MRI Brain Tumor Detection Using Boosted Crossbred Random Forests and Chimp Optimization Algorithm Based Convolutional Neural Networks

Omar Abdullah Murshed Farhan Alnaggar[1]*   Basavaraj Ningappa Jagadale[1]
Swaroopa Hebbar Narayan[1]

[1] *Department of PG Studies and Research in Electronics, Kuvempu University,
Jnanasahyadri, Shankaragatta, Shimoga District, Karnataka 577451, India*
* Corresponding author's Email: alnaggar1994@gmail.com

Abstract: Early detection of the brain tumors can help in providing sufficient treatment to completely cure the disease. Machine Learning (ML) and Deep Learning (DL) algorithms are utilized predominantly in recent years for the tumor detection process from the brain Magnetic Resonance Imaging (MRI). In this paper, an effective tumor detection framework is proposed by utilizing a hybrid segmentation algorithm and a new hyper-parameter optimized Convolutional Neural Networks (CNN) based classifier using Chimp Optimization Algorithm (COA). The framework includes pre-processing, segmentation, feature extraction and classification stages. The pre-processing stage performs the denoising using wavelet filter and image enhancement using adaptive histogram equalization. Then the Boosted Crossbred Random Forests (BCRF) is used to segment the regions of interest to identify skull, tumor and lesions. The features are extracted using the Grey-Level Co-occurrence Matrix (GLCM) and Gabor filter. Finally, the tumor classification is achieved using the COA-CNN. The COA-CNN is developed by utilizing the COA to tune the hyper-parameters of CNN to obtain optimal CNN architecture. Experimental evaluation was performed on a dataset of 3264 MRI brain images containing normal and tumor images. No tumor, Glioma, Meningioma, and Pituitary tumor are classes of the MRI images which are correctly classified by the proposed BCRF and COA-CNN based detection model with a testing accuracy of 95.18% and reduced processing time. The results indicate the efficacy of the proposed method in classifying brain tumors for assisting physicians in early tumor diagnosis and treatments.

Keywords: Brain tumor, Magnetic resonance imaging, Tumor detection, Convolutional neural networks, Boosted crossbred random forests, Chimp optimization algorithm.

1. Introduction

The brain tumor is one of the hot research topics in the modern academic community due to its vital contribution to the medical fields as well as humanity as a whole. The brain is the center of the human nervous system and hence the tumor in the brain causes life-threatening scenario that accounts for about 18,000 adult deaths in 2020 [1]. Brain tumors are formed as unrestricted masses of abnormal cells accumulating at a point and eradicating the brain's control mechanism. Brain tumors can be cancerous or non-cancerous based on the degradation it causes to the brain tissues [2]. The gravity inside the skull can quicken the tumor growth in the brain and cause degradation in body health; worst case leading to death. MRI data are commonly utilized to visualize the different appearances and classes of brain tumors. Therefore, MRI scan images can be utilized in detecting the tumor size, region, type and grade which can be utilized to evaluate the tumors such that further planning can be done for the treatment of the tumor [3]. The shape, size, region, type and grade of the tumor differs based on the condition of the patient and are influential in determining the treatment process. Hence the accurate detection of the tumor can help in proper treatment procedures.

Processing the MRI manually and detecting the tumors are highly tedious and error-prone tasks

since the possibility of human mistakes and low precision are common. Hence the automated detection systems were developed to replace the conventional manual approaches. The statistical and filtering methods were the traditionally used automated systems for tumor detection [4]. However, they did not provide the expected performance due to the poor learning of the tumor features. The ML and DL algorithms have been extensively utilized in the last decade for multiple applications. Brain tumor detection studies have also utilized these algorithms and achieved significantly higher performance. Compared to the ML algorithms [5], the DL algorithms have provided superior performance for large image datasets such as BRATS challenge datasets, indicating their superior effectiveness for tumor detection. DL algorithms achieved a high detection rate due to their behaviour of deep learning and precise class determination [6]. It also utilized the transfer learning technique to reduce the computation time so that the smaller MRI datasets are also effectively processed. The use of better segmentation and feature extraction models also help in improving the accuracy and decreasing the processing time. Hence this paper focuses on developing an efficient DL algorithm-based brain tumor detection system to harvest these benefits.

There are many challenges in multi-class brain tumor classification task which reduce the overall system performance. In the segmentation task, the irregular shape and confusing boundaries of the tumor regions are highly challenging. The probability of inhomogeneity of tumorous tissue is also a challenge in tumor region segmentation which many traditional algorithms fail to overcome. Similarly, the performance of classifiers is subject to the power of the extracted features. In ML and DL based classifiers, all features are generally not required for accurate classification [7]. Feature selection algorithms can improve the ML and DL classifiers performance but might increase the computational complexity. Hence the DL classifiers adopted the plan of extracting the features themselves and utilizing the high informative features without separate feature selection stage. CNN has been the most appropriate classifier that inhibits this behaviour through self-learning property and achieved high performance [8]. However, the CNN classifier does not achieve the maximum accuracy in the testing stage since the configuration of CNN is not optimal. Therefore, the proposed method in this paper aims at developing a COA-optimized CNN classifier that has optimal configuration through the assignment of the optimal value for the hyper-parameters.

The major contributions of this paper are the efficient segmentation and classifiction stages. The development of Boosted Crossbred Random Forests (BCRF) by combining the boosting and crossbreeding strategies to the random forests for accurate segmentation is the first contribution. This integration of the boosting algorithm constructs complementary classifier structure to improve the accuracy even when the decision trees are reduced. Likewise, the crossbreeding strategy reduces the space and improves the processing speed of the random forests. The application of Chimp Optimization Algorithm to tune the vital hyper-parameters of the standard CNN classifier to form the hybrid COA-CNN for detecting the tumor classes is the second contribution. This hybrid model of CNN with hyper-parameter optimization reduces the architecture complexity without reducing the accuracy while the benefits of COA also improve the convergence rate of the classifier. The rest of this paper is structured as: related works in Section 2. The proposed brain tumor detection system is given in Section 3 and the experimental results in section 4 with the conclusions in section 5.

2. Related works

ML and DL algorithms have outperformed the other models for brain tumor detection. Recent studies have focused on these algorithms to improve tumor classification. Various MRI brain image datasets were utilized in the studies which means the size and quality of the image datasets are also significant in deciding the classifier performance. Many algorithms provided high training accuracy while the testing accuracy was very poor. This section compares some of the recent prominent research studies using ML and DL algorithms for tumor classification. Minz and Mahobiya [9] utilized the AdaBoost classifier for brain tumor type classification and achieved 89.4 % accuracy and 11 % false-positive rate (FPR). However, this model has increased FPR when applied for larger datasets. Kavin Kumar et al. [10] proposed a brain tumor detection approach using texture features and an SVM classifier. In this approach, the texture features are extracted using a combination of modified multi-texton histogram (MMTH) and multi-texton microstructure descriptor (MTMD). Then the SVM classifier is used to classify the tumor images. This approach achieved 95 % accuracy which was better than the KNN and extreme learning machines (ELM) models. However, this approach has higher false positives which reduce the effectiveness of classification. Gokulalakshmi et al. [11] developed a

discrete wavelet transform-based feature extraction and SVM classifier (TFE-SVM) algorithms based tumor detection approach. In this approach, the filtering techniques are applied for pre-processing while GLCM and discrete wavelet transform are used for feature extraction. SVM is used for binary as well as multiclass classification with 96.2 % and 94.4 % accuracies, respectively and an FPR rate of 6.67 %. However, this approach has limited performance with the lack of an efficient segmentation model.

Ahmed et al. [12] developed a hybrid grey wolf optimizer–artificial neural network (HGWO-ANN) classification approach to tumor detection. This approach classified the tumor types with an accuracy of 94.45 % and FPR of 8.5 % but it incurs high computation complexity. Pereira et al. [13] developed CNN based tumor detection model in which small kernels are used to replace the larger kernels. This approach increases the deep learning property and reduces the weights for avoiding the over-fitting problem. This approach achieved 94.6 % detection accuracy and less FPR of 5 % for tumor type classification. However, this model has higher complexity and higher false negatives. Zhao and Jia [14] proposed multi-scale CNN (MCNN) based brain tumor detection by analysing the top-three scales of the image sizes and combine information to detect the tumor regions. Tested on the BRATS dataset, this approach provided only 81 % testing accuracy and 17 % FPR due to the lack of specialized pre-processing methods and non-extraction of richer boundary information. Zhou et al. [15] proposed a Holistic brain tumor classification approach based on DenseNet and recurrent neural network model of LSTM. In this approach, three RNN models were utilized. Among them, the DenseNet-LSTM model achieved high accuracy of 92.13 % with less computation time. However, this model does not utilize the feature labels which degrade the overall accuracy. Pashaei et al. [16] proposed an ensemble model of CNN and kernel ELM (CNN-KELM) for tumor detection. This ensemble model achieved 93.68 % detection accuracy and 0.062 FPR for Glioma, Meningioma, and Pituitary tumor type classification. However, this model consumed high computation time.

Thaha et al. [17] designed enhanced CNN (ECNN) based brain tumor detection model with loss function optimization by BAT algorithm. It also used Skull stripping and image enhancement algorithms to pre-process the input tumor images. This model achieved a detection accuracy of 92 % which was better than the standard CNN (89 %). However, this model also suffers from high computation complexity since the smaller kernels functions increase the feature learning time. Anaraki et al. [18] proposed brain tumor grades classification model using a genetic algorithm optimized CNN (GA-CNN). This model utilized the GA to optimize the CNN architecture evolution for improving accuracy. To decrease the variance of prediction error, bagging as an ensemble algorithm was also utilized. This model was tested on two test cases and achieved 90.9 % accuracy and 0.0223 FPR for classifying three Glioma grades and 94.2 % accuracy and 0.0581 FPR for Glioma, Meningioma, and Pituitary tumor type classification. Although efficient, this model was evaluated only on smaller datasets so that its performance will vary for larger datasets. Sajjad et al. [19] developed a tumor detection approach using the CNN algorithm with an extensive data augmentation approach for training the network when the lack of data problem occurs. This approach achieved 94.58 % detection accuracy and 3.4 % FPR for Glioma, Meningioma, and Pituitary tumor type classification. However, this CNN architecture is highly complex and hence the lightweight models must be designed.

Irmak [20] developed a multi-classification approach using the deep convolutional neural network whose structure is an optimized grid search optimization algorithm (GSO-DCNN). This model was tested for three classification tasks. Among them, the brain tumor type classification task used normal, glioma, meningioma, pituitary and metastatic types. In this task, the GSO-DCNN achieved 92.66 % accuracy and also reduced the FPR and model complexity. However, this model utilizes all the deep features for classification among which some are completely not needed. Ghassemi et al. [21] utilized the deep neural network with generative adversarial networks (GAN) pre-training to improve the tumor type classification accuracy. This model achieved 93 % accuracy without split while achieved 94.78 % accuracy when the random split was applied. However, the network input image size was 64×64 due to GAN architecture limitation while standard input is often given as either 128×128 or 256×256.

The above studies in the literature show that the brain tumor type detection models using the ML and DL algorithm provided significant results. However, there are limitations in segmentation, feature extraction and classifier optimization that resulted in lower detection rate, high false positives and increased time and model complexities. Considering these limitations, the proposed brain tumor detection framework includes efficient algorithms for segmentation, feature extraction and classifier

optimization so that superior performance is ensured.

3. Methodology

The proposed methodology performs the pre-processing, segmentation, feature extraction and classification tasks. The input dataset contains four classes of images namely no tumor, Glioma, Meningioma, and Pituitary tumor. For reducing the noise, the wavelet denoising filter is used and adaptive histogram equalization is used for image enhancement. BCRF segments the tumor regions and the GLCM and Gabor features are used in COA-CNN for classification. The functional flow diagram of the proposed system is shown in Fig. 1.

3.1 Dataset

Brain tumor images from a public dataset are used in this study to evaluate the performance of the proposed system. The dataset is available at the Kaggle repository [22]. The dataset contains 3264 MRI brain images collected from healthy and tumor patients. It includes four classes of images namely No tumor, Glioma, Meningioma, and Pituitary tumor. The dataset is already split into training and testing datasets for all four classes. The class distribution of the images is shown in Table 1.

Fig. 2 shows the sample images for each class in the input dataset.

Table 1. Class distribution of the input dataset

Class	Total images	Training	Testing
No Tumor	500	395	105
Glioma	926	826	100
Meningioma	937	822	115
Pituitary tumor	901	827	74
Total	**3264**	**2870**	**394**
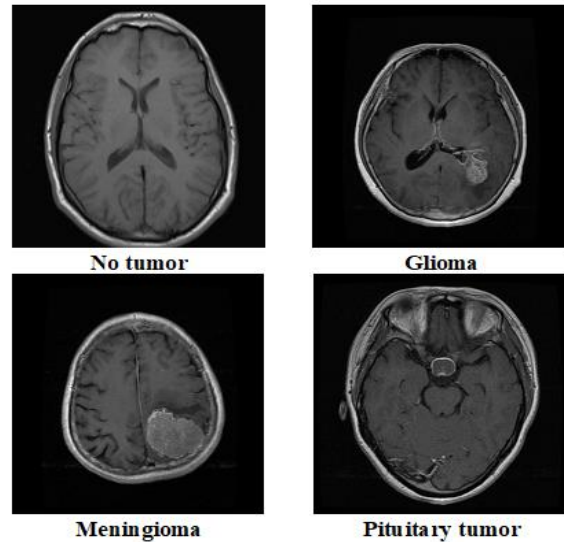
Figure. 2 Sample MRI for all four classes in the input dataset

Figure. 3 Pre-processing results of sample Glioma tumor image

3.2 Pre-processing

The pre-processing task in this study is to remove the noise and enhance the image. Since the images are already grouped as respective classes, duplication is avoided. The adaptive histogram equalization [23] is used to enhance the contrast rate in the brain images. Unlike the standard histogram equalization, this approach computes many histograms corresponding to each section of the image and applies them adaptively in each section to redistribute the contrast values for attaining the best possible image quality.

The images are denoised using a wavelet filter [24]. The basic principle behind the wavelet filter is the wavelet transform. In this process, the input
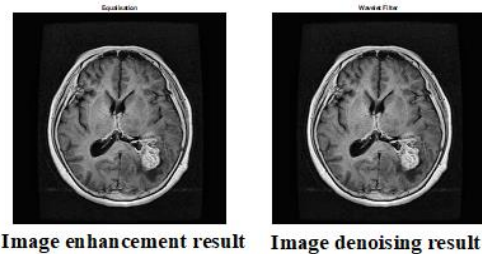
Figure. 1 Functional flow diagram of proposed brain tumor detection system

40

brain images are transformed into sparse representation to eradicate the noisy large-magnitude wavelet coefficients. Initially, a wavelet threshold is determined by analysing the pixels. Then the pixels' wavelet coefficients are compared with the threshold to remove the noisy pixels without degrading the image information. Fig. 3 shows the pre-processed results of the input Glioma image.

3.3 Segmentation using BCRF

The tumor segmentation is performed by the boosted crossbred random forests which is a combination of boosting [25] and crossbreeding [26] strategies in the standard random forests. Random forest is an ML algorithm that utilizes multiple decision trees to form a forest (group) to perform ensemble training. As the random forests are robust against the noise due to the random training process, their reliability is high. However, the random forests require numerous decision trees to perform the segmentation and the use of a reduced number of decision trees will reduce the generality and overall performance. Similarly, space and time complexity of random forests is high due to the complex architecture of numerous decision trees. For resolving these issues, boosting and crossbreeding strategies are incorporated. The boosting enhances the generality even with a small number of decision trees and also reduces the training time. Crossbreeding utilizes the best tree branches and crossbreeds them to attain simpler RF architecture with few best trees for increasing the performance of RF in space and time. Fig. 4 shows the initial stages in BCRF based segmentation on Glioma tumor image.

The BCRF involves the two training processes based on the sample weighting such that the training is performed at both when weights are updated and not updated. Initially, the re-processed images are assigned to the nodes of the decision trees in RF. Then the training image sample with local feature dimension d and class labels $y \in M$ are prepared. The training sample weights are initialized as $w =$
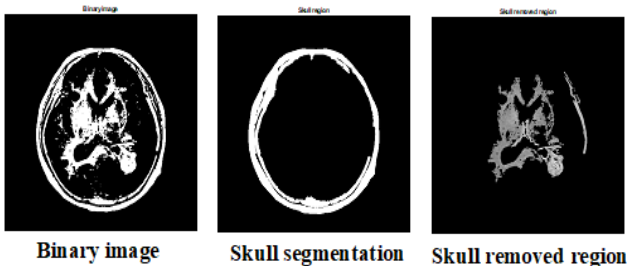

Binary image   **Skull segmentation**   **Skull removed region**

Figure. 4 Initial segmentation stages over Glioma image

$\frac{1}{N}$,where N demotes the nodes. Decision trees are built using sample sets through random sampling for the RF. The local features are extracted for segmentation based on local intensity and edge textures at different scales.

After the tree construction, the splitting function chooses the combinations of randomly generated features and thresholds that have the highest information gain. The information gain ΔG can be computed as

$$\Delta G = E(S_n) - \frac{|S_l|}{|S_n|} E(S_l) - \frac{|S_r|}{|S_n|} E(S_r) \tag{1}$$

here S denotes the training sample arrived at any node, S_n denotes the training sample at node n, S_l denotes the training sample at the left child node S_r denotes the training sample at right child node, and $E(S)$ represents the entropy. $E(S)$ can computed as

$$E(S) = -\sum_{j=1}^{M} P(C_j) \log P(C_j) \tag{2}$$

here M denotes the number of class and $P(C_j)$ is the probability of class C_j,which can be computed as

$$P(C_j) = \frac{\sum_{i \in S, y=c_j} w_i}{\sum_{i \in S} w_i} \tag{3}$$

here w_i is the sample weight of the i-th sample and y is the class label. From Eq. (1) to (3), it can be seen that the information gain is dependent on the priority of sample weights and $P(C_j)$. Thus, the depth of the decision tree is increased with a leaf node created whenever the recursive splitting is performed or the value of ΔG for a training sample becomes zero.

Then the boosting-based process is applied to determine the decision tree weight, α_t.

$$\alpha_t = \frac{1}{2} \log \frac{(M-1)(1-\varepsilon_t)}{\varepsilon_t} \tag{4}$$

here ε_t denotes the error rate of sample data in the decision tree. The expected rate of successful segmentation in RF is $1/M$. If the segmentation error goes beyond $1 - \frac{1}{M}$, the value of α becomes negative and the decision tree must be rejected. The error rate ε_t can be computed from the weights of wrongly segmented samples.

$$\varepsilon_t = \frac{\sum_{i=1;y_i \neq \hat{y}_i}^{N} w_i^{(t)}}{\sum_{i=1}^{N} w_i^{(t)}} \tag{5}$$

here y_i denotes the exact class labels, $\hat{y}_i$ denotes the estimated class labels and $w_i^{(t)}$ denotes the weights of the current step of processing. Based on these values, the weights of the incorrect segments in the next step can be estimated as

$$w_i^{(t+1)} = \begin{cases} w_i^{(t)} \exp(\alpha_t) & if\ y_i \neq \hat{y}_i \\ w_i^{(t)} \exp(-\alpha_t) & otherwise \end{cases} \quad (6)$$

here $\hat{y}_i$ is computed as $\hat{y}_i = \arg\max_C P_t(C|l)$. After the updating process, the weights are normalized to N.

Then the crossbreeding [26] is performed on the best branches of the boosted RF. In BCRF, each branch is defined as base learners and a group forecaster f is estimated

$$f(X) = \frac{1}{Z'}\sum_{z=1}^{Z} \begin{cases} b_z(x) & if\ b_z(x)\ process\ X \\ 0 & if\ b_z(x)\ not\ process\ X \end{cases} \quad (7)$$

here $Z = \{1,2,...,z\}$ denotes the number of branches in BCRF, Z' is the number of base learners that can process X and $b_z(x)$ is the base learner. When the grouping is interlinked, $f(X)$ becomes

$$f(X) = \begin{cases} \arg\max_{y \in \alpha} \sum_{z=1}^{Z} I(y = b_z(x)) & if\ y \in \alpha b_z\ to\ process\ X \\ g(X) & otherwise \end{cases} \quad (8)$$

here I is the group index and $g(X)$ is the decision function of the samples in nodes that are not under any branch. $g(X)$ can either be a random decision or a clustering-based decision based on the splitting process.

Generating new decision trees using boosting and crossbreeding, and updating the training sample weights are repeated to obtain T decision trees and T weighted trees for normalization. The segmentation process finishes with the estimation of the outputs $P_t(C|f(X))$ of the decision trees which are given as

$$P(C|f(X)) = \frac{1}{T}\sum_{t=1}^{N} \alpha_t P_t(C|f(X)) \quad (9)$$

The class which has the highest probability $\hat{y}$ is given as the output of the segmentation process

$$\hat{y} = \arg\max_C P(C|f(X)) \quad (10)$$

This process is repeated until the entire tumor regions of the images are segmented and then the global features will be extracted. Fig. 5 shows the Final segmentation results while the orientation varied segmentation results are shown in Fig. 6.
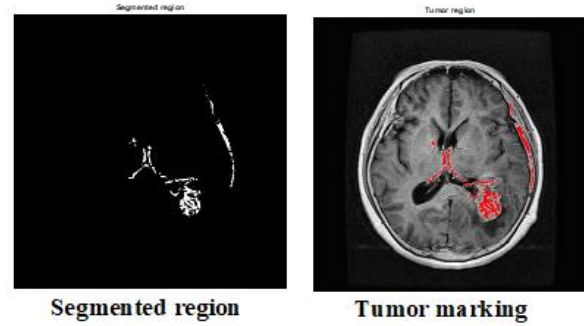

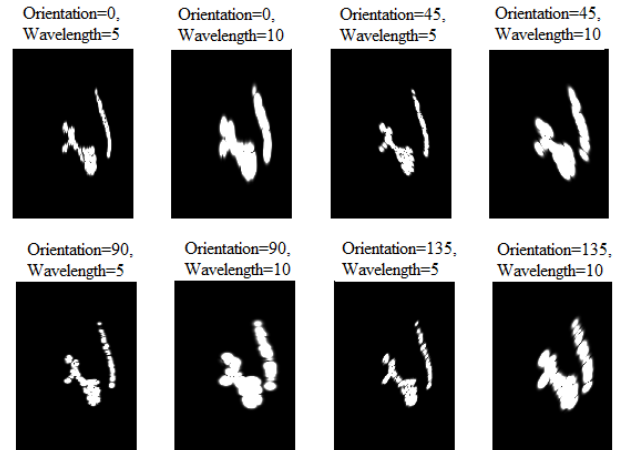Figure. 5 Final segmentation output of Glioma image

Figure. 6 Orientation varied segmentation results of Glioma

3.4 Feature extraction

Feature extraction is an important process in brain tumor classification. CNN algorithm can self-learn the features and select the informative features. However, in this study, CNN is used only for the selection and classification tasks since the extraction of features by separate processes will reduce the storage complexity of CNN. The spectral features and spatial features are the two main categories of the features. Among these features, spatial features are the commonly used features in image processing. For brain image processing, the spatial features are of two categories namely shape and texture features which are highly important in tumor classification. GLCM and Gabor filters are used to extract the texture features [27]. The GLCM method is highly time-consuming and hence the extraction of features might increase the overall processing time. Likewise, the GLCM provide poor features in the class border regions. Gabor filters can overcome these two limitations of GLCM; however, Gabor filters have a high number of redundant features. GLCM can tackle this problem. Hence the Gabor filter and GLCM are used together to compensate for the limitations of one another and provide highly informative features in less time.

3.5 Classification using COA-CNN

CNN is utilized in this proposed system for classification by optimizing the hyper-parameters using COA. The architecture, layers and parameters are the major components of the CNN. Convolution, pooling layer, and fully connected layers are the three main layers of the CNN with the additional optimizer, activation function, etc. In the convolutional layer, different kernels are used to convolve the input brain images into various feature maps. This layer will help in effectively reducing the number of parameters and improving the learning rate of correlation between the neighbor pixels. In the COA-CNN two stages of training are applied; the inputs are fed to the network in the feed-forward stage and the COA training to optimize the parameter values. The feed-forward stage feeds the input to the CNN and performs the dot product of the input and parameter vectors. Then the convolution operator is applied and finally, the output is computed along with the loss function. The size of the output is unchanged by applying zero-padding around the boundaries. From the valid padding and same padding methods, the same padding is used in this study for all the convolutional layers.

Initial weights are assigned properly to speed up the network convergence, and hence the 'He' initializer has been employed for setting the initial weights for the CNN layers. Then the Convolution Layer (CL) analyses the desired features and performs the convolution operation by acquiring the aspects by kernel functions. The outcome of the CL will be the convolved aspect plot. The kernel points are updated automatically based on the optimal structure configuration. After the convolution operation, the additional nonlinear function is used before the creation of feature maps to compromise the negative points of the aspects. The NLA can either be tanh, sigmoid or Rectified Linear Unit (ReLU). In most cases, the sigmoid or ReLU provided better performance; so, the ReLU is used in this study.

$$ReLU\,(x) = \begin{cases} x & if\ x \geq 0 \\ 0 & if\ x < 0 \end{cases} \qquad (11)$$

The pooling layer (PL) will be present next to the CL and it is used to minimize the size of the feature maps. It also minimizes the number of parameters so that the computational costs. In this study, a max-pooling layer is used with filters of size 2×2. Then the dropout method is used to regularize the network with lesser computational

costs. It is applied in the fully-connected Layer (FCL) to remove some nodes to prevent over-fitting. FCL employs a softmax function in the output layer for labeling the data into classes using past learning knowledge. The output of the last FCL is returned as the output of the CNN. The output is better when the loss function is minimized. The cross-entropy function (H) is used as the loss function in this study which can be computed for two distributions q_1 and q_2 as

$$H(q_1, q_2) = \sum_x q_1(x) \ln(q_2(x)) \qquad (12)$$

here $q_2(x)$ is the estimate of the true distribution of $q_1(x)$. Combining all these operators, a CNN is formed. For training the CNN, the learning rate is optimized using Adam optimizer in this study.

COA [28] is based on the intelligent characteristics of chimps in their group hunting. In a chimp group, all individuals are not similar in terms of intelligence and behaviour but they perform the group tasks properly. Likewise, the chimps try to obtain meat chaotically so that they could get social favours such as grooming and sex. These two characteristics are vital in the formulation of the COA. COA converges fast and provides an effective optimal solution. The chimps are assigned as the different configurations of CNN and the chimp with

Table 2. Optimum hyper-parameter values for CNN using COA

Parameters	Range of parameters	Optimum Value
Input feature dimension	-	1x32x1
No. of CL	[1, 2, 3, 4]	1 {Stride [1 1]; Padding [0 0]}
Dropout rate	[0.1, 0.2, 0.3, 0.4, 0.5]	0.3
No. of max PL	[1, 2, 3, 4]	1 {Stride [2 2]; Padding [0 0]}
No. of FCL	[1, 2, 3, 4]	4
Number of filters	[16, 24, 32, 48, 64, 96, 128]	32, 64
Filter size	[3, 4, 5, 6, 7]	4, 4
Kernel size	[2, 3, 4, 5, 6, 7]	2*2, 3*3
Activation function	[SELU, ReLU, Leaky ReLU]	ReLU
Mini-Batch Size	[4, 8, 16, 32, 64]	64
Momentum	[0.80, 0.85, 0.9, 0.95]	0.9
Learning Rate	[0.0001, 0.0005, 0.001, 0.005]	0.001
L2 Regularization	[0.0001, 0.0005, 0.001, 0.005]	0.0001

the best fitness is selected. The learning error rate is used as the fitness function. The COA analyses and determines the optimal configuration of CNN.

Based on the COA, the CNN architecture is modified. While many configurations for the CNN are found by the tuning process, COA selects the configuration with a minimum error rate. The CNN configuration obtained using COA are shown in table 2.

This configuration of CNN has less error rate which has been selected by the COA. This CNN can learn the spatial features from the brain images and classify the tumors more accurately.

4. Experimental results

The experiments are conducted over the described input brain tumor images to evaluate the performance of the proposed BCRF and COA-CNN based tumor detection system. The evaluations are performed in MATLAB R2016b with the necessary library files and toolboxes. The system configurations used for the implementation are Intel Pentium i5 processor with 4 GB RAM and 1 TB ROM storage with Windows 10 operating system. The performance of the proposed model is evaluated in terms of accuracy, precision, recall, f-measure, specificity, true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). The training and testing time is also estimated for the input dataset. The proposed COA-CNN model has achieved higher accuracy and reduced error rate which is shown in Fig. 7 and Fig. 8.

Classification accuracy is above 95 % for a maximum of 10 iterations while the error rate less than 5 %.

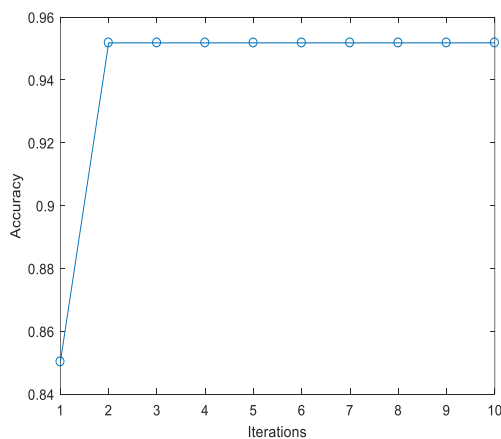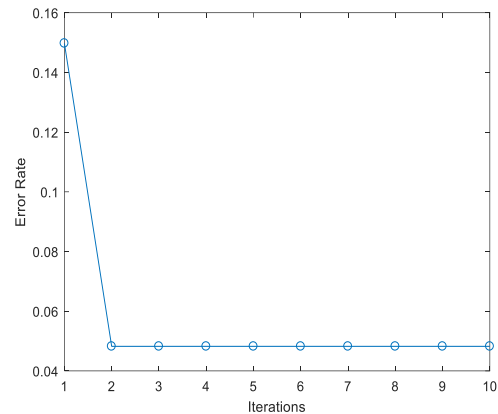Fig. 9 shows the AUC-ROC curve of the proposed BCRF and COA-CNN model.
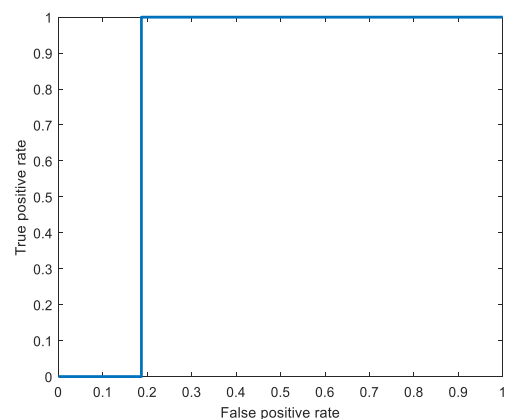

Figure. 8 Error rate of COA-CNN

Figure. 9 AUC-C curve of BCRF and COA-CNN classifier

Table 3 shows the classification results in which it is obtained that the proposed model achieved higher accuracy for both training and testing. The number of correctly classified images is 3237 from a total of 3264 input images. The manual calculation shows that about 100 % of images were classified correctly in No tumor, Glioma and Meningioma classes from the training set. Similarly, the no tumor and Meningioma class images were perfectly classified in the testing set. These results indicate that the proposed system is 100 % effective in classifying the tumor and non-tumor images.

Table 4 shows the performance metrics results achieved for the proposed BCRF and COA-CNN system for tumor detection. The proposed system achieved an accuracy of 98.94 % for the no tumor class, 95.68 % for Glioma, 92.89 % for Meningioma and 92.64 % for Pituitary tumor classes.

The overall accuracy of the proposed system is 95.18 % which is a sufficiently good performance. Likewise, the false positive and false negative values are also very less, thus indicating the effectiveness of the proposed system. The proposed system also achieved less processing time. It

Figure. 7 Accuracy of COA-CNN

Table 3. Classification results

Classes	Number of images			Number of correctly classified images		
	Training	Testing	Total	Training	Testing	Total
No tumor	395	105	500	395	105	500
Glioma	826	100	926	826	91	917
Meningioma	822	115	937	822	115	937
Pituitary	827	74	901	819	64	891
Total	**2870**	**394**	**3264**	**2862**	**375**	**3237**

Table 4. Performance results of BCRF and COA-CNN based system

Classes	Total	TP	TN	FP	FN	Accuracy (%)	Precision (%)	Recall (%)	F-measure (%)	Specificity
No tumor	105	105	285	4	0	98.94	92.92	100	94.34	0. 8651
Glioma	100	91	286	8	9	95.68	90.20	81	90.60	0.9320
Meningioma	115	115	251	18	10	92.89	84.68	100	85.5	0.8602
Pituitary	74	64	301	9	20	92.64	88.97	82.97	87.47	0.9406
Overall	394	375		9	10	95.18	89.44	90.75	90.61	0.8995

Table 5. Performance comparison against existing tumor detection systems

Methods	Accuracy (%)	FPR (%)	Total test images	Correctly classified images	Accuracy (%)			
					No tumor	Glioma	Meningioma	Pituitary
AdaBoost [9]	89.4	11	394	352	93.4	89.7	88.1	86.4
SVM [10]	95	3.8	394	374	98.4	95.25	93.8	92.55
TFE-SVM [11]	94.2	6.67	394	371	95.7	95.15	93.89	92.06
HGWO-ANN [12]	94.45	8.5	394	372	97.25	95.7	92.8	92.05
CNN [13]	94.6	5.0	394	373	96.9	95.45	93.5	92.55
MCNN [14]	81	17	394	320	84.89	82.4	80.69	76.02
DenseNet-LSTM [15]	92.13	5.67	394	363	93.67	93.5	91.56	89.79
CNN-KELM [16]	93.68	6.2	394	369	95.75	94.6	92.0	92.37
ECNN [17]	92	7.66	394	362	94.8	92.4	90.6	90.2
GA-CNN [18]	94.2	5.8	394	371	97.55	95.1	92.11	92.04
CNN [19]	94.6	3.4	394	373	97.99	95.1	91.85	90.46
GSO-DCNN [20]	92.66	5.13	394	365	96.86	95.12	89.6	89.06
GAN [21]	94.7	4.75	394	373	97.33	95.6	92.5	93.37
BCRF & COA-CNN	95.18	2.284	394	375	98.94	95.68	92.89	92.64

consumed an average of 54.5917 seconds for training and 2.6275 seconds for the testing process.

Table 5 shows the results of the accuracy, false positive rate, total tested images, and correctly classified images of the proposed BCRF and COA-CNN based tumor detection system compared against some of the popular existing state-of-the-art methods from literature. The popular methods in literature were implemented in our experiments along with the proposed system with the 3264 BRATS 2018 MRI brain images described in Table 1 obtained from [22]. The evaluations were performed in MATLAB R2016b with the necessary library files and toolboxes. The experimental settings include the maximum number of iterations (1000), time-out limit (1000 seconds) and adjusted size of images (512 x 512). With these common setting, the existing methods and proposed method are trained using 2870 images and tested using 394

images in this study.

The comparison of the proposed BCRF and COA-CNN against the existing methods in the literature also showed that the proposed system has outperformed the other methods with high accuracy and fewer FPR values. This clearly shows that the proposed system is efficient for brain tumor type classification.

5. Conclusion

This paper presented an efficient brain tumor detection system from the MRI data using hybrid BCRF segmentation and COA-CNN based classifier. In this proposed system, the pre-processing and feature extraction stages are also performed to improve detection accuracy. The evaluation results showed that the proposed system achieved high accuracy of 95.18 %, less FPR of 2.284 % and less

45

training time of 54.5917 seconds. Thus, the proposed system has also outperformed the other popular existing models. In future, the proposed BCRF and COA-CNN based model can be used to detect the grades of the brain tumor. Also, the individual class detection accuracy will be increased along with a possible reduction of the initial training time using parallel processing methods.

Conflicts of interest

The authors declared no conflict of interest.

Author contributions

The paper conceptualization, methodology, software, formal analysis, and initial manuscript preparation have been done by Omar Abdullah Murshed Farhan Alnaggar. Writing—review and editing have been done by Omar Abdullah Murshed Farhan Alnaggar, and Basavaraj Ningappa Jagadale. Validation have been done by Omar Abdullah Murshed Farhan Alnaggar, Basavaraj Ningappa Jagadale, and Swaroopa Hebbar Narayan.

References

[1] M. L. Bondy, M. E. Scheurer, B. Malmer, J. S. B. Sloan, F. G. Davis, D. I. yasova, C. Kruchko, B. J. McCarthy, P. Rajaraman, J. A. Schwartzbaum, S. Sadetzki, B. Schlehofer, T. Tihan, J. L. Wiemels, M. Wrensch, and P. A. Buffler, "Brain tumor epidemiology: Consensus from the Brain Tumor Epidemiology Consortium", *Cancer*, Vol. 113, No. 7, pp. 1953-1968, 2008.

[2] M. C. Mabray, R. F. Barajas, and S. Cha, "Modern Brain Tumor Imaging", *Brain Tumor research and treatment*, Vol. 3, No. 1, pp. 8-23, 2015.

[3] N. Gordillo, E. Montseny, and P. Sobrevilla, "State of the art survey on MRI brain tumor segmentation", *Magnetic Resonance Imaging*, Vol. 31, No. 8, pp. 1426-1438, 2013.

[4] A. Wadhwa, A. Bhardwaj, and V. S. Verma, "A review on brain tumor segmentation of MRI images", *Magnetic Resonance Imaging*, Vol. 61, pp. 247-259, 2019.

[5] G. Mohan and M. M. Subashini, "MRI based medical image analysis: Survey on brain tumor grade classification", *Biomedical Signal Processing and Control*, Vol. 39, pp. 139-161, 2018.

[6] K. Muhammad, S. Khan, J. D. Ser, and V. H. C. D. Albuquerque, "Deep Learning for Multigrade Brain Tumor Classification in Smart Healthcare Systems: A Prospective Survey", *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 32, No. 2, pp. 507-522, 2020.

[7] M. Ghaffari, A. Sowmya, and R. Oliver, "Automated Brain Tumor Segmentation Using Multimodal Brain Scans: A Survey Based on Models Submitted to the BraTS 2012-2018 Challenges", *IEEE Reviews in Biomedical Engineering*, Vol. 13, pp. 156-168, 2019.

[8] Chahal, P. Kaur, S. Pandey, and S. Goel, "A survey on brain tumor detection techniques for MR images", *Multimedia Tools & Applications*, Vol. 79, pp. 21771-21814, 2020.

[9] A. Minz and C. Mahobiya, "MR image classification using adaboost for brain tumor type", In: *Proc. of IEEE 7th International Advance Computing Conference (IACC)*, pp. 701-705, 2017.

[10] K. K. Kumar, T. M. Devi, and S. Maheswaran, "An efficient method for brain tumor detection using texture features and SVM classifier in MR images", *Asian Pacific Journal of Cancer Prevention (APJCP)*, Vol. 19, No. 10, pp. 2789-2794, 2018.

[11] A. Gokulalakshmi, S. Karthik, N. Karthikeyan, and M. S. Kavitha, "ICM-BTD: improved classification model for brain tumor diagnosis using discrete wavelet transform-based feature extraction and SVM classifier", *Soft Computing*, Vol. 24, No. 24, pp. 18599-18609, 2020.

[12] H. M. Ahmed, B. A. B. Youssef, A. S. Elkorany, A. A. Saleeb, and F. A. E. Samie, "Hybrid gray wolf optimizer–artificial neural network classification approach for magnetic resonance brain images", *Applied optics*, Vol. 57, No. 7, pp. B25-B31, 2018.

[13] S. Pereira, A. Pinto, V. Alves, and C. A. Silva, "Brain Tumor Segmentation Using Convolutional Neural Networks in MRI Images", *IEEE Transactions on Medical Imaging*, Vol. 35, No. 5, pp. 1240-1251, 2016.

[14] L. Zhao and K. Jia, "Multiscale CNNs for Brain Tumor Segmentation and Diagnosis", *Computational and Mathematical Methods in Medicine*, Vol. 2016, 2016.

[15] Y. Zhou, Z. Li, H. Zhu, C. Chen, M. Gao, K. Xu, and J. Xu, "Holistic brain tumor screening and classification based on densenet and recurrent neural network", In: *Proc. of International MICCAI Brainlesion Workshop, Springer, Cham*, Vol. 11383 LNCS, pp. 208-217, 2018.

[16] A. Pashaei, H. Sajedi, and N. Jazayeri, "Brain tumor classification via convolutional neural network and extreme learning machines", In: *Proc. of IEEE 8th International conference on*

computer and knowledge engineering (ICCKE), pp. 314-319, 2018.

[17] M. M. Thaha, K. P. M. Kumar, B. S. Murugan, S. Dhanasekeran, P. Vijayakarthick, and A. S. Selvi, "Brain Tumor Segmentation Using Convolutional Neural Networks in MRI Images", *Journal of Medical Systems*, Vol. 43, No. 9, pp. 1-10, 2019.

[18] A. K. Anaraki, M. Ayati, and F. Kazemi, "Magnetic resonance imaging-based brain tumor grades classification and grading via convolutional neural networks and genetic algorithms", *Biocybernetics and Biomedical Engineering*, Vol. 39, No. 1, pp. 63-74, 2019.

[19] M. Sajjad, S. Khan, K. Muhammad, W. Wu, A. Ullah, and S. W. Baik, "Multi-grade brain tumor classification using deep CNN with extensive data augmentation", *Journal of Computational Science*, Vol. 30, pp. 174-182, 2019.

[20] E. Irmak, "Multi-Classification of Brain Tumor MRI Images Using Deep Convolutional Neural Network with Fully Optimized Framework", *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, Vol. 9, pp. 1-22, 2021.

[21] N. Ghassemi, A. Shoeibi, and M. Rouhani, "Deep neural network with generative adversarial networks pre-training for brain tumor classification based on MR images", *Biomedical Signal Processing and Control*, Vol. 57, p. 101678, 2020.

[22] Kaggle   Repository: https://www.kaggle.com/sartajbhuvaji/brain-tumor-classification-mri.

[23] H. Kaur and J. Rani, "MRI brain image enhancement using Histogram Equalization techniques", In: *Proc. of IEEE International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pp. 770-773, 2016.

[24] D. Gupta and M. Ahmad, "Brain MR image denoising based on wavelet transform", *International Journal of Advanced Technology and Engineering Exploration*, Vol. 5, No. 38, pp. 11-16, 2018.

[25] Y. Mishina, R. Murata, Y. Yamauchi, T. Yamashita, and H. Fujiyoshi, "Boosted random forest", *IEICE Transactions on Information and Systems*, Vol. 98, No. 9, pp. 1630-1636, 2015.

[26] A. Nadi, H. Moradi, and K. Taheri, "Crossbreeding in Random Forest", *arXiv preprint*, arXiv: 2101.08585, 2021.

[27] A. R. Mathew, P. B. Anto, and N. K. Thara, "Brain tumor segmentation and classification using DWT, Gabour wavelet and GLCM", In: *Proc. of IEEE International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, pp. 1744-1750, 2017.

[28] M. Khishe and M. R. Mosavi, "Chimp optimization algorithm", *Expert systems with applications*, Vol. 149, p. 113338, 2020.