Multimedia Tools and Applications (2023) 82:44623–44653
https://doi.org/10.1007/s11042-023-15239-7

Accurate detection of brain tumor using optimized feature selection based on deep learning techniques

Praveen Kumar Ramtekkar[1] · Anjana Pandey[1] · Mahesh Kumar Pawar[1]

Received: 6 September 2021 / Revised: 12 March 2023 / Accepted: 30 March 2023 /
Published online: 25 April 2023

Abstract

An unusual increase of nerves inside the brain, which disturbs the actual working of the brain, is called a brain tumor. It has led to the death of lots of lives. To save people from this disease timely detection and the right cure is the need of time. Finding of tumor-affected cells in the human brain is a cumbersome and time- consuming task. However, the accuracy and time required to detect brain tumors is a big challenge in the arena of image processing. This research paper proposes a novel, accurate and optimized system to detect brain tumors. The system follows the activities like, preprocessing, segmentation, feature extraction, optimization and detection. For preprocessing system uses a compound filter, which is a composition of Gaussian, mean and median filters. Threshold and histogram techniques are applied for image segmentation. Grey level co-occurrence matrix (GLCM) is used for feature extraction. The optimized convolution neural network (CNN) technique is applied here that uses whale optimization and grey wolf optimization for best feature selection. Detection of brain tumors is achieved through CNN classifier. This system compares its performance with another modern technique of optimization by using accuracy, precision and recall parameters and claims the supremacy of this work. This system is implemented in the Python programming language. The brain tumor detection accuracy of this optimized system has been measured at 98.9%.

Keywords Brain tumor · CNN · GLCM · Histogram segmentation · Particle swarm optimization (PSO) · Whale optimization algorithm (WOA) · Gray wolf optimization (GWO)

✉ Praveen Kumar Ramtekkar
pramtekkar@rediffmail.com

Anjana Pandey
apeeshukla77@gmail.com

Mahesh Kumar Pawar
mkpawar24@gmail.com

[1] University Institute of Technology, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, Madhya Pradesh, India

1 Introduction

The brain is a complex organ which controls every process that regulates our body. Together, the brain and spinal cord that extends from it make up the central nervous system. The cells of brain grow unusually and the mechanism which controls normal cells is not able to stabilize progress of brain's cells, which causes brain tumor. Tumor lodges space in skull, which disturbs usual working of brain and makes stress on brain. Because of higher stress on brain, several brain tissues are moved, pushed against skull and becomes liable for harm of nerves of other healthy brain's tissues [16]. Brain tumors are mainly classified as benign and malignant tumors. As per the investigation of the World Health Organization (WHO), 120 types of brain tumors are reported based on cell origin and cell behavior from less aggressive to more aggressive. Brain tumors are mainly classified as benign and malignant tumors. Benign tumors grow inside the brain and are non-cancerous and malignant or cancerous.

Today, the involvement of Information Technology (IT) and e-health care techniques in the medical arena support physicians to deliver superior quality health services to the diseased. Tumor in brain disturbs the human brain critically, due to unusual increasing of cells in the brain. It disturbs the regular working of the brain and might be fatal for life. A brain tumor is typical cancer in human beings, which is directly related to death. Therefore, timely detection of this disease is the need of time and plays a significant part in dropping the death rate.

The exact method to understand the brain tumor and its stages is a main job to prevent and to carry out the steps in curing the disease. For this magnetic resonance imaging (MRI) is widely used by physicians to analyze brain tumors. The result of the analysis carried out in this paper reveals whether the brain is normal one or tumor infected by applying the deep learning techniques. Several techniques of medical imaging such as Ultra Sound, Single-Photon Emission Computerized Tomography (SPECT), Computed Tomography (CT) Scan, X-Rays, Positron Emission Tomography (PET) and Magnetic Resonance Imaging (MRI) have been applied for the identification of brain tumors [5]. In relation to another imaging process, MRI is commonly used since it provides better contrast of brain MRI images and malignant tissues. Hence, identification of brain tumors is frequently done via MRI images [3]. Analyzing the MRI scan image is the modern technique to detect the presence of a brain tumor at its early stage. Preprocessing also performs an important role in identifying the tumor in brain cancer detection. Conversion of segmented MRI image into a group of features are known as feature extraction. Pre-processing is applied to this system to reduce the noise and artifact from the image. Next, histogram and threshold techniques are used to segment the tumor area which is located on a non-uniform basis. GLCM is used for feature extraction after the segmentation. Optimization is the process of obtaining the best results in minimum effort. A WOA and GWO algorithms are proposed for the selection of the best image features from the features obtained by GLCM.

In this research paper, the detection of brain tumor is done by using deep learning techniques. When these techniques are applied on the MRI images, the detection of brain tumor is done very fast and a higher accuracy helps in providing the treatment to the patients. These prediction also assists the physicians in making quick decisions. In this proposed system CNN is applied with whale optimization in detecting the presence of brain tumor. In CNN the dimension of the image is reduced at every CNN layer without the loss of information needed for training. Different processing task like zero padding, convolution, batch normalization, ReLU, max pooling, flatten and dense are applied for creating the CNN model. Finally the performance of this system is compared with other systems that uses deep learning and optimization techniques for brain tumor identification. This work justifies that system based on deep neural network give promising results in detection accuracy of brain tumor via image processing techniques. Optimization techniques have been applied with deep neural network models to set optimized values for model parameters like accuracy, precision, recall and F-Score. This combination of the use of optimization techniques with deep neural network enhances the performance of the system by converging it faster. These optimization algorithms proves their efficacy in tumor detection. Therefore, to take the advantage of optimization algorithm, this system uses the whale optimization technique with CNN to improve the brain tumor detection accuracy and processing speed.

This research work is divided into seven sections. Section 1, Introduction, briefs this proposed research work. Section 2, Literature Review, describes the existing work related to this proposed work. Section 3, Methods and Materials, includes details of MR images, datasets, and proposed architecture of brain tumor detection, WOA, GWO algorithm and CNN. Section 4, Results and discussion, describes the results obtained by using PSO, genetic algorithm (GA), GWO and WOA algorithms along with CNN classifier based on accuracy, precision, recall and F-score parameters. Section 4, Performance Comparison, compares the results based on the accuracy of proposed and exiting work. At last, conclusion has been drawn.

2 Literature survey

Tumor detection in an MRI brain image in the field of medical science is a challenging and hard task. Researchers are involved in this field to acquire the best results and techniques to identify tumors. Today deep learning-based optimization techniques contribute excellent results and the practice of using CNN is increasing day by day.

Geetha A, et al. (2019) [7] proposes a new accurate brain tumor detection model. The model includes certain processes such as preprocessing, segmentation, feature extraction and classification. Initially, two main processes contrast enhancement and skull stripping have been processed. The fuzzy C-means clustering (FCM) algorithm has been used for the segmentation. GLCM and grey-level run-length matrix (GRLM) features were extracted in the feature extraction phase. Moreover, this system uses a deep belief network (DBN) for classification. The optimized DBN concept is used here, for which GWO is used. The proposed model is termed as GW-DBN model. This proposed model compares its performance over other conventional methods in terms of accuracy, specificity, sensitivity, precision, negative predictive value (NPV), F1-Score and Matthews correlation coefficient (MCC), false negative rate (FNR), false positive rate (FPR) and false discovery rate (FDR). The results obtained by this model show that the accuracy of GW-DBN was 0.39%, 67.32%, 15.83% and 7.56% better than conventional DBN, naïve Bayesian (NB), support vector machine (SVM) and neural network (NN), respectively. The sensitivity of the proposed model was 18.51%, 42.22% and 1.58% superior to NB, SVM and NN, respectively. FNR of GW-DBN was 55.55% and 70.37% better than NB and SVM models. The F1-Score of the proposed model was 49.01%, 22.35% and 7.56% better than NB, SVM and NN, respectively. Thus, the performance of proposed model for detecting brain tumor was proven over other methods.

Sindhu A, et al. (2020) [26] presents a system which applies five steps, i.e., preprocessing, segmentation, feature extraction, feature selection (optimization) and classification. This system performs preprocessing as an initial step to remove the noises. The segmentation has been used to identify the tumor location. The features extracted from segmented images consist of irrelevant features, which reduces the classification accuracy in disease identification. Therefore, efficient feature selection method was introduced in this system to improve the classification performance. For the improvement of results obtained by feature selection, initially, image segmentation is carried out by using saliency-based k-means clustering segmentation and then feature extraction is done by using First Order and Second Order Statistical features by GLCM and GLRM. Further PSO and whale optimization techniques are applied for best feature selection. The results obtained by these methods indicate the potential advantages of using feature selection techniques to improve the classification accuracy with a smaller number of feature subset. From the result, it is concluded that the performance of whale optimization is superior to PSO method for classification. The machine learning classifiers such as DT, KNN, SVM and AdaBoost with ensemble KNN-SVM classifier are utilized to classify the tumor as normal or abnormal. The proposed framework achieves a classification accuracy of 98.3%.

Rammurthy D, et al. (2020) [20] describes an optimization based technique, viz. Whale Harris Hawks optimization (WHHO) for brain tumor detection using MRI. In this system segmentation is performed using cellular automata and rough set theory and the image features are extracted from the segments, which include tumor size, Local Optical Oriented Pattern (LOOP), Mean, Variance and Kurtosis. Further, the brain tumor detection is carried out using deep convolution neural network (DCNN), wherein the training is performed using proposed WHHO which has been designed by integrating Whale optimization algorithm (WOA) and Harris hawks optimization (HHO) algorithm. The WHHO-based DCNN performed well with accuracy of 0.816, specificity of 0.791 and sensitivity of 0.974, respectively.

Ramtekkar PK, et al. (2020) [21] designs a model for the development of an automatic system that works on brain MRI images to detect and classify tumors using CNN classifier and deep learning techniques. This proposed model includes series of stages namely preprocessing, segmentation, feature extraction, tumor detection, tumor classification and tumor location for detecting and classifying brain tumor. Here, CNN classifier has been combined with GLCM for the extraction of tumor region. The model also proposes the use of Partial Differential Equation (PDE) for image clustering, K-means and Otsu thresholding for image segmentation and multi histogram equalization for image enhancement. This paper reviews various techniques of image classification and concludes their advantages and disadvantages. The performance of this proposed model has been expected well than other manual and semi-automatic systems that uses SVM and Artificial Neural Network (ANN) classifiers. Finally, the model also illustrates that DNN is preferred for brain tumor detection and classification since it provides higher accuracy than modern techniques.

Hossain T, et al. (2020) [11] proposes a system to distinguish normal and abnormal pixels, based on texture based and statistical features of an MRI. This system also suggests a method to extract brain tumor from 2D MRI by FCM algorithm which was followed by traditional classifiers and CNN. The experimental work of this system has been carried out on a real-time dataset with diverse tumor sizes, locations, shapes and different image intensities. Initially, the system uses traditional classifiers namely SVM, K-Nearest Neighbor (KNN), Multilayer Perceptron (MLP), Logistic Regression, NB and Random Forest to classify the tumor and this work was implemented in scikit-learn. Further, the CNN has been used to classify the tumor, which was implemented in Keras and Tensorflow. This system with CNN yields better performance than the traditional classifier. The use of CNN gained an accuracy of 97.87%, which is quite compelling.

Mishra PK, et al. (2021) [18] recommends a model which observes that earlier DCNN models do not consider the weights as of learning instances which may decrease accuracy levels of the segmentation procedure. This paper suggested the framework for optimizing the network parameters such as weight and bias vector of DCNN models using swarm intelligent based algorithms like Genetic Algorithm (GA), PSO, GWO and WOA. The simulation results of this model reveals that the WOA optimized DCNN segmentation model is outperformed than other three optimization based DCNN models i.e., GA-DCNN, PSO-DCNN, GWO-DCNN. DCNN classifier with whale optimization for the detection of brain tumors with an accuracy of 98%. This paper compares the accuracy obtained by PSO, GWO, WAO and GA.

Irmak E (2021) [12] creates a system to make multi classification of brain tumors for the initial diagnosis purposes using CNN. For this three different CNN models are proposed for three different classification of tumor. The first CNN model achieves 99.33% accuracy. Second model classifies the brain tumor into five types as normal, glioma, meningioma, pituitary and metastatic with an accuracy of 92.66%. The third model classifies the tumors into three grades as Grade II, Grade III and Grade IV with an accuracy of 98.14%. The important hyper parameters of CNN models are automatically designated using the grid search optimization algorithm. This CNN models are compared with other popular state-of-the-art CNN models such as AlexNet, Inceptionv3, ResNet-50, VGG-16 and GoogleNet. Satisfactory classification results are obtained using large and publicly available clinical datasets. The proposed CNN models can be employed to assist physicians and radiologists in validating their initial screening for brain tumor multi-classification purposes.

Sharma AK et al. (2022) [25] develops the DL architectures such as recurrent networks and CNN (ConvNets), which are proven appropriate for non-handcrafted extraction of complex features in skin image. To additional expand the efficiency of the ConvNet models, a cascaded ensemble network that uses an integration of ConvNet and handcrafted features based multi-layer perceptron has proposed in this paper. This model utilizes the CNN model to mine non-handcrafted image features, color moments and texture features as handcrafted features. It is demonstrated that accuracy of ensemble DL model is improved to 98.3% from 85.3% of CNN model.

Verma SS et al. (2021) [27] The Coronavirus Disease 2019 (COVID-19) outbreak has a devastating impact on health and the economy globally, that's why it is critical to diagnose positive cases rapidly. Currently, the most effective test to detect COVID-19 is Reverse Transcription-polymerase chain reaction (RT-PCR) which is time-consuming, expensive and sometimes not accurate. It is found in many studies that, radiology seems promising by extracting features from X-rays. COVID-19 motivates the researchers to undergo the deep learning process to detect the COVID- 19 patient rapidly. This paper has classified the X-rays images into COVID- 19 and normal by using multi-model classification process. This multi-model classification incorporates Support Vector Machine (SVM) in the last layer of VGG16 Convolution network. For synchronization among VGG16 and SVM we have added one more layer of convolution, pool, and dense between VGG16 and SVM. Further, for transformations and discovering the best result, we have used the Radial Basis function. CovXmlc is compared with five existing models using different parameters and metrics. The result shows that our proposed CovXmlc with minimal dataset reached accuracy up to 95% which is significantly higher than the existing ones. Similarly, it also performs better on other metrics such as recall, precision and F-score.

Khan AH, et al. (2022) [14] Observed that brain tumors may be one of the serious of psychiatric complications like, depression and panic attacks. Early detection of a brain tumor is more effective in tumor healing. The medical image processing plays an important role in assisting humans in identifying different diseases. Classification of brain tumors is a significant part that depends on the expertise and knowledge of the physician. An intelligent system for detecting and classifying brain tumors is essential to help physicians. The novel feature of the study is the division of brain tumors into glioma, meningioma, and pituitary using a hierarchical deep learning method. The diagnosis and tumor classification are significant for the quick and productive cure and medical image processing using a CNN is giving excellent outcomes in this capacity. CNN uses the image fragments to train the data and classify them into tumor types. Hierarchical Deep Learning-Based Brain Tumor (HDLBT) classification is proposed with the help of CNN for the detection and classification of brain tumors. The proposed system categorizes the tumor into four types: glioma, meningioma, pituitary and no-tumor and achieves 92.13% precision and miss rate of 7.87%.

Gunasekara SR et al. (2021) [9] States that the main requirements of tumor extraction are the annotation and segmentation of tumor boundaries correctly. To fulfill this purpose, the paper presents a threefold DL architecture. First, the classifiers are implemented with a DCNN and second a region-based CNN (R-CNN) is performed on the classified images to localize the tumor regions of interest. At third stage, the concentrated tumor boundary is contoured for the segmentation by using the Chan-Vese segmentation algorithm. As the typical edge detection algorithms based on gradients of pixel intensity tend to fail in the medical image segmentation process, an active contour algorithm defined with the level set function is proposed. Specifically, the Chan-Vese algorithm was applied to detect the tumor boundaries for the segmentation process. To evaluate the performance of the overall system, Dice Score, Rand Index (RI), Variation of Information (VOI), Global Consistency Error (GCE), Boundary Displacement Error (BDE), Mean Absolute Error (MAE), and Peak Signal to Noise Ratio (PSNR) were calculated by comparing the segmented boundary area which is the final output of the proposed, against the demarcations of the subject specialists which is the gold standard. Overall performance of the proposed architecture for both glioma and meningioma segmentation is with an average Dice Score of 0.92 (also, with RI of 0.9936, VOI of 0.0301, GCE of 0.004, BDE of 2.099, PSNR of 77.076, and MAE of 52.946), pointing to the high reliability of the proposed architecture.

Avsar E (2019) [2] observed that the MRI is a useful method for the diagnosis of tumors in human brain. Under this system, MRI images have been analyzed to detect the regions containing tumor and classify these regions into three different tumor categories like, meningioma, glioma and pituitary. DL is a relatively recent and powerful method for image classification. Therefore, this system has used faster Region-based CNN, which is based on DL techniques. The proposed system was implemented using TensorFlow library. A publicly available dataset containing 3064 MRI brain images (708 meningioma, 1426 glioma, 930 pituitary) of 233 patients has been used to train and test classifier. This system yields an accuracy of 91.66% which is higher than other system using the same dataset.

Ramtekkar PK, et al. (2023) [22] proposes novel, precise and optimized system to detect brain tumors. This system includes preprocessing, segmentation, feature extraction, optimization and detection steps. The compound filter has been designed by applying Gaussian, mean and median filters for preprocessing. Threshold and histogram techniques have been applied for image segmentation. GLCM has used for feature extraction. The optimized CNN was applied along with ant colony optimization (ACO), bee colony optimization (BCO), PSO, GA, GWO and WOA for best feature selection. Detection of brain tumors is achieved through CNN classifiers. Performance of this system has been compared using accuracy, precision and recall parameters. The accuracy of this optimized system was measured at 98.9%.

Based on the literature survey presented above, the following drawbacks are observed by the previous methodologies used:

- The basic difficulties of the previous methods were binary categorization of tumors which creates further ambiguity for the physicians. Due to a lack of data, physicians fail to get reliable results.
- It is observed that tumor detection using brain MRI is a difficult task due to two reasons. The first, brain tumors exhibit a great degree of variability in size, severity and form. The second, tumors come in various pathological varieties, showing the same symptoms. These difficulties are not covered by the previous methods.
- The former methods which applied deep learning had improved tumor detection accuracy but they needed a large amount of training data for analysis. However, the computational cost and training time associated with brain tumor detection were substantial.
- In the earlier methods segmentation and detection of tumor areas from brain MRIs was complicated and time consuming for analysis. However, the accuracy is significantly improved for the identification of brain tumors.
- The prior methods which used genetic algorithms needed less data to detect brain tumors, but it may be challenging to eliminate the objective function. Genetic algorithms are time consuming and non-deterministic. Therefore, each time you run the algorithm on the same samples of data, the solutions they provide can differ.

Following are the main contributions of this novel optimized system:

- Making familiar scientific community and new researchers with brain tumor detection techniques based on optimization and deep learning concept.
- Spreading awareness with the specific use of optimization techniques in brain tumor detection.
- Developing the idea for a new technique which can be a combination of deep learning and optimization algorithm for brain tumor detection.
- Providing information regarding optimization techniques to detect brain tumor in human.
- Presenting a combination of CNN and WOA for the detection of brain tumor.
- Providing sufficient information to young researchers, scientists, technocrats about different optimization techniques such as WOA, GWO, PSO, GA, etc. and working of CNN for brain tumor detection.
- The proposed system will be helpful to develop the Binary and multi-objective variants of this system for brain tumor detection in future.
- Finally, the proposed CNN models can be employed to assist physicians in validating their initial screening for brain tumor multi-classification purposes.

3 Methods and materials

3.1 Magnetic resonance imaging (MRI) scanning

Lots of scanning methods such as CT scan, PET, MRI, etc. are used to study brain tumors. At present, MRI is mostly the preferred imaging method, as it produces images of high resolution, which helps to extract features clearly. It is also useful for the visualization of pathological and physiological changes of live tissues. Aims behind the use of MRIs for brain tumors imaging are [17].

- It does not use Ionizing radiation.
- Its resolution is high.
- It generates 3D images, which helps to generate complex tumor localization.
- Power of achievement of structural and functional data of tumor throughout the same scan.

These days, MRI is the utmost demanded radiological imaging technique, because it helps the internal structure of the brain to be viewed with few details. It helps in detecting, descent contrast among dissimilar soft tissues of the brain, which makes it suitable to offer brain images of moderate quality as compared to alternate techniques of imaging (https://case.edu/med/neurology/NR/MRI%20Basics.htm).

3.2 Dataset

For the practical work of this proposed work, we collected 253 brain MRI images online from Kaggle (https://www.kaggle.com) dataset for brain tumor detection. Our dataset contains two classes – Yes and No. Class Yes and No represents tumor and non-tumor images. There are 155 images in class Yes and 98 in Class No. All these images are MRI images from various modalities, such as T1, T2, Fluid-attenuated inversion recovery (FLAIR) [10]. The size of each image is (128, 128) exactly in axial view. All the images of the dataset are available at: https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection. Further, the dataset size is augmented to 2318 images. Out of these images, 1622 have been used for training, 348 for testing and 348 have been utilized for validation.

3.3 Proposed architecture of brain tumor detection

Brain tumor detection in the early phase is an important task for the reduction of casualties of patients. In this system architecture based on deep learning techniques has been introduced, which is a combination of image classification and deep learning techniques. Convolution neural network with whale and wolf optimization is used for brain tumor detection. The detection of brain tumors is done in four steps: preprocessing, segmentation, feature extraction, CNN-based optimization and detection. The proposed architecture is depicted in Fig. 1.

3.3.1 Preprocessing

Processing the MRI is cumbersome task. Before processing MRI, it is necessary to remove unnecessary artifacts in MRI. After the elimination of unnecessary noise, MRI becomes ready for further processing [10]. Preprocessing includes grayscale conversion and filtering. After conversion to grayscale, extra noise is eliminated by applying filtering techniques. This proposed system defines a compound filter, which is a composition of Gaussian, mean and median filter and is used for elimination of Gaussian, salt and pepper and speckle noises in a greyscale images. The advantage of compound filter is that it preserves the edges and boundaries in MRI.

• **Greyscale conversion** Greyscale conversion of MRI is the common preprocessing practice [13]. RGB MRI contains unnecessary information that is not required for processing of the image. Such information can be discarded by converting RGB MRI into grayscale MRI [4]. RGB scale presents MRI into 3 channels, B, G, R and each channel consists of 8 bits.

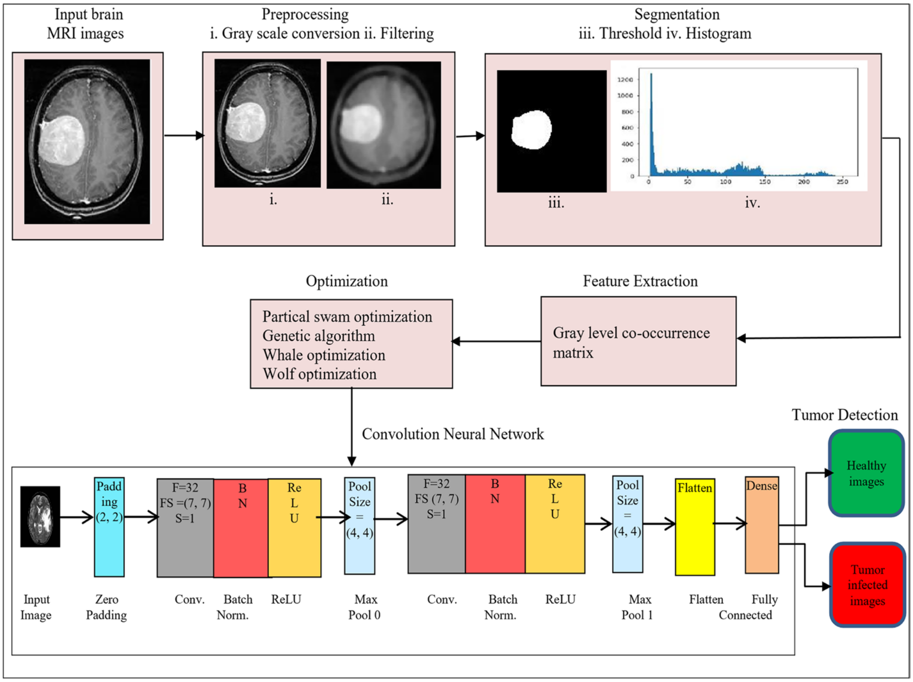


Fig. 1 Proposed architecture of brain tumor detection

For each B, G, R component, the image holds different intensity levels. That is why for a color image we have intensities for each scale. Therefore, RGB needs large amount of data to store and manipulate. Figure 1 shows a sample of conversion of RGB MRI image into greyscale Image.

- **Filtering** After conversion to greyscale, extra noise is eliminated by applying filtering techniques. This proposed system defines a compound filter, which is a composition of Gaussian, mean and median filter and is used for elimination of Gaussian, salt and pepper and speckle noises in greyscale images. The advantage of a compound filter is that it preserves the edges and boundaries in MRI.

3.3.2 Segmentation

MRI Image Segmentation [4] is very important since numbers of images are produced at the time of scanning and it is very difficult for physicians to divide these images manually in a reasonable time. Image segmentation means, the partition of MRI images in several non-overlapping sections. Segmentation partitions the image into groups of pixels that are more important and easier for analysis. Segmentation is performed to identify boundaries or objects in an image and the resulting sections together cover the whole image. Segmentation methods work on two features of image intensity, similarity and discontinuity. Many segmentation techniques are available, like threshold based segmentation and histogram-based methods, region-based, edge-based and clustering methods. Threshold and histogram-based segmentation is a most likely technique for processing the MRI images. This work uses threshold and histogram-based segmentation methods for the analysis of brain tumors and calculation of area of tumor in MRI images. The sample result of threshold and histogram segmentation is shown in Fig. 1.

3.3.3 Feature Extraction

Feature extraction means shortening a number of resources needed to define a big group of data correctly. During the analysis of the large set of data, the problems stem if a huge number of variables are used. Analysis of the huge number of variables needs more memory and computation time. Feature extraction is the process of creating groupings of variables to solve these complications and then describing the data with enough accuracy. In this proposed work, GLCM [15] is used to obtain both statistical and texture-based features.

The number of columns and number of rows in GLCM remains the same as the number of gray levels of the image. Assume (P_i: Δx, Δy) is an element of GLCM. Here, (P_i: Δx, Δy) represents the relative frequency with which two pixels, which are separated by a pixel distance (Δx, Δy) occurs within a given neighborhood, where the intensity of the first pixel is i and the other's is j. The matrix element (P_i: d, $\varnothing$) contains the second- order statistical probability values for the changes between gray level i and j at a displacement distance d and angle $\varnothing$.

The proposed system extracts the following features using GLCM are shown in Table 3.

i. Contrast: Contrast splits the darkest and brightest area of an image. It is calculated using the following formula:

$$Contrast = \sum_{i,j=0}^{n-1} P_{i,j}(i-j)^2 \tag{1}$$

ii. Correlation: Correlation is calculated as the correlation coefficient between -1 and + 1.

$$Correlation = \sum_{i,j=0}^{n-1} P_{i,j}\frac{(i-\mu)(j-\mu)}{\sigma^2} \tag{2}$$

iii. Homogeneity: Homogeneity means the quality or state of being homogeneous.

$$Homogeneity = \sum_{i,j=0}^{n-1} \frac{P_{i,j}}{1+(i-j)^2} \tag{3}$$

iv. Entropy: the degree of uncertainty in a random variable is termed entropy.

$$Entropy = \sum_{i,j=0}^{n-1} P_{i,j}\log(P_{i,j}) \tag{4}$$

v. Energy: Energy is used in GLCM to calculate the total of squared elements. Energy measures homogeneity. A high level of energy indicates that the image has excellent homogeneity or pixels of an image are very similar.

$$Energy = \sum_{i,j=0}^{n-1} {P_{i,j}}^2 \tag{5}$$

vi. Smoothness: Smoothness is a measure of grey level contrast that is used to establish descriptors of relative smoothness.

$$Smoothness = 1 - \frac{1}{1 + s^2} \tag{6}$$

Here, s is the standard deviation of an image.

vii. Kurtosis: Kurtosis is used to compute the flatness of the distribution which is relative to a normal distribution.

$$Kurtosis = \frac{1}{mn} \sum\nolimits_{i=1}^{m} \sum\nolimits_{j=1}^{n} \left\{ \left[\frac{P_{i,j} - \sigma}{\sigma} \right]^4 \right\} - 3 \tag{7}$$

Here, $P_{i,}$ is the pixel value at point(i, j), m and s are the mean and standard deviation respectively.

viii. Root Mean Square (RMS): RMS calculates the value of each row or column of the input matrix of the given dimension of the input or whole input.

$$RMS = \frac{\sqrt{\overline{\left|P_{i,j}\right|}^2}}{m} \tag{8}$$

Eight GLCM textural features, Contrast, Correlation, Homogeneity, Entropy, Energy, Smoothness, Kurtosis and RMS of each image out of 253 images are calculated, as shown in Table 1.

3.3.4 Optimization

The optimization phase deals with the optimizers for the selection of image features. For the implementation of the proposed optimizer, the whale and wolf optimization algorithms are presented here. Feature selection is to be converted in to a more reliable and suitable form for the classifier implementation to detect brain tumors. This paper introduced the whale and wolf optimized features for deep learning-based CNN algorithms to detect the healthy and tumor-infected brain in MRI images. Meanwhile, the proposed algorithm has been compared in terms of accuracy, precision and recall with another modern optimization algorithm is shown below:

• Particle swam optimization (PSO) algorithm PSO [24] is a swarm intelligence-based optimization algorithm. It follows the behavior of particles of the swarm and their interactions. The birds move from one place to another for food. The bird which is nearby to food can smell it. PSO algorithm uses n swarm particles and the position of each particle stands for the possible solution. The particle changes its position as per given below three rules:

- particle keeps its inertia
- particle updates the condition w.r.t. its optimal position
- particle updates the condition w.r.t. the most optimal position of swarm.

Table 1 Extracted features of MRI images

Image	Contrast	Correlation	Homogeneity	Entropy	Energy	Smoothness	Kurtosis	RMS
I001	1.7326	0.1075	1.0987	2.4589	0.8927	0.9450	15.255	0.0845
I002	0.2414	0.1237	0.8876	3.8976	0.3686	0.9335	12.544	0.0933
I003	0.2725	0.1465	0.8745	3.1765	0.3644	0.9194	15.280	0.0919
I004	1.1327	0.1253	0.9178	2.8345	0.4168	0.9607	15.438	0.0860
I005	1.2434	0.1344	1.1984	2.6623	0.7424	0.9316	13.231	0.0931
I006	1.2723	0.1147	1.5873	3.4078	0.3683	0.9445	17.285	0.0944
I007	1.0326	0.1195	1.4742	3.4412	0.3246	0.9493	18.164	0.0849
I008	0.1417	0.0964	1.3177	2.4534	0.4268	0.9522	16.173	0.0852
I009	0.2736	0.1126	0.0985	3.2223	0.7923	0.9509	17.148	0.0950
I010	1.7325	0.1183	1.8875	2.5565	0.7680	0.9348	18.145	0.0934
I011	0.4414	0.0942	1.8744	3.7643	0.4649	0.6915	19.154	0.0891
I012	1.5720	0.1166	1.9172	3.1178	0.6165	0.9524	14.175	0.0952
I013	1.4411	0.1347	2.3329	2.9065	0.3423	0.3233	16.254	0.0823
I014	1.2526	0.1565	1.4378	2.6634	0.5685	0.9387	15.264	0.0938
I015	1.0625	0.1344	1.3246	2.8834	0.4243	0.9464	12.858	0.0946
I016	0.1414	0.1183	1.1274	2.4376	0.8262	0.9362	19.243	0.0936
I017	0.4723	0.1222	1.1180	3.6712	0.5527	0.9229	17.233	0.0922
I018	1.3326	0.1337	1.1871	2.4534	0.4485	0.9261	12.208	0.0926
I019	1.5415	0.1446	0.8743	2.8967	0.3684	0.9396	15.676	0.0939
I020	0.5924	0.1115	1.2174	3.2754	0.4163	0.9145	15.917	0.0914
I021	1.8413	0.1623	1.3985	2.5834	0.7422	0.9554	12.133	0.0955
I022	1.6727	0.1067	1.4576	2.9623	0.3680	0.9183	13.615	0.0918
I023	1.5328	0.1332	1.5747	3.1076	0.3241	0.9205	12.859	0.0920
I024	0.4459	0.1571	1.6173	3.1454	0.4263	0.9530	17.447	0.0953
I025	0.2445	0.1730	0.7984	2.7834	0.7924	0.9368	19.355	0.0936
I026	1.3424	0.1358	1.7875	3.1287	0.7685	0.9555	18.412	0.0855
I027	1.4667	0.1036	2.5749	2.5167	0.4646	0.9634	13.287	0.0863
I028	0.7726	0.1145	1.4170	3.7156	0.6167	0.9493	17.433	0.0949
I029	1.4313	0.1974	0.8328	3.1756	0.3424	0.9506	21.363	0.0950
I030	0.4722	0.1583	1.4377	2.5023	0.5683	0.9315	17.298	0.0931
I031	1.3367	0.1060	0.7246	2.6056	0.4248	0.9644	15.136	0.0964
I032	0.3548	0.1128	1.1275	2.4834	0.8266	0.9593	12.124	0.0959
I033	0.5726	0.1781	1.7174	3.2378	0.5464	0.9322	19.411	0.0932
I034	1.3233	0.0993	1.6680	3.4732	0.7653	0.9707	12.909	0.0870
I035	1.2212	0.0924	1.4576	3.1758	0.3752	0.9545	13.142	0.0854
I036	1.2224	0.1040	1.4545	2.284	0.3240	0.6311	18.152	0.0631
I037	0.2545	0.1168	1.4174	3.3614	0.3369	0.9724	21.225	0.0972
I038	0.5436	0.1747	0.9793	3.3036	0.6741	0.3835	14.204	0.0383
I039	0.5518	0.1486	1.5876	3.3486	0.7236	0.9783	20.798	0.0978
I040	1.4436	0.1225	0.8745	2.9845	0.7845	0.9667	18.243	0.0966
I041	1.5933	0.1934	0.9174	3.2224	0.4564	0.9566	14.214	0.0956
I042	1.6585	0.1143	1.8323	2.6195	0.3883	0.9623	22.212	0.0862
I043	1.6383	0.1167	0.8872	3.9137	0.5382	0.9462	18.613	0.0846
I044	0.1642	0.0738	0.7249	3.0754	0.8949	0.9899	18.972	0.0889

Table 1 (continued)

Image	Contrast	Correlation	Homogeneity	Entropy	Energy	Smoothness	Kurtosis	RMS
I045	1.3753	0.1054	0.9278	2.9034	0.4560	0.9340	13.195	0.0934
I046	1.4337	0.1383	1.5177	2.6720	0.8651	0.9651	18.142	0.0965
I047	1.4336	0.1592	1.2326	2.8474	0.6685	0.9583	12.859	0.0958
I048	0.7869	0.1711	1.4371	3.1322	0.7646	0.9304	19.647	0.0830
I049	0.8418	0.1307	1.3240	3.1740	0.6564	0.9736	22.154	0.0873
I050	2.0026	0.1048	0.9276	3.4265	0.7643	0.9667	20.527	0.0966

The algorithm follows these steps:

- Initialize the best neighborhood particle pbest and the best global particle gbest with random values to calculate a fitness value for each particle, which is then compared with pbest and gbest values.
- If the particle has a better value than pbest and gbest, then we update the values of pbest and gbest as the values of that particle.
- Otherwise, update the particles' position and velocity according to Eqs. (9) and (10) respectively, so as to follow the best particle.

$$v_{idk} = wv_{idk} + c_1 r_1 \left(pbest_{idk} - x_{idk}\right) + c_2 r_2 \left(pbest_{idk} - x_{idk}\right) \tag{9}$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \tag{10}$$

Where c_1 and c_2 are the learning factors and w is the initial weight. After visiting all the particles, the particle with the best fitness factor is selected as an initial particle of the CNN algorithm.

- **Genetic algorithm (GA)** Genetic algorithm [1] uses the following data for its working:

- genotype: the segmented result of an image is considered as an individual described by the class of each pixel.
- initial population: is a group of individuals categorized by their genotypes.
- fitness function: allows computing the fitness of an individual to the environment by seeing its genotype.
- operators: defines alterations on genotypes in order to make the population develop during generations. There are three operators:

Individual mutation: allows the individual's genes to modify for better adaption to the environment. Here the non-uniform mutation process is used, which randomly selects one chromosome and sets it equal to the non-uniform random number:

$$\begin{gathered} x_i' = x_i + \left(b_i - x_i\right)(G), if r_1 < 0.5 \\ x_i' = x_i - \left(x_i + a_i\right) f(G), if r_1 < 0.5 \end{gathered} \tag{11}$$

where

$$f(G) = \left(r_2\left(1 - \frac{G}{G_{max}}\right)\right)^b \tag{12}$$

r_1, r_2	numbers in range [0, 1]
a_{i}, b_{i}	lower and upper bound of chromosome r_i
G	current generation
G_{max}	maximum number of generations
b	shape parameter

- Selection of individual: for individual selection normalized geometric ranking selection method is applied in the next generation of individuals. The probability P_i for each individual i to be selected as given below:

$$P_i = \frac{q(1-q)^{r-1}}{1-(1-q)^n} \tag{13}$$

Where:

q	the probability of selecting the best individual
r	rank of the individual, where 1 is the best
n	population size

- Crossover: by the combination of genes of two individuals third individual can reproduce. Here, the arithmetic crossover produces two complementary linear combinations of the parents as given below:

$$X^{'} = aX + (1-a) \tag{14}$$

$$Y^{'} = (1-a) + aY \tag{15}$$

Where:

X, Y	genotype of parents
a	number in the range [0, 1]
X', Y'	genotype of the linear combinations of the parents

- Stopping criterion: it ends the evolution of the population.

• **Humpback whale optimization algorithm (HWOA)** The proposed HWOA based on deep learning [23] helps to enhance the speed of the training procedure using optimally selecting the parameters pixel resolution. In the whale optimization technique, the humpback whales hunt the prey by applying three processes namely search, encircle and forming a bubble net for hunting the prey.

Mathematical modeling of HWOA For the mathematical presentation of HWOA processes, encircling the prey, spiral bubble-net feeding activities and searching for prey are explained below.

Phase 1: Initialization The initialization phase of HWOA is used to develop the initial solution randomly. After preprocessing of MRI image of brain tumor, its pixel size generated by the parameters of CNN is optimally selected by HWOA algorithm. The CNN parameters such as number of kernels, padding, pooling type, number of feature maps and whale population are randomly initialized. The random value in the search space is given in Eq. (16):

$$(u) = \left(e_1, e_2, e_3, \ldots, e_h\right) \tag{16}$$

Here, E is the original population of the whale and h is the number of interconnection layers for optimization.

Phase 2: Fitness calculation For brain tumor detection, the fitness function is used for the best classification measure by maximizing its accuracy and evaluated using Eq. (17):

$$(u) = maxi(Accuracy) \tag{17}$$

Phase 3. Encircling prey stage Humpback whale (agent) knows the position of prey and encircle them. Whale considers that the current best candidate solution is the best-obtained solution and close to the optimal solution. After assigning the best candidate solution, other agents try to update their positions towards the best search agent by using the Eq. (18) and (19):

$$D = |C.X^{*}(t) - X(t)| \tag{18}$$

$$D = (t + 1) = X^{*}(t) - AD \tag{19}$$

where t is current iteration, A and C are coefficient vectors, X* and X are position vectors of the best solution and solution respectively, | | represents absolute value.

The value of vectors A and C are calculated by using Eq. (20) and (21):

$$A = 2a.r.a \tag{20}$$

$$C = 2.r \tag{21}$$

where, components of a are linearly decreased from 2 to 0 over the course of iterations and r is a random vector in [0 1].

Phase 4. Exploitation or attack stage The humpback whales use a bubble-net mechanism for attacking the prey. There is two mechanisms of attack as given below:

- Shrinking encircling mechanism: In this mechanism, the value of A is a random value in the interval [-a, a] and it is decreased from 2 to 0 over the course of iterations as shown in Eq. (3).
- Spiral updating position mechanism: In this mechanism, the distance between the location of the whale and prey is computed. Thereafter, the helix-shaped movement of the humpback is created by using Eqs. (22) and (23).

$$(t+1) = D^{'}.c^{bl}.\cos(2\pi l) + X^{*}(t) \tag{22}$$

Where,

$$D^{'} = |X^{*}(t) - X(t)| \tag{23}$$

is the distance between the prey or best solution and i$^{\text{th}}$ whale, b is a constant, l is a random number in the range [-1 1].

The humpback whales apply the above two mechanisms when they come nearby the prey. There is a probability of 50% to select these two mechanisms to update the position of whales as shown in Eqs. (24) and (25):

$$(t+1) = X^{*}(t) - AD, ifp < 0.5 \tag{24}$$

or

$$(t+1) = D^{'}.c^{bl}.\cos(2\pi l) + X^{*}(t), ifp \geq 0.5 \tag{25}$$

where, p is a random number in the range [0 1].

Phase 5. Exploration or search stage The humpback whales or search agents hunt for prey or the best solution randomly and alter their locations as per the location of another whale. To force the search agent to move away from the referenced whale, the value of A > 1 or A < 1 is used. The arithmetic representation of the exploration phase is given in Eqs. (26) and (27):

$$D = \left|C.X_{rand} - X\right| \tag{26}$$

$$(t+1) = \left|X_{rand} - AD\right| \tag{27}$$

where X_{rand} is a random position vector selected from the current population?

At the time of updating each solution, the fitness calculation is measured to find the best solution among them. Among the best solution, a set of new solutions is selected and then the fitness function is computed for continuing the above solution updating process.

Phase 6: Termination criteria At the last, it satisfies the finest parameters of CNN by the hunting behavior of the whale. As a result of finding the optimal solution, the prediction model is trained. Since the objective function is to improve the accuracy of training data, the prediction model obtained for the best fitness structure is well trained to predict unknown data.

• **Gray wolf optimization (GWO) algorithm** GWO [28] is applied to optimize the features along the hidden neurons of CNN. It has four levels: the first level alpha ($\propto$) represents the leader of the troop. Alpha may be male or female and have the power of taking decisions. The second level beta (β) helps alpha in taking decisions. The third level delta (δ) is subordinates and the fourth level omega (ω) is known as the scapegoat. All the three levels $\propto$, β and δ guide in the hunting procedure. The encircling behavior is shown in Eq. (28) and (29).

$$A = (tr) - (tr) \tag{28}$$

$$(tr + 1) = (tr) - CA) \tag{29}$$

where tr indicates present iteration, C and B represent coefficient vectors, X^P refers the position vector of the prey, X specifies the position vector of the grey wolf. The assessment of C and B are defined in Eqs. (30) and (31), where, m_i is linearly minimized from 2 to 0, v_1 and v_2 are the random vectors in the range [0, 1].

$$C = 2m_i.v_1 - m_i \tag{30}$$

$$B = 2v_2 \tag{31}$$

Generally, $\propto$ guides the hunting process. The leading three best solutions are saved from the search space and the respective update strategy is assessed as per Eq. (32–34).

$$\begin{aligned} \mathrm{A}_{\propto} &= \left|\mathrm{B}_1.\mathrm{X}_{\propto} - \mathrm{X}\right| \\ \mathrm{A}_{\upbeta} &= \left|\mathrm{B}_2.\mathrm{X}_{\upbeta} - \mathrm{X}\right| \end{aligned} \tag{32}$$

$$\begin{aligned} \mathrm{A}_{\delta} &= \left|B_1.X_{\delta} - X\right| \\ X_1 &= X_{\propto} - C_1(A_{\propto}) \end{aligned} \tag{33}$$

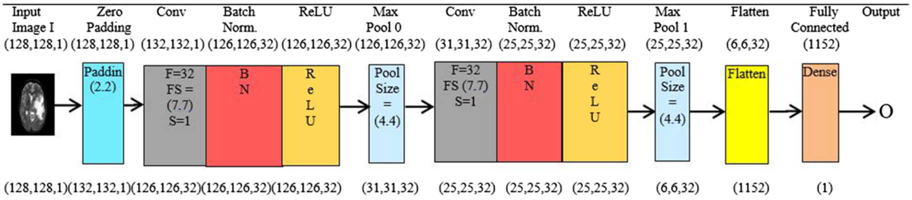
$$\begin{aligned} X_2 &= X_{\beta} - C_2(A_{\beta}) \\ X_3 &= X_{\delta} - C_3(A_{\delta}) \\ X(tr + 1) &= \tfrac{X_1+X_2+X_3}{3} \end{aligned} \tag{34}$$

The final $\propto$, β and δ maybe a random position in search space. The final position will be the optimal feature and hidden neuron that works for the detection of a brain tumor.

3.3.5 Convolution neural network (CNN)

The proposed system uses deep learning-based CNN for the detection of brain tumors by using the features extracted from pre-processed brain tumor images in step 3. This deep learning CNN includes many hidden layers known as zero paddings, convolution, batch normalization, ReLU, max pooling, flatten and fully connected layer as explained below.

CNN [19] is generally used in image processing to detect and classify brain tumors. In this proposed work Nine Layers CNN is used and practically implemented using Python programming language for the identification of brain tumors from brain MRI images. Figure 2 describes the working of CNN.



0: Represents Healthy image
1: RepresentsTumor infected image
F: Number of Filters
FS: Filter Size
S: Strides

Fig. 2 Working flow of CNN

Steps of CNN algorithm There are four types of layers for a CNN. These are convolutional layer, pooling layer, ReLU correction layer and fully-connected layer. Details of theses layers are given below.

i. Convolutional layer

The convolutional layer is the key component of CNN and is always the first layer. Its purpose is to detect the presence of a set of features in the images received as input. This is done by convolution filtering: the principle is to “drag” a window representing the feature on the image and to calculate the convolution product between the feature and each portion of the scanned image. A feature is then seen as a filter, these two terms are equivalent in this context.

The convolutional layer thus receives several images as input and calculates the convolution of each of them with each filter. The filters correspond exactly to the features we want to find in the images.

We get for each pair (image, filter) a feature map, which tells us where the features are in the image: the higher the value, the more the corresponding place in the image resembles the feature.

Unlike traditional methods, features are not pre-defined according to a particular formalism, but learned by the network during the training phase. Filter kernels refer to the convolution layer weights. They are initialized and then updated by back-propagation using gradient descent.

ii. The pooling layer

This layer is often placed between two layers of convolution. It receives several feature maps and applies the pooling operation to each of them. The pooling operation consists in reducing the size of the images while preserving their important features.

To do this, we cut the image into regular cells, then we keep the maximum value within each cell. In practice, small square cells are often used to avoid losing too much information. The most common choices are 2×2 adjacent cells that don’t overlap, or 3×3 cells, separated from each other by a step of 2 pixels (thus overlapping).

We get in output the same number of feature maps as input, but these are much smaller. The pooling layer reduces the number of parameters and calculations in the network. This improves the efficiency of the network and avoids over-learning. The maximum values are spotted less accurately in the feature maps obtained after pooling than in those received in input.

iii. The Rectified Linear Units (ReLU) correction layer

ReLU refers to the real non-linear function defined by

$$\mathrm{ReLU}(x) = \max(0, x).$$

The ReLU correction layer replaces all negative values received as inputs by zeros. It acts as an activation function.

iv. The fully-connected layer

The fully-connected layer is always the last layer of a neural network. This layer receives an input vector and produces a new output vector. To do this, it applies a linear combination and then possibly an activation function to the input values received. The last fully-connected layer classifies the image as an input to the network: it returns a vector of size N, where N is the number of classes in our image classification problem. Each element of the vector indicates the probability for the input image to belong to a class.

To calculate the probabilities, the fully-connected layer, multiplies each input element by weight, makes the sum, and then applies an activation function (logistic if $N=2$, softmax if $N>2$). This is equivalent to multiplying the input vector by the matrix containing the weights. The fact that each input value is connected with all output values explains the term fully-connected.

The convolutional neural network learns weight values in the same way as it learns the convolution layer filters: during the training phase, by back-propagation of the gradient.

The fully connected layer determines the relationship between the position of features in the image and a class. Indeed, the input table being the result of the previous layer, it corresponds to a feature map for a given feature: the high values indicate the location (more or less precise depending on the pooling) of this feature in the image. If the location of a feature at a certain point in the image is characteristic of a certain class, then the corresponding value in the table is given significant weight.

The parameterization of the layers A CNN differs from another networks by the way the layers are stacked, but also parameterized. The convolution and pooling layers have hyper parameters, that is to say parameters whose you must first define the value. The size of the output feature maps of the convolution and pooling layers depends on the hyper parameters.

Each image (or feature map) is $W \times H \times D$, where W is its width in pixels, H is its height in pixels and D the number of channels (1 for a black and white image, 3 for a color image).

The convolutional layer has four hyper parameters:

1. The number of filters K.
2. The size F filters: each filter is of dimensions $F \times F \times D$ pixels.
3. The S step with which you drag the window corresponding to the filter on the image. For example, a step of 1 means moving the window one pixel at a time.
4. The Zero padding: adds a black contour of P pixels thickness to the input image of the layer. Without this contour, the exit dimensions are smaller. Thus, the more convolu-

tional layers are stacked with $P = 0$, the smaller the input image of the network is. We lose a lot of information quickly, which makes the task of extracting features difficult.

For each input image of size $W \times H \times D$, the pooling layer returns a matrix of dimensions $W_C \times H_C {\times} D_C$, where:

$$W_C = \frac{W-F+2P}{S} + 1$$

$$H_C = \frac{H-F+2P}{S} + 1$$
$$D_C = K$$

Choosing, $F = \frac{F-1}{2}$, $S = 1$ gives feature maps of the same width and height as those received in the input.

The pooling layer has two hyper parameters:

1. The size F of the cells: the image is divided into square cells of size $F \times F$ pixels.
2. The S step: cells are separated from each other by S pixels.

For each input image of size $W \times H \times D$, the pooling layer returns a matrix of dimensions $W_P \times H_P {\times} D_P$ where:

$$W_P = \frac{W-F}{S} + 1$$
$$H_P = \frac{H-F}{S} + 1$$
$$D_P = D$$

The choice of hyper parameters is made according to a classic scheme:

For the convolution layer, the filters are small and dragged on the image one pixel at a time. The zero-padding value is chosen so that the width and height of the input volume are not changed at the output. We choose $F = 3, P = 1, S = 1$ or $F = 5, P = 2, S = 1$

For pooling layer, $F = 2, S = 2$ is a choice. This eliminates 75% of the input pixels. We can also choose $F = 3, S = 2$. In this case, the cells overlap. Choosing larger cells causes too much loss of information and results in less good.

In Fig. 2 each input image **I** of the shape of (128, 128, 3) is served to CNN, which follows the sequence of layers.

Here the size of the image **I** is (128, 128, 3). We need to add an additional dimension to process multiple batches in one epoch. Since size of batches can change, therefore, this additional dimension is represented here as None. Hence the changed shape of input image is (None, 128, 128, 3).

Zero Padding layer packs the border of the image **I** with 0 $_s$. A zero padding layer of pool size, 2*2 is used here to perfectly fit the input image by padding the image **I** with zeros. After padding zero the size of image becomes (132, 132, 3) which is fed to convolution layer.

Convolution of the size of an image (132,132, 3) with the filter size of (7, 7), stride 1, dilation rate 1 and valid padding, gives the output of size of image:

$$(132 - 7 + 1, 132 - 7 + 1) = (126, 126)$$

Here convolution layer uses 32 filters, therefore the output shape becomes (126, 126, 32). The filter is applied here for the detection of the presence of particular features in the input image **I**. the output of the convolution layer (126, 126, 32) becomes the input of the batch normalization layer.

The output of the batch normalization layer is (126, 126, 32). This is fed to the activation layer. The output of the activation layer is (126, 126, 32) which is fed to max-pooling layer 0, It selects the biggest element from the rectified feature map. This layer uses 4 filters, 4 strides and a pool size of (4, 4) for its processing. Max pooling layer 0 gives output image size using the formula as given below:

$$\frac{N-F}{S}+1$$

Here:

N is the dimension of input to pooling layer, F is the dimension of filter and S is stride. Then the size of the output image is

$$\left(\frac{126-4}{4}+1,\frac{126-4}{4}+\right)=(31,31)$$

This CNN includes another max-pooling layer to reduce the computation cost. Here stride indicates the shifting of the number of pixels in the input matrix. If the value of stride is 1, it allows to move filter 1 pixel at a time, if the stride is 2, means moving the filters 2 pixels at a time and so on.

The image size of (31, 31, 32) is fed to max-pooling layer 1. The max-pooling layer 1 uses 4 filters, 4 strides and a pool size of (4, 4). Therefore, max-pooling layer 1 produces an output image of size

$$\left(\frac{31-4}{4}+1,\frac{31-4}{4}+\right)=(7,7)$$

The image size of (7, 7, 32) is fed to the next layer, which is flattened.

An activation function represents the non-linear transformation that is applied on an input signal to convert into output. Here, the ReLU layer works as an activation function. Therefore, it does not corroborate with the output. Non-linear ReLU layer is used to apply nonlinear operation in CNN since non-negative linear values are needed to learn CNN. Here, the output of ReLU is:

$$\begin{gathered}\mathrm{f(x)=max(0,x).Where,}\\ \mathrm{if x<0then f(x)=0, andif x>then f(x)=x.}\end{gathered}$$

After pooling, we use flatten layer to transform the 3-dimensional matrix, which represents the input images into a 1-dimensional vector for processing. Thereafter, it is fed to the neural network to process further. Flatten layer takes all the pixels and channels of an image which creates a 1-dimensional vector without batch size. An input of size of an image (7, 7, 32) is flattened to:

$$7*7*32=1568\mathrm{values.}$$

At last, a fully connected layer along with one neuron and sigmoid activation function is known as the dense layer is used for binary classification. The final output of our model is 1, it outputs 1 value per sample in the batch.

Each layer contains parameters. Trainable parameters can be updated with CNN and non-trainable parameters remain static. The number of parameters of the convolution layer is represented by Eq. (35):

Table 2 Layer-Wise Output of CNN

Layer no	Name of layer	Inputs of layers	Outputs of layers	No. of parameters
1	Input	(128, 128, 1)	(None, 128, 128, 1)	0
2	Zero Padding	(128, 128, 1) with pool size (2, 2)	(None, 132, 132, 1)	0
3	Convolution	(132, 132, 1) with filter size (7, 7), stride 1, dilation rate 1 and valid padding	(None, 126, 126, 32)	4736
4	Batch Normalization	(126, 126, 32)	(None, 126, 126, 32)	128
5	Activation	(126, 126, 32)	(None, 126, 126, 32)	0
6	Max Pooling 0	(126, 126, 32) with filters 4, stride 4, pool size (4, 4)	(None, 31, 31, 32)	0
7	Convolution	(31, 31, 32) with filter size (7, 7), stride 1, dilation rate 1 and valid padding	(None, 25, 25, 32)	4736
8	Batch Normalization	(25, 25, 32)	(None, 25, 25, 32)	128
9	Activation	(25, 25, 32)	(None, 25, 25, 32)	0
10	Max Pooling 1	(25, 25, 32) with filters 4, stride 4, pool size (4, 4)	(None, 6, 6, 32)	0
11	Flatten	(6, 6, 32)	(None, 1152)	0
12	Dense	1152	(None, 1)	1569

Total parameters: 6433
Trainable parameters: 6369
Non-trainable parameters: 64

$$\mathrm{H} * \mathrm{W} * \mathrm{NIC} * \mathrm{NOC} + \mathrm{NOC}\ \text{(if bias is issued)} \tag{35}$$

Here,

H	Height of kernel,
W	Width of kernel,
NIC	Number of input channels,
NOC	Number of output channels

For our convolution layer total parameters calculated, are 6433. Out of 6433 parameters, the trainable parameters are 6369 and 64 are non-trainable.

The complete output of each layer of CNN is depicted in Table 2.

Calculation of parameters of CNN has been described as below.

1. The first layer input is used to read the image. So, no parameters are required by the input layer.
2. In CNN, zero-padding refers to surrounding a matrix with zeros. This can help preserve features that exist at the edges of the original matrix and control the size of the output feature map. Therefore, the zero padding layer requires no parameters.
3. The convolution layer Conv, where CNN learns and certainly has weight matrices. To calculate the numbers of learnable parameters in a Conv we use following formula: ((shape of width of the filter * shape of height of the filter * number of filters in the previous layer + 1) * number of filters in the current layer). Here 1 is added, because of the bias term for each filter.The number of parameters in the first convolution layer Conv with filter shape = 7*7, stride = 1, dilation rate = 1 and valid padding is = (7*7*3 + 1)*32) = 4736.
4. Batch normalization is a layer that allows every layer of the network to learn independently. It is used to normalize the output of the previous layers. The Batch Normalization layer has 128 parameters.
5. An activation function in a CNN defines how the weighted sum of the input is transformed into an output from a node in a layer of the network. Activation layer uses the ReLU function which requires no parameters. ReLU performs an element-wise operation and sets all the negative pixels to 0.
6. There are no parameters to learn in max pooling layer 0. This layer has used to reduce the image dimension size. This has got no learnable parameters because all it does is calculate a specific number and no backprop learning involved.
7. The total number of parameters in the second convolution layer Conv with filter shape = 7*7, stride = 1, dilation rate = 1 and valid padding is = (7*7*3) + 1)*32) = 4736.
8. Again the second batch normalization layer has 128 parameters.
9. A second activation layer uses ReLU function which requires no parameters.
10. There are also no parameters to learn in max pooling layer 1. This layer reduces the image dimension size and has no learning parameters because all it does is calculate a specific number and no backprop learning involved.
11. Flattening is used to convert all the two dimensional arrays from pooled feature maps into a single dimensional array, which requires no parameters. Therefore, flatten layer has no parameters.

Table 3 Parameters and values used in CNN

Algorithm	Parameter	Values
CNN	Layer	8
	Activation function	ReLu
	Optimizer	ADAM
	Loss	binary_crossentropy
	Epochs	20
	validation_split	2
	Batch Size	8

12. In Dense or fully-connected layer all inputs units have a separable weight to each output unit. For n inputs and m outputs, the number of weights is $n*m$. This layer has the bias for each output node, so There are $(n+1)*m$ parameters. Here n = 6, The formula to calculate number of parameters are: (7*7*32)+1= 1569., Here 1 is added, because of the bias term for each filter. Dense layer has 1569 parameters (Table 3).

In this system we have used 8 layers of CNN such as input, zero padding, convolution, batch normalization, activation, max pooling, flatten and dense to detect brain tumor. Activation layer uses the ReLU function to accelerate the training speed of deep neural networks. Adaptive Moment Estimation (ADAM) algorithm applied here for optimization because this system uses a lot of data and parameters and requires minimum memory. ADAM is a combination of gradient descent with momentum algorithm and the Root Mean Square Propagation. The single epoch represents one single pass of all the data through the CNN, here 20 epochs have been used. In this system CNN uses the batch size of 8. This means that 8 MRI images of the brain will be passed as a group to CNN at a time. The binary_ crossentropy loss function is applied here to calculate the loss to train a CNN model for the classification of brain tumor which reduces the classification of a binary choice i.e. yes or no, A or B, 0 or 1.

4 Results and discussion

To test and assess the proposed system, 696 images are taken as an input for testing the whole system. The whale and wolf features, which are extracted from the datasets, are utilized for testing system. For valuation, the presented approach can be estimated by some statistical measurements, such as true negative (TN), true positive (TP), False Negative (FN), and False positive (FP). These values are represented as a confusion matrix in as shown in below table. Confusion matrix provides accuracy of detection of tumor of any classifier. Confusion matrix (Table 4) is created by using actual and predicted values.

Accuracy: Accuracy refers to the ratio of the true patterns to the summation of entire patterns. It can be expressed as

$$\begin{gathered} Accuracy = \frac{TN+TP}{TP+FN+TN+FP} \times 100 \\ Accuracy = \frac{277+412}{696} = 98.9 \end{gathered} \tag{36}$$

Table 4 Confusion Matrix

No. of images 696	Output		
	Healthy	Tumor infected	
Healthy	TP = 412	FP = 5	417
Tumor infected	FN = 2	TN = 277	279
	414	282	696

Therefore, the accuracy for the proposed system 98.9.

Precision: Precision can be described as the ratio of true positive to the summation of entire positive patterns. It can be expressed as

$$\begin{aligned} Precision &= \frac{TP}{TP+FP} \times 100 \\ Precision &= \frac{412}{412+5} = 98 \end{aligned} \tag{37}$$

Thus, precision for the proposed system is 98.

Recall: The recall parameter is used for calculating the false positive and true positive values, and also it can be called sensitivity. It can be expressed as

$$\begin{aligned} Recall &= \frac{TP}{TP+FN} \times 100 \\ Recall &= \frac{412}{412+3} = 100 \end{aligned} \tag{38}$$

Thus, precision for the proposed system is 100.

F-Score: To measure the value of F-Score, it requires both precision and recall values. It can be expressed as

$$\begin{aligned} F - Score &= 2 \times \frac{Precision \times Recall}{Precision + Recall} \\ F - Score &= 2 \times \frac{98 \times 100}{98+100} = 99 \end{aligned} \tag{39}$$

Thus, precision for the proposed system is 99.

Where:

TP means both actual as well as predicted value is true.
FP means predicted value is true but actual value is false.
TN means both actual as well as predicted value is false.
FN means actual value is true and predicted value is false.

In this part, the performance analysis of proposed system is assessed by using accuracy, precision, recall and F-Score parameters. For performance analysis the existing CNN and deep learning based approaches which uses CNN and optimization based approaches such as PSO + CNN, GA + CNN, Wolf + CNN, Whale + CNN are considered Observations values of the proposed and current methods are given in Table 5.

Table 5 describes the results of the current and proposed approach. The accuracy obtained for the brain tumor detection from our proposed approach which has used

Table 5 Comparison of various optimization techniques with proposed system

Techniques	Algorithm	Accuracy	Precision	Recall	F-Score
Deep Learning	CNN	93.9%	95	94	87
	PSO + CNN	95.6%	96	97	89
	GA + CNN	95.9%	97	98	94
Optimization	Wolf + CNN	96.4	97	100	97
	Whale + CNN	98.9%	98	100	99

the Whale + CNN combination is 98.9%. The accuracy of the competitive algorithms, such as CNN, PSO + CNN, GA + CNN and Wolf + CNN are 93.9%, 95.6%, 95.9%, and 96.4% respectively. The precision level for the proposed approach is 95 which uses Whale + CNN combination, whereas for the existing techniques like CNN, PSO + CNN, GA + CNN, Wolf + CNN the values of precision are 95, 95, 96 and 97, respectively. This shows that the precision level for the proposed approach is high. The recall value for the suggested method is 100, whereas for the existing algorithms like CNN, PSO + CNN, GA + CNN and Wolf + CNN are 94, 97, 98, and 100, respectively. At last, the F-Score value for the proposed method is 99, whereas for the existing methods, CNN, PSO + CNN, GA + CNN and Wolf + CNN are 87, 89, 94 and 97 respectively. Therefore, the presented approach based on Deep learning and Whale + CNN based optimization techniques is more efficient in accuracy, precision, recall and F-Score. Table 5 also clearly shows that the Whale + CNN classifier gives better results of accuracy, precision, recall and F-Score than other algorithms. The feature optimization and detection of brain tumors using CNN are implemented in Python which also calculates the number of healthy and tumor infected images. 696 images were taken for analysis. Out of that, 696 has been used for testing and the 70% for training. With this the accuracy of Whale + CNN algorithm has achieved 98.9%. Whale + CNN algorithm reached the maximum accuracy when compared with other algorithms. The confusion matrix indicates that 414 images have been identified as normal and 282 are under tumor infection. Comparative analysis of table accuracy, precision, recall and F-Score is shown in Figs. 3, 4, 5 and 6.

The x-axis represents brain tumor detection performance methods like CNN, PSO + CNN, GA + CNN, Wolf + CNN and Whale + CNN in the Fig. 3, The y-axis represents the parameter values of Accuracy in Fig. 1, Precision in Fig. 2, Recall in

Fig. 3 Comparison with Accuracy

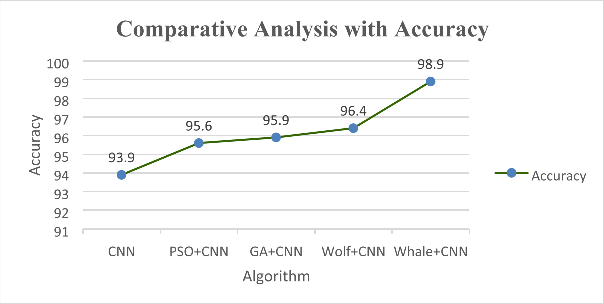


Fig. 4 Comparison with Precision

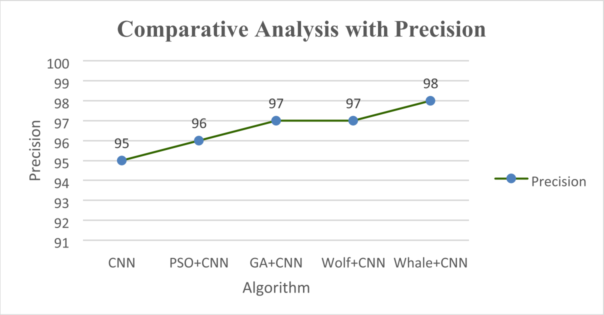


Fig. 5 Comparison with Recall

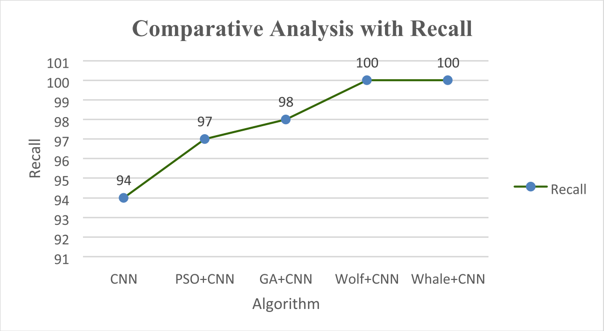


Fig. 6 Comparison with F-Score

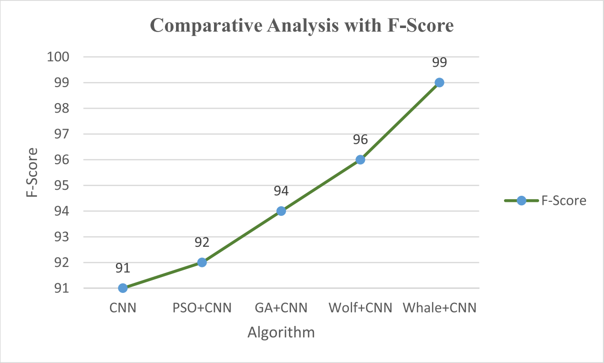


Fig. 3 and F-Score in Fig. 4, respectively. The Figs. 1, 2, 3 and 4 respectively clearly shows that the presented approach with Whale + CNN has highest value for its accuracy, precision, recall and F-Score. This indicates that that the proposed approach is performed efficiently in accuracy, precision, recall and F-Score with other existing algorithms. Therefore, the proposed method may be the most preferred method for detecting brain tumors.

5 Performance comparison

Performance comparison of various existing optimization techniques of brain tumor detection with proposed system is presented in Table 6

Geetha et al. [7] utilized a DBN classifier with GWO to detect the tumor in MRI images and calculated 94.11% accuracy in the year 2019. They proposed a model which includes certain processes such as preprocessing, segmentation, feature extraction and classification. Preprocessing applies two main processes, contrast enhancement and skull stripping. The segmentation process, FCM algorithm has been used. GLCM and GRLM are used for feature extraction. Moreover, this system uses a DBN for classification. The optimized DBN concept is used here, for which GWO is used. Then, Sindhu A et al. [26] designed an Adaboost ensemble KNN-SVM model which uses WOA to identify brain tumors and achieved 98.3% accuracy in the year 2020. This model presents a system which applies five steps, preprocessing, segmentation, feature extraction, feature selection (optimization) and classification. This system performs preprocessing as an initial step to remove the noises by using HE/ACWM. The segmentation uses K Means clustering to identify the tumor location. Feature extraction is done by using First Order and Second Order Statistical features by GLCM and GLRM. Further PSO and WOA are applied for best feature selection. The machine learning classifiers such as DT, KNN, SVM and AdaBoost with ensemble KNN-SVM classifiers are utilized to classify the tumor as normal or abnormal. The Mishra PK et al. [18] applied a DCNN classifier with WOA for the detection of brain tumors with the accuracy of 98% in the year 2021. This system applies histogram alignment and median filter for preprocessing, Otsu thresholding and morphological operations for image segmentation, WOA for feature selection and DCNN for classification. Yin B et al. [28] used a multilayer perceptron neural network classifier with WOA for the detection of brain tumors with the accuracy of 96.5% in the year 2020. This system applies MICO for preprocessing, CNN for segmentation and WAO for feature selection and multilayer perceptron neural network for classification. Fouad A et al. [6] applied Ensemble Learning Classifier with WAO for the detection of brain tumors with the accuracy of 96.4% in the year 2020. This system applied Haar Discrete Wavelet transforms hybrid with the Histogram of Oriented Gradients for preprocessing. Important feature selection is done by using WAO and tumor classification is done by using Ensemble Learning Classifier. Gong S et al. [8] applied radial basis function (RBF) network with WAO for the detection of brain tumors with the accuracy of 88% in the year 2020. This system applied a median filter for preprocessing, for segmentation thresholding and Otsu methods, feature selection is done by WAO and tumor detection uses RBF network. Compared to the above systems, this proposed system obtained 98.9% accuracy with the CNN classifier and WAO for the brain tumor detection in MRI images. Our proposed system uses a combination of Gaussian, mean and median filter for preprocessing of image, threshold and histogram techniques for segmentation and WOA for the best feature selection and finally CNN is applied for tumor detection.

6 Conclusion

This research paper proposed a novel accurate optimized system based on deep learning techniques to detect brain tumors. The system uses preprocessing, segmentation, feature extraction, optimization and detection activities to detect the tumor in MRI images by

Table 6 Performance comparison of existing optimization techniques with proposed system

Author	Year	Preprocessing	Segmentation	Feature extraction and selection	Classification	Accuracy
Geetha A et al. [7]	2019	Contrast enhancement, skull stripping	FCM	Gray wolf optimization	DBN	94.11
Sindhu A et al. [26]	2020	HE/ACWM	K. Means clustering	Whale optimization algorithm	Adaboost ensemble KNN-SVM	98.3
Mishra PK et al. [18]	2021	MICO	CNN	Whale optimization algorithm	DCNN	98
Yin B et al. [28]	2020	histogram alignment, median filter	Otsu thresholding and morphological operations	Whale optimization algorithm	multilayer perceptron neural network	96.5
Fouad A et al. [6]	2019	Haar Discrete Wavelet transforms hybrid with the Histogram of Oriented Gradients	--	Whale optimization algorithm	Ensemble Learning Classifier	96.4
Gong S et al. [8]	2019	median filter	Thresholding, Otsu method	Whale optimization algorithm	radial basis function (RBF) network	88
Proposed Work	2021	Gaussian, mean and median filter	Threshold & Histogram	Whale optimization algorithm	CNN	98.9

using CNN. For preprocessing compound filter is used which is a composition of Gaussian, mean and median filter. For segmentation of MRI images threshold and histogram, methods are used. Image features are extracted by GLCM. Deep learning-based optimized technique that uses whale and grey wolf optimization for the selection of best features of an image. Detection of brain tumors is achieved by using CNN classifier. The performance of this system is compared with other modern methods of optimization in terms of accuracy, precision and recall parameters and proves the superiority of this work. This system is implemented on Python programming language. For the Identification of brain tumors 253 MRI images selected from the Kaggle dataset. Further, the dataset size is augmented to 2318 images. Out of these images, 1622 have been used for training, 348 for testing and 348 have been utilized for validation. The results of implementation and evaluations with the other related system show that this system can reach better detection performance than another modern system. The brain tumor detection accuracy of this optimized system has been measured at 98.9%

It might be hoped that this work will help physicians to take final decisions for further treatment.

Funding There is no funding.

Data availability All the data for brain tumor detection created or used during the study are openly available from the kaggle dataset at**:** https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection.

Declarations

Ethics approval This article contains no studies related to animals.

Conflicts of interest Author declares there is no conflict of interest.

References

1. Abdel-Gawad AH, Lobna A, Ahmed S, Radwan G (2020) Optimized Edge Detection Technique for Brain Tumor Detection in MR Images. IEEE Access 8:136243–136259
2. Avsar E, Salcin K (2019) Detection and classification of brain tumors from MRI images using faster R-CNN. Tehnički Glasnik 13(4):337–342
3. Bahadure NB et al (2017) Image analysis for MRI based brain tumor detection and feature extraction using biologically inspired BWT and SVM. Int Biomed Imaging 2017:1–12
4. Bhima K, Jagan A (2016) Analysis of MRI based brain tumoridentification using segmentation technique. 2016 International Conference on Communication and Signal Processing (ICCSP), 2109-2113
5. Borole VY et al (2015) Image Processing Techniques for Brain Tumor Detection: A Review. Int J Emerging Trends Technol Comput Sci 4:28–32
6. Fouad A, Moftah HM, Hefny HA (2020) Brain Diagnoses Detection Using Whale Optimization Algorithm Based on Ensemble Learning Classifier. Int J Intell Eng Syst 13:2
7. Geetha GA, Gomathi, N (2019) A robust grey wolf-based deep learning for brain tumour detection in MR images. Biomed Eng Biomed Tech 1:9–23
8. Gong S, Gao W, Abza F (2019) Brain tumor diagnosis based on artificial neural network and a chaos whale optimization algorithm. Comput Intell 1–17
9. Gunasekara SR, Kaldera HNTK, Dissanayake MB (2021) A Systematic Approach for MRI Brain Tumor Localization and Segmentation Using Deep Learning and Active Contouring. Hindawi J Healthcare Eng 2021:13
10. Hebli AP, Gupta S (2016) Brain Tumor Detection Using Image processing: A Survey. In: Proceedings of 65th IRF International Conference

11. Hossain T, Shadmani Shishir F, Ashraf M, Abdullah Al Nasim MD, Muhammad Shah F (2019) Brain Tumor Detection Using Convolutional Neural Network, 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT-2019)
12. Irmak E (2021) Multi-Classification of Brain Tumor MRI Images Using Deep Convolutional Neural Network with Fully Optimized Framework. Iranian J Sci Technol Trans Electr Eng 45:1015–1036
13. Kapoor L, Thakur S (2017) A survey on brain tumor detection using image processing techniques. 2017 7th International Conference on Cloud Computing, Data Science & Engineering Confluence. 582–585
14. Khan AH, Abbas S, Khan MA, Farooq M, Khan WA, Siddiqui SY, Ahmad A (2022) Intelligent Model for Brain Tumor Identification Using Deep Learning. Hindawi Appl Comput Intell Soft Comput 2022:10
15. Kumar P, Vijayakumar B (2016) An Efficient Brain Tumor MRI Segmentation and Classification Using GLCM Texture Features and Feed Forward Neural Networks. World J Medical Sci 13:8592
16. Louis DN, Ohgaki H, Wiestler OD, Cavenee WK (2007) WHO Classification of Tumors of the Central Nervous System. International Agency for Research on Cancer, Lyon, France
17. Medical Imaging in Cancer Care: Charting the Progress, US Oncology and National Electrical Manufacturers Association (2012)
18. Mishra PK, Satapathy SC, Rout M (2021) Segmentation of MRI Brain Tumor Image using Optimization based Deep Convolutional Neural networks (DCNN). Open Comput Sci 11:380–390
19. Parihar AS (2017) A Study on Brain Tumor Segmentation Using Convolution Neural Network. 2017 International Conference on Inventive Computing and Informatics (ICICI)
20. Rammurthy D, Mahesh PK (2020) Whale Harris Hawks optimization based deep learning classifier for brain tumor detection using MRI images. J King Saud Univ Comput Inf Sci 1–14.
21. Ramtekkar PK, Pandey A, Pawar MK (2020) A proposed model for automation of detection and classification of brain tumor by deep learning. 2020 2nd International Conference on Data, Engineering and Applications (IDEA).
22. Ramtekkar PK, Pandey A, Pawar MK (2023) Innovative brain tumor detection using optimized deep learning techniques. Int J Syst Assur Eng Manag 14:459–473
23. Shankar K, Lakshmanaprabu SK, Khanna A, Tanwar S, Joel J, Rodrigues PC, Roy NR (2019) Alzheimer detection using Group Grey Wolf Optimization based features with convolutional classifier. Comput Electr Eng 77:230–243
24. Sharifa M, Amina J, Razaa M, Yasmina M, Satapathy SC (2020) An Integrated Design of Particle Swarm Optimization (PSO) with Fusion of Features for Detection of Brain Tumor. Pattern Recogn Lett 129:150–157
25. Sharma AK, Tiwari S, Aggarwal G,Goenka N, Kumar A, Chakrabarti P, Chakrabarti T, Gono R, Leonowicz Z, Jasiński M (2022) Dermatologist-Level Classification of Skin Cancer Using Cascaded Ensembling of Convolutional Neural Network and Handcrafted Features Based Deep Neural Network. IEEE Access. https://doi.org/10.1109/ACCESS.2022.3149824
26. Sindhu A, Radha V (2020) An Optimal Feature Selection with Whale Algorithm and Adaboost Ensemble Model for Pancreatic Cancer Classification in PET/CT Images. Biosc Biotech Res Comm 13(4):18861894
27. Verma SS, Prasad A, Kumar A (2021) CovXmlc: High performance COVID-19 detection on X-ray images using Multi-Model classification. Biomed Signal Process Control
28. Yin B, Wang C, Abza F (2020) New brain tumor classification method based on an improved version of whale optimization algorithm. Biomed Signal Process Control 56:1–10

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.