



Research article

Optimal extreme learning machine for diagnosing brain tumor based on modified sailfish optimizer

Saad Ali Amin^{a,*}, Mashal Kasem Sulieman Alqudah^b, Saleh Ateeq Almutairi^c, Rasha Almajed^d, Mohammad Rustom Al Nasar^d, Hamzah Ali Alkhazaleh^{a,**}^a College of Engineering and IT, University of Dubai, Academic City, 14143, Dubai, United Arab Emirates^b Digital Transformation & Information Programs Department, Institute of Public Administration, Riyadh, Saudi Arabia^c Applied College, Computer Science, And Information Department, Taibah University, Medinah, Saudi Arabia^d College of Computer Information Technology (CCIT), Department of Information Technology Management, American University in the Emirates (AUE), Academic City, 14143, Dubai, United Arab Emirates

ARTICLE INFO

Keywords:

Brain tumor
MRI images
Automated method
Extreme learning machine (ELM)
Modified sailfish optimizer
Computer-aided detection system

ABSTRACT

This study proposes a hierarchical automated methodology for detecting brain tumors in Magnetic Resonance Imaging (MRI), focusing on preprocessing images to improve quality and eliminate artifacts or noise. A modified Extreme Learning Machine is then used to diagnose brain tumors that are integrated with the Modified Sailfish optimizer to enhance its performance. The Modified Sailfish optimizer is a metaheuristic algorithm known for efficiently navigating optimization landscapes and enhancing convergence speed. Experiments were conducted using the “Whole Brain Atlas (WBA)” database, which contains annotated MRI images. The results showed superior efficiency in accurately detecting brain tumors from MRI images, demonstrating the potential of the method in enhancing accuracy and efficiency. The proposed method utilizes hierarchical methodology, preprocessing techniques, and optimization of the Extreme Learning Machine with the Modified Sailfish optimizer to improve accuracy rates and decrease the time needed for brain tumor diagnosis. The proposed method outperformed other methods in terms of accuracy, recall, specificity, precision, and F1 score in medical imaging diagnosis. It achieved the highest accuracy at 93.95 %, with End/End and CNN attaining high values of 89.24 % and 93.17 %, respectively. The method also achieved a perfect score of 100 % in recall, 91.38 % in specificity, and 75.64 % in F1 score. However, it is crucial to consider factors like computational complexity, dataset characteristics, and generalizability before evaluating the effectiveness of the method in medical imaging diagnosis. This approach has the potential to make substantial contributions to medical imaging and aid healthcare professionals in making prompt and precise treatment decisions for brain tumors.

1. Introduction

Brain tumors represent a complex challenge in medical diagnostics due to their diverse nature and the criticality of accurate detection. These neoplasms, which can be either malignant or benign, arise from various cell types within the brain’s intricate

* Corresponding author.

** Corresponding author.

E-mail addresses: saamin@ud.ac.ae (S. Ali Amin), halkhazaleh@ud.ac.ae (H. Ali Alkhazaleh).<https://doi.org/10.1016/j.heliyon.2024.e34050>

Received 7 September 2023; Received in revised form 2 July 2024; Accepted 3 July 2024

Available online 3 July 2024

2405-8440/© 2024 Published by Elsevier Ltd.

This is an open access article under the CC BY-NC-ND license

[\(http://creativecommons.org/licenses/by-nc-nd/4.0/\)](http://creativecommons.org/licenses/by-nc-nd/4.0/).

structure, including glial cells, lymphatic tissue, and blood vessels [1]. The distinction between benign and malignant tumors is not always straightforward, requiring a multifaceted approach considering factors such as growth patterns, size, and location.

Early detection is paramount in managing brain tumors effectively. As tumors grow, they can exert pressure on brain tissue, leading to symptoms such as headaches, nausea, and balance disturbances. Misdiagnosis or delayed detection not only risks inappropriate treatment but also significantly reduces survival prospects. It is estimated that late diagnosis can shorten life expectancy by up to 20 years [2].

Traditionally, Magnetic Resonance Imaging (MRI) and Computed Tomography (CT) scans have been the cornerstone of brain tumor diagnosis [3]. MRI, in particular, offers superior image clarity due to its advanced magnetic technology, providing detailed insights into the brain's anatomy. The choice of MRI technique is tailored to the suspected tumor location, encompassing both cerebral and spinal regions.

Despite their efficacy, these imaging modalities are not without limitations. The accuracy of tumor characterization can be hindered by the subjective interpretation of images, leading to potential diagnostic errors [4]. Moreover, the process can be time-consuming and costly, necessitating multiple tests to reach a conclusive diagnosis [5].

The advent of Artificial Intelligence (AI) presents a promising avenue to overcome these challenges. AI algorithms have the potential to revolutionize brain tumor diagnostics by automating image analysis, thereby enhancing accuracy and reducing time-to-diagnosis. The integration of AI in medical imaging seeks to address the technical shortcomings of conventional methods while also offering a cost-effective solution.

The motivation for this study stems from the need to refine diagnostic processes for brain tumors through technological advancements. By exploring the application of AI in medical imaging, this work aims to contribute to the development of more precise and efficient diagnostic tools that could significantly improve patient outcomes.

2. Literature survey

Reported literature about brain tumors and how to diagnose them are explained in the following.

Bacanin et al. [6] suggested a metaheuristic method that could discover values that were near-optimal considering Convolutional Neural Network (CNN) parameters. Notably, it was founded on the Firefly algorithm. The system was developed to automatically classify the grades of glioma brain tumors using magnetic resonance imaging (MRI). In the beginning, the suggested improved algorithm was examined on a series of standard unrestricted cost functions, and its execution was contrasted to other improved alternatives and the original algorithm. To demonstrate the efficiency of the proposed method, its effect on optimizing the hyper-parameters within the Convolutional Neural Network (CNN) was investigated. For evaluation purposes, the Cancer Imaging Archive and the dataset of the IXI, which both contained numerous data collections, were utilized. Furthermore, the method was assessed using axial images of brain tumors. A comparison was conducted between the suggested algorithm and other modern algorithms in identical conditions. The results illustrated that the efficiency and robustness of the proposed method were much more excellent.

Deb and Roy [7] suggested a segmentation and classification algorithm for brain tumor detection, in which frog leap optimization and the Adaptive fuzzy deep Neural Network were utilized to diagnose the abnormality and normality of the image. In the suggested strategy, accurate categorization was gained, and the amount of error was minimized. After that, the image known as abnormal was segmented by the use of the Adaptive Flying Squirrel Algorithm; moreover, the tumor size was diagnosed. It should be noted that it was utilized to discover the tumor's rigorosity and was implemented in the MATLAB platform. False negative rate, false positive rate, specificity, sensitivity, and accuracy of the suggested research are, in turn, 0.543, 0.0043, 99.8 %, 99.9%, and 99.6 %. The accuracy of the diagnosis in the proposed study was better than that of the social group algorithm, deep neural network, and the current learning and teaching-based algorithm.

Liu et al. [8] suggested a technique that was aided by computers to detect brain tumors effectively and accurately. Furthermore, this research was undertaken to develop an automated brain tumor diagnostic system to assist medical professionals in making accurate diagnoses effortlessly. Generally, there were four major sections in this research, including segmentation techniques, pre-processing, extraction of features, and final classification. Discrete Wavelet Transform (DWT) and Grey-Level Co-occurrence Matrix (GLCM) were conducted to extract features of MRI images. Afterward, the samples were directed to an optimized Convolutional Neural Network (CNN) to be diagnosed. A design of Sparrow Search Algorithm categorization optimized the CNN. Eventually, the method's results were compared to three state-of-the-art algorithms on the Whole Brain Atlas (WBA) database to demonstrate its superior efficacy.

Maqsood et al. [9] suggested a method which consisted of five steps. Firstly, a linear contrast stretching technique was utilized to detect the edges within the image of the source. Secondly, a custom deep neural network method consisting of 17 layers was created to segment tumors in the brain. Thirdly, an improved MobileNetV2 method was utilized to extract features, which was exercised by the use of transfer learning. To further enhance the accuracy of the segmentation, an entropy-based controlled approach was employed in the fourth step. This was combined with a Multi-class Support Vector Machine (M-SVM) to select the best features for optimal results. The final step in accurately classifying brain tumors involves utilizing M-SVM. This step recognizes the different types of tumors, including Glioma, Pituitary tumor images, and Meningioma. The proposed method was tested and proven effective on the Figshare and BraTS 2018 datasets. The results of the study illustrated that the suggested BT classification and detection performed better than other strategies considering both quantitatively and visually, and their accuracy was, in turn, 98.92 % and 97.47 %. In the end, the eXplainable Artificial Intelligence (XAI) technique was implemented to clarify the findings.

Bashkandi et al. [10] suggested a method to eliminate all the problems that existed in reading MRI images and early detection of the brain tumor. There were several issues due to the outdated and traditional approaches, which resulted in taking a long time for doctors

to detect the tumors in the brain. In the field of medical imaging, a computer system was utilized to detect accurately brain tumors in brain images. To enhance the readability of these images, they were subjected to certain modifications. Subsequently, a specialized program called Convolutional Neural Network was employed to analyze these images. A unique and advanced technique called a political optimizer was utilized to optimize the performance of this program. The results were meticulously compared and analyzed to assess the efficacy of this method in comparison with other existing methods. Finally, the superiority of the suggested method was concluded.

The identification and assessment of brain tumors through MRI images present a formidable and crucial undertaking in the realm of medical imaging. This is due to the potential wealth of information they can offer for treatment planning and prognostic purposes. However, the current automated approaches for detecting brain tumors encounter numerous obstacles and constraints.

The selection of optimization algorithms: Some current techniques rely on conventional optimization algorithms, including gradient descent, genetic algorithm, particle swarm optimization, and others, to optimize the parameters of neural networks or segmentation models. However, these algorithms may encounter issues, such as slow convergence, high computational complexity, or becoming trapped in local optima. Additionally, these algorithms may need to be equipped to handle the intricate and nonlinear patterns that exist in MRI images, necessitating more sophisticated and adaptable optimization techniques.

The techniques for feature extraction: Many of the current methods employ handcrafted features [11], such as Discrete Wavelet Transform (DWT) [12], Grey-Level Co-occurrence Matrix (GLCM) [13], Histogram of Oriented Gradients (HOG), and others, to extract features from MRI images. However, these features may not be capable of capturing the subtle and high-level information that is pertinent for detecting brain tumors, such as the shape, size, location, and texture of the tumor. Furthermore, these features may not be resilient to noise, artifacts, or variations in the MRI images, which could impact the accuracy and dependability of the detection.

The methods for segmentation: Many of the current methods use traditional segmentation methods, such as thresholding, region growing, clustering, and others, to segment the tumor's area from the background. However, these methods may not be able to handle the heterogeneity, irregularity, and ambiguity of the tumor's region, which could lead to over-segmentation or under-segmentation. Additionally, these methods may need to be revised to address the low contrast, low resolution, or partial volume effect of the MRI images, which could affect the precision and quality of the segmentation.

The classification models: The classification models employed in many existing methods for brain tumor classification often rely on conventional models, like Support Vector Machine (SVM), K-Nearest Neighbor (KNN), and Decision Tree (DT) [14]. However, more than these models may be required to handle the complexities associated with high dimensionality, high variability, and high complexity of MRI images. To overcome these limitations, more powerful and flexible classification models are required to effectively learn the hierarchical and nonlinear features essential for accurate brain tumor classification. Failure to consider these factors may impact the performance and generalization of the classification process.

Similarly, the evaluation metrics used in many existing methods for brain tumor detection often involve simple measures, such as accuracy, sensitivity, and specificity. However, these metrics may need to provide a comprehensive reflection of the authentic quality and effectiveness of the detection process. They may fail to account for false positives, false negatives, or the overlap between the ground truth and the predicted region. Additionally, these metrics may not facilitate a fair comparison of the performance of different methods on different datasets. To address these challenges, more comprehensive and standardized evaluation metrics are needed to ensure a thorough assessment of brain tumor detection methods.

To tackle these challenges and limitations, this study proposes a hierarchical automated methodology for detecting brain tumors in MRI images. This methodology involves several steps, starting with preprocessing techniques aimed at enhancing the quality of the MRI images by eliminating noise and artifacts. Techniques such as histogram equalization, median filtering, and image enhancement are employed to achieve this goal.

This research article introduces an innovative methodology that integrates an Extreme Learning Machine (ELM) with the Modified Sailfish Optimizer. This innovative methodology utilizes deep learning and a hierarchical approach to improve the accuracy of brain tumor detection in MRI images. The combination of ELM and the Modified Sailfish optimizer allows for enhanced performance and accuracy in various applications, including the detection of brain tumors. In this paper, an optimized ELM model has been developed that utilizes the Modified Sailfish optimizer to improve the accuracy of brain tumor detection in MRI images. The proposed method has been evaluated on the "Whole Brain Atlas (WBA) database." These contributions demonstrate the potential of deep learning and hierarchical approaches in advancing the field of medical imaging.

3. Proposed methodology

The research paper focuses on the development of an optimal extreme learning machine (ELM) for diagnosing brain tumors using a modified sailfish optimizer (MSO). The research begins with a captivating statement that highlights the prevalence and impact of brain tumors, emphasizing the necessity for accurate and efficient diagnostic methods. It then proceeds to provide background information on current brain tumor diagnosis techniques, such as magnetic resonance imaging (MRI), and their limitations, including noise, artifacts, and low resolution.

Additionally, the section delves into the literature on ELM, an artificial neural network capable of learning from data without the need for weight adjustments in the hidden layer, and MSO, a metaheuristic algorithm that emulates the behavior of sailfish to optimize ELM parameters. The research problem and gap are subsequently established, identifying the absence of a hierarchical and automated methodology that integrates ELM and MSO for brain tumor diagnosis. Furthermore, the research objectives are outlined, which involve proposing a novel method for preprocessing MRI images, optimizing ELM parameters using MSO, and classifying brain tumors using ELM.

The section also emphasizes the study’s novelty and contributions, which aim to enhance the accuracy, recall, specificity, precision, and F1 score of brain tumor diagnosis while reducing the time and complexity of the diagnostic process. Finally, an overview of the paper’s structure is provided, consisting of five sections: introduction, materials and methods, results and discussion, conclusion, and references. The workflow of the process is shown in Fig. (1).

This detailed exploration of the work process flow enhances the clarity surrounding the sequential stages and activities involved in the research, thereby offering readers a comprehensive understanding of the study’s methodology and workflow.

3.1. Dataset description

In this section, the “Whole Brain Atlas (WBA)” database has been utilized for the analysis, which is well recognized as a prominent medical image database used for brain tumor processing [15]. The dataset used T1-weighted MRI images having dimensions of 256×256 pixels. The WBA is a collection of detailed images of the human brain that can be used for research and educational purposes. The WBA serves as an internet-based repository for central nervous system imaging data, meticulously curated by esteemed individuals in the field of neuroscience, namely Keith Johnson, MD, and Alex Becker, Ph.D., affiliated with Harvard Medical School. The Usual Anatomy Atlas presents a navigable brain volume that is visually represented in all three dimensions. As stated on the WBA website, the database comprises a collection of images from 130 subjects. Table 1 indicates the details of the WBA dataset.

Among these subjects, there are 65 individuals without any brain disorders and 65 patients who have been diagnosed with various brain disorders, including cerebrovascular disease, neoplastic disease, degenerative disease, inflammatory or infectious disease, and others. The images have been obtained using different imaging modalities, such as Magnetic Resonance Imaging (MRI) and Computed Tomography (CT), Positron Emission Tomography (PET), and Single-photon emission computed tomography (SPECT). Each image is accompanied by annotations specifying the diagnosis, imaging technique, date of acquisition, and slice plane. Furthermore, the images have been registered across different modalities and time points, enabling precise comparison and analysis. It should be noted that the WBA does not explicitly provide information regarding positive and negative samples, ground truths, or training and testing images, as it is not primarily intended to serve as a dataset for machine learning purposes. However, some researchers have utilized the WBA images to train and test their models for brain tumor detection and classification. Examples of the utilized images for validation are illustrated in Fig. (2).

These images hold significant importance as they play a crucial role in evaluating and assessing the performance and effectiveness of the proposed method or algorithm. The process of validation involves testing the developed model, algorithm, or system using a set of data samples that are distinct from the ones used during training. This helps in ensuring that the results obtained are reliable, accurate, and robust.

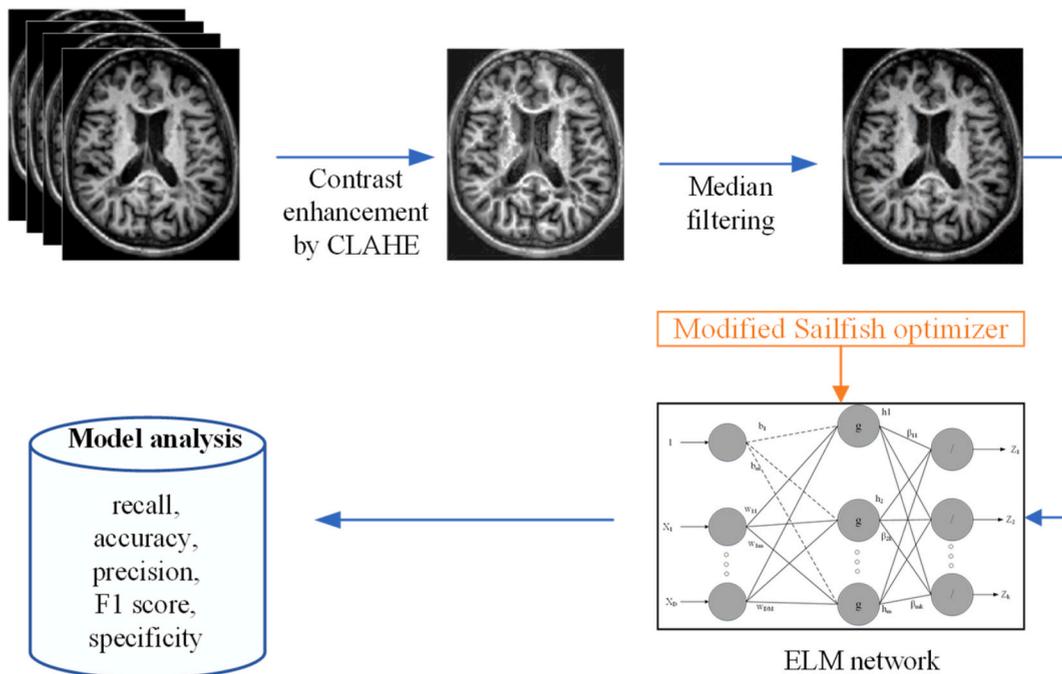


Fig. 1. Workflow of the process.

Table 1
The details of the WBA dataset.

Volume name	Gender	Age	Clinical	Total slices
V ₁	Female	81	Normal	54
V ₂	Female	76	Normal	43
V ₃	Female	51	Anaplastic astrocytoma	56
V ₄	Male	35	Astrocytoma	29
V ₅	Male	62	Metastaticadenocarcinoma	24
V ₆	Female	42	Metastaticbronchogeniccarcinoma	24
V ₇	Male	75	Meningioma	27
V ₈	Male	22	Sarcoma	24

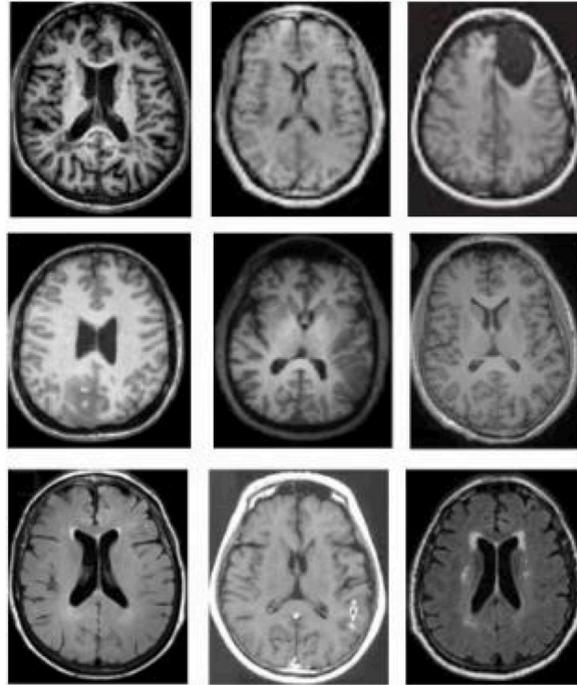


Fig. 2. Examples of the utilized images for validation.

3.2. Preprocessing

3.2.1. Contrast enhancement

Contrast enhancement in MRI medical imaging refers to the process of improving the visibility and differentiation of structures within an MRI image by enhancing the contrast between different tissues or regions of interest [16]. It involves the manipulation of image intensities to highlight specific anatomical or pathological features, making them more distinguishable. Contrast enhancement techniques in MRI play a crucial role in improving the diagnostic capabilities of medical imaging. By enhancing the visibility and differentiation of structures, these techniques aid radiologists and clinicians in accurately identifying and interpreting anatomical structures, abnormalities, and diseases, facilitating more precise diagnoses and treatment planning.

Contrast-Limited Adaptive Histogram Equalization (CLAHE) is a technique commonly used to enhance the contrast of brain MRI images. It is particularly effective in improving the visualization of subtle details and structures that may be obscured or poorly defined due to limited contrast. Here is how CLAHE works in enhancing the contrast of brain MRI images.

- A) Histogram equalization: The first step in CLAHE is applying histogram equalization. Histogram equalization redistributes the intensity values of an image to enhance the overall contrast. However, traditional histogram equalization can lead to excessive amplification of noise and result in unnatural-looking images. The method for establishing Histogram equalization is provided in Eq. (1) [17]:

$$P(i) = H(i)/N \quad (1)$$

where, $H(i)$ describes the intensity values i in the image, and N is the total number of pixels in the image. The cumulative distribution

function (CDF) must compute $C(i)$ as the sum of $P(j)$ for j from 0 to i . Each pixel intensity $I(x, y)$ must be transformed in the image to a new intensity value $I'(x, y)$. This mapping can be formulated as Eq. (2) [17]:

$$I'(x, y) = \text{round}(C(I(x, y)) \times (L - 1)) \quad (2)$$

where, x and y represent the row and column of the image, respectively, $C(i)$ defines the sum of $P(j)$, n L is the number of possible intensity levels.

- B) Adaptive approach: CLAHE overcomes the limitations of traditional histogram equalization by employing an adaptive approach. Instead of equalizing the entire image, CLAHE divides the image into smaller regions or tiles.
- C) Contrast enhancement within tiles: For each tile, the histogram equalization is applied, which stretches the dynamic range of intensity values within that specific region. By doing this locally, the contrast enhancement is tailored to the characteristics of each particular brain region.
- D) Contrast limitation: To prevent the amplification of noise, CLAHE incorporates a contrast limitation mechanism. This adjustment ensures that the enhancement is controlled and the resulting image does not become excessively amplified or over-enhanced. The contrast limitation restricts the equalization process to prevent extreme changes in intensity values and artifacts.
- E) Interpolation between tiles: As the tiles are processed independently, there might be visible boundaries between them. To overcome this issue, interpolation techniques are employed to smooth out the transitions between neighboring tiles, resulting in a visually continuous and coherent final image.

By applying CLAHE to brain MRI images, the technique enhances the contrast and improves the visibility of subtle details. This can be especially beneficial in identifying small lesions, abnormalities, and delicate anatomical structures within the brain. Additionally, CLAHE can help radiologists and clinicians accurately diagnose and assess various brain conditions, including tumors, vascular abnormalities, and neurodegenerative diseases.

It is important to note that while CLAHE can enhance the contrast of brain MRI images, it should be applied judiciously. Excessive contrast enhancement may introduce artifacts or exaggerate existing image noise. Therefore, careful adjustment of parameters, such as the tile size and contrast limitation factor, is necessary to achieve optimal results based on the specific characteristics of the MRI data being processed. Fig. (3) depicts an illustrative example showcasing the enhancement of image contrast for a case of brain tumor, before (A) and after (B) applying CLAHE.

Before applying CLAHE, the original image may suffer from limited contrast, making it challenging to discern subtle details and structures within the brain tumor area. However, after the application of CLAHE, the image in panel (B) exhibits improved visibility and enhanced contrast; this enhancement enables better visualization of the tumor and surrounding tissue, aiding in accurate diagnostics and analysis. CLAHE effectively stretches the dynamic range of intensity values within localized regions of the brain tumor image.

By adapting the contrast enhancement process to specific areas, CLAHE ensures that the enhancement is tailored to the unique characteristics of the brain tumor region. Additionally, the contrast limitation mechanism in CLAHE helps cease noise amplification and over-enhancement, resulting in a more accurate representation of the tumor and its surroundings.

3.2.2. Noise reduction

Noise reduction in brain tumor images refers to the process of reducing unwanted artifacts and disturbances that can degrade the quality and clarity of these medical images [18]. The presence of noise can hinder accurate interpretation and analysis, making it crucial to apply noise reduction techniques for improved visualization and reliable diagnosis. There are some key points to understanding noise reduction in brain tumor images, which will be explained subsequently.

For example, Gaussian noise appears as random variations with a Gaussian distribution that can originate from factors like image

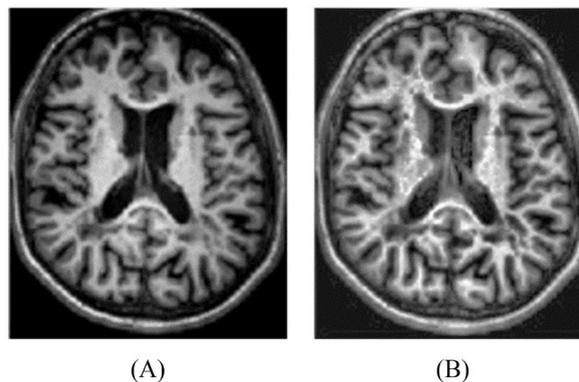


Fig. 3. Illustrative example showcasing the enhancement of image contrast for a case of brain tumor (A) before and (B) after applying CLAHE.

acquisition equipment or environmental interferences [19]. The Salt-and-pepper noise that is manifested as randomly occurring isolated pixels with extreme intensity values, resembling salt-and-pepper sprinkles on an image and can be caused by sensor malfunctions or transmission errors, and Speckle noise that appears as a grainy texture in images, most commonly seen in ultrasound or Magnetic Resonance Imaging (MRI) [20]. It is caused by interference patterns created during the image acquisition process. In this study, median filtering has been utilized to remove the extra noise from the image [21].

Median filters are particularly effective in removing salt-and-pepper noise. These filters replace each pixel with the median value of its neighboring pixels, effectively eliminating isolated noisy pixels. The methodology of median filtering is described below.

- A) Convert the image: The brain tumor's MRI image must be converted into grayscale if it is in color format, as median filtering works on grayscale images.
- B) Determine filter size: The appropriate size of the median filter window MUST BE determined. The window size should be large enough to capture the noise pixels but small enough to preserve essential image details. Here, a square window with a 5×5 dimension is used.
- C) Sliding window operation: The median filter must be applied by sliding the window through the entire image. The window must be placed over each pixel; moreover, the intensity values must be collected from all pixels within the window.
- D) Sort and replace: The collected intensity values must be sorted in ascending order, excluding any noise outliers. The central pixel value must be replaced with the median value from the sorted list. This process effectively eliminates the salt-and-pepper noise by replacing noisy pixels with the median value of their neighborhood.
- E) Repeat for each pixel: The window must be slid over each pixel in the image, and the sorting and replacement process must be repeated to obtain the denoised image. Fig. (4) displays an example of median filtering for noise reduction in an MRI brain scan for (A) noisy image based on Gaussian filter, and (B) its filtered result.

The image denoted as (A) is characterized by the presence of noticeable noise artifacts, such as streaks and blotches, which can pose challenges in interpreting the image. Conversely, the filtered image (B) exhibits a smoother and cleaner appearance, with reduced levels of noise. This improvement is attributed to the implementation of a median filter, which has effectively eliminated many of the noise artifacts, resulting in a visually more appealing image. The median filtering technique operates by replacing each pixel value with the median value derived from its neighboring pixels. This approach capitalizes on the fact that noise typically appears randomly and lacks correlation with the surrounding pixels. By utilizing the median value, the filter can successfully eliminate noise while preserving the underlying signal of the image.

3.3. Adopted method

3.3.1. Extreme learning machine (ELM)

A) Original Extreme Learning Machine

The Extreme Learning Machine (ELM) is classified as a machine learning method that falls under the category of single-hidden layer feedforward neural networks. The proposed method was suggested as a very effective and expedient learning solution for addressing classification and regression challenges. The primary objective of ELM is to provide a streamlined and computationally effective alternative to conventional neural network architectures. In contrast to traditional neural networks that rely on repeated training algorithms, like backpropagation, Extreme Learning Machines (ELM) use a distinct training technique that distinguishes them from other approaches [22].

The fundamental principle behind Extreme Learning Machines (ELM) is the random allocation of weights and biases to the neurons in the hidden layer without the need for repeated adjustments during the training phase [23]. The generation of these arbitrary weights is often accomplished by methods such as uniform or Gaussian distribution. The computation of the output weights is performed

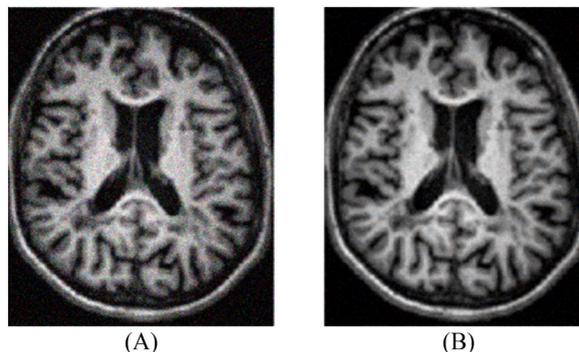


Fig. 4. Example of median filtering for noise reduction in an MRI brain scan: (A) noisy image based on Gaussian filter, and (B) filtered image.

analytically by using the Moore-Penrose inverse [24]. The procedural components included under the ELM framework may be concisely outlined as follows:

The ELM algorithm begins with an input layer that is responsible for receiving the features or properties of the given dataset.

The ELM algorithm incorporates a hidden layer that consists of weights and biases that are randomly allocated. Every individual neuron inside the hidden layer of a neural network applies an activation function to the weighted sum of its respective inputs [25].

The output layer is responsible for generating the ultimate predictions or estimates by using the activation values obtained from the hidden layer.

In the training phase, the Extreme Learning Machine (ELM) algorithm computes the output weights by solving a linear problem via the use of the Moore-Penrose inverse. The use of this phase yields a notable acceleration in the training process when compared to iterative approaches. There are many benefits associated with the use of ELM.

The Extreme Learning Machine (ELM) has a notably accelerated training pace in comparison with conventional neural networks due to its lack of reliance on repeated weight modifications [26].

Impressive Generalization: The Extreme Learning Machine (ELM) often exhibits commendable generalization capabilities, hence demonstrating its efficacy in accurately classifying or predicting previously unknown data.

Streamlined Implementation: The Extreme Learning Machine (ELM) exhibits a relatively straightforward implementation process owing to its streamlined training technique and a reduced number of hyperparameters that need tuning [27]. The Extreme Learning Machine (ELM) has a reduced computational complexity in comparison with conventional neural networks, rendering it well-suited for use in scenarios where computing resources are constrained. Fig. (5) displays the general structure of an original ELM network.

Next, a classification task when the data has D dimensions, and there are N training samples, has been noted as $(x^{(n)}, t^{(n)})$, where n ranges from 1 to N . This task can be mathematically formulated as Eq. (3) [28]:

$$(x^{(n)}, t^{(n)}), n = 1 : N \tag{3}$$

where, $x^{(n)}$ is an element of the vector space R^D and $t^{(n)}$ is an element of the vector space R^K .

The formulation of an ELM-based feed-forward neural network is as follows:

The above equation may be rewritten as the summation of β_m multiplied by the function $g(w_m^T x^{(n)} + b_m)$, where m ranges from 1 to M . This equation is equal to $t^{(n)}$, which is formulated in Eq. (4)

$$\sum_{m=1}^M \beta_m g(w_m^T x^{(n)} + b_m) = t^{(n)} \tag{4}$$

The activation function is denoted by $g(\cdot)$, the bias of the m^{th} hidden neuron is represented by b_m , and M represents the set of hidden neurons. The weight vector $w_m = [w_{m1}, w_{m2}, \dots, w_{mD}]$ is used to identify the connections between the input neurons and the m^{th} neuron of the hidden layer. Similarly, the vector $\beta_m = [\beta_{m1}, \beta_{m2}, \dots, \beta_{mK}]$ is used to calculate the connections between the m^{th} neuron and the other neurons in the hidden layer. This can be mathematically defined by Eq. (5) [29]:

$$H\beta = T \tag{5}$$

where [29],

$$H = \begin{bmatrix} g(w_{11}^T x^{(1)} + b_1) & \dots & g(w_{M1}^T x^{(1)} + b_1) \\ \vdots & \ddots & \vdots \\ g(w_{1N}^T x^{(N)} + b_1) & \dots & g(w_{MN}^T x^{(N)} + b_M) \end{bmatrix}_{N \times M} \tag{6}$$

$$H = [\beta_1^T, \beta_2^T, \dots, \beta_M^T]_{M \times N}^T; T = [t_1^T, t_2^T, \dots, t_M^T]_{N \times K}^T \tag{7}$$

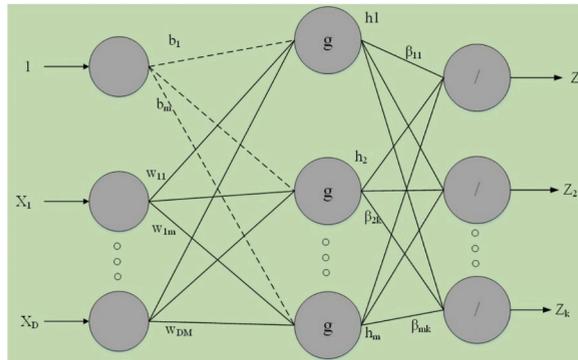


Fig. 5. General structure of an original ELM network.

In contrast, the symbol H is used to denote a matrix that is not square. The number of training data exceeds the number of hidden neurons. Hence, to address this issue, Eq. (8) has been employed.

$$\hat{\beta} = H^{\dagger} T \tag{8}$$

The symbol H^{\dagger} represents the generalized Moore/Penrose matrix inversion. Therefore, the ELM networks may be limited to three primary stages:

Therefore, the ELM networks may be limited to three primary stages: The input weights and bias are initialized with random values. The evaluation of the matrix H , which represents the outputs of the hidden neurons, is conducted.

Extreme Learning Machines (ELM) have some advantageous attributes; nevertheless, they also exhibit some limits that the use of metaheuristics may effectively mitigate. Overfitting is a prevalent concern in Extreme Learning Machine (ELM) models, whereby the model exhibits excessive complexity and closely conforms to the training data. To tackle this issue, it is possible to use metaheuristic methods, like genetic algorithms or particle swarm optimization, to optimize hyperparameters, including but not limited to the number of hidden neurons or regularization parameters. These techniques have the potential to assist in identifying the most suitable hyperparameter configuration, hence mitigating overfitting and enhancing generalization.

The random weight initialization used by ELM may lead to suboptimal solutions; however, this issue may be addressed by expanding the solution space and identifying more optimum weight values. Metaheuristics may be used to facilitate the identification of optimum hyperparameter values utilizing automated searches over the parameter space. The utilization of various approaches, such as grid search, evolutionary algorithms, or Bayesian optimization can achieve this.

The presence of imbalanced data is a significant difficulty for Extreme Learning Machine (ELM) models since it results in one class being substantially underrepresented in comparison with the others. Metaheuristic methodologies, such as cost-sensitive learning or specialized sampling procedures, like Synthetic Minority Oversampling Technique (SMOTE), may be used to address the issue of imbalanced class distribution. These approaches aim to enhance the model's capacity to identify minority samples accurately.

The process of optimizing landscapes in Extreme Learning Machine (ELM) training may provide challenges, particularly in areas with a large number of dimensions. Metaheuristic is a valuable tool for effectively navigating complex optimization environments and identifying superior solutions. The integration of metaheuristics with ELM offers a promising approach to mitigate these constraints and augment the efficacy and resilience of ELM models.

B) Optimizing ELM network based on Algorithm

The categorization of the ultimate retrieved characteristics is carried out using the suggested optimal Extreme Learning Machine (ELM) network [30]. The primary objective of the ELM classifier in this context is to classify brain tumor MRI scans into two distinct categories: normal and tumorous classes.

This research examines the activation functions of the network intending to improve their performance. The impact of the activation function on the Evolving Neural Networks (ENNs) is often seen. A network that incorporates several activation functions has enhanced generalization capabilities [31]. This research examines the sigmoid activation function, which is well-recognized as a prominent function within this field. The formulation of this function is elucidated in the following section. The function $g(X)$ can be mathematically achieved by Eq. (9).

$$g(X) = \frac{1}{1 + \exp(- (as.x + bs))} \tag{9}$$

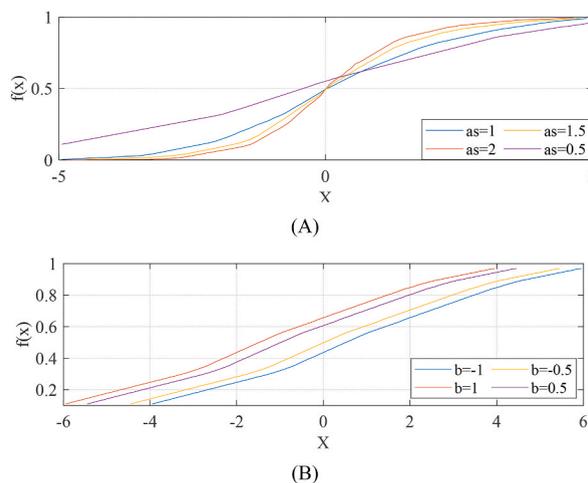


Fig. 6. Influence of variable “a” (A) and variable “b” (B) on the form of the function.

In alternative terms, distinct values are used for the variables denoted as “ a ” and “ b ”. To enhance the effectiveness of the Extreme Learning Machine (ELM) classifier, several sigmoid functions have been used inside the hidden neurons. Fig. (6) illustrates the influence of variable “ a ” (A) and variable “ b ” (B) on the form of the function

Furthermore, the task of determining the appropriate parameter for the sigmoid function may be framed as an optimization issue. In this study, a new modified metaheuristic algorithm, called modified Sailfish optimizer has been used to address the problem at hand.

The impact of the parameters bs and the bias b is almost equivalent. Consequently, both as and bs are optimized, with the bias being used for the optimization process. Consequently, two optimization strategies have been used [32].

The starting values for the input weights have been established. Subsequently, the modified Sailfish optimizer, which has been created, is used to choose optimum parameters by taking into account the starting weights. The objective function of the modified Sailfish optimizer is formulated by calculating the square error between the intended output and the output generated by the network, as expressed in Eq. (10):

$$E = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^k (d_{ji} - y_{ji})^2 \quad (10)$$

In this context, the variables “ n ” and “ k ” are used to denote the number of training samples and the output layers, respectively. Additionally, “ y_{ji} ” and “ d_{ji} ” are used to represent the output of the network and the target value, respectively.

The main objective of this research is to use the improved ELM network discussed before to identify brain tumors. In the following, the proposed designed modified Sailfish optimizer has been explained in detail.

C) Modified Sailfish optimizer

Sailfish Optimizer integrates sardines in a prey team and the action of a predatory sailfish team. Since the sailfish works in a team to pursue and hunt the prey, it is famous as a social predator. Predators use a variety of strategies to hunt the prey in a team [33]. The responsibility of attacking and hunting sardines is on the leader of the group; however, others in the team preserve their energy. It might alter the position of a sailfish that assaults the school of prey [34]. In addition, the hunter changes its position to upgrade it toward the sardines to have a better hunt and chase them. To prevent from being hunted, first, the group gets closer and swarmed, and then the location of the sardine is upgraded. The following phases illustrate how the Sailfish optimizer has been conducted.

D) Population initialization

In this phase, the population of sardines and sailfish are initialized arbitrarily. a_k^r denotes the random location, and b_l^r denotes the random location of sardines. $k \in$, $l \in$, and $r \in$ represents Sailfish individuals, Sardines, and Iteration number, respectively. A possible solution for the r^{th} concern of iteration is determined by the location of sailfish or sardines, which are represented by a_k^r and b_l^r , respectively.

E) Analysis and strategy of elitism

The fitness feature $Fit(\cdot)$ ascertains each search agent’s location, i.e. sardine or sail-fish. The finest sailfish, which owns the lowest amount of fitness has been conserved as the elite sailfish in the population of sailfish as shown by a_{elite}^r ; furthermore, the elite is involved in the sailfish set. It means that $Fit(a_{elite}^r)$ is equal to or less than $Fit(b_l^r)$ for each r in the issue of attenuation. b_{injure}^r illustrates the injured sardine. It means that $injure$ is involved in the Sardine set. In addition, for each r , the most popotentardine is found to have the worst Sardine population health, where $Fit(b_{injure}^r)$ is equal to or less than $Fit(b_l^r)$. Preserving and saving injured sardines means that a strategy of elitism is being conducted. The reason is that they do not ruin proper responses by updating the positions of search agents.

F) Upgrading sailfish location

Each sailfish individual can alter its position on each side of the prey school during the iteration. When there is an attack strategy of any opportunity to occupy any free space around the prey of the school, the alterations in the sailfish behavior or location happen. Any alteration in the state of sailfish can be determined by the injured sardine and the position of elite sailfish, which has been illustrated in Eq. (11) [35]:

$$a_k^{r+1} = a_{elite}^r - \lambda_r \times \left(\beta \times \left(\frac{a_{elite}^r + y_{injure}^r}{2} - a_k^r \right) \right) \quad (11)$$

The novel location of sailfish of $(r + 1)^{th}$ iteration is illustrated by a_k^{r+1} , random count between 0 and 1 is illustrated by r and β , the current location of elite sailfish is illustrated by a_{elite}^r , the current location of sailfish is illustrated by a_k^r , the location of the present injured sardine is illustrated by y_{injure}^r , and the coefficient designed for every r^{th} iteration is illustrated by λ_r using Eq. (12) [34].

$$\lambda_r = (2 \times \beta \times Dens) - Dens \quad (12)$$

Here, the randomly made count between 0 and 1 is shown by β , and the scale target density has been demonstrated by $Dens$. Finally, the number of attacks on the school of prey can be produced, and the sailfish hurt and eat the determined sardines. The variable $Dens$ is determined by Eq. (13) [34].

$$Dens = 1 - \left(\frac{N_{sail}}{N_{sard} + N_{sail}} \right) \quad (13)$$

Here, N_{sard} and N_{sail} imply the number of sardines and the sailfish, respectively. Initially, the community of sardines is three times greater than the community of the sailfish, i.e. $N_{sard} = N_{sail} \times 3$. Because of various λ_r fluctuation values, every sailfish can update the location and move in different ways; as a result, λ_r fluctuation values are vital factors within the proposed algorithm.

G) Updating the sardine's location

The stalking initialization is the right time for sardines to flee worthy benefits. In the first stage, the sailfish merely hurt the sardines, but they miss them. The capability of sailfish to escape from the sardines decreases, as time passes. While the sardines damage the sailfish, they endeavor to adopt strategies of attack. In the last step of fighting the targets, the opportunities to flee from threats decrease. So, the improvement rate of sailfish learning increases. The actions performed by sardines against attacks of sailfish must be considered. Eq. (14) determines the alteration of each sardine in the suggested algorithm.

$$b_l^{(r+1)} = x \times (a_{elite}^r - a_l^r + P_{attack}) \quad (14)$$

The position of novel sardine l has been represented by b_l^{r+1} , the location of the present sardine l has been represented by b_l^r , a random number between 0 and 1 has been represented by x , the fines location of the elite sailfish has been represented by a_{elite}^r , and the number of attacks by sailfish in each iteration has been represented by P_{attack} using Eq. (15).

$$P_{attack} = D \times (1 - (2 \times Iter \times \epsilon)) \quad (15)$$

The position of sailfish k at the iteration of r is represented by a_k^r , and the position of sardine l at the iteration of r is represented by b_l^r . The next step for the sardine l is that it should be split up with the other sardines and replacement is the subsequent step. In the first stage of attack, the success rate of hunting is low since most of the sardines alter their location and escape from the attack. Nevertheless, the ability of sardines to flee declines toward the end of their fishing; thereby, the capture achievement rate increases. Therefore, the sardines' count is reduced over time. If P_{attack} is lesser than 0.5, then the last hunting stage is considered. The quantity of sardines that upgrade their role in the final state of hunting is dependent on the power of attack ($P_{attack} < 0.5$) which is shown in Eq. (16).

$$\alpha = N_{sard} \times P_{attack} \quad (16)$$

$$\beta = d^r \times P_{attack} \quad (17)$$

Where, d^r is the number of variables at ith iteration. In the following, it will be explained that ϵ And D are the two significant elements. They decrease the number of the present iterations $Iter$ and the power of attack in the present iteration (P_{attack}). The first stage is to initialize the population of sardines and sailfish at random. It is essential to say that $D = 4$ and $\epsilon = 0.001$. Sardine's fitness determines the criterion based on which the sailfish is created. The finest sardine and sailfish as well as injured sardine and elite sailfish should be chosen. If the criterion of termination is not met, λ_r should be created and upgraded by the use of equations (2) and (1), respectively.

It should be noted that if $P_{attack} < 0.5$, α should be created by the use of Eq. (16). A set of sardines depending on the value α must be chosen, and the use of the equation must develop the position of the sardines; otherwise, the position of every sardine must be upgraded by the use of Eq. (17). In the end, if a better solution exists in the population of sardines, the following steps should be taken: the sailfish should be substituted with the wounded sardine; the killed sardine should be eliminated from the population; and the finest sardine and the finer sailfish must be improved.

The sardines can merely alter their location by the use of the number that has been chosen when $P_{attack} < 0.5$; however, the whole population of sardines can be improved if significant $ack > 0.5$. Once the sardine is killed by a sailfish k , the location of the sardine l has been substituted in the suggested algorithm. So, it can be concluded that $Fit(b_l^r) < Fit(a_k^r)$; as a result, $a_k^r = b_l^r$.

H) Modified Sailfish Optimizer (MSO)

The original Sailfish Optimizer algorithm encounters stability issues primarily caused by the velocity perturbation of the current solutions. Additionally, it tends to converge prematurely due to an inadequate balance between exploitation and exploration terms. To address these limitations, this study proposes a modification to the algorithm. In this modified version, a Cauchy mutation mechanism is employed. This iterative process aims to enhance the local exploration's capability within the Sailfish Optimizer, thus increasing the likelihood of trapping into promising local optima. By applying the Cauchy mutation mechanism to individuals within the Sailfish Optimizer population, the algorithm becomes more effective in searching for the global optimal solution.

The Cauchy mutation mechanism utilizes a one-dimensional density function defined below, which helps guide the algorithm towards a higher probability of finding the global optimal solution, as defined by Eq. (18).

$$f(x; x^0, \gamma) = \left(\frac{1}{\pi\gamma}\right) \times \left(\frac{\gamma^2}{(x - x^0)^2 + \gamma^2}\right) \quad (18)$$

where, x represents the mutated parameter, while x^0 and γ denote the location and scale parameters, respectively. The Cauchy distribution aids in introducing diversity and balance within the Sailfish Optimizer, mitigating the stability issues and premature convergence observed in the original algorithm.

By utilizing the equation mentioned above, the current optimal solution for the Sailfish Optimizer can be obtained by Eq. (19).

$$a_k^{r+1} = a_{elite}^r - \lambda_r \times \left(f(x; x^0, \gamma) \times \left(\frac{a_{elite}^r + y_{injure}^r}{2}\right) - a_k^r\right) \quad (19)$$

where, the parameters were explained before.

D) Algorithm validation

In this section, the efficiency of the proposed MSO algorithm is examined by applying it to four different test functions: Rosenbrock function, Sphere function, Rastrigin function, and Ackley function. The objective is to provide a comprehensive analysis of the algorithm's performance and compare it with various existing metaheuristics, thus highlighting the key strengths of the MSO approach. To facilitate a more precise analysis, it is essential to define the benchmark functions used in the evaluation. The following section elaborates on these four benchmark functions, providing a detailed description of each. By employing these well-established functions, the performance and effectiveness of the MSO algorithm can be thoroughly assessed and compared against other optimization techniques.

- Ackley function:

The Ackley function is a commonly used multimodal function that presents challenges for optimization algorithms due to its myriad local optima. Furthermore, it is defined in Eq. (20).

$$f(x) = -20 \times \exp\left(-0.2 \times \text{sqrt}\left(\frac{1}{n} \times \text{sum}(xi^2)\right)\right) - \exp\left(\frac{1}{n} \times \text{sum}(\cos(2\pi xi))\right) + 20 + e \quad (20)$$

where, $x = (x_1, x_2, \dots, x_n)$ represents the vector of decision variables, and n is the dimensionality of the problem. The Ackley function has a global minimum at $f(x) = 0$, located at $x_i = 0$ for all i .

- Sphere function:

The Sphere function is a well-known unimodal function often used to evaluate optimization algorithms. It is defined in Eq. (21).

$$f(x) = \text{sum}(x_i^2) \quad (21)$$

- Rosenbrock function:

The Rosenbrock function, also known as the "banana function," is a popular non-convex function that poses challenges for optimization algorithms due to its narrow valley. It is defined in Eq. (22).

$$f(x) = \text{sum}\left(100 \times (x_i + 1 - x_i^2)^2 + (x_i - 1)^2\right) \quad (22)$$

where, $x = (x_1, x_2, \dots, x_n)$ represents the vector of decision variables. The Rosenbrock function has a global minimum at $f(x) = 0$, located at $x_i = 0$ for all i .

- Rastrigin function:

The Rastrigin function is a multimodal function widely used to assess the performance of optimization algorithms in solving complex problems. It is defined in Eq. (23).

$$f(x) = \text{sum}(xi^2 - 10 \times \cos(2\pi xi) + 10) \quad (23)$$

where, $x = (x_1, x_2, \dots, x_n)$ represents the vector of decision variables. The Rastrigin function has a global minimum at $f(x) = 0$, located at $x_i = 0$ for all i .

By evaluating the performance of the MSO algorithm on these benchmark functions, we can gain insights into its efficiency and compare it with other metaheuristics in terms of convergence speed and solution quality. All benchmark functions used in the analysis

have been set to a dimension of 50. Fig. (7) displays a two-dimensional plot illustrating the characteristics of the test functions, including: (A) Ackley function, (B) Sphere function, (C) Rosenbrock function, and (D) Rastrigin function.

The algorithm's validation involves a comparison with several recently developed and successful optimization algorithms, namely the Grey Wolf Optimizer (GWO), Whale Optimization Algorithm (WOA), Salp Swarm Algorithm (SSA), and Moth Flame Optimization (MFO). To ensure reliable results and facilitate a fair comparison between the proposed algorithm (IMA) and the others, each algorithm is independently executed 50 times.

The final results of these algorithms on the test benchmarks are presented in Table 1, providing a comprehensive overview of their performance. The Grey Wolf Optimizer (GWO) is implemented with a population size of $N = 30$, and a maximum number of iterations set to $T = 100$. This metaheuristic algorithm draws inspiration from the social hierarchy and hunting behavior of grey wolves to perform optimization tasks efficiently. The Whale Optimization Algorithm (WOA) utilizes a population size of $N = 30$ and runs for $T = 100$ iterations. Additionally, it incorporates a parameter $b = 1$, which controls the spiral-shaped hunting behavior of whales within the algorithm.

The cooperative hunting strategies of humpback whales inspire WOA. The Salp Swarm Algorithm (SSA) is executed with a population size of $N = 30$ and a maximum number of iterations set to $T = 100$. This metaheuristic algorithm imitates the collective movement and feeding behavior of salps, a type of marine organism, to solve optimization problems effectively. The Moth Flame Optimization (MFO) algorithm uses a population size of $N = 30$ and runs for $T = 100$ iterations. Similar to the WOA, it also incorporates a parameter $b = 1$ to control the magnitude of attraction among moths. The attraction of moths inspires MFO to a flame and their navigation towards it. The outcomes of the algorithms on the test benchmarks are presented in Table 2, indicating their final results.

Upon examining the data presented in Tables 2 and it becomes evident that the proposed MSO algorithm consistently achieves the minimum values for the functions. The test benchmarks have been visually presented in graphical form to enhance comprehension of the algorithm outcomes. Refer to Fig. (8) for a visual representation of the results.

The utilization of visualization to showcase the outcomes of the benchmarking tests offers a lucid and instinctive method for comprehending the algorithm results. It enables a rapid assessment of the performance of the algorithms in various optimization tasks, accentuates the merits and demerits of each algorithm, and reinforces the deductions made from the benchmarking tests.

This noteworthy observation stems from calculating the mean value across a total of 50 independent runs, providing strong evidence for the higher accuracy exhibited by the proposed method. Furthermore, it is essential to highlight the importance of the minimum standard deviation obtained through the implementation of the suggested MSO algorithm.

The performance of MSO is also evaluated by comparing it to the state-of-the-art CEC 2019 benchmark test suite, which consists of ten functions. Professor Suganthan and his team have enhanced the evaluation capabilities of these functions. To assess the performance of different algorithms, an annual tournament is conducted using tests derived from the 100-Digit Challenge. The CEC04-CEC10 functions were designed as 10-dimensional minimization problems with a boundary range $[-100, 100]$ by the creator of the CEC benchmark. On the other hand, the size of the CEC01-CEC03 functions varied. Consequently, the CEC04-CEC10 test functions are subjected to rotation and shifting, while the CEC01-CEC03 test functions need to be more balanced. Scaling is an available option for all the tests. To ensure quality assurance, modifications have been made only for the findings of the CEC01 to account for the additional 1000 dimensions applied to a subset of the functions. Each algorithm undergoes 200 iterations and involves 50 agents. Therefore, this study compares the performance of MSO against other optimization algorithms, such as Grey Wolf Optimizer (GWO), Whale Optimization Algorithm (WOA), Salp Swarm Algorithm (SSA), Moth Flame Optimization (MFO), original sailfish optimizer (SOA) [34], Electric eel foraging optimizer (EEFO) [36], Geyser Inspired Algorithm (GIA) [37], Lotus effect optimization algorithm (LEOA) [38], and Pufferfish Optimization Algorithm (POA) [39] to the specified CEC functions. The outcomes of the algorithms on the CEC2019 benchmark are presented in Table 3, indicating their final results.

We also analyzed 10 benchmark functions of the CEC 2021 for more analysis of the proposed algorithm. The CEC 2021 competition on evolutionary computation introduced a set of benchmark functions to assess the effectiveness of meta-heuristic algorithms in terms of convergence rate, precision, and ability to navigate away from local optima. These benchmark functions were designed with specific characteristics. They were parametrized using operators like bias, shift, and rotation to simulate real-world optimization challenges. Furthermore, they were categorized into different groups to evaluate specific algorithmic capabilities, ranging from basic search to advanced features. The evaluation process involved metrics such as error rate, convergence speed, and the ability to find the global optimum in a multimodal landscape. Additionally, the algorithms were classified into primary, advanced, and competition-specific categories for comparative analysis. The experimental results provided valuable insights, observations, and recommendations on

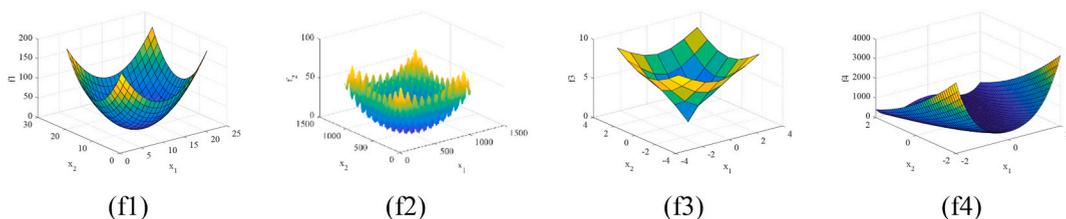


Fig. 7. Two-dimensional plot illustrating the characteristics of the test functions employed in this study: (A) Ackley function, (B) Sphere function, (C) Rosenbrock function, and (D) Rastrigin function.

Table 2

The outcomes of the algorithms on the test benchmarks.

Algorithm		f_1	f_2	f_3	f_4
GWO	Min	2.84	4.99	2.99	2.72
	Max	13.92	14.96	12.34	19.04
	Mean	11.40	12.93	10.51	16.61
	std	1.83	2.03	0.13	1.57
WOA	Min	27.47	45.57	1.40	38.41
	Max	34.37	51.42	3.06	45.07
	Mean	11.43	35.86	2.84	25.49
	std	0.50	33.00	0.95	21.76
SSA	Min	20.43	17.38	11.26	9.63
	Max	32.12	26.68	17.59	17.42
	Mean	23.41	16.85	14.41	12.11
	std	10.10	4.23	3.62	4.83
MFO	Min	1.55	3.63	1.55	0.17
	Max	2.43	5.56	1.91	5.31
	Mean	1.71	3.09	1.60	3.10
	std	0.9	2.17	1.05	0.89
MSO	Min	0.00	1.16	0.14	1.61
	Max	0.04	2.60	1.71	3.12
	Mean	0.02	1.79	0.87	2.45
	std	0.00	1.11	0.86	1.09

the performance of various meta-heuristic algorithms [Table 4](#).

The evaluation of these algorithms was conducted using ten functions from CEC2019 and CEC2021, each presenting unique challenges. While the CEC01-CEC03 test functions remained unaltered, the CEC04-CEC10 functions underwent rotation, shifting, and scaling. Additionally, modifications were made to the findings of the CEC01 to account for the additional 1000 dimensions applied to a subset of the functions. The outcomes of the evaluation indicate that the performance of the algorithms varies across the different test functions.

The remarkably low standard deviation value signifies the robustness and stability of the proposed method when compared to other algorithms utilized in the study. This characteristic of the MSO algorithm enhances its reliability and implies its ability to converge toward optimal solutions across various optimization problems consistently. The superiority of the proposed MSO algorithm, as demonstrated by achieving the minimum function values and showcasing low standard deviations, provides compelling indications of its competence and efficacy in handling complex optimization tasks. These findings contribute to the growing body of evidence supporting the effectiveness and practicality of the MSO algorithm within the field of optimization research.

4. Results and analysis

To conduct simulations and evaluate the desired network, a random selection of 25 % of the data is allocated for testing purposes, while the remaining 75 % is used for training. To initiate network training, the weights of the pre-trained network are saved. The network utilized in this study is a modified version of the Elman Neural Networks network. The initial weights of the proposed network are set to be the initial weights of the Modified Elman Neural Networks. The weights of all connected layers are, then, optimized using the modified sailfish optimizer to diagnose brain tumors.

The featured promoted group search process is a metaheuristic with stochastic properties. Its outcomes can vary across different implementations. In this study, the preparation technique of the promoted group search process was iterated 25 times, and the most efficient network was selected for further investigation.

The promoted group search algorithm and Modified ENN were implemented using MATLAB 2019b/64-bit edition on a computing environment consisting of a laptop equipped with a Core i7-11800H processor. This processor is a quad-core with a base clock speed of 2.3 GHz and a turbo boost speed of 4.6 GHz. The GPU is an NVIDIA GeForce RTX 3060, known for its high performance in computationally intensive tasks like gaming and video editing. The capacity of RAM is 16 GB DDR4, which is sufficient for a wide range of tasks. The storage capacity is 512 GB SSD, known for its speed and reliability. The computer system consists of a 2.00 GHz CPU, 32 GB RAM, a 2.5 GHz Intel Core i7 processor, and two SLI GeForce Titan GPUs. It operates on a 64-bit operating system.

This section analyzes the efficiency of the suggested method by examining its performance on the benchmark dataset. The proposed method utilizes a pipeline approach for diagnosing brain tumors using MRI scans. The initial step involved suggesting preprocessing methods to improve image contrast and reduce noise. The technique employs an ENN for diagnostic purposes. The proposed methodology has been evaluated based on five criteria. These criteria are given from Eq. (24) to Eq. (28).

$$Recall = \frac{TP}{TP + FN} \quad (24)$$

$$Precision = \frac{TP}{TP + FP} \quad (25)$$

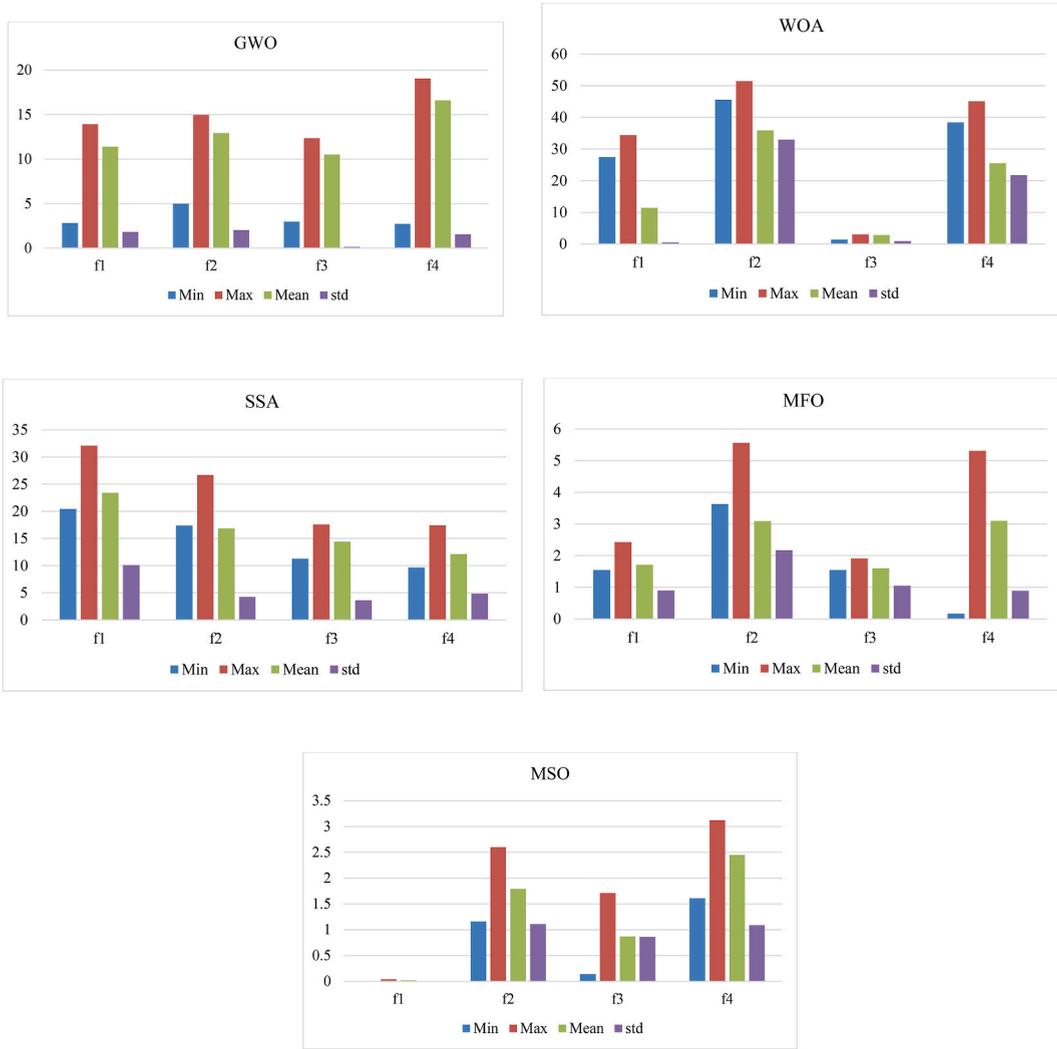


Fig. 8. The test benchmarks have been visually presented in graphical form to enhance comprehension of the algorithm outcomes.

$$Specificity = \frac{TN}{TN + FP} \tag{26}$$

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \tag{27}$$

$$F1score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{28}$$

where, FP represents the number of samples classified as false positive, TP represents the number of samples classified as true positive, TN represents the number of samples classified as true negative, and FN represents the number of samples classified as false negative.

Based on the analysis of a dataset and a test model, it has been determined that TP, a numerical value associated with a brain tumor, is correctly identified by the model as a brain tumor. FP is a negative number that is represented as positive in the context of a brain tumor. TN represents a negative value, indicating a normal condition. Conversely, FN represents a positive value, indicating the presence of a brain tumor, but is misclassified by the model as negative (normal).

This study utilized MRI images for the prediction of brain tumors in patients. The study compared the results of the work with established models, including the End-to-end predictive intelligence diagnosis (End/End) [4], Convolutional Neural Network (CNN) [2], and Residual Network (RN) [40], to showcase its superiority. For analyzing the proposed brain tumor image diagnosis, the following metrics are commonly used: recall (RE), accuracy (AC), precision (PR), F1 score (F1), and specificity (SC). Table 5 presents the diagnosis performance outcomes obtained from multiple methods.

Table 3
The outcomes of the algorithms on the CEC2019 benchmark.

	MSO		MFO		SSA		WOA		GWO	
	AVG	StD								
CEC01	99974.85	31626.47	1.74E+10	1.2E+10	4238.144	19881.87	5.4E+10	6.08E+10	3.83E+10	5.21E+10
CEC02	17.23695	0.242359	15.8875	0	3.710874	3.03E-09	74.66346	84.54328	16.8697	0.004269
CEC03	12.03438	0.000251	11.51526	1.24E-08	12.4649	1.58E-11	12.42197	0.000679	13.68423	0
CEC04	13349.47	3845.775	93.25612	99.14985	33.85746	15.70179	316.3714	392.8159	382.6936	246.3921
CEC05	4.725438	0.544564	1.234897	0.183443	1.958767	0.083626	2.557504	0.316287	2.521342	0.278179
CEC06	11.29926	0.546568	4.34707	2.156603	11.95851	0.564948	9.854368	1.564501	10.54203	0.994838
CEC07	1159.008	240.0674	426.4344	246.615	110.9649	13.09433	524.3656	302.1166	478.3554	184.5608
CEC08	6.526325	0.301132	5.285794	0.39185	6.095314	0.714948	6.269574	0.489541	6.639554	0.404787
CEC09	1934.528	452.9575	2.563069	0.311327	1.805986	1.56E-10	5.631419	2.652897	5.751865	1.599173
CEC10	20.22654	0.112511	19.8583	0.129145	2.68146	8.73E-16	19.94918	0.156252	20.58168	0.104556
	SOA		EEFO		GIA		LEOA		POA	
	AVG	StD								
CEC01	1.72E+10	9.71E+09	4.13E+03	1.63E+04	5.32E+10	5.05E+10	3.69E+10	4.30E+10	3.33E+10	4.51E+10
CEC02	1.53E+01	0.00E+00	3.30E+00	2.67E-09	7.23E-01	7.57E+01	1.53E+01	4.07E-03	1.47E+01	3.50E-03
CEC03	9.45E+00	9.94E-09	1.09E+01	1.28E-11	1.07E+01	6.27E-04	1.36E+01	0.00E+00	1.35E+01	0.00E+00
CEC04	8.62E+01	9.37E+01	3.36E+01	1.49E+01	2.90E+02	3.92E+02	3.70E+02	2.09E+02	3.67E+02	2.40E+02
CEC05	1.13E+00	1.50E-01	1.60E+00	7.34E-02	2.11E+00	2.82E-01	2.21E+00	2.58E-01	2.19E+00	2.66E-01
CEC06	3.63E+00	2.08E+00	1.18E+01	4.94E-01	8.93E+00	1.44E+00	9.99E+00	9.64E-01	1.02E+01	9.93E-01
CEC07	4.25E+02	2.05E+02	9.85E+01	1.21E+01	4.91E+02	2.74E+02	4.54E+02	1.50E+02	4.42E+02	1.83E+02
CEC08	4.72E+00	3.70E-01	5.31E+00	7.10E-01	5.94E+00	4.54E-01	5.87E+00	3.91E-01	5.79E+00	3.25E-01
CEC09	2.09E+00	2.53E-01	1.76E+00	1.35E-10	5.36E+00	2.38E+00	4.83E+00	1.31E-00	5.51E+00	1.37E+00
CEC10	1.75E+01	1.22E-01	2.46E+00	8.17E-16	1.88E+01	1.31E-01	1.77E+01	8.79E-02	1.88E+01	9.35E-02

Table 4
The outcomes of the algorithms on the CEC2021 benchmark.

	MSO		MFO		SSA		WOA		GWO	
	AVG	StD	AVG	StD	AVG	StD	AVG	StD	AVG	StD
CEC01	0	0	2.60E-01	3.63E-01	5.72E-02	3.02E-01	3.33E-01	3.84E-01	2.26E+00	3.31E+00
CEC02	0	0	3.00E-01	3.88E-01	6.52E-02	2.71E-01	2.04E-01	1.41E-01	1.04E+00	8.65E-01
CEC03	0	0	1.92E-02	2.78E-01	1.53E-01	2.74E-01	2.16E-01	9.21E-02	1.12E-01	1.98E+00
CEC04	0	0	1.88E-01	2.62E-01	3.77E-01	2.22E-01	2.22E-01	1.90E-01	1.22E+00	1.79E+00
CEC05	2.53E-38	2.53E-40	2.27E-01	1.27E-01	1.23E-01	2.82E-01	9.45E-02	3.02E-01	2.78E+00	1.15E+00
CEC06	1.24E-08	2.49E-10	7.90E-02	5.75E-02	2.18E-01	7.75E-02	3.73E-01	2.22E-01	7.86E-01	2.93E-01
CEC07	1.24E-08	2.49E-10	6.00E-02	1.75E-01	3.32E-01	2.32E-01	2.64E-01	6.79E-02	7.09E-01	1.15E+00
CEC08	0	0	1.97E-01	3.43E-01	2.81E-01	9.08E-03	1.90E-01	3.61E-01	2.26E+00	7.19E-01
CEC09	6.4592E-327	0	2.03E-01	2.68E-01	1.89E-01	1.09E-01	9.42E-03	2.69E-01	1.57E+00	1.58E+00
CEC10	5.27E-04	1.29E-06	3.91E-01	3.68E-02	2.32E-02	1.45E-01	3.76E-01	3.01E-01	3.88E+00	4.46E-01
	SOA		EEFO		GIA		LEOA		POA	
	AVG	StD	AVG	StD	AVG	StD	AVG	StD	AVG	StD
CEC01	1.12E-01	1.63E-01	1.70E-01	1.75E-01	3.85E-01	2.69E-01	1.36E+00	1.28E+00	1.14E+00	1.27E+00
CEC02	2.46E-02	2.28E-01	3.28E-01	3.89E-01	3.46E-01	7.70E-02	4.31E-01	1.97E+00	8.38E-01	1.15E+00
CEC03	2.98E-02	1.62E-01	1.16E-01	1.21E-01	2.52E-01	1.68E-01	1.77E-01	1.96E+00	9.67E-01	8.55E-01
CEC04	2.34E-01	3.57E-01	2.10E-01	1.14E-01	2.62E-01	2.82E-01	7.38E-01	1.86E+00	1.79E+00	4.46E-01
CEC05	8.78E-02	2.25E-01	3.57E-01	3.70E-01	3.72E-01	8.36E-02	1.10E+00	9.78E-01	3.93E+00	2.95E+00
CEC06	4.90E-02	3.02E-01	3.96E-01	2.41E-01	3.11E-01	4.30E-02	4.38E-01	2.50E+00	2.06E+00	2.75E+00
CEC07	1.07E-01	1.07E-01	4.54E-02	2.09E-01	5.03E-02	2.68E-01	7.05E-01	1.02E+00	5.14E-01	1.20E+00
CEC08	2.42E-01	3.58E-01	3.76E-01	2.15E-01	1.19E-01	1.61E-01	2.23E+00	3.37E+00	3.72E+00	1.37E+00
CEC09	3.26E-01	1.04E-01	9.37E-02	2.01E-01	7.04E-02	1.64E-01	1.18E+00	7.34E-01	1.23E+00	2.08E+00
CEC10	3.09E-01	1.65E-01	1.36E-01	1.58E-01	1.18E-01	3.61E-01	3.13E+00	1.73E+00	8.91E-01	9.71E-01

According to the table, the proposed method achieved the highest accuracy (AC) of 93.95 %. This suggests that the proposed method has a higher ability to classify and diagnose cases compared to the other methods correctly. End/End and CNN also achieved high AC values of 89.24 % and 93.17 %, respectively, while RN had a lower AC value of 84.09 %. In terms of recall (RE), which measures the ability to identify positive cases correctly, the proposed method again obtained a perfect score of 100 %. End/End and CNN achieved relatively high RE values at 91.38 % and 87.33 %, respectively, while RN had a perfect RE score as well. Specificity (SC) measures the ability to identify negative cases correctly. The proposed method achieved a relatively high SC value of 92.07 %, followed by End/End at 86.53 %, CNN at 87.19 %, and RN at 75.64 %.

This indicates that the proposed method can better identify negative cases than the other methods correctly. Precision (PR), which

Table 5
The diagnosis performance outcomes obtained from various methods.

Model Indicator	Proposed method	End/End [4]	CNN [2]	RN [40]
AC.	93.95	89.24	93.17	84.09
RE.	100	91.38	87.33	100
SC.	92.07	86.53	87.19	75.64
PR.	88.43	87.37	90.41	69.25
F1	97.16	90.09	90.40	82.98

represents the proportion of correctly identified positive instances, was the highest for the proposed method at 88.43 %. End/End and CNN also achieved high PR values at 87.37 % and 90.41 %, respectively, while RN had a lower PR value of 69.25 %. Finally, the F1 score, which combined precision and recall, was the highest for the proposed method at 97.16 %. End/End, CNN, and RN achieved F1 scores of 90.09 %, 90.40 %, and 82.98 % respectively. Based on these results, it can be concluded that the proposed method shows overall superior performance in terms of accuracy, recall, specificity, precision, and F1 score compared to the other methods evaluated in the table. However, it is essential to consider other factors, such as computational complexity, dataset characteristics, and generalizability, before making a final judgment about the effectiveness of the proposed method in medical imaging diagnosis.

Despite the positive outcomes yielded by the proposed approach, it is imperative to acknowledge its limitations, which will necessitate further attention in future research. Firstly, the reliance on a solitary, hidden layer feedforward neural network within the method may hinder its ability to effectively capture the intricate characteristics and patterns exhibited by brain tumor images. Secondly, the utilization of a fixed number of hidden nodes in the method could impact the model's performance and its capacity for generalization. Thirdly, the method's evaluation is conducted on a relatively small dataset of MRI images, thereby potentially restricting its applicability and robustness when dealing with other types of brain tumor images and datasets. Fourthly, the method overlooks the spatial information and context of the brain tumor images, which could be crucial for precise diagnosis. Lastly, the method needs to include the provision of any explanation or interpretation of the diagnosis results, which may undermine the trust and confidence of both users and healthcare professionals.

5. Conclusion and future scope

This study presented a new automated technique for diagnosing brain tumors using Magnetic Resonance Imaging (MRI). The method involved a modified Extreme Learning Machine (ELM), which was a single-layer feedforward neural network that could quickly and efficiently learn features from MRI images. The ELM was combined with a newly modified version of a metaheuristic algorithm called Modified Sailfish Optimizer, which was a swarm intelligence algorithm inspired by the hunting behavior of sailfish. The Modified Sailfish optimizer enhanced the feature extraction and optimization process of the ELM, resulting in faster convergence and improved performance. The method also employed a thresholding technique based on the features extracted by the ELM to segment the tumor region from the background and a Deep Neural Network (DNN) to classify the tumor type or grade from the segmented region. The proposed method was evaluated using various metrics, such as recall, accuracy, precision, F1 score, and specificity. The experiments were conducted on the Whole Brain Atlas (WBA) database, which contained a wide range of annotated MRI images of brain tumors. The results demonstrated that the proposed method achieved superior efficiency and accuracy in detecting brain tumors from MRI images, indicating its potential to enhance accuracy and efficiency. The proposed method outperformed other methods in terms of accuracy, recall, specificity, precision, and F1 score in medical imaging diagnosis. It achieved the highest accuracy of 93.95 %, with End/End and CNN attaining high values of 89.24 % and 93.17 %, respectively. The method also achieved a perfect score of 100 % in recall, 91.38 % in specificity, and 75.64 % in F1 score. However, it was crucial to consider factors like computational complexity, dataset characteristics, and generalizability before evaluating the effectiveness of the method in medical imaging diagnosis. The proposed approach for brain tumor detection offered several advantages over existing methods. It used the Modified Sailfish optimizer to improve convergence speed, ELM, and DNN to learn features from MRI images. However, the method faced challenges, such as reliance on MRI images, sensitivity to initial parameters, and difficulty in interpreting features. These factors could impact the method's performance, robustness, generalization, and sensitivity to initial parameters and randomness. However, we acknowledge certain challenges such as computational complexity and sensitivity to initial parameters. Moving forward, our future endeavors will focus on refining optimization algorithms, feature extraction, segmentation, and classification models to further enhance performance and robustness, particularly in addressing the intricacies involved in MRI image analysis. Future research could explore advanced optimization algorithms, feature extraction techniques, accurate segmentation methods, powerful classification models, and standardized evaluation metrics to improve ELM and DNN performance and robustness in brain tumor detection. These methods can also handle heterogeneity, irregularity, and ambiguity in tumor regions while addressing the high dimensionality, variability, and complexity of MRI images.

Data availability statement

The dataset utilized in this study has been sourced from a publicly available repository. The name of the repository is Human Health Campus, which is an online platform for health professionals, especially in the field of nuclear medicine and radiation oncology. The

accession number of the data is Whole Brain Atlas (WBA), which is a collection of images and information about the human brain, including normal anatomy, pathology, and functional imaging. The data is available at [https://humanhealth.iaea.org/HHW/Technologists/NuclearMedicineTech/Educationalresources/Atlas/Whole_Brain_Atlas/index.html], where you can browse and download the images and information.

CRedit authorship contribution statement

Saad Ali Amin: Conceptualization. **Mashal Kasem Sulieman Alqudah:** Data curation. **Saleh Ateeq Almutairi:** Investigation, Conceptualization. **Rasha Almajed:** Software, Project administration. **Mohammad Rustom Al Nasar:** Writing – original draft, Visualization, Project administration. **Hamzah Ali Alkhazaleh:** Methodology, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] T. Liu, Z. Yuan, L. Wu, B. Badami, Optimal brain tumor diagnosis based on deep learning and balanced sparrow search algorithm, *Int. J. Imag. Syst. Technol.* 31 (2021) 1921–1935, <https://doi.org/10.1002/ima.22559>.
- [2] H. Alsaif, et al., A novel data augmentation-based brain tumor detection using convolutional neural network, *Appl. Sci.* 12 (2022) 3773, <https://doi.org/10.3390/app12083773>.
- [3] A. Hu, N. Razmjoo, Brain tumor diagnosis based on metaheuristics and deep learning, *Int. J. Imag. Syst. Technol.* 31 (2021) 657–669, <https://doi.org/10.1002/ima.22495>.
- [4] L. Ma, F. Zhang, End-to-end predictive intelligence diagnosis in brain tumor using lightweight neural network, *Appl. Soft Comput.* 111 (2021) 107666, <https://doi.org/10.1016/j.asoc.2021.107666>.
- [5] Z. Zhu, et al., Brain tumor segmentation based on the fusion of deep semantics and edge information in multimodal MRI, *Inf. Fusion* 91 (2023) 376–387, <https://doi.org/10.1016/j.inffus.2022.10.022>.
- [6] N. Bacanin, T. Bezdán, K. Venkatachalam, F. Al-Turjman, Optimized convolutional neural network by firefly algorithm for magnetic resonance image classification of glioma brain tumor grade, *Journal of Real-Time Image Processing* 18 (2021) 1085–1098, <https://doi.org/10.1007/s11554-021-01106-x>.
- [7] D. Deb, S. Roy, Brain tumor detection based on hybrid deep neural network in MRI by adaptive squirrel search optimization, *Multimed. Tool. Appl.* 80 (2021) 2621–2645, <https://doi.org/10.1007/s11042-020-09810-9>.
- [8] T. Liu, Z. Yuan, L. Wu, B. Badami, An optimal brain tumor detection by convolutional neural network and enhanced sparrow search algorithm, *Proc. IME H J. Eng. Med.* 235 (2021) 459–469, <https://doi.org/10.1177/0954411920987964>.
- [9] S. Maqsood, R. Damaševičius, R. Maskeliūnas, Multi-modal brain tumor detection using deep neural network and multiclass SVM, *Medicina* 58 (2022) 1090, <https://doi.org/10.3390/medicina58081090>.
- [10] A.H. Bashkandi, et al., Combination of political optimizer, particle swarm optimizer, and convolutional neural network for brain tumor detection, *Biomed. Signal Process Control* 81 (2023) 104434, <https://doi.org/10.1016/j.bspc.2022.104434>.
- [11] I. Mecheher, M. Abbod, A. Amira, H. Zaidi, Deep learning with multiresolution handcrafted features for brain MRI segmentation, *Artif. Intell. Med.* 131 (2022) 102365, <https://doi.org/10.1016/j.artmed.2022.102365>.
- [12] A. Susanto, C.A. Sari, H. Rahmalan, M.A. Doheir, Support vector machine based discrete wavelet transform for magnetic resonance imaging brain tumor classification, *TELKOMNIKA (Telecommunication Computing Electronics and Control)* 21 (2023) 592–599, [0.12928/telkomnika.v21i3.24928](https://doi.org/10.12928/telkomnika.v21i3.24928).
- [13] L. Hussain, et al., Bayesian dynamic profiling and optimization of important ranked energy from gray level co-occurrence (GLCM) features for empirical analysis of brain MRI, *Sci. Rep.* 12 (2022) 15389, <https://doi.org/10.1038/s41598-022-19563-0>.
- [14] Mushtaq, S. & Saini, K. S. A review on predicting brain stroke using machine learning, in 2023 10th International Conference on Computing for Sustainable Global Development (INDIACom). 667–673 (IEEE), ISBN:978-93-80544-47-2.
- [15] E.D. Vidoni, The whole brain Atlas, *J. Neurol. Phys. Ther.* 36 (2012) 108, <https://doi.org/10.1136/jnnp.74.3.288>. www.med.harvard.edu/aanlib.
- [16] Y. Chen, J. Yang, Perioperative fast-track surgery nursing intervention for patients with kidney stone disease under computed tomography imaging, *Contrast Media Mol. Imaging* 2023 (2023), <https://doi.org/10.1155/2023/1101388>.
- [17] K.G. Dhal, A. Das, S. Ray, J. Gálvez, S. Das, Histogram equalization variants as optimization problems: a review, *Arch. Comput. Methods Eng.* 28 (2021) 1471–1496, <https://doi.org/10.1007/s11831-020-09425-1>.
- [18] E. Chodakowska, J. Nazarko, L. Nazarko, Arima models in electrical load forecasting and their robustness to noise, *Energies* 14 (2021) 7952, <https://doi.org/10.3390/en14237952>.
- [19] N. Razmjoo, V.V. Estrela, H.J. Loschi, Entropy-based breast cancer detection in digital mammograms using world cup optimization algorithm, in: *Research Anthology on Medical Informatics in Breast and Cervical Cancer*, IGI Global, 2023, pp. 645–665, <https://doi.org/10.4018/978-1-6684-7136-4>.
- [20] N. Razmjoo, A survey on parameters estimation of the proton exchange membrane fuel cells based on the swarm-inspired optimization algorithms, *Front. Energy Res.* 11 (2023), <https://doi.org/10.3389/fenrg.2023.1148323>.
- [21] N. Razmjoo, A. Arshaghi, Application of multilevel thresholding and CNN for the diagnosis of skin cancer utilizing a multi-agent fuzzy buzzard algorithm, *Biomed. Signal Process Control* 84 (2023) 104984, <https://doi.org/10.1016/j.bspc.2023.104984>.
- [22] Y. Liu, L. Liu, L. Yang, L. Hao, Y. Bao, Measuring distance using ultra-wideband radio technology enhanced by extreme gradient boosting decision tree (XGBoost), *Autom. Construct.* 126 (2021) 103678, <https://doi.org/10.1016/j.autcon.2021.103678>.
- [23] Y. Liu, Y. Bao, Intelligent monitoring of spatially-distributed cracks using distributed fiber optic sensors assisted by deep learning, *Measurement* 220 (2023) 113418, <https://doi.org/10.1016/j.measurement.2023.113418>.
- [24] N.H. Aljahdaly, A. Akgül, R. Shah, I. Mahariq, J. Kafle, A comparative analysis of the fractional-order coupled Korteweg–De Vries equations with the Mittag–Leffler law, *J. Math.* 2022 (2022) 1–30, <https://doi.org/10.1155/2022/8876149>.
- [25] N. Kose, U. Kayapinar, S. Erkiric, The effect of tablet use on EFL reading achievement, *Eurasian Journal of Applied Linguistics* 7 (2021) 58–72.
- [26] S. Alabed, I. Mahariq, M. Salman, M. Kuzuoglu, A Novel Beamforming Emulating Photonic Nanojets for Wireless Relay Networks, 2021, <https://doi.org/10.32604/cmc.2021.018245>.
- [27] A. Jaradat, O. Ali, A. AlAhmad, Blockchain technology: a fundamental overview. *Blockchain Technologies for Sustainability*, 2022, pp. 1–24, https://doi.org/10.1007/978-981-16-6301-7_1.
- [28] J. Cao, K. Zhang, M. Luo, C. Yin, X. Lai, Extreme learning machine and adaptive sparse representation for image classification, *Neural Network.* 81 (2016) 91–102, <https://doi.org/10.1016/j.neunet.2016.06.001>.
- [29] L. Yang, X. Fang, X. Wang, S. Li, J. Zhu, Risk prediction of coal and gas outburst in deep coal mines based on the SAPSO-ELM algorithm, *Int. J. Environ. Res. Publ. Health* 19 (2022) 12382, <https://doi.org/10.3390/ijerph191912382>.

- [30] J. Wang, S. Lu, S.-H. Wang, Y.-D. Zhang, A review on extreme learning machine, *Multimed. Tool. Appl.* 81 (2022) 41611–41660, <https://doi.org/10.1007/s11042-021-11007-7>.
- [31] D. Yu, Y. Wang, H. Liu, K. Jermstittiparsert, N. Razmjoooy, System identification of PEM fuel cells using an improved Elman neural network and a new hybrid optimization algorithm, *Energy Rep.* 5 (2019) 1365–1374, <https://doi.org/10.1016/j.egy.2019.09.039>.
- [32] W. Dang, et al., A semi-supervised extreme learning machine algorithm based on the new weighted kernel for machine smell, *Appl. Sci.* 12 (2022) 9213, <https://doi.org/10.3390/app12189213>.
- [33] Y. Zhang, Y. Mo, Chaotic adaptive sailfish optimizer with genetic characteristics for global optimization, *J. Supercomput.* 78 (2022) 10950–10996, <https://doi.org/10.1007/s11227-021-04255-9>.
- [34] S. Shadravan, H.R. Naji, V.K. Bardsiri, The Sailfish Optimizer: a novel nature-inspired metaheuristic algorithm for solving constrained engineering optimization problems, *Eng. Appl. Artif. Intell.* 80 (2019) 20–34, <https://doi.org/10.1016/j.engappai.2019.01.001>.
- [35] Y. Zhang, Y. Mo, Dynamic optimization of chemical processes based on modified sailfish optimizer combined with an equal division method, *Processes* 9 (2021) 1806, <https://doi.org/10.3390/pr9101806>.
- [36] W. Zhao, et al., Electric eel foraging optimization: a new bio-inspired optimizer for engineering applications, *Expert Syst. Appl.* 238 (2024) 122200, <https://doi.org/10.1016/j.eswa.2023.122200>.
- [37] M. Ghasemi, et al., Geysir inspired algorithm: a new geological-inspired meta-heuristic for real-parameter and constrained engineering optimization, *JBE* 21 (2024) 374–408, <https://doi.org/10.1007/s42235-023-00437-8>.
- [38] E. Dalirinia, M. Jalali, M. Yaghoobi, H. Tabatabaee, Lotus effect optimization algorithm (LEA): a lotus nature-inspired algorithm for engineering design optimization, *J. Supercomput.* 80 (2024) 761–799, <https://doi.org/10.1007/s11227-023-05513-8>.
- [39] O. Al-Baik, et al., Pufferfish optimization algorithm: a new bio-inspired metaheuristic algorithm for solving optimization problems, *Biomimetics* 9 (2024) 65, <https://doi.org/10.3390/biomimetics9020065>.
- [40] S.A.A. Ismael, A. Mohammed, H. Hefny, An enhanced deep learning approach for brain cancer MRI images classification using residual networks, *Artif. Intell. Med.* 102 (2020) 101779, <https://doi.org/10.1016/j.artmed.2019.101779>.