Journal of Innovative Image Processing (ISSN: 2582-4252)
www.irojournals.com/iroiip/

Controlled Search Diversification-Based Catch Fish Optimization-Driven Feature Selection for Brain Tumor Classification

Rama Krishna P.[1*], Prasad V.V.K.D.V.[2], Krishna B.T.[3]

[1,3]Department of Electronics and Communications Engineering, Jawaharlal Nehru Technological University, Kakinada, India.
[2]Department of Electronics and Communications Engineering, Seshadri Rao Gudlavalleru Engineering College, Vijayawada, India.

E-mail: [1*]pramakrishna441@gmail.com, [2]varaprasadvvkd@gmail.com, [3]tkbattula@jntucek.ac.in

Orcid ID: [1*]0000-0002-6762-3099, [2]0000-0001-9554-3814, [3]0000-0002-8316-7310

Abstract

Brain tumors are defined as the abnormal growth of cells originating from various central nervous system regions, which leads to patient risk. However, existing tumor detection and classification models face difficulties in identifying precise tumor boundaries in Magnetic Resonance Imaging (MRI) data due to the presence of noise, low contrast, and complex tissue structures. To address this limitation, an automated brain tumor classification model, namely Controlled Search Diversification-based Catch Fish Optimization Algorithm and Visual Geometry Group-19 (CSD-CFOA-VGG19), is proposed in this research. Initially, the raw MRI data are acquired from benchmark datasets and preprocessed to enhance quality for accurate detection of tumor boundaries. Next, the Davies–Bouldin Index-based Fuzzy C-Means (DBI-FCM) clustering method is used to localize and segment the affected brain regions. After that, discriminative features are extracted using a CNN based model, and relevant features are selected by the proposed Controlled Search Diversification-based Catch Fish Optimization Algorithm (CSD-CFOA) for differentiating tumor features from normal tissues. Finally, the selected features are fed to a modified VGG19 approach for multi-class brain tumor classification. Experimental results show that the proposed model reported an accuracy of 98.53% on the Figshare dataset, which is a competitive performance when compared to benchmarks such as the Chronological Jaya Honey Badger Algorithm (CJHBA) under identical conditions.

Keywords: Brain Tumor Classification, Catch Fish Optimization Algorithm, Controlled Search Diversification, Davies–Bouldin Index, Fuzzy C-Means, Visual Geometry Group-19.

1. Introduction

Brain tumors are characterized by abnormal cell growth originating from a diverse group of central nervous system conditions. Based on tumor size, growth rate, and specific location, these tumors cause a wide range of neurological symptoms, including cognitive impairment and speech loss. Therefore, early detection is essential for effective diagnosis and to provide timely treatment for mitigating symptom severity [1]. Some specific symptoms of

*** Corresponding Author**

DOI: https://doi.org/10.36548/jiip.2026.3.023
Received: 20.05.2026, received in revised form: 24.06.2026, accepted: 12.07.2026, published: 24.07.2026

brain tumors include behavioral changes, memory loss, seizures, confusion, or vision and balance problems [2,3]. Brain tumors are generally grouped into two categories: benign (non-cancerous) and malignant (cancerous). The latter type of tumors exhibits aggressive behavior and tends to invade surrounding brain tissue and other parts of the brain [4]. Common medical imaging techniques for the diagnosis and treatment planning of brain tumors include Computed Tomography (CT), Ultrasound, X-rays, and Magnetic Resonance Imaging (MRI). Among these techniques, MRI is commonly used for brain tumor imaging due to its high spatial resolution and better soft tissue contrast [5, 6]. Moreover, MRI is a safer method compared to traditional techniques as it uses non-ionizing radiation instead of hazardous X-ray radiation to obtain high-resolution sequences [7]. Using four basic imaging sequences, these sequences make use of unique tissue properties to help detect brain anomalies [8]. Yet, proper MRI analysis requires thorough visual inspection along with high-level expertise, which proves difficult for non-experts [9-10].

Brain tumor imaging methods can generally be categorized into three groups based on human effort: manual methods, semi-automatic methods involving user intervention, and automatic methods [11, 12]. The emergence of advanced ML and computer vision techniques has facilitated the development of architectures like CNN that enable the effective analysis of medical images. These architectures are of utmost importance in Computer-Aided Diagnosis (CAD) for assisting in the segmentation, detection, and classification of brain abnormalities [14, 15]. In early tumor diagnosis, manual methods are usually applied for tumor tracing and structural element detection to classify specific anatomical regions [16]. However, manual techniques are quite time-consuming, tedious, and subject to fatigue. Traditional classification models can also suffer from detection biases, making them less reliable. Furthermore, manual methods might not possess the necessary precision required for detecting brain tumors [17, 18]. It is evident from recent findings that there has been increased research interest in the application of AI and computer vision techniques to enhance the analysis of medical images. Computer vision techniques are usually classified into supervised and unsupervised techniques, all with the objective of enhancing tumor detection and classification [19, 20].

1.1 Problem Statement and Objective

Brain tumor detection and classification prove difficult because of the complicated structures, image noise, and poor contrast of MRI images. To enhance classification accuracy, this research proposes an optimization-based approach for feature selection using the Controlled Search Diversification-Based Catch Fish Optimization Algorithm (CSD-CFOA). Furthermore, the Davies–Bouldin Index-Based Fuzzy C-Means (DBI-FCM) is used for accurate tumor localization. Lastly, a Visual Geometry Group-19 (VGG19) neural network model will be used to classify multiple tumor classes. This combination will not only reduce the amount of computational resources but also address the problems of overfitting and high-dimensional data. The paper's main contribution is the combination of tumor localization and feature selection techniques. Unlike previous works, which considered the two tasks separately, this research uses the DBI-FCM algorithm for tumor localization, and subsequently, the CSD-CFOA is applied for selecting the most relevant features. This technique helps in removing irrelevant data, thus increasing the accuracy of diagnosis on BraTS and Figshare datasets.

The primary contributions of this research are summarized below:

- This research introduces a feature selection method based on the CSD-CFOA to improve brain tumor classification. The CSD model finds the perfect balance

between exploitation and exploration, which helps in avoiding the problem of getting trapped in local optima.

- A modified VGG19 architecture is employed for classifying tumors into various classes. The network is capable of making hierarchical decisions and improving feature extraction.
- The tumor regions segmented using the DBI-FCM method are fed as input to the modified VGG19 architecture, which extracts hierarchical features through sequential convolutional and pooling operations. The output of the final convolutional block is processed by the proposed tree-pooling module and is subsequently passed to the fully connected layers, where the Softmax layer performs the final brain tumor classification.

The rest of this manuscript is structured as follows. Section 2 presents a literature review on brain tumor classification. The proposed methodology is detailed in Section 3. Section 4 discusses the experimental results. Finally, the conclusion is provided in Section 5.

2. Literature Survey

In this section, previous studies on brain cancer gene expression using Deep Learning (DL) approaches are discussed. Roy et al. [21] presented binary CNN approaches, including S-Net and S-Net with attention mechanisms, for classifying segmented brain tumor images. These approaches assumed of a U-shaped Convolutional Network (U-Net) as a common method. The presented approach used a "Merge Block" to combine local and global backgrounds, and an "Attention Block" to focus on Regions of Interest (RoI) where specific objects were present. Additionally, the approaches used data augmentation to leverage annotated instances effectively. Metlek and Çetiner [22] designed a convolution-assisted hybrid approach for brain tumor segmentation. In this method, convolution was specifically applied to the RoI rather than the entire sample for tumor detection across different modalities. The hybrid approach addressed the semantic gap in U-Net by minimizing the impact of irrelevant information through the use of residual blocks. Almufareh et al. [23] utilized deep learning-based object detection frameworks, specifically You Only Look Once version-5 (YOLOv5) and YOLOv7, to detect and classify brain tumors. To improve target localization, the preprocessing pipeline integrated mask-alignment techniques to isolate the tumor regions. This segmented data was subsequently utilized to analyze the classification performance of the DL approaches.

The PDCNN architecture, proposed by Rahman and Islam [24], has a topology designed to capture multi-scale local and global features through dual parallel layers to classify brain tumors. Parallel processing capability is achieved through the use of two binary DCNNs simultaneously with different window sizes so that both local and global information can be captured. Abboodi et al. [26] proposed a CNN-based attention mechanism to classify brain tumors. Attention mechanisms were incorporated to highlight only important features while capturing all background information. The U-Net-attention technique was used to improve traditional segmentation techniques. This strategy employed complexity feature extraction to improve image quality. The features extracted using the RCSF-CNN were then used for classification. Topannavar et al. [28] proposed a CNN architecture using the Global Self-Attention Module (GSAM) for brain tumor classification. The preprocessed images were segmented using Otsu Thresholding and the Adaptive Kernel Fuzzy C-Means (AKFCM). The Sine Cosine Reptile Search Algorithm (SCRSA) was applied for feature selection after

segmentation. The selected features were fed into the CNN designed using the Global Self-Attention Module for classification. Maedeh Mohammadi and Majid Ziaratban [29] proposed a network named Transformer-Guided Cross-Scale Attention and Deep Semantic Fusion Network (TCAFNet) for brain tumor classification. TCAFNet uses the pre-trained EfficientNetB0 backbone along with multiscale attention and transformer-guided refinement. The extracted features are combined with an attention based fusion technique and further processed by a lightweight transformer to analyze the long-range spatial relations among them. The Fine-Tuning Vision Transformer (FTVT) was introduced by C. Kishor Kumar Reddy et al. [30] for the detection of brain tumors. The FTVT proposed a custom classifier head that included components such as Batch Normalization (BN), a dense layer, ReLU activation, and a dropout layer.

2.1 Research Gap

The conventional techniques used for brain tumor detection and classification employ either handcrafted pre-processing or classic CNN models, which are quite complicated, redundant, and high dimensional. Furthermore, conventional meta-heuristic techniques suffer from poor convergence and low exploration capacity, leading to inferior feature subsets. Also, the conventional segmentation techniques often unsuccessful in retainingfeature boundaries and lack generalization capability across diverse datasets. In the proposed method, these problems are addressed using CSD-CFOA, which promotes search diversification and maintains population diversity. Compared to previous single-objective metaheuristics, CSD implements a roulette-based weighted selection and a controlled reinitialization policy to reduce the risk of premature convergence during optimization iterations. The proposed CSD-CFOA employs a mathematically defined dual-criteria evaluation mechanism. Each solution (fisherman) is assessed based on (i) fitness value and (ii) positional diversity relative to the global best agent. This dual-criteria score is used in a roulette-weighted selection process that adaptively preserves higher-quality solutions while reinitializing weaker agents only when population diversity drops below a certain threshold. Additionally, a weighted-average positional update replaces the traditional arithmetic mean used in CFOA, ensuring consistent convergence without trapping the swarm in a small region of the solution space.

3. Proposed Methodology

This article aims to improve classification effectiveness by selecting significant features from the collected dataset using the CSD-CFOA approach. The proposed framework comprises preprocessing, segmentation, feature selection, and classification. For the BraTS2020 and BraTS2021 datasets, DBI-FCM is applied to segment tumor regions for segmentation evaluation. The Figshare dataset does not provide voxel-level ground-truth masks, so segmentation or ROI extraction is not performed. Each MRI image is resized to the required VGG19 input resolution after preprocessing and is directly used for classification. Consequently, the segmentation and classification experiments remain independent throughout the proposed framework. At the final block, the $7 \times 7 \times 512$ feature map is recursively aggregated through the Tree-Pooling module and flattened into a 512-dimensional deep feature vector. The proposed CSD-CFOA optimizer acts on this 512-D vector, generating an optimal binary feature-selection mask to eliminate redundant deep features. The resulting subset vector is then forwarded to the fully connected layers for classification by the Softmax layer. Figure 1 illustrates the workflow of the CSD-CFOA method in brain tumor classification.

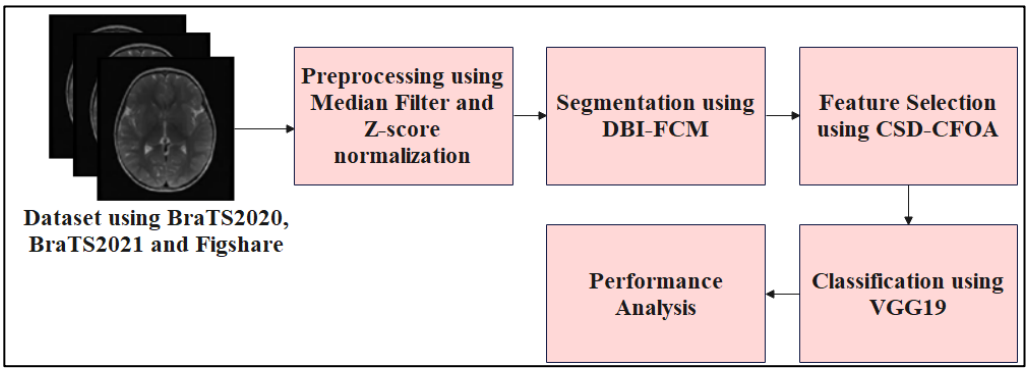

Figure 1. Workflow of the CSD-CFOA method in brain tumor classification

3.1 Data Acquisition

To train and validate the performance of the proposed model, three benchmark datasets are utilized in this research such as BraTS2020 [31], BraTS2021 [32], and Figshare [33]. Both BraTS datasets consist of multimodal three-dimensional MRI volumes such as T1-weighted (T1), T2-weighted (T2), Fluid-Attenuated Inversion Recovery (FLAIR) and T1-weighted contrast-enhanced (T1ce) images with expert-labeled tumor masks. These datasets are mainly used for segmentation and volumetric evaluation. In contrast, the Figshare dataset exists in two publicly referenced versions. The actual Figshare dataset includes 3,064 MRI scans from 233 patients, comprising 1,426 gliomas, 708 meningioma, and 930 pituitary tumors. To ensure a balanced experimental evaluation, the complete Figshare dataset, containing 3,064 brain MRI images is split using a stratified 80:20 train-test split, as summarized in Table 1. This strategy preserves the original class distribution by allocating 2,451 images for model training and 613 images for independent testing.

Table 1. Stratified 80:20 train-test split of Figshare dataset

Dataset split / Class	Gliomas	Meningioma	Pituitary	Total
Total	1426	708	930	3064
Train (80%)	1141	566	744	2451
Test (20%)	285	142	186	613

In the current research, the volume-based BraTS datasets are used for segmentation purposes using the DBI-FCM method, whereas the Figshare dataset is used for classification purposes using the VGG19 method. This ensure no confusion between datasets and that data collection is performed according to the proper tasks. Figure 2 depicts some MRI data from the collected dataset.

BraTS is a 3D MRI dataset, but 2D slices are taken from the BraTS dataset in diagnostic regions due to computational efficiency. These 2D slices are taken from the BraTS dataset because of computational constraints, and the selected 2D slices maintain the important features of the tumors such as their shapes and intensities.

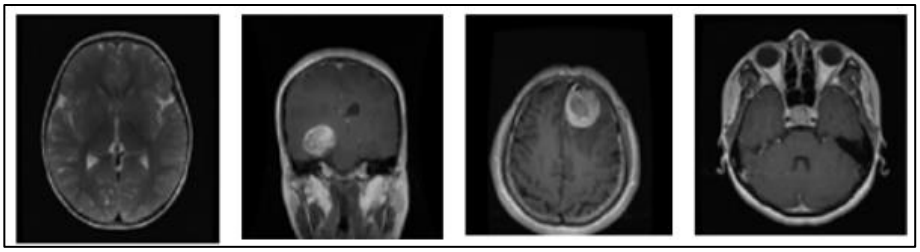

Figure 2. Sample images of collected dataset

The segmentation and classification experiments are performed as two independent evaluation tasks using different benchmark datasets. The BraTS2020 and BraTS2021 datasets are employed to evaluate the effectiveness of the proposed DBI-FCM segmentation method, whereas the Figshare dataset is independently utilized to validate the classification performance of the proposed CSD-CFOA integrated modified VGG19 framework. The BraTS2020 and BraTS2021 datasets comprise 369 and 1,251 multi-modal MRI patient volumes, respectively. For slice-wise segmentation, one representative 2D axial slice containing the maximum tumor area is selected from each patient volume, resulting in 369 slices for BraTS2020 and 1,251 slices for BraTS2021.

3.2 Preprocessing

This research was conducted based on the development of an effective pre-processing framework through the implementation of z-score normalization for pixel scaling and the use of a median filter for noise reduction.

3.2.1 Noise Removal and Image Enhancement

A median filter is applied to the uncleaned MRI image with a 3 × 3 kernel window to remove impulse noise and keep edges intact. A 3 × 3 median filter is chosen for noise reduction to effectively eliminate impulse noise and retain considerable tumor edges and structural boundaries; larger kernel windows are not considered to avoid oversmoothing of diagnostically useful regions. For the BraTS2020 and BraTS2021 databases, the binary tumor mask obtained via the DBI-FCM algorithm is only employed to evaluate segmentation performance relative to the ground-truth annotations. For the Figshare database, MRI images are resampled to the needed VGG19 input resolution without extraction of RoIs due to the lack of voxel-level segmentation maps. These RoI patches are then resized to 224×224 pixel resolution and provided as input to the VGG19 network.

3.2.2 Z-score Normalization

This method standardizes MRI image intensities by scaling each pixel value according to the standard deviation (σ) and mean (μ) of the entire dataset. The mathematical representation of Z-score is given in Equation (1).

$$Z(x) = \frac{x-\mu}{\sigma} \tag{1}$$

In this case, x denotes an input density. Individual normalization is applied to every pixel in the image domain to ensure a is consistent statistical distribution among all MRI inputs. Preprocessing parameters have been precisely defined to ensure repeatability.

3.3 Segmentation Using FCM

The FCM algorithm allows an input to be assigned to more than one cluster during classification. FCM is very effective because continuously updates the cluster center and memberships to achieve the minimum value of the objective function according to the distance measure. In the FCM clustering method, the sum of a data point's membership in all clusters is always one [34]. With the inclusion of DBI in the FCM clustering method, clusters will form with minimum intra-cluster variation and maximum inter-cluster separation. This differs from

the use of the U-Net encoder–decoder architecture, where DBI can add more advantage to the proposed framework. The definition of DBI is given in Equation (2).

$$DBI = \frac{1}{C}\sum_{i=1,i\neq j}^{C} max\left(\frac{S_i+S_j}{M_{ij}}\right) \quad (2)$$

Here, $S_i = \frac{1}{|C_i|}\sum_{x\in C_i}\|x - v_i\|_2$ denotes intra-cluster dispersion of cluster i; while $M_{ij} = \|v_i - v_j\|_2$ denotes inter-cluster separation. C represents the number of clusters; S_i and S_j are the standard deviations of point distance v from the center g_i within the clusters G_i and G_{ij}, respectively. Configuration C=4 has been selected to achieve optimal accuracy in identifying the tumor region. The segmentation performance of the proposed DBI-FCM approach was evaluated using binary tumor masks created by the proposed DBI-FCM algorithm against the ground truth masks available in the BraTS2020 and BraTS2021 datasetsBoth predicted masks and their corresponding reference masks were resized to the same resolution prior to performing the performance analysis. Segmentation performance was calculated in terms of Intersection over Union (IoU) and Dice Similarity Coefficient (DSC). The values presented here were acquired through an average of five independent experimentations.

Figure 3 shows the qualitative segmentation results on representative BraTS MRI samples: Original MRI Image, Ground Truth, and Overlay of Segmented Tumor in Original MRI Image. The results show good correlation between the segmented region and reference annotations.

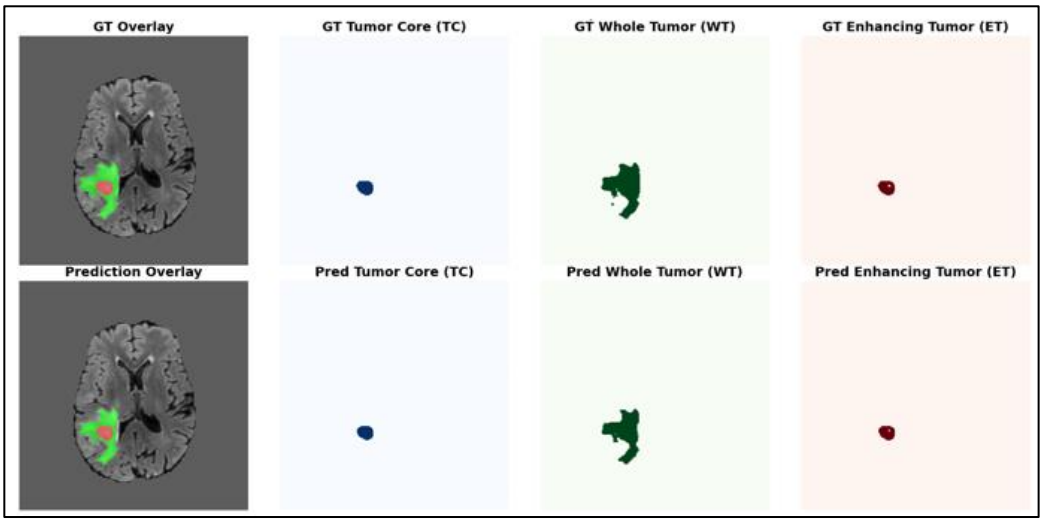

Figure 3. Qualitative segmentation comparison on representative BraTS MRI samples showing original MRI image, ground truth annotation, and overlay of the segmented tumor on the original MRI image

3.4 Feature Selection Using Catch Fish Optimization Algorithm

The feature selection technique based on optimization not only eliminates irrelevant and duplicated descriptors but also effectively reduces dimensionality and narrows the focus of the classification technique to more discriminative signals. This paper presents a CFOA-based technique for extracting important features for brain tumor classification. Optimization techniques like ACO, GWO, and WOA typically face problems such as premature convergence, poor population diversity, and an imbalance between exploration and exploitation processes when used with high-dimensional medical image features. The proposed CSD-CFOA technique improves upon the traditional CFOA technique using the CSD technique. Through

this, the estimation of candidate solutions is done based on quality and position diversity. Thus, population diversity can be maintained while achieving convergence stability. CFOA stands for Cooperative Fishing Optimization Algorithm, which is a metaheuristic optimization algorithm based on the idea of fishermen catching fish through cooperation and adaptation.

3.4.1 Initialization Process

Like other metaheuristic algorithms, CFOA begins by randomly initializing a population. Assume that N fishermen operate in an N-dimensional solution space. An initial population matrix is created where each row represents a potential solution for each fisherman. Each fisherman corresponds to a candidate solution with a set of decision variable values. An aim value, which estimates the quality of each solution, is calculated based on these values and is represented by Equation (3).

$$F(X) = [fit_1, fit_2, \dots, fit_N] = \begin{bmatrix} function(X_1), \\ function(X_2), \dots, function(X_N) \end{bmatrix} \quad (3)$$

Where fit_1 and $function(X_1)$ denotes target function value evaluated for the ith fishermen. Every X_i denotes a fisher location, which is initialized in Equation (4).

$$X_{i,j} = lb_j + rand \times (ub_j - lb_j), i = 1,2, \dots, N \quad (4)$$

Here $X_{i,j}$ represents the location of the ith fisherman in the jth dimension. ub_j and lb_j represent the upper and lower boundaries of the jth dimension in the solution space, respectively. Optimization approaches involve two stages: exploration and exploitation. This is controlled through the parameter S. When S is less than 0.5, CFOA explores, allowing it to search for new areas within the solution space. Conversely, when S is greater than or equal to 0.5, the algorithm shifts to exploitation, focusing on refining solutions in already promising regions. The parameter S is calculated using t/T, Where t and T indicate the current iteration count and the maximum number of iterations, respectively.

3.4.2 Independent Search and Group Capture

CFOA involves binary solution approaches in the exploration stage, such as self-determination and group apprehension. A fisherman will select a solution approach for catching a fish in each iteration, as well as switch between different approaches based on the catch rate parameter α, which is formulated in Equation (5).

$$\alpha = \left(1 - \frac{3}{2}s\right)^{\frac{3}{2}S} \quad (5)$$

When the catch rate α is higher, the fisherman chooses to catch the fish through individual search. The fisherman temporarily interrupts water during fishing to surface breathe. A fisherman identifies the position of a fish by perceiving the swells generated by the fish's breathing. Simultaneously, a fisherman will change their location according to the catch of others; this mechanism is represented in Equations (6) and (7).

$$X_{i,j}(t+1) = X_{i,j}(t) + \left(X_{r,j}(t) - X_{i,j}(t) \times Exp + rand \times s_j \times R\right) \quad (6)$$

$$Exp = \frac{fit_1 - fit_r}{fit_{max} - fit_{min}} \quad (7)$$

Here, $X_{r,j}(t)$ specifies that the arbitrary collection of the fisherman's positions varies from $X_{i,j}(t)$. Exp demonstrates the experience acquired through the ith fisherman $X_{i,j}(t)$ through $X_{r,j}(t)$ as a reference person. $fit_i(i = 1,2, \dots, N)$ indicates each fish's fitness. fit_{max} and fit_{min} denote the lowest and highest fitness values in a population at the current iteration. dis is the Euclidean distance from the ith fisherman to a situation point. S is an arbitrary unit vector of dimension D. Once α≤p, a fisherman chooses to work with a group to apprehend the fish. Fishermen coordinate with others to increase their chances of catching fish using nets. Three or more fishermen collaborate during fishing. This mechanism is formulated in Equations (8) and (9).

$$X_{i,j}(t+1) = X_{i,j}(t) + rand \times \left(Center_c - X_{c,j}(t)\right) + (1-2S)^2 \times r_1 \quad (8)$$

$$Center_c = mean\left(X_{c,j}(t)\right) \quad (9)$$

Where $Center_c$ denotes the target point of cth grip group. c means the group of candidates. r_1 is a random number [-1, 1]. $mean\left(X_{c,j}(t)\right)$ denotes an average location of $X_c(t)$.

3.4.3 Collective Capture

In a subsequent phase of catching fish, each fisherman adopts an integrated search strategy that guides both free and hidden fish towards a central gathering point for squeezing. The distribution of fishermen is focused around the school of fish, with the density gradually decreasing from the center to the edges. As the circulation narrows outward, fishermen in the center focus on capturing the main school of fish, while marginal fishermen target those attempting to escape. The corresponding updated equations are given in Equations (10) and (11).

$$X_i\binom{t}{+1} = X_{Gb} + normrnd\left(\begin{matrix}0,\\ \frac{r_2 \times \sigma \times}{\frac{|mean(X) - X_{Gb}|}{3}}\end{matrix}\right) \quad (10)$$

$$\sigma = \sqrt{\left(\frac{2\times(1-S)}{(1-S)^2+1}\right)} \quad (11)$$

Where X_{Gb} represents the location of the fisher, based on the optimal fitness value. GD distribution function has a mean of 0 and a standard deviation of σ. r_2 is an arbitrary integer between 1 and 2. $|.|$ denotes absolute values that are considered. S illustrates the search progression parameter and $(1 - S)$ defines the complementary search progression.

3.4.4 Boundary Control

Once the fisherman's location is determined, it is checked for boundary violations. If any dimension exceeds the predefined bounds, that dimension is regenerated using Equation (5), ensuring that each solution remains within the lower and upper bounds of the search space.

3.4.5 Controlled Search Diversification

The CFOA technique uses random searches, which were beneficial in thoroughly investigating the solution space. However, these random searches cannot be controlled, making

it difficult for the algorithm to converge on any particular point. To overcome the weaknesses of the CFOA technique, this study presents a new technique to enhance the balance, diversity, and adaptiveness of the algorithm. In this regard, the suggested CSD-CFOA method uses the objective fitness and positional diversity of candidates to make their estimate. In high dimensional spaces of MRI features, neighboring feature subsets have similar fitness values although they possess structurally distinct spatial characteristics.

Hence, the proposed research integrates the CSD mechanism into the traditional CFOA method. Contrary to most metaheuristic methods, which only rely on the agent's fitness score as the measure of its performance, the proposed model measures an agent on two fronts: the fitness score and positional distance. Through such dual measurement, a more accurate measure of the impact each fisherman brings to the system becomes possible. In the process of selecting the optimal fishermen, this research adopted the roulette wheel technique, whereby higher performing agents are selected more often than low-performing agents. Furthermore, instead of computing the simple mean value, the researchers compute the weighted position of each fisherman to more accurately represent their impacts on the system. Such measures are enabled by the use of fitness and distance as estimation techniques. The formula for CSD is presented in Equation (12).

$$Score(X_i) = w_1 \times norm(fit_i) + w_2 \times norm(dis_i) \tag{12}$$

Where $Score(X_i)$ represents the value for the ith fisherman. $norm(.)$ illustrates the normalized estimation. dis_i is the Euclidean distance between the ith fisherman and an optimal fisherman. w_1 and w_2 illustrate weights of binary matrices. The weighting parameters w_1 and w_2 determine the contribution of objective fitness and positional diversity in candidate estimation. In this research, both weights are empirically set to 0.5 to prioritize solution quality and search diversification uniformly. Applying balanced weights reduces bias toward single optimization criterion and enables consistent convergence. After computing the fitness value of each fisherman, the fisherman $X_{CSD}(t)$ is selected as the optimal fisherman using a roulette-wheel selection mechanism. A weighted average location is computed using Equations (13) and (14).

$$X_{wmean}(t) = \sum_{i=1}^{N} \gamma_i \times X_i(t) \tag{13}$$

$$\gamma_i = In(N+1) / \left(\sum_{i=1}^{N} \left(In \frac{(N+1)}{i} \right) \right) \tag{14}$$

Here, $X_{wmean}(t)$ represents an average position. An improved cooperative detention mechanism through CSD is formulated in Equation (15).

$$X_i(t+1) = X_{CSD}(t) + GD\left(0, \frac{r_2 \times \sigma \times |X_{wmean}(t) - X_{Gb}|}{3}\right) \tag{15}$$

In the CFOA rules, $r \in [0,1]$ illustrates the random exploration coefficient. $X_i(t+1)$ is the position of the ith fisherman at iterations t and $t+1$ individually. CSD-CFOA differs from recent CFOA variants by introducing a dual-criteria evaluation mechanism where both normalized fitness and positional diversity jointly determine solution quality. Compared to chaotic or multi-strategy CFOA extensions, which depend on arbitrary perturbations to increase diversity, CSD performs a controlled reinitialization guided by diversity thresholds and roulette-weighted selection probabilities. This approach stabilizes convergence and reduces the risk of stagnation, making a clear methodological difference from earlier CFOA improvements.

The enhancements provided by CSD are computationally justifiable by estimating its impact on population variance. Assume σ_t^2 denotes the positional variance of fishermen at iteration t. In traditional CFOA, σ_t^2 monotonically decreases due to recurrent exploitation phases, leading to early inactivity. In contrast, CSD maintains a lower bound on variance, as formulated in Equation (16).

$$\sigma_{t+1}^2 \geq \gamma . \sigma_t^2 \quad \gamma = \frac{1}{N}\sum_{i=1}^{N} \eta_i \tag{16}$$

Where η_i denotes the dual criteria weight combining fitness and positional distance. Since $\eta_i > 0$ the swarm recollects non-zero spatial spread, it stops premature convergence. Analytically, this stabilization speeds up global-optimum detection and reduces local-minimum trapping, providing a computational explanation for CSD-CFOA's improved convergence behavior. The fitness of each candidate feature subset is evaluated only using validation accuracy, while the test set remains completely unseen during optimization. After selecting a feature subset, the final model is trained on the training data and then evaluated on the test set. As the number of features increases, evaluating candidate solutions requires more computational effort for fitness calculation and diversity assessment. The CSD-CFOA selected feature dimension of 184, effectively filtering out 328 redundant deep feature channels before processing by the fully connected dense blocks.

3.5 Classification Using VGG19

In this research, the VGG19 [35] convolutional neural network is used as the main framework for both feature extraction with the help of convolutional layers and classification with the help of the softmax layer. The standard VGG19 architecture comprises 16 convolutional layers followed by three fully connected layers, resulting in a total of 19 trainable layers. Each convolutional layer employs a 3×3 kernel with a stride of 1 and appropriate zero-padding to preserve the spatial dimensions of the feature maps. In the proposed framework, the modified VGG19 architecture preserves the original sequence of convolutional and max-pooling operations in Blocks 1–4. In Block 5, the network contains four convolutional layers (Conv 1 to Conv 4). As compared to the standard VGG19 architecture, only the final max-pooling layer (MaxPool5) immediately following the Conv 4 layer is replaced by the proposed Tree-Pooling module, whereas all preceding max-pooling layers remain unchanged. The output of the Tree-Pooling module is subsequently forwarded to the fully connected layers, and the integrated Softmax layer performs the final three-class brain tumor classification such as (glioma, meningioma, and pituitary). Figure 4 presents the architecture of VGG19, which involves different layers: Max-Pooling, convolutions, and Fully-Connected [36].

The VGG19 model functions as the primary feature extraction from the spatial image tensor input. The CSD-CFOA algorithm does not operate on raw pixel feature spaces as a preprocessing stage. It is positioned downstream as an internal deep-feature selection mechanism located between the Tree-Pooling module and the first fully connected block.

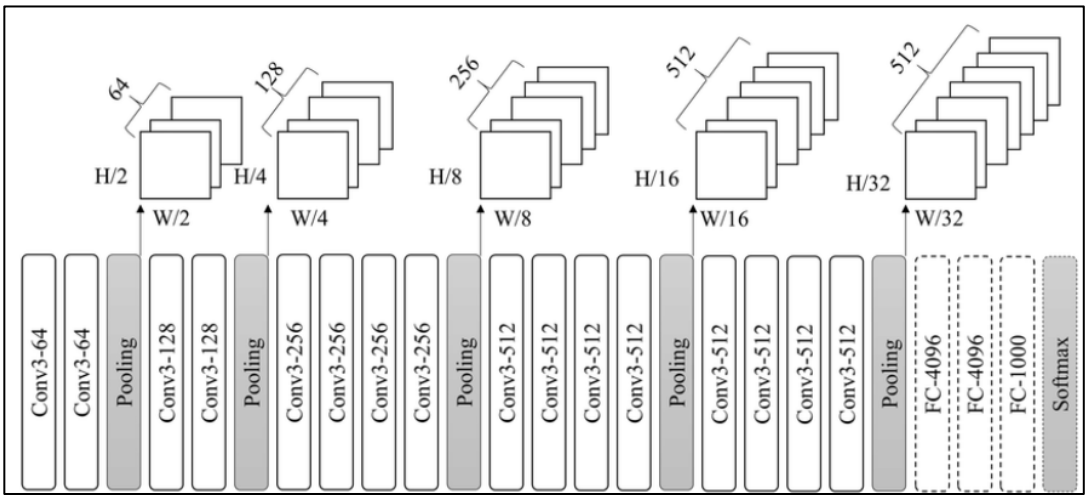

Figure 4. Architecture of VGG19, which involves different layers: pooling, convolutions and fully-connected [36]

These parameters are specifically tailored for MRI-based tumor classification to ensure task-specific optimization, rather than simply replicating ImageNet [33]. Figure 4 outlines VGG19's architecture. For a given convolutional layer output F_l, tree-pooling produces a pooled response P, as described in Equation (17).

$$P = \sum_{l=1}^{L} w_l . Pool(F_l) \tag{17}$$

Where w_l denotes the learnable significance weights. $Pool(.)$ is the dynamic pooling selection at each level. Tree-Pooling module is incorporated after the final convolutional block of the modified VGG19 network and before the fully connected layers. The resulting feature tensor of size 7 × 7 × 512 is partitioned into 2 × 2 local regions with stride 2 and hierarchically aggregated using a two-level binary tree. Trainable pooling weights are defined for the internal nodes in the tree and optimized alongside the convolutional parameters via backpropagation by using the Adam optimizer. In contrast to traditional max-pooling technique, the introduced trainable pooling mechanism learns the optimal feature aggregation such that the spatial discriminative information is retained while redundant feature responses are suppressed. These aggregated features are then fed into the fully connected layers and Softmax classifier calculates the posterior probability of each tumor category according to Equation (18).

$$P(C_k \mid F^*) = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}} \tag{18}$$

Where, $P(C_k \mid F^*)$ means posterior probability of class C_k for the selected feature subset F^*, z_k represents the output score of the fully connected layer, and K is the total number of brain tumor classes. Finally, the predicted brain tumor class is obtained by selecting the class with the maximum posterior probability, as expressed in Equation (19).

$$C = \arg\ \max P(C_k \mid F^*) \tag{19}$$

Where, denotes the predicted brain tumor class corresponding to the highest posterior probability. Algorithm 1 summarizes the complete workflow by integrating preprocessing, DBI-FCM segmentation, CSD-CFOA feature selection, Tree-Pooling-based VGG19 classification, and validation to improve the reproducibility of the proposed framework.

Algorithm 1: Pseudocode of the proposed DBI-FCM segmentation, CSD-CFOA feature selection, and Tree-Pooling-based VGG19 classification framework

Input: MRI images I (BraTS2020 and BraTS2021 for segmentation; Figshare dataset for classification)
Output: Predicted brain tumor class C
1: Load MRI image I
2: Preprocess I using median filtering, CLAHE, Z-score normalization by Eq. (1), and resizing
3: // DBI-FCM segmentation
4: Initialize the fuzzy cluster centers, membership matrix $U(0)$ with $C = 4$ clusters, and set the convergence tolerance $\varepsilon = 1\times10^{-5}$ (standard FCM threshold)
5: While $\|U(t+1) - U(t)\| > \varepsilon$ **Do**
  6: Update the fuzzy membership values $U(t+1)$ and cluster centers to minimize the clustering objective function
  7: Compute the Davies-Bouldin Index using Eq. (2)
8: End While
9: Select the cluster partition that minimizes the Davies-Bouldin Index in Eq. (2) as the optimal tumor segmentation
10: Feed the segmented image into the modified VGG19 network
11: Extract the deep feature vector F from Block 5 of the modified VGG19
12: // CSD-CFOA feature selection
13: Initialize the population of N fishermen $X_{i,j}$ using Eq. (4), $i = 1,2,...,N$
14: Evaluate the fitness $F(X)$ of each candidate solution using Eq. (3)
15: Set $t = 1$
16: While $t \leq T$ **Do**
  17: Compute the search progression parameter $S = t/T$
  18: For each fisherman (X_i) **Do**
    19: If $S < 0.5$
      20: // Exploration phase
      21: Compute the catch rate α using Eq. (5)
      22: If $\alpha > p$
        23: Choose a reference fisherman X_r randomly
        24: Compute the experience Exp using Eq. (7)
        25: Update the position of the i-th fisherman using Eq. (6)
      26: Else If $\alpha \leq p$
        27: Form a capture group c and compute $Center_c$ using Eq. (9)
        28: Update the position of the i-th fisherman using Eq. (8)
      29: End If
    30: Else
      31: // Exploitation phase: collective capture
      32: Compute the standard deviation σ using Eq. (11)
      33: Update the position of the i-th fisherman using Eq. (10)
    34: End If
    35: Check the boundary violations and regenerate the exceeding dimensions using Eq. (4)
  36: End For
  37: // Controlled Search Diversification (CSD)
  38: For each fisherman (X_i) **Do**

39: Compute $Score(X_i)$ using Eq. (12)
40: End For
41: Select the optimal fisherman $X_{CSD}(t)$ via roulette-wheel selection based on $Score(X_i)$
42: Compute the weighting coefficient γ_i using Eq. (14)
43: Compute the weighted mean position $X_{wmean}(t)$ using Eq. (13)
44: Update the position of the i-th fisherman using Eq. (15)
45: Convert the continuous positions into binary feature masks using a sigmoid-binarization threshold
46: Recompute the fitness of each candidate feature subset using Eq. (3)
47: Update the global best solution X_{Gb}
48: $t = t + 1$
49: End While
50: Obtain the optimal feature subset $F^* = F \odot X_{best}$
51: Apply Tree-Pooling to the selected feature maps using Eq. (17)
52: Pass the pooled features through the fully connected layers
53: Compute the Softmax probabilities $P(C_k|F^*)$ using Eq. (18)
54: Assign the predicted class C = argmax $P(C_k|F^*)$ using Eq. (19)
55: Return C

4. Experimental Results

All experiments for the proposed method are conducted using an Intel Core i7 processor, an NVIDIA RTX 3090 GPU (24 GB), 64 GB RAM, Python 3.10, and TensorFlow 2.12. Segmentation performance is assessed using DSC and IoU. The experimental evaluation followed a two-stage validation strategy: primary and secondary. In the primary validation strategy, the dataset was partitioned using an 80:20 ratio. In the secondary validation strategy, 5-fold cross-validation was conducted on the training subset to optimize model parameters and assess the robustness of the proposed framework. To prevent data leakage during segmentation evaluation, the BraTS datasets were processed on a patient-wise basis. Classification performance is measured with accuracy (Acc), precision (Pre), sensitivity (Se), and F1-score. All experiments are conducted using a fixed random seed of 42 to ensure reproducibility. The proposed VGG19 with the tree-pooling module is trained using a batch size of 32, an initial learning rate of 0.0001, and the Adam optimizer. The network is trained for 100 epochs, and a dropout rate of 0.5 is applied to the fully connected layers. For feature selection, the CSD-CFOA algorithm used hyperparameters such as a population size of 40, a maximum of 150 iterations, an adaptive catch factor, and a sigmoid-binarization threshold of 0.5 for feature-mask generation. All hyperparameters remained fixed across folds to ensure reproducibility. This unified training and optimization setup resulted in consistent convergence and less variance over repeated runs, as evidenced by the ablation study results.

4.1 Performance Estimation

This section describes the performance of the suggested brain tumor segmentation and classification model by means of a thorough experimental investigation. In the current study, the BraTS datasets are employed only for tumor segmentation, and the three-class tumor classification experiment is carried out using only the Figshare dataset. Thus, Figshare MRI scans are resized and then fed to the VGG19 classifier following preprocessing, while DBI-

FCM segmentation is executed on BraTS2020 and BraTS2021 datasets for evaluation of segmentation. The ROI masks obtained via DBI-FCM are not used in Figshare classification.

Table 2. Performance analysis of the segmentation results

Dataset	Methods	IoU (%)	DSC (%)
BraTS2020	Otsu Thresholding	71.23	73.88
	Watershed	76.35	78.22
	Region-based	83.67	85.11
	DBI-FCM	94.39	97.11
BraTS2021	Otsu Thresholding	69.80	71.23
	Watershed	74.94	76.56
	Region-based	82.15	83.92
	DBI-FCM	93.68	96.74

Table 2 represents the segmentation accuracy obtained through different conventional techniques and DBI-FCM technique. Otsu Thresholding, Watershed, and Region-Based techniques fail to perform well due to overlapping structures and noise in MRI images, resulting in low values of IoU and DSC scores. On the other hand, better segmentation results can be obtained with the help of DBI-FCM as a result of the compactness of its DBI, which helps retain the spatial information necessary for tumor boundary detection.

Table 3. Analysis of feature selection results for brain tumor classification using the Figshare dataset

Methods	Acc (%)	Pre (%)	Se (%)	F1-score (%)	AUC (%)
WOA	85.77	83.24	86.91	85.03	91.62
GWO	88.62	86.33	89.48	87.87	93.41
ACO	90.11	89.29	91.55	90.40	95.08
Proposed CSD-CFOA	98.53	98.57	97.58	98.07	98.84

Results for feature selection on the Figshare dataset are shown in Table 3. Conventional algorithms like WOA, GWO, and ACO provide satisfactory outcomes; however, CFOA surpasses them with the best scores under all criteria. The reason is that CSD-CFOA is a customized and improved algorithm with the CSD mechanism, ensuring that there will be no stagnation in the local optima.

Table 4. Performance evaluation of the VGG19 with tree pooling compared with existing brain tumor classification method

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)	Mean Accuracy (%)	Standard deviation	95% Confidence interval
VGG16	88.11	87.02	89.22	88.10	91.34	87.94	±0.30	[87.64-88.24]
ResNet50	90.64	89.33	91.77	90.53	91.56	90.48	±0.25	[90.23-90.73]
InceptionNet	91.98	91.01	93.14	92.06	92.57	91.83	±0.23	[91.60-92.06]
ViT-B/16	94.43	91.12	92.59	91.84	93.38	94.27	±0.20	[94.07-94.47]
Swin Transformer	96.48	91.48	93.48	92.46	95.75	96.31	±0.18	[96.13-96.49]
CrossViT	96.87	94.28	94.56	94.41	96.88	96.73	±0.16	[96.57-96.89]
EfficientFormer	97.48	96.47	95.38	95.92	97.91	97.34	±0.15	[97.19-97.49]
TCAFNet	95.92	94.39	94.32	94.35	96.12	95.81	±0.19	[95.62-96.00]

FTVT-b32	97.49	97.38	96.21	96.79	98.06	97.35	±0.14	[97.21-97.49]
VGG19 with tree pooling	98.53	98.57	97.58	98.07	98.84	98.29	±0.15	[98.14-98.44]

Note: The Accuracy (%) indicates the accuracy of classification achieved for the independent 80:20 test set. The Mean Accuracy (%), Standard Deviation (±), and 95% Confidence Interval were computed based on five independent experiments that were performed in exactly the same training environment.

The performance evaluation results of the VGG19 with Tree Pooling approach are provided in Table 4. Various existing approaches, including VGG16, Residual Network (ResNet50), InceptionNet, Base with 16 × 16 patches (ViT-B/16), Swin Transformer, Cross Scale Transformer (CrossViT), and Efficient Attention Transformer (EfficientFormer), are investigated in the experiment and compared to the VGG19 model. All baseline classification models, namely VGG16, ResNet50, InceptionNet, ViT-B/16, Swin Transformer, CrossViT, EfficientFormer, TCAFNet, and FTVT-b32, have been independently reimplemented with the corresponding architectural configuration from the publications. Each architecture is trained and tested on the same Figshare dataset, preprocessing procedure and comparable hyperparameters. The ViT-B/16 architecture is used for transformer-based comparison. While transformer-based architectures, such as ViT-B/16 and Swin Transformer, rely on self-attention mechanisms for global feature learning, these networks retain sensitivity to the size and diversity of the training dataset. On the contrary, the proposed framework provides stability with respect to optimization-based feature selection and tree-pooling aggregation.

Table 5 shows the k-fold cross-validation results of the suggested framework using the Figshare dataset, which shows the model's performance with different validation setups. The evaluation criteria do not vary much regardless of the K value, implying that the architecture's generalization ability is guaranteed. Particularly, the K=5 setup ensures a balanced bias-variance trade-off and thus confirms the reliability of the framework under such validation settings. The results from the K-fold cross-validation are shown below to determine the robustness and generalization ability of the suggested framework. Small variations may be observed in the cross-validation results and the independent test since both use different data partitions.

Table 5. Performance from fold-wise validation for five-fold cross-validation on the training subset using Figshare dataset

Dataset	K-Indices (K=5)	Pre (%)	Se(%)	Acc (%)	F1-score (%)	AUC (%)
Figshare	1	98.05	97.98	98.12	98.01	98.01
	2	98.38	98.31	98.45	98.34	98.34
	3	98.22	98.15	98.34	98.18	98.18
	4	97.98	97.92	98.08	97.94	97.95
	5	98.42	98.34	98.46	98.37	98.38
	Mean ± SD	98.21±0.174	98.14±0.169	98.29±0.161	98.16±0.171	98.17±0.171

Table 6. Comparison of computational complexity for the proposed method across two datasets

Dataset	Segmentation models	Inference time (s)	Processing time (s)	Memory usage (MB)	Inference FLOPs (G)
	Segmentation results				
BraTS2020	Otsu Thresholding	0.016	198	186	0.12
	Watershed	0.013	193	224	0.18
	Region-based	0.021	187	276	0.26
	DBI-FCM	0.010	181	318	0.34
BraTS2021	Otsu Thresholding	0.028	196	189	0.13
	Watershed	0.016	191	228	0.19

	Region-based	0.023	186	281	0.27
	DBI-FCM	0.011	179	324	0.35
	Classification results				
Figshare	**Feature selection and Classification models**	**Inference time (s)**	**Training time (s)**	**Memory usage (MB)**	**Inference FLOPs (G)**
	WOA-VGG19 with tree pooling	0.124	1842	2146	20.84
	GWO-VGG19 with tree pooling	0.113	1836	2178	21.32
	ACO-VGG19 with tree pooling	0.101	1831	2215	21.78
	CSD-CFOA-VGG19 with tree pooling	0.086	1825	2168	20.63

To measure the complexity of computations, the performance of the proposed model was measured using inference time, training time, memory consumption, and Floating-Point Operations (FLOPs) as shown in Table 6. The computational complexity of the entire classification architecture consisting of VGG19 forward propagation and the respective feature selection algorithm is expressed in FLOPs. Computational complexity evaluation was done using the same hardware and training conditions for all models. Since each approach employs the same VGG19 backbone, only slight differences in training time and memory usage are observed, mainly due to the computational overhead introduced by the respective feature selection algorithms. The proposed CSD-CFOA-VGG19 framework achieves competitive computational efficiency while maintaining superior classification performance. Computational efficiency is assessed using four metrics:inference time, training time, memory usage, and FLOPs. Inference and training time are used to measure the execution latency during testing and the total duration of the learning process, respectively.

Table 7. Statistical significance analysis using paired two-tailed t-test with Holm-Bonferroni correction (α = 0.05, n = 30, df = 29) for all three datasets

Methods	**Paired t-statistic**	**Two-tailed p-value**	**Rank (i)**	**Holm-Bonferroni threshold ($\frac{\alpha}{m-i+1}$)**	**Significant (α = 0.05)**
Segmentation results					
DBI-FCM vs Region-based	6.02	0.000002	1	0.016	Yes
DBI-FCM vs Watershed	5.28	0.000012	2	0.025	Yes
DBI-FCM vs Otsu	4.73	0.000054	3	0.05	Yes
Classification results					
VGG19 with tree pooling vs EfficientFormer	6.14	0.000001	1	0.0071	Yes
VGG19 with tree pooling vs CrossViT	5.89	0.000002	2	0.0083	Yes
VGG19 with tree pooling vs Swin Transformer	5.34	0.000010	3	0.01	Yes
VGG19 with tree pooling vs ViT-B/16	4.92	0.000032	4	0.0125	Yes
VGG19 with tree pooling vs ResNet50	4.90	0.000033	5	0.0167	Yes
VGG19 with tree pooling vs InceptionNet	4.38	0.000141	6	0.0250	Yes
VGG19 with tree pooling vs VGG16	4.36	0.000154	7	0.05	Yes

Table 7 illustrates the statistical analysis based on IoU (%) for segmentation and classification accuracy (%) for classification, obtained from 30 paired observations. For the segmentation experiments on the BraTS2020 and BraTS2021 datasets, the statistical analysis was performed using the IoU (%) values obtained from 30 paired observations generated

through five-fold cross-validation repeated 6 times. The statistical significance analysis is conducted using paired two-tailed t-tests. The number of paired observations (n = 30) is calculated using 5 folds × 6 repetitions, resulting in 29 degrees of freedom (df = 29). The corresponding paired t-statistic values quantify the statistical differences between the proposed and baseline methods, while the reported raw and Holm-Bonferroni threshold indicate the statistical significance of the observed performance differences, where m denotes number of compared models. All two-tailed p-values remained below the significance level of $\alpha = 0.05$, indicating statistically significant improvements. The null hypothesis (H_0) states that there is no significant difference in performance between the proposed model and the baseline methods. The alternative hypothesis (H_1) asserts that the proposed model achieves a statistically significant improvement.

Table 8. Mean and standard deviation of average convergence behavior of CFOA and CSD-CFOA

Iterations	CFOA – Fitness (Mean ± Std)	CSD-CFOA – Fitness (Mean ± Std)
10	0.892 ± 0.018	0.915 ± 0.012
20	0.906 ± 0.016	0.938 ± 0.010
40	0.921 ± 0.015	0.964 ± 0.008
60	0.948 ± 0.014	0.976 ± 0.006
80	0.961 ± 0.014	0.981 ± 0.005
100	0.973 ± 0.013	0.984 ± 0.004

Table 8 illustrates the mean and standard deviation of the average convergence behavior of CFOA and CSD-CFOA. Compared with the baseline optimization approach, the proposed method exhibits reduced fluctuations during later optimization iterations, indicating improved search stability and adaptive feature selection. The primary advantage of the proposed CSD-CFOA lies in its faster convergence rate and lower variability across iterations rather than a substantial increase in the final fitness value.

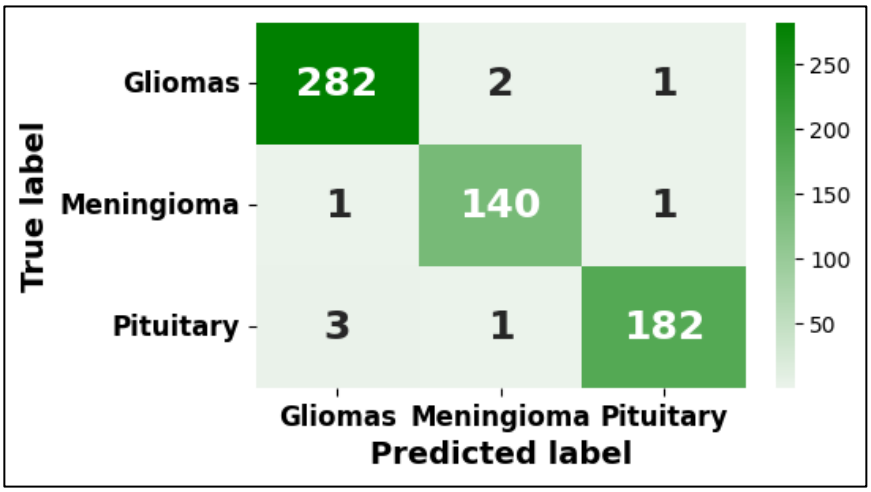

Figure 5. Confusion matrix of the proposed CSD-CFOA-VGG19 method obtained from the independent test set using a stratified 80:20 train–test split

Figure 5 shows the confusion matrix for the CSD-CFOA-VGG19 model, tested using an independent dataset comprising 613 MRI images (285 Glioma, 142 Meningioma, and 186 Pituitary) through a stratified 80:20 split method. It demonstrates good class-wise separation without any misclassification, indicating that the optimized feature representation is quite efficient. The low rates of false positives and false negatives also confirm the high degree of classification accuracy. As all 613 test data points are considered, the confusion matrix ensures complete consistency between the dataset splitting approach used and the final result presented.

Figure 6 depicts the ROC curve of the CSD-CFOA-VGG19 model with the Figshare dataset. The curves, being close to the top-left corner, show good class separation. Additionally, the high AUC values indicate the model's strong classification capability.

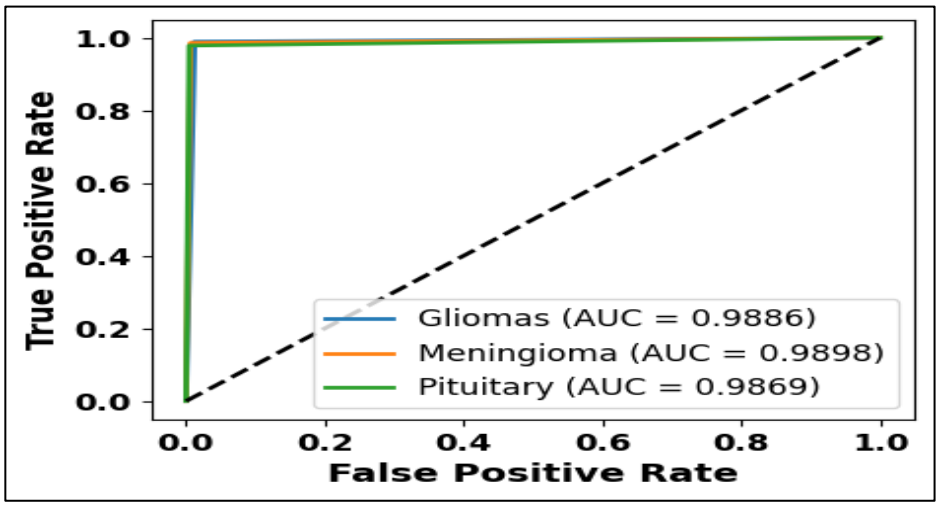

Figure 6. ROC curve analysis of the proposed framework using the Figshare dataset

The results of ablation experiments evaluating DBI-FCM-based segmentation and CSD-CFOA feature selection, both individually and together, are presented in Table 9. Table 10 shows the results of ablation experiments to evaluate the proposed tree-pooling method across three different settings: VGG19 as the baseline, CSD-CFOA feature selection on VGG19, and tree-pooling on VGG19. From Tables 9 and 10, it is observed that the baseline VGG19 network without enhancements demonstrates the minimum accuracy, which points to a limited capacity to process complex, high-dimensional brain MRI data. The integrated architecture combining optimization-based feature selection and tree-pooling achieves the maximum classification accuracy. Results demonstrate that tree-pooling improves spatial feature preservation and boosts classification accuracy by effectively aggregating discriminative tumor features.

Table 9. Component-wise ablation study of the proposed segmentation method

Dataset	Methods	IoU (%)	DSC (%)
BraTS2020	K-means clustering	84.23	91.44
	Traditional FCM	86.33	92.66
	Spatial FCM	87.32	93.23
	Probabilistic C-Means	92.12	95.90
	DBI-FCM	94.39	97.11
BraTS2021	K-means clustering	82.01	90.12
	Traditional FCM	84.78	91.76
	Spatial FCM	86.11	92.54
	Probabilistic C-Means	89.34	94.37
	DBI-FCM	93.68	96.74

Table 10. Ablation study highlighting the individual performance contributions of pipeline components on the Figshare dataset

Dataset	Methods	Pre (%)	Se (%)	F1-score (%)	Acc (%)	AUC (%)
Figshare	Baseline VGG19	90.22	91.19	90.70	91.31	93.21
	VGG19 + Tree-Pooling	91.14	92.91	92.01	92.42	95.45
	VGG19 + Tree-Pooling + CFOA	93.48	94.29	93.88	95.58	96.67
	VGG19 + Tree-Pooling + CSD-CFOA	98.57	97.58	98.07	98.53	98.84

4.2 Comparative Analysis

Table 11 presents a comparative analysis of segmentation for brain tumors across the BraTS2020 and BraTS2021 datasets. Table 12 illustrates a comparison of classification results using Figshare datasets. Existing methods, such as Spatial Attention Network (SA-Net) [21], Residual U-Net+ (ResUNet+) [22], Resize + Grayscale conversion Augmentation + PDCNN [24], CJHBA-based DRN [25], TCAFNet [29], and FTVT-b32 [30] are evaluated and compared with the proposed CSD-CFOA combined with the VGG19 approach. To ensure a fair comparison, all baseline models were independently reimplemented and trained under the same experimental conditions such as dataset splits, preprocessing techniques, and experimental settings identical to the proposed framework. For comparative analysis, the proposed method and all traditional models are reimplemented under a uniform dataset, preprocessing techniques, and hyperparametric settings. Additionally, these traditional models are evaluated under identical evaluation settings adapted for the proposed model.

Table 11. Comparative analysis of segmentation performance for brain tumor using BraTS2020 and BraTS2021 datasets

Dataset	Methods	IoU (%)	DSC (%)	Source Referenced/ Reimplemented
BraTS2020	SA-Net [21]	76.61	77.79	Reimplemented
	ResUNet+ [22]	91.42	92.80	Reimplemented
	DBI-FCM	94.39	97.11	-
BraTS2021	SA-Net [21]	76.94	77.25	Reimplemented
	ResUNet+ [22]	93.57	94.25	Reimplemented
	DBI-FCM	93.68	96.74	-

Table 12. Comparative analysis of classification performance using the Figshare dataset

Dataset	Methods	Accuracy (%)	Precision (%)	Recall (%)	Source Referenced/ Reimplemented
Figshare	Resize + Gray scale conversion +Augmentation + PDCNN [24]	96.39	95.39	95.23	Reimplemented
	CJHBA-based DRN [25]	91.14	90.67	90.31	Reimplemented
	TCAFNet [29]	95.92	94.39	94.32	Reimplemented
	FTVT-b32 [30]	97.49	97.38	96.21	Reimplemented
	Proposed CSD-CFOA with tree pooling based VGG19	98.53	98.57	97.58	-

4.3 Discussion

Despite the notable achievements of the proposed CSD-CFOA with the VGG19 model, some limitations are identified in the experiment. The proposed framework has three main limitations. First, because it mainly depends on supervised learning, it requires a large labeled dataset, which limits the model's performance on new, real-world data due to variations in image data types (domain shifts). Second, extensive preprocessing steps such as noise removal and normalization increase the computational workload before classification even starts. Finally, although the model performs well across the BraTS2020, BraTS2021, and Figshare datasets, different hospitals use different MRI settings, which may occasionally cause inconsistent results. The proposed framework presents three distinct limitations. First, the model's supervised learning framework depends on extensively labeled datasets, which limit generalization on unseen clinical data due to variations in image acquisition (domain shifts). Second, preprocessing steps such as noise reduction and intensity normalization create additional computational overhead before classification. Additionally, differences in MRI acquisition parameters cause performance variations across the BraTS2020, BraTS2021, and

Figshare datasets. To optimize classification accuracy and efficiency, this research presents a feature selection method based on the CSD-CFOA. This paper introduces an integrated pipeline combining DBI-FCM- regularized unsupervised segmentation and optimization-driven feature selection CSD-CFO). Subsequently, the selected features are processed by a VGG19 classification model with an adaptive multi-level tree-pooling mechanism to categorize the tumors. A limitation of the proposed framework is that evaluation relies on a single representative 2D axial slice containing the largest tumor region from each BraTS patient volume. Consequently, the reported IoU and DSC values represent slice-wise segmentation performance rather than complete three-dimensional volumetric segmentation. Extending the proposed framework to full-volume segmentation will be considered as future work.

5. Conclusion

The primary objective of this research is to propose an automated brain tumor classification system to identify brain tumors and facilitate appropriate treatment. Thus, a DBI-FCM based segmentation, CSD-CFOA based feature selection and modified VGG19 for classification are proposed to enhance accurate classification of brain tumors. The DBI-FCM method is used to localize and segment tumor regions from the BraTS2020 and BraTS2021 datasets. The FCM method handles overlapping tissue boundaries by assigning soft membership percentages to individual image pixels. By integrating FCM with the DBI, the system automatically calculates the optimal number of tissue clusters, reduces sensitivity to image noise, and delivers automated, highly accurate tissue segmentation. Following this, CSD-CFOA is proposed to address premature convergence by utilizing CSD for maintaining high population diversity and preventing local optima stagnation. This mechanism effectively preserves critical, subtle tumor markers while isolating a minimal, highly discriminative feature subset that improves classification accuracy. Finally, the VGG19 model classifies brain tumors into multiple categories based on the selected features. The proposed CSD-CFOA defines structural feature selection parameters and tumor boundaries, providing a foundation for future medical image analysis tools. Experimental results show that combining CSD-CFOA with VGG19 achieves a 98.53% accuracy rate, which is higher than benchmark methods like the CJHBA. Future work will focus on testing the network's flexibility under large-scale, real-time conditions, and on integrating mechanisms to improve overall performance.

References

[1] Yurtsever, M. M. E., Y. Atay, B. Arslan, and S. Sagiroglu. "Development of Brain Tumor Radiogenomic Classification Using GAN-Based Augmentation of MRI Slices in the Newly Released Gazi Brains Dataset." BMC Medical Informatics and Decision Making 24, no. 1 (2024): 285.

[2] Ramprasad, M. V. S., M. Z. U. Rahman, and M. D. Bayleyegn. "SBTC-net: Secured Brain Tumor Segmentation and Classification Using Black Widow with Genetic Optimization in IoMT." IEEE Access 11 (2023): 88193–88208.

[3] Dash, S., M. Siddique, S. Mishra, D. J. Gelmecha, S. Satapathy, D. S. Rathee, and R. S. Singh. "Brain Tumor Detection and Classification Using IFF-FLICM Segmentation and Optimized ELM Model." Journal of Engineering 2024, no. 1 (2024): 8419540.

[4] Sahoo, A. K., P. Parida, K. Muralibabu, and S. Dash. “An Improved DNN with FFCM Method for Multimodal Brain Tumor Segmentation.” Intelligent Systems with Applications 18 (2023): 200245.

[5] Xu, Y., K. Yu, G. Qi, Y. Gong, X. Qu, L. Yin, and P. Yang. “Brain Tumour Segmentation Framework with Deep Nuanced Reasoning and Swin-T.” IET Image Processing 18, no. 6 (2024): 1550–1564.

[6] Zhang, Q., W. Qi, H. Zheng, and X. Shen. “Cu-net: A U-net Architecture for Efficient Brain-Tumor Segmentation on BraTS 2019 Dataset.” In 2024 4th International Conference on Machine Learning and Intelligent Systems Engineering (MLISE), IEEE, (2024): 255-258.

[7] Fu, Z., Z. Li, Y. Li, and H. Chen. “MICFOA: A Novel Improved Catch Fish Optimization Algorithm with Multi-Strategy for Solving Global Problems.” Biomimetics 9, no. 9 (2024): 509.

[8] Kanna, R.K. and Salau, A.O., 2025. New Cognitive Computational Strategy for Optimizing Brain Tumour Classification Using Magnetic Resonance Imaging Data. Intelligence-Based Medicine, 11, p.100215.

[9] Sangui, S., T. Iqbal, P. C. Chandra, S. K. Ghosh, and A. Ghosh. “3D MRI Segmentation Using U-Net Architecture for the Detection of Brain Tumor.” Procedia Computer Science 218 (2023): 542–553.

[10] Balamurugan, T., and E. Gnanamanoharan. “Brain Tumor Segmentation and Classification Using Hybrid Deep CNN with LuNetClassifier.” Neural Computing and Applications 35 (2022): 4739–4753.

[11] Aly, M., and A. S. Alotaibi. “EMU-Net: Automatic Brain Tumor Segmentation and Classification Using Efficient Modified U-Net.” Computers, Materials & Continua 77, no. 1 (2023): 557–582.

[12] Jabbar, A., S. Naseem, T. Mahmood, T. Saba, F. S. Alamri, and A. Rehman. “Brain Tumor Detection and Multi-Grade Segmentation through Hybrid Caps-VGGNet Model.” IEEE Access 11 (2023): 72518–72536.

[13] Asiri, A. A., A. Shaf, T. Ali, M. Aamir, M. Irfan, S. Alqahtani, K. M. Mehdar, H. T. Halawani, A. H. Alghamdi, A. F. A. Alshamrani, and S. M. Alqhtani. “Brain Tumor Detection and Classification Using Fine-Tuned CNN with ResNet50 and U-Net Model: A Study on TCGA-LGG and TCIA Dataset for MRI Applications.” Life 13, no. 7 (2023): 1449.

[14] Shreeharsha, J. “Brain Tumor Segmentation and Classification Using Binomial Thresholding-Based Bidirectional Long Short-Term Memory.” International Journal of Intelligent Engineering & Systems 17, no. 3 (2024): 149–158.

[15] Agrawal, P., N. Katal, and N. Hooda. “Segmentation and Classification of Brain Tumor Using 3D-UNet Deep Neural Networks.” International Journal of Cognitive Computing in Engineering 3 (2022): 199–210.

[16] Kurdi, S. Z., M. H. Ali, M. M. Jaber, T. Saba, A. Rehman, and R. Damaševičius. "Brain Tumor Classification Using Meta-Heuristic Optimized Convolutional Neural Networks." Journal of Personalized Medicine 13, no. 2 (2023): 181.

[17] Bompem, G., and D. Pandluri. "Batch Normalization Based Convolutional Neural Network for Segmentation and Classification of Brain Tumor MRI Images." International Journal of Intelligent Engineering & Systems 17, no. 2 (2024): 39–49.

[18] Khan, S. U. R., S. Asif, M. Zhao, W. Zou, Y. Li, and C. Xiao. "ShallowMRI: A Novel Lightweight CNN with Novel Attention Mechanism for Multi Brain Tumor Classification in MRI Images." Biomedical Signal Processing and Control 111 (2026): 108425.

[19] Arif, M., A. Jims, O. Geman, M. D. Craciun, and F. Leuciuc. "Application of Genetic Algorithm and U-Net in Brain Tumor Segmentation and Classification: A Deep Learning Approach." Computational Intelligence and Neuroscience 2022, no. 1 (2022): 5625757.

[20] Aggarwal, M., A. K. Tiwari, M. P. Sarathi, and A. Bijalwan. "An Early Detection and Segmentation of Brain Tumor Using Deep Neural Network." BMC Medical Informatics and Decision Making 23 (2023): 78.

[21] Roy, S., R. Saha, S. Sarkar, R. Mehera, R. K. Pal, and S. K. Bandyopadhyay. "Brain Tumour Segmentation Using S-Net and SA-Net." IEEE Access 11 (2023): 28658–28679.

[22] Metlek, S., and H. Çetıner. "ResUNet+: A New Convolutional and Attention Block-Based Approach for Brain Tumor Segmentation." IEEE Access 11 (2023): 69884–69902.

[23] Almufareh, M. F., M. Imran, A. Khan, M. Humayun, and M. Asim. "Automated Brain Tumor Segmentation and Classification in MRI Using YOLO-Based Deep Learning." IEEE Access 12 (2024): 16189–16207.

[24] Rahman, T., and M. S. Islam. "MRI Brain Tumor Detection and Classification Using Parallel Deep Convolutional Neural Networks." Measurement: Sensors 26 (2023): 100694.

[25] Deepa, S., J. Janet, S. Sumathi, and J. P. Ananth. "Hybrid Optimization Algorithm Enabled Deep Learning Approach Brain Tumor Segmentation and Classification Using MRI." Journal of Digital Imaging 36, no. 3 (2023): 847–868.

[26] Al Abboodi, H. M., A. W. Al-Funjan, A. Aldhahab, W. M. S. Abedi, A. H. Abdullah, and A. H. Siraj. "High-Resolution Model for Segmenting and Predicting Brain Tumor Based on Deep UNet with Multi Attention Mechanism." International Journal of Intelligent Engineering & Systems 17, no. 2 (2024): 294–310.

[27] Kesav, O. H., and G. K. Rajini. "Enhancing Brain Tumor Detection and Classification with Reduced Complexity Spatial Fusion Convolutional Neural Networks." International Journal of Intelligent Engineering & Systems 17, no. 1 (2024): 263.

[28] Topannavar, P. S., D. M. Yadav, and V. Bendre. "Global Self-Attention Module Based Convolutional Neural Network for Brain Tumor Classification." International Journal of Intelligent Engineering & Systems 17, no. 3 (2024): 647–657.

[29] Mohammadi, M. and M. Ziaratban. "TCAFNet: Transformer-Guided Cross-Scale Attention and Deep Semantic Fusion for Brain Tumour Classification from MRI." IET Image Processing, Vol. 20, no. 1 (2026): 70303.

[30] Reddy, C., P.A. Reddy, H. Janapati, B. Assiri, M. Shuaib, S. Alam, and A. Sheneamer, "A Fine-Tuned Vision Transformer Based Enhanced Multi-Class Brain Tumor Classification Using MRI Scan Imagery." Frontiers in oncology, Vol. 14, (2024): 1400341.

[31] BraTs2020 Dataset Link: https://www.kaggle.com/datasets/awsaf49/brats20-dataset-training-validation. (Accessed on August 2025)

[32] BraTs2021 Dataset Link: https://www.kaggle.com/datasets/syedsajid/brats2021. (Accessed on August 2025)

[33] Figshare Dataset Link: https://www.kaggle.com/datasets/rahimanshu/figshare-brain-tumor-classification. (Accessed on August 2025)

[34] Bhardwaj, M., N. U. Khan, V. Baghel, S. K. Vishwakarma, and A. Bashar. "Brain Tumor Image Segmentation Using K-Means and Fuzzy C-Means Clustering." In Digital Image Enhancement and Reconstruction, Academic Press, (2023): 293–316.

[35] Sumaya, A. I., S. Forhad, M. Al Rafi, H. Rahman, M. H. Bhuyan, and Q. Tareq. "Comparative Analysis of AlexNet, GoogLeNet, VGG19, ResNet50, and ResNet101 for Improved Plant Disease Detection Using Convolutional Neural Networks." In 2024 2nd International Conference on Artificial Intelligence, Block chain, and Internet of Things (AIBThings) IEEE, (2024): 1-6.

[36] Brahmareddy, A., Arigela, A.K. and Nagalakshmi, M., 2024. Dffmd a Deepfake Face Mask Dataset for Infectious Disease Era with Deepfake Detection Algorithms.