.33	5.56	4.76	4.76	4.76	4.56	4.56
F1-score	Best	95.81	89.11	95.43	91.91	95.27	92.74	93.63	95.27	92.74	92.74	93.63	93.63	93.63	95.56	96.12	96.12	96.12	96.59	96.59
	Mean	91.12	85.34	89.49	86.27	88.62	87.50	88.09	88.62	87.50	87.50	88.09	88.09	88.09	90.47	91.98	91.98	91.98	93.12	93.12
	Variance	18.71	9.27	22.52	15.29	22.03	16.68	17.59	22.03	16.68	16.68	17.59	17.59	17.59	20.20	16.37	16.37	16.37	11.21	11.21
	Standard deviation	4.33	3.04	4.75	3.91	4.69	4.08	4.19	4.69	4.08	4.08	4.19	4.19	4.19	4.49	4.05	4.05	4.05	3.35	3.35

BT , Brain tumor; CJHBA-DRN, Chronological Jaya Honey Badger Algorithm-based Deep Residual Network; CNN, convolutional neural network; DE-MCBM, Dolphin swarm algorithm-based multihed bidirectional convolutional memory; SPO-MCBM, Sonar Prey Optimized Convolutional Bidirectional Memory; SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network; SSA-DTCBiNet, Sparrow Search Algorithm-based distributed tetra head attention-based convolutional bidirectional network.

Table 6: BraTS 2018 dataset-based statistical analysis

Methods/metrics	Dense-CNN	EDN-SVM	CJHBA-DRN	PatchResNet	2D-CNN-CAE	BA-MCBM	DE-MCBM	SSA-DTCBiNet	SPO-MCBM	SEnO-DTCBiNet
Accuracy										
Best	89.11	89.81	91.60	92.62	93.17	93.75	94.22	94.59	94.78	95.51
Mean	84.85	85.64	87.00	87.87	88.95	89.39	89.94	90.23	90.96	92.12
Variance	7.40	7.53	11.51	12.33	12.91	13.83	15.52	15.21	14.40	11.72
Standard deviation	2.72	2.74	3.39	3.51	3.59	3.72	3.94	3.90	3.79	3.42
Precision										
Best	90.35	91.25	91.85	93.03	93.66	94.04	94.56	94.83	94.95	95.37
Mean	85.40	86.28	87.42	87.93	89.14	89.58	90.22	90.48	91.17	91.98
Variance	10.32	10.30	11.23	12.63	13.35	13.60	15.39	14.57	15.85	14.38
Standard deviation	3.21	3.21	3.35	3.55	3.65	3.69	3.92	3.82	3.98	3.79
Recall										
Best	86.64	86.95	91.10	91.81	92.18	93.16	93.54	94.13	94.43	95.80
Mean	83.76	84.37	86.17	87.73	88.57	89.02	89.36	89.73	90.55	92.40
Variance	3.09	3.44	12.44	12.27	12.36	14.46	16.18	16.68	11.99	7.68
Standard deviation	1.76	1.85	3.53	3.50	3.52	3.80	4.02	4.08	3.46	2.77
F1-score										
Best	88.46	89.05	91.48	92.42	92.92	93.60	94.05	94.48	94.69	95.58
Mean	84.57	85.31	86.79	87.83	88.85	89.30	89.79	90.10	90.86	92.19
Variance	6.06	6.23	11.71	12.23	12.73	13.96	15.64	15.57	13.74	10.58
Standard deviation	2.46	2.50	3.42	3.50	3.57	3.74	3.95	3.95	3.71	3.25

BA-MCBM, Bat-Algorithm-based multitask convolutional bidirectional memory; CJHBA-DRN, Chronological Java Honey Badger Algorithm-based Deep Residual Network; CNN, convolutional neural network; DE-MCBM, Dolphin swarm algorithm-based multitask bidirectional convolutional memory; SPO-MCBM, Sonar Prey Optimized Convolutional Bidirectional Memory; SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network; SSA-DTCBiNet, Sparrow Search Algorithm-based distributed tetra head attention-based convolutional bidirectional network.

Table 7: Real-time dataset-based statistical analysis

Methods/metrics	Dense-CNN	EDN-SVM	CJHBA-DRN	Patch ResNet	2D-CNN-CAE	BA-MCBM	DE-MCBM	SSA-DTCBiNet	SPO-MCBM	SEnO-DTCBiNet
Accuracy	Best	88.00	90.46	91.23	92.92	93.66	94.52	95.52	95.58	96.48
	Mean	84.17	85.45	86.68	87.59	88.59	89.34	90.90	91.68	93.11
	Variance	6.30	11.08	11.89	14.74	16.74	15.91	12.30	9.84	6.77
	Standard deviation	2.51	3.33	3.45	3.84	4.09	3.99	3.51	3.14	2.60
Precision	Best	89.65	90.03	90.83	92.41	93.50	93.54	94.55	94.61	95.78
	Mean	84.94	85.55	86.29	87.12	88.20	88.70	90.54	90.71	92.31
	Variance	9.44	9.69	10.04	13.57	17.21	13.70	10.22	9.99	6.81
	Standard deviation	3.07	3.11	3.17	3.68	4.15	3.70	3.20	3.16	2.61
Recall	Best	84.71	91.33	92.04	93.96	93.96	96.47	97.45	97.51	97.87
	Mean	82.62	85.25	87.47	88.54	89.36	90.61	91.62	93.62	94.71
	Variance	1.98	14.53	16.85	17.50	15.99	20.95	17.64	9.58	6.74
	Standard deviation	1.41	3.81	4.10	4.18	4.00	4.58	4.20	3.09	2.60
F1-score	Best	87.73	90.61	91.35	92.75	93.31	94.24	96.31	96.53	96.63
	Mean	84.76	85.81	86.73	87.59	88.29	88.96	90.61	91.09	93.16
	Variance	4.37	10.01	12.19	15.03	12.07	13.60	15.50	13.60	9.17
	Standard deviation	2.09	3.16	3.49	3.88	3.47	3.69	3.94	3.69	3.03

BA-MCBM, Bat-Algorithm-based multithead convolutional bidirectional memory; CJHBA-DRN, Chronological Jaya Honey Badger Algorithm-based Deep Residual Network; CNN, convolutional neural network; DE-MCBM, Dolphin swarm algorithm-based multithead bidirectional convolutional Memory; SPO-MCBM, Sonar Prey Optimized Convolutional Bidirectional Memory; SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network; SSA-DTCBiNet, Sparrow Search Algorithm-based distributed tetra head attention-based convolutional bidirectional network.

Table 8: *T*-test analysis based on the BraTS 2020 dataset

Methods	<i>T</i> -test analysis							
	Accuracy		Precision		Recall		F1-score	
	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic
EDN-SVM	0.12	2.20	0.10	2.39	0.16	1.87	0.13	2.11
Dense-CNN	0.09	2.42	0.09	2.53	0.12	2.19	0.10	2.36
CJHBA-DRN	0.11	2.21	0.11	2.27	0.12	2.13	0.12	2.19
PatchResNet	0.11	2.24	0.12	2.14	0.10	2.40	0.11	2.28
BA-MCBM	0.13	2.11	0.11	2.25	0.18	1.76	0.13	2.04
2D-CNN-CAE	0.13	2.07	0.11	2.22	0.19	1.70	0.14	2.00
DE-MCBM	0.14	2.02	0.13	2.11	0.16	1.86	0.14	1.98
SSA-DTCBiNet	0.12	2.20	0.11	2.22	0.12	2.16	0.12	2.19
SPO-MCBM	0.11	2.21	0.12	2.17	0.11	2.28	0.11	2.22
SEnO-DTCBiNet	0.08	2.58	0.11	2.24	0.06	3.00	0.07	2.71

BA-MCBM, Bat-Algorithm-based multihead convolutional bidirectional memory; CJHBA-DRN, Chronological Jaya Honey Badger Algorithm-based Deep Residual Network; CNN, convolutional neural network; DE-MCBM, Dolphin swarm algorithm-based multihead bidirectional convolutional Memory; SPO-MCBM, Sonar Prey Optimized Convolutional Bidirectional Memory; SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network; SSA-DTCBiNet, Sparrow Search Algorithm-based distributed tetra head attention-based convolutional bidirectional network.

Table 9: *T*-test Analysis based on the BT dataset

Methods/ metrics	<i>T</i> -test analysis							
	Precision		F1-score		Accuracy		Recall	
	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic
DE-MCBM	0.09	2.54	0.10	2.36	0.09	2.41	0.11	2.24
EDN-SVM	0.14	1.98	0.14	1.96	0.14	1.97	0.15	1.92
CJHBA-DRN	0.10	2.37	0.11	2.25	0.11	2.29	0.12	2.17
PatchResNet	0.09	2.42	0.11	2.22	0.11	2.28	0.13	2.08
Dense-CNN	0.13	2.05	0.12	2.20	0.12	2.15	0.10	2.35
BA-MCBM	0.10	2.38	0.11	2.25	0.11	2.29	0.12	2.15
2D-CNN-CAE	0.10	2.31	0.12	2.16	0.12	2.20	0.13	2.04
SSA-DTCBiNet	0.08	2.62	0.08	2.56	0.08	2.58	0.09	2.51
SPO-MCBM	0.07	2.77	0.07	2.71	0.07	2.73	0.08	2.66
SEnO-DTCBiNet	0.10	2.31	0.07	2.70	0.08	2.62	0.07	2.83

BA-MCBM, Bat-Algorithm-based multihead convolutional bidirectional memory; BT, Brain tumor; CJHBA-DRN, Chronological Jaya Honey Badger Algorithm-based Deep Residual Network; CNN, convolutional neural network; DE-MCBM, Dolphin swarm algorithm-based multihead bidirectional convolutional Memory; SPO-MCBM, Sonar Prey Optimized Convolutional Bidirectional Memory; SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network; SSA-DTCBiNet, Sparrow Search Algorithm-based distributed tetra head attention-based convolutional bidirectional network.

Table 10: BraTS 2018 dataset-based *T*-test analysis

Methods/ metrics	<i>T</i> -test analysis							
	Precision		Accuracy		F1-score		Recall	
	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic
SSA-DTCBiNet	0.07	2.73	0.07	2.72	0.07	2.72	0.07	2.69
EDN-SVM	0.13	2.08	0.13	2.09	0.13	2.10	0.13	2.10
CJHBA-DRN	0.10	2.37	0.11	2.23	0.12	2.15	0.15	1.94
PatchResNet	0.11	2.23	0.11	2.29	0.10	2.31	0.10	2.37
Dense-CNN	0.17	1.80	0.17	1.78	0.18	1.77	0.19	1.68
BA-MCBM	0.08	2.66	0.08	2.67	0.08	2.67	0.08	2.67
DE-MCBM	0.07	2.78	0.07	2.75	0.07	2.73	0.08	2.66
2D-CNN-CAE	0.08	2.64	0.08	2.68	0.07	2.69	0.07	2.72
SPO-MCBM	0.07	2.78	0.07	2.72	0.08	2.68	0.08	2.54
SEnO-DTCBiNet	0.07	2.72	0.07	2.73	0.07	2.72	0.08	2.66

BA-MCBM, Bat-Algorithm-based multihead convolutional bidirectional memory; CJHBA-DRN, Chronological Jaya Honey Badger Algorithm-based Deep Residual Network; DE-MCBM, Dolphin swarm algorithm-based multihead bidirectional convolutional Memory; CNN, convolutional neural network; SPO-MCBM, Sonar Prey Optimized Convolutional Bidirectional Memory; SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network; SSA-DTCBiNet, Sparrow Search Algorithm-based distributed tetra head attention-based convolutional bidirectional network.

Table 11: Real-time dataset-based *T*-test analysis

Methods/ metrics	<i>T</i> -test analysis							
	Precision		F1-score		Accuracy		Recall	
	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic	<i>p</i> -value	<i>T</i> -statistic
PatchResNet	0.10	2.30	0.10	2.35	0.10	2.34	0.10	2.39
EDN-SVM	0.11	2.25	0.14	2.03	0.13	2.10	0.17	1.83
CJHBA-DRN	0.11	2.29	0.09	2.51	0.09	2.45	0.08	2.63
SPO-MCBM	0.09	2.44	0.10	2.39	0.10	2.41	0.10	2.34
2D-CNN-CAE	0.10	2.35	0.10	2.34	0.10	2.34	0.10	2.32
BA-MCBM	0.10	2.32	0.11	2.25	0.11	2.27	0.12	2.19
DE-MCBM	0.11	2.29	0.12	2.20	0.11	2.23	0.12	2.12
SSA-DTCBiNet	0.09	2.44	0.12	2.18	0.11	2.26	0.14	1.96
Dense-CNN	0.13	2.04	0.14	1.97	0.14	2.00	0.17	1.79
SEnO-DTCBiNet	0.12	2.13	0.12	2.17	0.12	2.16	0.11	2.22

BA-MCBM, Bat-Algorithm-based multihead convolutional bidirectional memory; CJHBA-DRN, Chronological Jaya Honey Badger Algorithm-based Deep Residual Network; CNN, convolutional neural network; DE-MCBM, Dolphin swarm algorithm-based multihead bidirectional convolutional Memory; SPO-MCBM, Sonar Prey Optimized Convolutional Bidirectional Memory; SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network; SSA-DTCBiNet, Sparrow Search Algorithm-based distributed tetra head attention-based convolutional bidirectional network.

Table 12: Results of the SEnO-DTCBiNet and baseline models based on training percentage

Methods /metrics	DE-MCBM	Patch ResNet	EDN-SVM	Dense-CNN	2D-CNN-CAE	CJHBA-DRN	BA-MCBM	SSA-DTCBiNet	SPO-MCBM	SEnO-DTCBiNet
Brats 2020 dataset	Precision (%)	87.90	85.33	83.42	89.36	87.33	90.22	90.67	95.49	97.30
	Recall (%)	93.73	92.92	91.90	95.48	93.37	96.07	96.15	97.77	97.88
	Accuracy (%)	89.89	87.40	86.19	90.72	88.81	91.24	92.46	96.53	97.17
	F1-Score (%)	90.72	88.96	87.45	92.32	90.25	93.05	93.33	96.62	97.25
BT dataset	Precision (%)	89.04	86.73	85.09	90.12	86.74	90.57	95.55	96.71	96.86
	Recall (%)	93.14	92.51	91.74	94.74	92.97	95.83	97.33	98.52	98.72
	Accuracy (%)	91.35	89.05	88.57	91.78	91.13	91.82	96.11	97.91	98.48
	F1-Score (%)	91.05	89.53	88.29	92.37	89.75	93.128	96.43	97.61	97.98
Brats 2018 dataset	Precision (%)	91.65	89.95	89.33	92.76	90.09	93.08	93.49	93.89	94.16
	Recall (%)	92.82	90.60	90.15	93.01	92.72	94.06	95.45	97.03	97.42
	Accuracy (%)	94.05	89.91	88.16	94.13	91.21	94.30	95.69	96.80	97.76
	F1-Score (%)	92.23	90.27	89.74	92.88	91.39	93.56	94.46	95.43	95.76
Real-time dataset	Precision (%)	93.27	91.62	90.45	93.96	91.73	94.24	95.29	95.31	95.99
	Recall (%)	94.57	92.93	92.56	94.88	93.85	95.13	96.89	97.03	97.10
	Accuracy (%)	93.70	92.06	91.15	94.27	92.44	94.54	95.82	95.88	96.36
	F1-Score (%)	93.92	92.27	91.49	94.42	92.78	94.68	96.08	96.16	96.54

BA-MCBM, Bat-Algorithm-based multihead convolutional bidirectional memory; BT, Brain tumor; CJHBA-DRN, Chronological Jaya Honey Badger Algorithm-based Deep Residual Network; CNN, convolutional neural network; DE-MCBM, Dolphin swarm algorithm-based multihead bidirectional convolutional Memory; SPO-MCBM, Sonar Prey Optimized Convolutional Bidirectional Memory; SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network; SSA-DTCBiNet, Sparrow Search Algorithm-based distributed tetra head attention-based convolutional bidirectional network.

Table 13: Results of the SEnO-DTCBiNet model and comparative methods based on K-fold

Methods/metrics	2D- CNN-CAE	DE- MCBM	Dense- CNN	CJHBA- DRN	Patch ResNet	EDN- SVM	BA- MCBM	SSA- DTCBiNet	SPO- MCBM	SEnO- DTCBiNet
BrATS 2020 dataset	Precision (%)	92.50	93.65	87.57	89.81	92.24	89.33	93.22	95.39	95.53
	Recall (%)	94.14	97.38	87.89	92.94	93.26	91.92	95.28	97.69	97.75
	Accuracy (%)	93.05	94.89	87.68	90.85	92.58	90.19	93.91	96.16	96.27
	F1-Score (%)	93.31	95.48	87.73	91.35	92.75	90.61	94.24	96.53	96.63
BT dataset	Precision (%)	93.53	93.73	89.73	91.05	91.37	90.62	93.59	94.60	95.49
	Recall (%)	97.07	97.47	88.49	94.49	96.00	93.23	97.34	97.69	97.72
	Accuracy (%)	94.71	94.97	89.32	92.19	92.91	91.49	94.84	95.63	96.24
	F1-Score (%)	95.27	95.56	89.11	92.74	93.63	91.91	95.43	96.12	96.59
BrATS 2018 dataset	Precision (%)	93.66	94.56	90.35	91.85	93.03	91.25	94.04	94.95	95.37
	Recall (%)	92.18	93.54	86.64	91.10	91.81	86.95	93.16	94.43	95.80
	Accuracy (%)	93.17	94.22	89.11	91.60	92.62	89.81	93.75	94.78	95.51
	F1-Score (%)	92.92	94.05	88.46	91.48	92.42	89.05	93.60	94.69	95.58
Real-time dataset	Precision (%)	93.50	94.39	89.65	90.83	92.41	90.03	93.54	94.61	95.78
	Recall (%)	93.96	97.34	84.71	92.04	93.96	91.33	96.47	97.51	97.87
	Accuracy (%)	93.66	95.38	88.00	91.23	92.92	90.46	94.52	95.58	96.48
	F1-Score (%)	93.73	95.84	87.11	91.43	93.17	90.67	94.99	96.04	96.82

BA-MCBM, Bat-Algorithm-based multihead convolutional bidirectional memory; BT, brain tumor; CJHBA-DRN, Chronological Jaya Honey Badger Algorithm-based Deep Residual Network; CNN, convolutional neural network; DE-MCBM, Dolphin swarm algorithm-based multihead bidirectional convolutional Memory; SPO-MCBM, Sonar Prey Optimized Convolutional Bidirectional Memory; SEnO-DTCBiNet, sonar energy optimized distributed tetra head attention-based convolutional bidirectional network; SSA-DTCBiNet, Sparrow Search Algorithm-based distributed tetra head attention-based convolutional bidirectional network.

Declaration Statements

Acknowledgments

I would like to express my very great appreciation to the co-authors of this manuscript for their valuable and constructive suggestions during the planning and development of this research work.

Funding

This research did not receive any specific funding.

Author contribution

Aasha Mahesh Chavan has made substantial contributions to conception and design, revising the manuscript, and final approval of the version to be published. In addition, Dr. Vanita Mane agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

Conflict of interest

The authors declare no conflict of interest.

Informed consent

Not Applicable.

Ethical approval

Not Applicable.

Data availability statement

In case of benchmark data:

The data underlying this article are available in

- BraTS 2020 dataset, <https://www.kaggle.com/datasets/awsaf49/brats20-dataset-training-validation/data>.
- BT dataset, https://figshare.com/articles/dataset/brain_tumor_dataset/1512427.
- BraTS 2018 dataset, <https://www.kaggle.com/datasets/sanglequang/brats2018/data>

References

[1] Ozkaraca, O., Bagriacik, O.İ., Guruler, H., Khan, F., Hussain, J., Khan, J., and Laila, U.E., 2023.

Multiple brain tumor classification with dense CNN architecture using brain MRI images. *Life*, 13(2), p.349.

[2] Babu Vimala, B., Srinivasan, S., Mathivanan, S.K., Mahalakshmi, Jayagopal, P., and Dalu, G.T., 2023. Detection and classification of brain tumors using hybrid deep learning models. *Scientific Reports*, 13(1), p.23029.

[3] Hosny, K.M., Mohammed, M.A., Salama, R.A., and Elshewey, A.M., 2025. Explainable ensemble deep learning-based model for brain tumor detection and classification. *Neural Computing and Applications*, 37(3), pp.1289-1306.

[4] Deepa, S., Janet, J., Sumathi, S., and Ananth, J.P., 2023. A hybrid optimization algorithm enabled a deep learning approach to brain tumor segmentation and classification using MRI. *Journal of Digital Imaging*, 36(3), pp.847-868.

[5] Ayadi, W., Charfi, I., Elhamzi, W. and Atri, M., 2022. Brain tumor classification based on a hybrid approach. *The Visual Computer*, 38(1), pp.107-117.

[6] Islam, M.M., Barua, P., Rahman, M., Ahammed, T., Akter, L. and Uddin, J., 2023. Transfer learning architectures with fine-tuning for brain tumor classification using magnetic resonance imaging. *Healthcare Analytics*, 4, p.100270.

[7] Muezzinoglu, T., Baygin, N., Tuncer, I., Barua, P.D., Baygin, M., Dogan, S., Tuncer, T., Palmer, E.E., Cheong, K.H., and Acharya, U.R., 2023. PatchResNet: multiple patch division-based deep feature fusion framework for brain tumor classification using MRI images. *Journal of digital imaging*, 36(3), pp.973-987.

[8] Sarhan, A.M., 2020. Brain tumor classification in magnetic resonance images using deep learning and wavelet transform. *Journal of Biomedical Science and Engineering*, 13(06), p.102.

[9] Díaz-Pernas, F.J., Martínez-Zarzuela, M., Antón-Rodríguez, M. and González-Ortega, D., 2021, February. A deep learning approach for brain tumor classification and segmentation using a multiscale convolutional neural network. In *Healthcare* (Vol. 9, No. 2, p. 153). MDPI.

[10] Younis, A., Qiang, L., Nyatega, C.O., Adamu, M.J. and Kawuwa, H.B., 2022. Brain tumor analysis using deep learning and VGG-16 ensembling learning approaches. *Applied Sciences*, 12(14), p.7282.

[11] Anantharajan, S., Gunasekaran, S. and Subramanian, T., 2024. MRI brain tumor detection using deep learning and machine learning approaches. *Measurement: Sensors*, 31, p.101026.

[12] Lamrani, D., Cherradi, B., El Gannour, O., Bouqentar, M.A., and Bahatti, L., 2022. Brain tumor detection using MRI images and a convolutional neural network. *International Journal of Advanced Computer Science and Applications*, 13(7).

- [13] Rahman, T. and Islam, M.S., 2023. MRI brain tumor detection and classification using parallel deep convolutional neural networks. *Measurement: Sensors*, 26, p.100694.
- [14] Ullah, Z., Jamjoom, M., Thirumalaisamy, M., Alajmani, S.H., Saleem, F., Sheikh-Akbari, A., and Khan, U.A., 2024. A deep learning based intelligent decision support system for automatic detection of brain tumors. *Biomedical Engineering and Computational Biology*, 15, p.11795972241277322.
- [15] Saeedi, S., Rezayi, S., Keshavarz, H., and R. Niakan Kalhori, S., 2023. MRI-based brain tumor detection using convolutional deep learning methods and chosen machine learning techniques. *BMC Medical Informatics and Decision Making*, 23(1), p.16.
- [16] Haque, R., Hassan, M.M., Bairagi, A.K., and Shariful Islam, S.M., 2024. NeuroNet19: an explainable deep neural network model for the classification of brain tumors using magnetic resonance imaging data. *Scientific Reports*, 14(1), p.1524.
- [17] Brain Tumor Dataset, "https://figshare.com/articles/dataset/brain_tumor_dataset/1512427", accessed on July 2025.
- [18] BraTS 2020 Dataset, "<https://www.kaggle.com/datasets/awsaf49/brats20-dataset-training-validation/data>", accessed on July 2025.
- [19] BraTS 2018 Dataset, "<https://www.kaggle.com/datasets/sanglequang/brats2018/data>", accessed on July 2025.
- [20] Wu, Z., Zhang, X., Li, F., Wang, S., Huang, L., and Li, J., 2023. W-Net: A boundary-enhanced segmentation network for stroke lesions. *Expert Systems with Applications*, 230, p.120637.
- [21] Park, J., Woo, S., Lee, J.Y., and Kweon, I.S., 2020. A simple and light-weight attention module for convolutional neural networks. *International Journal of Computer Vision*, 128(4), pp.783-798.
- [22] Roy, A.G., Navab, N. and Wachinger, C., 2018. Recalibrating fully convolutional networks with spatial and channel "squeeze and excitation" blocks. *IEEE transactions on medical imaging*, 38(2), pp.540-549.
- [23] Saini, R., Jha, N.K., Das, B., Mittal, S., and Mohan, C.K., 2020. Ulsam: Ultra-lightweight subspace attention module for compact convolutional neural networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 1627-1636).
- [24] Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., and Liu, W., 2019. Ccnet: Criss-cross attention for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 603-612).
- [25] Kaplan, K., Kaya, Y., Kuncan, M. and Ertunç, H.M., 2020. Brain tumor classification using modified local binary patterns (LBP) feature extraction methods. *Medical hypotheses*, 139, p.109696.
- [26] Islam, M.S., 2014. Local gradient pattern novel feature representation for facial expression recognition. *Journal of AI and Data Mining*, 2(1), pp.33-38.
- [27] Haloi, R., Hazarika, J., and Chanda, D., 2020, July. Selection of appropriate statistical features of EEG signals for the detection of Parkinson's disease. In *2020 International Conference on Computational Performance Evaluation (ComPE)* (pp. 761-764). IEEE.
- [28] Rout, J., Das, S.K., Mohalik, P., Mohanty, S., Mohanty, C.K., and Behera, S.K., 2022, December. GLCM-based feature extraction and medical X-ray image classification using machine learning techniques. In *International Conference on Intelligent Systems and Machine Learning* (pp. 52-63). Cham: Springer Nature Switzerland.
- [29] Yang, X.S. and Hossein Gandomi, A., 2012. Bat algorithm: a novel approach for global engineering optimization. *Engineering computations*, 29(5), pp.464-483.
- [30] Wu, T.Q., Yao, M. and Yang, J.H., 2016. Dolphin swarm algorithm. *Frontiers of Information Technology & Electronic Engineering*, 17(8), pp.717-729.
- [31] Xue, J. and Shen, B., 2020. A novel swarm intelligence optimization approach: Sparrow Search Algorithm. *Systems science & control engineering*, 8(1), pp.22-34.

AUTHOR BIOGRAPHY



Mrs. Aasha Chavan is a dedicated researcher and academician specializing in deep learning, machine learning, and medical image processing, residing in Navi Mumbai, India, with over 12 years of teaching experience. Currently pursuing a PhD, focusing on optimizing MRI imaging techniques, developing novel feature extraction models, and enhancing classification accuracy through advanced hyperparameter tuning. With a strong background in artificial intelligence and healthcare applications, she is passionate about leveraging AI to improve brain tumor diagnosis and medical decision-making. Her research integrates cutting-edge deep learning methodologies with real-world medical challenges, contributing to the development of intelligent diagnostic systems. Beyond research, she is actively engaged in mentoring students, collaborating with interdisciplinary teams, and publishing in reputed journals. She is deeply committed to nurturing young minds, guiding students toward academic excellence, and fostering innovation in AI-driven healthcare technologies. By creating a supportive and intellectually stimulating environment, she empowers students to develop critical thinking and problem-solving skills for future research and industry success.



Dr. Vanita Mane is an Associate Professor and researcher with 22 years and 08 months of experience teaching courses at both undergraduate and postgraduate levels. An organized professional with proven teaching, guidance, and counselling skills. Ability to be a team player and resolve problems and conflicts professionally. Skilled at communicating complex information simply and entertainingly. Supervised 50+ PG students' dissertations and guided five PhD students. Published 95 international conference papers, book chapters, and journal papers. Published three copyright. Filed nine patents, out of which two have been granted and five have been published.