Brain Tumor Segmentation and Identification Using Particle Imperialist Deep Convolutional Neural Network in MRI Images

Maahi Amit Khemchandani*, Shivajirao Manikrao Jadhav, B. R. Iyer

Dr. Babasaheb Ambedkar Technological University Vidyavihar, Lonere, Maharashtra 402103 (India)

Received 7 October 2020 | Accepted 24 December 2021 | Published 17 October 2022

Abstract

For the past few years, segmentation for medical applications using Magnetic Resonance (MR) images is concentrated. Segmentation of Brain tumors using MRI paves an effective platform to plan the treatment and diagnosis of tumors. Thus, segmentation is necessary to be improved, for a novel framework. The Particle Imperialist Deep Convolutional Neural Network (PI-Deep CNN) suggested framework is intended to address the problems with segmenting and categorizing the brain tumor. Using the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) Algorithm, the input MRI brain image is segmented, and then features are extracted using the Scatter Local Neighborhood Structure (SLNS) descriptor. Combining the scattering transform and the Local Neighborhood Structure (LNS) descriptor yields the proposed descriptor. A suggested Particle Imperialist algorithm-trained Deep CNN is then used to achieve the tumor-level classification. Different levels of the tumor are classified by the classifier, including Normal without tumor, Abnormal, Malignant tumor, and Non-malignant tumor. The cell is identified as a tumor cell and is subjected to additional diagnostics, with the exception of the normal cells that are tumor-free. The proposed method obtained a maximum accuracy of 0.965 during the experimentation utilizing the BRATS database and performance measures.

Keywords

Brain Tumor Segmentation, Deep Belief Network, LNS Descriptor, Scattering Transform, Tumor Level.

DOI: 10.9781/ijimai.2022.10.006

I. Introduction

The structure of the human brain is highly complex, which is bound inside the skull such that the diagnosis of the diseases becomes a hectic phenomenon. Brain tumors are abnormally growing clumps of brain cells that are visible [1]. Gliomas are a prevalent type of brain tumor that have significant fatality rates [2], [3]. After the diagnosis of Gliomas, the survival of patients is not more than 14 months [4]. A brain tumor diagnosis is made through surgery, radiotherapy, or a combination of all [5], [6], which is a hectic process. The tumors can be of two types benignor malignant [7]. For the benign tumor, the tumor mass does not affect the nearby healthy cells, and they are non-cancerous, whereas the malignant tumors develop as a cancerous mass leading to death when untreated [8], [1]. The modalities of X-ray, CT, Ultrasonography, and MRI are used to collect information on brain tumors so that clinicians can better understand the tumors' textural properties and choose the best course of treatment. The association of the manual analyzing and segmenting the brain tumor images can lead to computational segmentation and classification [9].

MRI offers a better imaging mechanism and assists in accessing the gliomas in such a way to obtain the MRI sequences that yield effective information [2], [6]. Using the diverse features of brain tumors, MRI renders viable information regarding the tumor of various tissues[10], [11]. With the advances in medical imaging technologies, tumor segmentation methods [12], [13] assure the automatic, repeatable, and accurate tumor segmentation algorithm rendering variability both in inter-and intra-rater, with difficulty to reproduce [14], [15]. There are numerous techniques for improving the accuracy and dependability of brain tumor segmentation in the literature [16]. Automatic segmentation [17] helps in better planning of the diagnostic followed by the surgical operations [18] because segmentation is a hectic and demanding component. The manual segmentation for planning in the clinical routines [19] results in human errors and it is time-consuming. Moreover, the radiologists require additional knowledge regarding the pathology and inflexible nature [20], [21] of the existing algorithms led to the real clinical practice because of the development of the non-reliable method using the ill-trained information and parameters that are inflexible in various datasets [10].

The segmentation method uses the brain tumor images for extracting the abnormal tissues for learning the shape and the growth with time. At the same time, tumor delineation using manual methods is a tedious process as a large number of incremental medical data is available without considering the time and visual perception of the health professionals. The above demerits are tackled using computational methods as they are highly accurate and faster [16]. For the last decade, several brain tumor segmentation methods, which are grouped into two categories, semi-automatic and automatic, are available. The first group of methods includes level-set models [22], active contour models [23], and Tumor Cut (TC) model [24] and these models

* Corresponding author.
E-mail address: maahiamitkhemchandani@gmail.com

operate independently without the need for humans, but the results are not satisfactory [15]. The automatic approaches for segmenting brain tumors make use of manually created characteristics, and the models incorporate the standard machine learning process [25]. An alternative approach to model the task-adapted feature understands the hierarchy based on the highly complex features straight from the in-domain data. On the other hand, deep neural networks [26] render better learning including feature hierarchies [25].

The paper's main aim engages in classifying the levels of the tumor using MRI brain images through a novel framework, and it consists of three steps. DBSCAN is initially used to segment the picture of the brain tumor. The SLNS descriptor, which combines the scattering transform and LNS descriptors, is then used to the feature extraction. The suggested PI Deep CNN, which uses feature vectors created using the histogram of the output from the SLNS descriptor and the grid-based shape features, is then used to advance the classification of brain tumors. The created descriptor effectively renders the extremely durable texture features for categorization. The classifier focuses at detecting cancers and its level is divided into four classes, such as Normal without tumor, Abnormal, Malignant tumor, and Non-malignant tumor. Deep CNN is optimally tuned using the proposed PI algorithm.

The contributions:

proposed classifier for the particle imperialist deep convolutional neural network (PI-deep CNN): The classifier's goal is to categorize the brain tumor using the features that were derived from the MRI input image segments in order to identify the different levels of the tumor. The suggested PI algorithm, which combines PSO and the Imperialist colony algorithm, is used to optimize the tuning of the DEEP CNN classifier.

Scatter Local Neighborhood Structure (SLNS) description that has been proposed: The suggested descriptor, SLNS, which combines the LNS and scattering transform descriptors, extracts texture information in a way that the robust features support accurate tumor categorization.

The remainder of this essay is structured as follows: The methods for segmenting and categorizing brain tumors that are currently used in the literature are reviewed in Section 2. The automatic segmentation utilizing DBSCAN for tumor-level categorization using a deep learning system is described in Section 3. The results and comments are explained in Section 4, and the paper is wrapped up in Section 5.

II. Motivation

The review of the existing methods for segmentation is demonstrated in this section that deliberates the need for the new model for effective segmentation. The methods developed by various authors and their demerits associated with tumor detection are discussed. Finally, the challenges of the methods are described laying out a smooth pathway to propose an effective brain tumor segmentation and classification method.

A. Related Works

This section is an overview of the literature on several techniques for MRI image segmentation of brain tumors. Based on segmentation of brain tumors, these research papers are selected and evaluated in accordance with the most recent years of publication. As a result, the following 8 research publications are chosen for reviews:

Convolutional Neural Networks (CNN) were utilized by Sergio Pereira et al. [6] to automatically segregate the brain tumors from the MRI images. Although the strategy reduced computing time, it performed poorly when there were several feature maps present. Elisee Ilunga-Mbuyamba and colleagues [16] created an alternative Active Contour Model (ACM) that is based on the Multi-population Cuckoo Search (MCSS) algorithm, which solved the segment's energy reduction issue. It rendered better accuracy with minimal computational times, but the method was more expensive than the existing technique. A total of five PCA algorithms were used by Irem Ersoz Kaya et al. [9] that aimed at dimensionality reduction but suffered from overfitting problems. Chaiyanan Sompong et al. [10] used a framework that improved brain tumor segmentation. The presence of the ambiguous tumor boundaries was tackled using the Gray-Level Co-Occurrence Matrix-based Cellular Automata (GLCM-CA) that transformed the original image as the target featured image, which is highly complex. Lubna Farhi et al. [1] designed an adaptive stochastic segmentation algorithm to model the energy-based stochastic segmentation that possessed a highly flexible topology with high speed and segmentation accuracy. It involved minimum iterations and less computation time with the help of the internal and external forces, but the invariability of the contour radius minimized the requirement of user intervention. A sparse representation-based algorithm developed by Yuhong Li et al. [15] offered better likelihood estimation but did not use deep learning and failed to render accurate results. Yuhong Li et al. [25] modeled an automatic method using the Deep Neural Networks (DNNs) that was fast and accurate in segments, but the presence of a large number of the outliers was a drawback. Elisee Ilunga-Mbuyamba et al. [27] used Localized Active Contour with Background Intensity Compensation (LACM-BIC) method for determining the abnormal tissue, but the assumption of the method regarding the tumor mass was a real failure.

B. Challenges

- GLCM-CA [10] offered better results inthe case of the regions of white matter that possess similar intensities of tumor region. Additionally, the patch weight distance based on the tumor voxels is exceedingly complicated to compute using the Improved Tumor Cut method (ITC).
- In an adaptive stochastic segmentation algorithm [1], the lack of the possibility to automatically change the contour radius insists on the need for human intervention. Moreover, the algorithm is inapplicable for a 3Dform of images.
- The existence of the invariability of the inter and intra intensities among the training and test data poses a huge challenge regarding the segmentation of the enhancing and core parts. Mainly these parts are not clear in the BRATS-HG0117 and HG0307 [15].
- The LACM-BIC framework [27] assumes that the images are initially grouped as tumor mass. In the case of the absence of the classified datasets, the segmentation results in inaccurate results.
- The segmentation is highly complex and time-consuming and does not focus on highly robust and accurate methods that need improving the computational speed in real-time applications.
- CNN [6] for segmenting the brain tumor suffered from the computational complexity and needed more training data.

III. Automatic Segmentation Using DBSCAN for Tumor-Level Classification Using Deep Learning Algorithm

The automatic classification of brain tumor reduces the time spend on the MRI images for segmenting and detecting the affected regions of the brain. Moreover, the detection of the affected region through traditional vision-based techniques is a tedious process. Therefore, the suggested method of PI-based Deep CNN is utilized to categorize the affected region utilizing the histogram and shape aspects of the segments, making it easier and more automated to detect brain tumors. The DBSCAN clustering technique is used to advance the segmentation, which groups the MRI brain image slices into tumor and non-tumor regions. Using the suggested grid-

based SLNS descriptor, which combines the effects of the scattering transform and the LNS descriptor, the characteristics of the segments are retrieved. The afflicted areas of the brain can be identified using Deep CNN, an artificial classifier that was developed using the Imperialist Competitive Algorithm. The architecture of the tumor-level classification is depicted in Fig. 1.

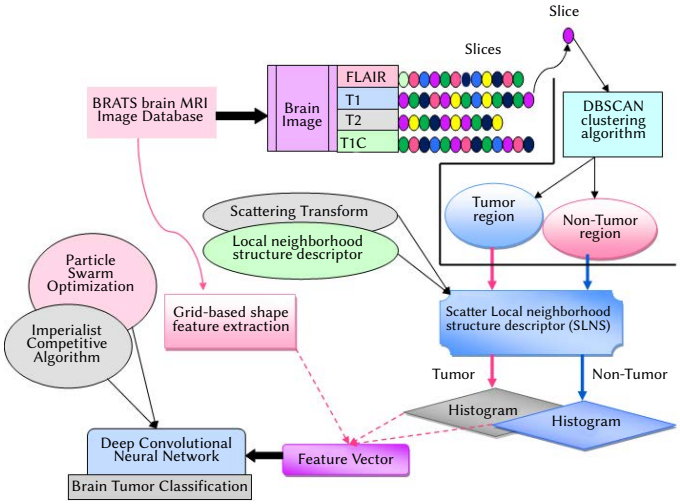


Fig. 1. Schematic diagram of the tumor classification.

For classification, MRI brain images are used that possess four modalities, FLAIR, T1, T2, and T1C, and let us denote the database as, D. Each modality possesses l_a slice and is subjected to segmentation using the DBSCAN clustering algorithm.

A. Formation of Segments Using DBSCAN Clustering

The goal of segmentation is to identify the affected regions in the MRI brain image using their modalities. The segments are generated from the image in such a way that these segments share identical properties, like contrast, color, boundary, and texture. The segmentation is done using the DBSCAN clustering algorithm [28], a density-based algorithm, which offers better segmentation accuracy through geometric constraints. The segmentation algorithm exhibits a faster generation of the super-pixels and renders improved boundary adherence in such a way that the pixels exhibiting homogeneous appearance (color and shape) form compact clusters. During clustering, two sets are initialized, candidate and unlabeled sets along with it.The top-down image pixel is named as the seed which is grouped with the labeled set. Thus, there are a total of three types of pixels, such as labeled, unlabeled, and seed. Initially, determine the four unlabeled pixels corresponding to the neighboring pixel of the labeledpixel such that the distance from the unlabelled pixel to the seed is computed. Whenever the distance is found to be less than the threshold, the unlabeled set is added to the candidate cluster that replaces the labeled set in such a way that the unlabelled pixel obtains the label of the seed. The stages are continued until the halting requirement, which is dependent on the total number of pixels and the empty set, is met. The threshold is calculated by dividing the image size by the total number of user-specified superpixels. Below is provided the DBSCAN clustering algorithm's pseudo-code.

The DBSCAN algorithm forms the segments using the input brain tumor image that is given in (1),

$$n = \{n_1, n_2, .n_3\} \tag{1}$$

where n signifies the segments that represent the tumor and the non-tumor region.

Algorithm for DBSCAN Clustering

DBSCAN clustering

Input: Image S, superpixel ρ, threshold T, labeled set L, Candidate set C
Output: Label of ρ, $L(\rho)$
Read the input image S
  Set the initial pixel as '0.'
  For a pixel with new label
  do
    Determine seed s such that $s \in L$
      While $L = \{\}$ or pixels in $\rho > T$ **do**
        For individual pixel i in L do
          For individual k pixel around i pixel do
  Compute the clustering distance $\partial_1^s(i,j)$ with seed s and pixel i
            If $\partial_1^s(i,j) < \delta$ then
              Set $k \in C$
            End if
          Enf For
        End For
          Set $L = C$
      End while
  End For

B. Extraction of the Features From the Segments Using the Proposed Scatter Local Neighborhood Structure Descriptor (SLNS)

Feature extraction ensures accurate and effective classification through the dimensionally-reduced significant features of the segments. The features used include the SLNS features and the shape features that are obtained using the proposed SLNS descriptor and the grid-based shape descriptor. The segments describe the tumor and the non-tumor region of the image in such a way that the texture features from both the regions are extracted separately using the proposed SLNS and grid-based descriptors.

The suggested SLNS descriptor, which combines the work of the scattering wavelet transform [29] and LNS descriptor [30], is used to extract features from the segments extracted using the DBSCAN clustering algorithm. The importance of SLNS lies in its ability to successfully extract textural information from the segments in order to facilitate the precise classification of the tumor level. Each of the segments of the image is subjected to the proposed SLNS descriptor, which yields the features by multiplying the output from scattering transform and LNS descriptor. The steps of the proposed SLNS descriptor are deliberated below.

Step 1: Application of scattering transform: The Morlet wavelet of different scales and orientation is extracted using the Scattering Transform (ST) [31] from separate segments to produce a highly resilient and locally invariant feature. To begin with, the segments' non-linear invariants are obtained using the modulus and average pooling functions. The ST steps are:

i) Preserving the image using local affine transformation: The local affine transformation, which is carried out by convoluting the individual image segments, and the low-pass filter, which functions based on the scaling factor, are used to safeguard the image against deformations. The components with high frequency are eliminated by the local affine transformation.

ii) Morlet filters for capturingthe high-frequency components: By averaging the coefficients generated from wavelet modulus, which are the unique results of the Morletor bandpass and the average filters, the high-frequency components are caught.

iii) Generation of the scattering coefficients: The wavelet modulus transformations are applied to the coefficients of wavelet modulus to produce the scattering coefficients. Convoluting the wavelet modulus coefficients with the generated wavelets of different scales and orientations yields the high-order scattering coefficients. Higher-order coefficients are required because they provide robustfeatures that are extremely stable and invariant locally. The features extracted from the segments using the scattering transform are denoted as, K^{scat}.

Step 2: Application of LNS descriptor:The segmented region is fed to the LNS descriptor [30] such that the individual pixel is thresholded using the neighborhood pixels of the image with the reference value. The reference value is fixed using the center pixel of the image and the mean absolute deviation computed using the difference in the neighboring pixels. The threshold is determined using equation (2).

$$H = G_c + \frac{1}{J}\sum_{r=0}^{J-1}\left|d_r - \frac{1}{J}\sum_{r=0}^{J-1}|d_r|\right| \tag{2}$$

$$d_r = \{G_r - G_c, r = (0,\ldots,J-1) \tag{3}$$

where G_n represents the gray values J that are spaced equally in the neighboring pixels and n vary between 0 and $J-1$.The gray value of the center pixel is denoted as G_c. d_r represents the difference in the local neighborhood and it is calculated using equation (3).

The mean absolute deviation offers robust features and renders the statistical features. The feature vector is obtained through the subtraction of the new threshold from the gray values in the circular neighborhood as given in (4).

$$W = \upsilon\left(G_0 - H, G_1 - H, \ldots, G_{J-1} - H\right) \tag{4}$$

where υ belongs to the joint distribution of J differences and W symbolizes the texture patterns obtained from the neighborhood pixels, whih is computed using (5). The representation is given in binary such that the gray values of the pixels exceeding the threshold are marked as '1' and the gray values lying below the threshold are filled as '0' as given in (6).

$$W = \upsilon\left(v[G_0 - H], v[G_1 - H], \ldots, v\left[G_{J-1} - H\right]\right) \tag{5}$$

$$v(x) = \begin{cases} 1 & ; x \geq 0 \\ 0 & ; x < 0 \end{cases} \tag{6}$$

where $v(x)$ specifies the sign function. To extract the pattern, the binomial 2^J is multiplied with the individual $v(x)$, and all the individuals are added together to represent the texture pattern. Thus, the LNS feature corresponding to each segment of the image is denoted as, K^{LNS}.

Step 3: SLNS feature generation: The third step is integrating the SLNS features using the proposed SLNS descriptor that is developed through the integration of the scattering features and LNS features. The generation of the SLNS features is described in (7).

$$K = \left[K^{Scat} \times K^{LNS}\right] \tag{7}$$

where K refers to the SLNS features. The features obtained from the ST and the LNS descriptor are denoted as, K^{Scat} and K^{LNS}, respectively.

Step 4: Histogram-based features using the SLNP-generated output:The histogram features of the SLNP are determined such that the histogram of the SLNP features belonging to the tumor and the non-tumor region enable the robust and accurate classification of the tumor level. The histogram features for the tumor segment and the non-tumor segment are denoted as p_1 and p_2.

3.2.2 Grid-based shape feature: The grid method is employed for extracting the shape features for which initially, the input image is subdivided with the grid lines. The image is scanned on the top-bottom and the left-right approach so that the grid space that covers the space partially or fully is assigned '1', whereas the grid space with no shape is filled with '0'. The shape feature is indicated as, g.

3.2.3 Feature vector: The feature vector represents the features obtained from the segments of the image, and is given in (8).

$$F = \{p_1, p_2, g\} \tag{8}$$

The dimension of the feature vector is denoted as, [1 x 128].

C. Brain Tumor Classification Using PI-Based Deep CNN Classifier

The Deep CNN classification module receives the extracted features from the segments, tumor and non-tumor, and the PI algorithm is used to optimize the classifier's weights. The PI algorithm, which is the modification of PSO [32] with the Imperial Competitive Algorithm [33] in such a way that the PI-Deep CNN classifies the input features corresponding to the input image to derive four classes. The classes include normal, abnormal, malignant, and non-malignant that are graded as four levels. The feature vector includes the histogram features of the tumor region, non-tumor region, and shape features of the input MRI brain image to facilitate the effective classification of the tumor level with better accuracy.

Comparing the PSO method to mathematical algorithms and other heuristic optimization techniques reveals that it is computationally efficient, simple to implement, robust to control factors, and has a straightforward idea.PSO follows the swarm behavior exhibiting easy implementation,and it requires only a few adjustable parameters compared with the Genetic Algorithm (GA). The particles in the search space are initialized randomly,and the optimal solution is derived through the frequent update of the solutions in the individual iterations. PSO follows the bird flocking behavior in such a way that the birds fly in groups in search of food randomly with certain velocities that direct the bird to fly to various locations. The optimal position of the particle is based on minimal error. The exploration abilities of PSO are high improving the optimal convergence faster. Moreover, PSO depends on intelligence and is applicable both in scientific and engineering applications. On the other hand, the Imperial Competitive Algorithm depends on the political-social developments of humans, and the competition among the imperials begins with the advent of the primary imperials. The decision is made based on the power such that the imperial with high power leaves the competition and the one with the highest power becomes the empire. The inclusion of the Imperial Competitive Algorithm increases the global optimum and minimal global solutions that assure the accuracy of classification.

a) Architecture of the Deep Convolutional Neural Network

Convolutional (Conv) layers, pooling (POOL) layers, and a fully connected (FC) layer make up the deep CNN [34], [35], whose structure is shown in Fig. 2. Each layer more accurately carries out its task of extracting the feature maps, sub-sampling, and classification. The pooling layer, which is used to generate the output maps in the FC layer, subsamples the feature maps created using the conv layer. The quantity of conv layers is used to improve classification accuracy by making the feature maps smaller when more conv layers are added.

Convolutional layers: Convolutional layers are the feature maps of neurons that extract patterns from the features of the input image. Using trainable weights, the neuron's receptive fields connect the neuron in the preceding layer to the neuron in the following convolutional layer. The trainable weights are used to convolute the input features and create a feature map, which is then applied to the subsequent layers using a non-linear activation function. Using the

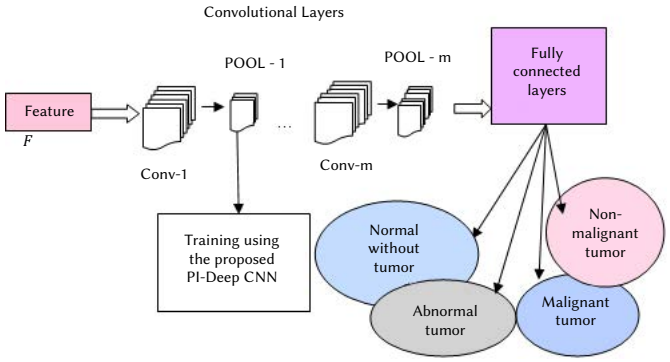


Fig. 2.The architecture of Deep CNN.

same neurons in a single feature map with varied weights in a single conv layer ensures the extraction of the variable features from diverse places. The feature vector serves as the input to the convolution layers of the deep CNN, and the convolution layers are represented in (9).

$$c = \{c_1, c_2, \ldots, c_j, \ldots, c_m\} \tag{9}$$

where m refers to the totalconv layers. The conv layers process the inputs and generate the output and the unit at (y, z) generates the output based on equation (10).

$$(V_u^j)_{y,z} = (b_u^j)_{y,z} + \sum_{q=1}^{f_1^{q-1}} \sum_{\upsilon=-w_1^j}^{w_1^j} \sum_{\kappa=-w_2^j}^{w_2^j} \left(\beta_{u,q}^j\right)_{\upsilon,\kappa} * \left(V_q^{j-1}\right)_{y+\upsilon,z+\kappa} \tag{10}$$

where, * denotes the convolutional operator for extracting the local patterns using the output of the alternate layers in conv layer, and $(V_u^j)_{y,z}$ symbolizes the fixed feature map. The feature maps from the previous conv layer f_1^{q-1} form the input to the j^{th} conv layer. The arbitrary filter $\left(\beta_{u,q}^j\right)_{\upsilon,\kappa}$ specifies the weights of j^{th} conv layer, which is trained using the proposed PI algorithm. The filter $\beta_{u,q}^j$ links the q^{th} feature map of $(j-1)^{th}$ conv layer and the u^{th} feature map in j^{th} conv layer. The bias matrix of the corresponding j^{th} conv layer is, b_u^j. ReLU is the activation function for eradicating the negative values toassist simplicity and effectiveness. The j^{th} non-linear layer is fed with the feature maps, and the output is given in (11).

$$V_u^j = fn\left(V_q^{j-1}\right) \tag{11}$$

where $fn(\,)$ denotes the activation function in convlayer j. The ReLU layer is significant because it allows Deep CNN to run faster and handle a large number of networks.

Pooling (POOL) layers

Complexity is reduced by the POOL layer, which includes neurons connecting the square-shaped space along its width and height of the preceding levels. The POOL layers perform a specified action even though they lack bias or training weights.

Fully connected layers: Abstract characteristics from the pooling and conv layers make up the input to the fully connected layer. The outcome of the fully connected layer can be seen in (12).

$$R_u^j = \delta(X_u^j) \text{ with } X_u^j = \sum_{q=1}^{f_1^{j-1}} \sum_{y=1}^{f_2^{j-1}} \sum_{z=1}^{f_3^{j-1}} \left(\varpi_{u,q,y,z}^j\right)\left(V_q^{j-1}\right)_{y,z} \tag{12}$$

where $\varpi_{u,q,y,z}^j$ refers to the weight that connects the unit (y, z) in the q^{th} feature map of the layer $(j-1)$ and u^{th} unit in the layer j.

b) Training Phase: Steps to Determine the Optimal Weights for Deep CNN

As shown in fig. 4, the algorithmic procedures for determining the best weights to train the deep CNN are explored further below.

a) Initialization: The swarm population is initialized in the first phase, and let's assume that there are a certain number of solutions, as shown in (13).

$$Z_a; (1 \le a \le N) \tag{13}$$

*b) Fitness Evaluation:*The fitness of the solutions is evaluated for individual iteration to choose the optimal solution. The fitness depends on the minimal error that is obtained by taking the square of the absolute difference of the estimated and the target output The MSE error is given in (14).

$$\varepsilon = \frac{1}{M}\sum_{e=1}^{M}|E - O|^2 \tag{14}$$

where E is the estimated output and O is classifier output. The total input samples are denoted as, M.

c) Computing the optimal position based on the minimum value of the error: The position of the particles using the minimal error. The position update follows either of the algorithms, namely the PI algorithm and Stochastic Gradient Descent (SGD) [36] algorithm. Whenever the error corresponding to the PI algorithm is less than the error corresponding to the SGD algorithm, the position of the particles is updated based on the PI algorithm, or else the position update follows the SGD algorithm. The position update is based on (15),

$$Z^{\tau+1} = \begin{cases} Z_{SGD}^{\tau+1} & ;i \quad f\varepsilon^{SGD} < \varepsilon^{PI} \\ Z_{PI}^{\tau+1} & ;O \quad therwise \end{cases} \tag{15}$$

where $Z_{SGD}^{\tau+1}$ refers to the position update based on SGD algorithm that is given in (16).

$$Z_{SGD}^{\tau+1} = \left(1 - \frac{1}{\tau}\right) Z^{\tau} + Y \tag{16}$$

where Y indicates the training sample and $Y \in (A, B)$. A specifies the input feature vector and B refers to the categories for the individual training sample. The position updated based on the PI algorithm is denoted as, $Z_{PI}^{\tau+1}$ and the derivation of the proposed equation follows as below:

i) Update equation for the PI algorithm: The optimal weights for training the deep CNNare derived using PI optimization. The position update of the particles is based on the PI algorithm and the update equation is obtained through modifying the PSO equation with the Imperialist Colony algorithm. The standard equation for the PSO algorithm is given in (17).

$$Z^{\tau+1} = Z^{\tau} + v^{\tau+1} \tag{17}$$

Substitute the velocity of the particle in (17), the position update is modified and it is given in (18).

$$Z^{\tau+1} = Z^{\tau} + \omega v^{\tau} + h_1 t_1 (P^{\tau} - Z^{\tau}) + h_2 t_2 (\Omega^{\tau} - Z^{\tau}) \tag{18}$$

where Z^{τ} refers to the position of the particle at a time τ. Ω^{τ} and P^{τ} are the global and the personal best solutions of the particles at iteration τ. h_1 and h_2 are the constants specifying the cognitive learning factor and acceleration factor, and t_1, t_2 are the random numbers. v^{τ} indicates the velocity of the particle at iteration τ and ω specifies the inertial weight. Equations (19), (20), and (21) are obtained by rearranging the equation (18), the position of the particle at iteration τ is derived based on the local and the global position of the particles as given in equation (21).

$$Z^{\tau+1} = Z^{\tau} + \omega v^{\tau} + h_1 t_1 P^{\tau} - h_1 t_1 Z^{\tau} + h_2 t_2 \Omega^{\tau} - h_2 t_2 Z^{\tau} \tag{19}$$

$$Z^{\tau+1} = Z^{\tau}[1 - h_1 t_1 - h_2 t_2] + \omega v^{\tau} + h_1 t_1 P^{\tau} + h_2 t_2 \Omega^{\tau} \tag{20}$$

$$Z^{\tau} = \frac{1}{[1 - h_1 t_1 - h_2 t_2]}\left[Z^{\tau+1} - \omega v^{\tau} - h_1 t_1 P^{\tau} - h_2 t_2 \Omega^{\tau}\right] \tag{21}$$

The standard equation of the Imperialist Colony algorithm is given as in equation (22) that depicts that the position at iteration $\tau + 1$ is based on the global best position and the position of the particle at τ.

$$Z^{\tau+1} = \Omega^{\tau} + 2\gamma\varphi Z^{\tau} \tag{22}$$

where γ denote the constant and its value exceeds 1 and ϕ denotes the distance among the colony and imperialist countries. Ω^{τ} signifies the global best solution and $Z^{\tau+1}$ is the position at the time $\tau+1$. Substituting equation (21) in equation (22), the equations (23) to (27) are obtained.

$$Z^{\tau+1} = \Omega^{\tau} + \frac{2\gamma\varphi}{[1-h_1t_1-h_2t_2]}\left[Z^{\tau+1} - \omega v^{\tau} - h_1t_1P^{\tau} - h_2t_2\Omega^{\tau}\right] \quad (23)$$

$$Z^{\tau+1} = \Omega^{\tau} + \frac{2\gamma\varphi\times Z^{\tau+1}}{[1-h_1t_1-h_2t_2]} - \frac{\omega v^{\tau}\times 2\gamma\varphi}{[1-h_1t_1-h_2t_2]} - \frac{h_1t_1P^{\tau}\times 2\gamma\varphi}{[1-h_1t_1-h_2t_2]} - \frac{h_2t_2\Omega^{\tau}\times 2\gamma\varphi}{[1-h_1t_1-h_2t_2]} \quad (24)$$

$$Z^{\tau+1} - \frac{2\gamma\varphi\times Z^{\tau+1}}{[1-h_1t_1-h_2t_2]} = \Omega^{\tau} - \frac{2\gamma\varphi}{[1-h_1t_1-h_2t_2]}\{\omega v^{\tau} + h_1t_1P^{\tau} + h_2t_2\Omega^{\tau}\} \quad (25)$$

$$Z^{\tau+1}\left[\frac{1-h_1t_1-h_2t_2-2\gamma\varphi}{1-h_1t_1-h_2t_2}\right] = \Omega^{\tau} - \frac{2\gamma\varphi}{[1-h_1t_1-h_2t_2]}\{\omega v^{\tau} + h_1t_1P^{\tau} + h_2t_2\Omega^{\tau}\} \quad (26)$$

$$Z^{\tau+1} = \frac{1-h_1t_1-h_2t_2}{1-h_1t_1-h_2t_2-2\gamma\varphi}\left\{\Omega^{\tau} - \frac{2\gamma\varphi}{[1-h_1t_1-h_2t_2]}\{\omega v^{\tau} + h_1t_1P^{\tau} + h_2t_2\Omega^{\tau}\}\right\} \quad (27)$$

The equation (27) makes it abundantly evident that the particle's position in the current iteration is dependent on its location, individual best, and overall best position in the previous iteration.

d) Stopping criterion: Updating positions is done repeatedly until the best overall solution is found.

Algorithm for deriving optimal weights to tune deep CNN

Optimal weight formulation for deep CNN

1 **Input**: Swarm Population Z_a; $(1 \le a \le N)$
2 **Output**: Best position of the particle $Z^{\tau+1}$
3 Swarm Initialization
4 Evaluation of the fitness ε
5 #Position Update
6 *If* $(\varepsilon^{SGD} < \varepsilon^{PI})$
7 {
8 Update the position of the particle using equation

$$Z_{SGD}^{\tau+1} = \left(1 - \frac{1}{\tau}\right) Z^{\tau} + Y$$

9 Else
10 Update the position based on equation

$$Z^{\tau+1} = \frac{1-h_1t_1-h_2t_2}{1-h_1t_1-h_2t_2-2\gamma\varphi}\left\{\Omega^{\tau} - \frac{2\gamma\varphi}{[1-h_1t_1-h_2t_2]}\{\omega v^{\tau} + h_1t_1P^{\tau} + h_2t_2\Omega^{\tau}\}\right\}$$

11 }
12 Repeat steps 4 to 11
13 End

IV. Results and Discussion

In order to demonstrate the efficiency of the suggested approach in estimating tumor levels, the section in question deliberates on the findings of the produced method when compared to the existing methods.

A. Setup Used

The PI-Deep CNN is implemented using MATLAB software operating on PC with Windows 8OS. The BRATS database, where the MRI images specific to each patient are maintained and there are four different modalities, including T1, T2, T1C, and FLAIR for each patient, is used to test and assess the approaches [37]. It also comprises feature and textural patterns, such as co-occurrence matrices, specified block sizes, mean and variance of slice or radial distance, etc. Performance is considerably improved. For every segmentation task, fixed groupings of algorithmic segmentations consistently outperformed the best individual segmentation algorithm. Online evaluation serves as the BRATS benchmark's main component.

B. Segmentation Output

The sample results are demonstrated in Fig.3 and Fig. 4, respectively that show the results obtained using the SLNS descriptor. Fig.3depicts the segmentation output of the SLNS descriptor using image 1. Slice 76 and 106 of the input image 1are depicted in fig.3 a) and fig.3 c), respectively. The segmented output for slice 76 and slice 106 are demonstrated in fig.3 b) and fig.3 d), respectively.

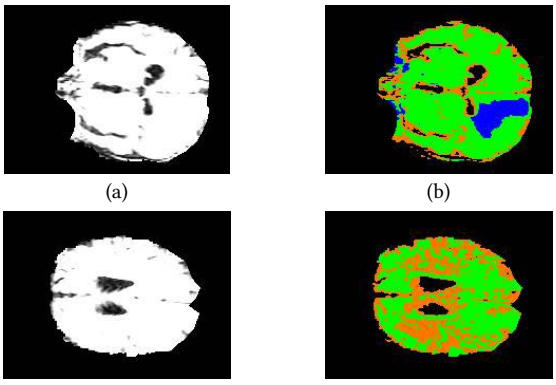


Fig. 3. Segmentation results using Input image 1 (a) Slice 76 of Input image 1, (b) Segmented output of (a), (c) Slice 106 of Input image 1, d) Segmented output of (c).

Fig.4 shows the segmentation results using image 2. The slices 116 and 81 of image 2are pictured in Fig.4 a) and Fig.4 c), respectively, whereas the segmented output of the slices is pictured in Fig.4b) and Fig. 4 d), respectively.

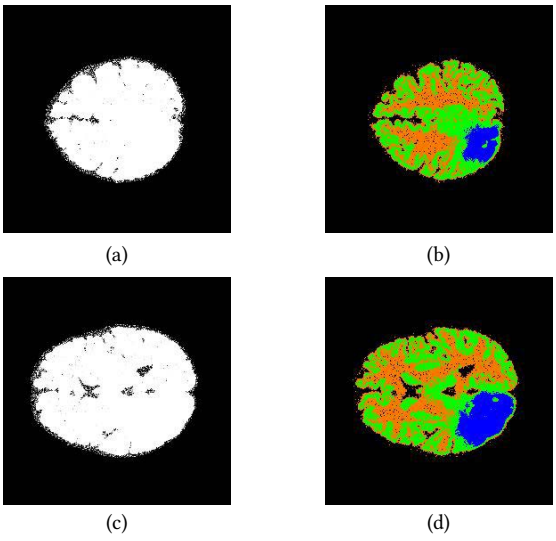
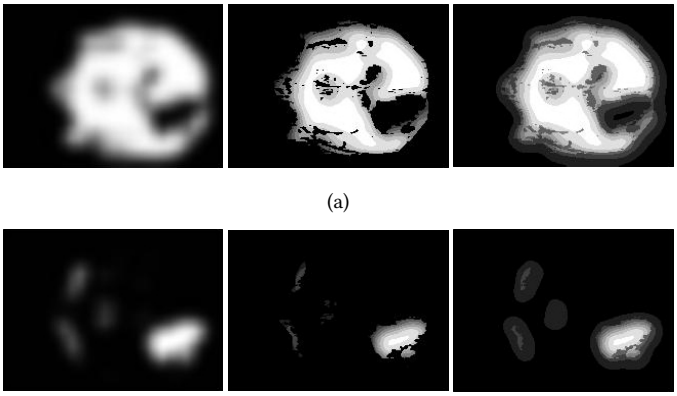


Fig. 4. Segmentation results using Input image 2 a) Slice 116 of Input image 2, b) Segmented output of (a), c) Slice 81 of Input image 2, d) Segmented output of (c).

The tumor and non-tumor zones that the proposed SLNS descriptor identified are shown in FIGS. 5A and 5B, respectively. To guarantee the classification accuracy of the classifiers, the input MRI image's tumor and non-tumor regions are subjected to feature extraction.



(a)

(b)

Fig. 5.Results of proposed SLNS descriptor a) Tumor region b) Non-tumor region.

C. Competing Methods

In order to demonstrate the effectiveness of the proposed method, it is compared with a number of other methods, including Convolutional Neural Networks (CNN) [6], PSSW+MCSS [16], Principle Component Analysis (PCA) [9], GLCM-CA [10], Markov-random field [15], Deep Neural Networks (DBN) [25], KNN [38], and NN [39].

D. Comparative Analysis

Sensitivity, specificity, accuracy, and ROC are used in a comparative examination of the suggested classifier's performance using the BRATS database. The suggested classifier is contrasted with techniques like KNN, NN, PCA, DBN, CNN, and PI-deep CNN in order to evaluate the performance effectively.

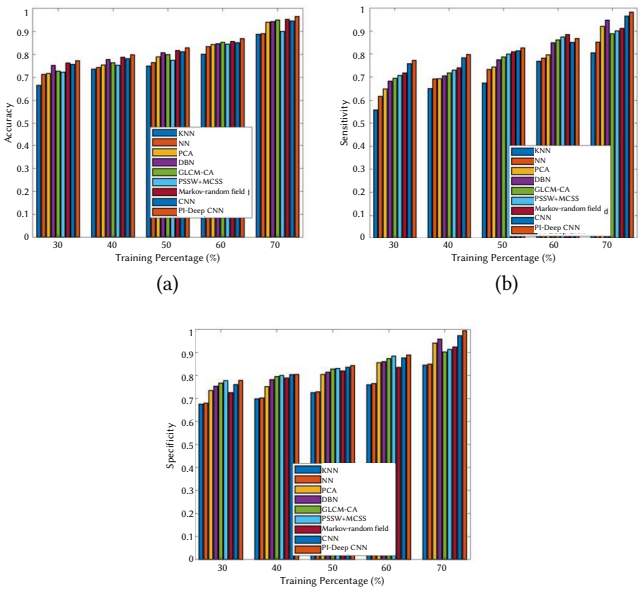
a) Analysis Using Image 1

Figures 6.a, 6.c, and 6.b, respectively, show the analysis of accuracy, specificity, and sensitivity using image 1 for the training percentage. For a training percentage of 70%, the accuracy of the comparative methods KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep CNN is 0.8871%, 0.89%, 0.94%, 0.942%, 0.95%, 0.899%, 0.952%, 0.945%, and 0.965%, respectively. This demonstrates that the proposed method is more accurate than all of the methods. The suggested technique is superior to the existing methods, as shown by the sensitivity of KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep CNN, which is 0.8046%, 0.8533%, 0.9214%, 0.9470%, 0.8896%, 0.9019%, 0.9122%, 0.965%, and 0.9821%, respectively, for 70% training data. For 70% training data, the specificity of the competing methods—KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep CNN—is, respectively, 0.8448%, 0.85%, 0.9413%, 0.958%, 0.9025%, 0.9131%, 0.9245%, 0.9782%, and 0.9951%.

Figures 7.a, 7.b, and 7.c show the analysis using picture 1, respectively. According to the cross-fold validation 10 results, the accuracy of the comparative methods KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep CNN is 0.8845, 0.8873, 0.9371, 0.9391, 0.9471, 0.8945, 0.9491, 0.9421, and 0.9621, respectively. This demonstrates that the proposed method is more accurate than the existing methods. The cross-fold validation10 shows that the proposed method outperforms the existing methods in terms of sensitivity with sensitivity values of 0.7749, 0.8465, 0.9140, 0.9394, 0.8923, 0.9046, 0.9149, 0.9572, and 0.9742, respectively, for the comparative methods of KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep. For the cross-fold validation 10, the specificities of KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep CNN are, respectively, 0.8397, 0.85, 0.9356, 0.9522, 0.9125, 0.9261, 0.9397, 0.9679, and 0.9892. Thus, it has been established that PI-Deep CNN performs better in terms of sensitivity, specificity, and accuracy than the current approaches.

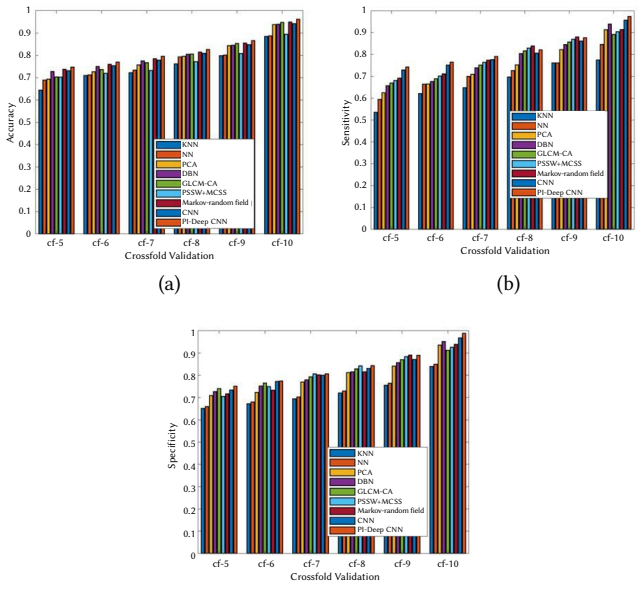
b) Analysis Using Image 2

The analysis based on the metrics, like accuracy, specificity, and sensitivity using image 2 based on the training percentage is depicted in Fig.8.a, Fig. 8.c, and Fig. 8.b,respectively. The accuracy of the comparative methods, KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep CNN, is 0.7670%, 0.89%, 0.94%, 0.9424%, 0.9374%, 0.9355%, 0.9305%, 0.945%, and 0.9469%, respectively, for 70% training data, which proves that the proposed method is superior over the existing methods in terms of accuracy.



(a) (b) (c)

Fig. 6. Comparative analysis based on training percentage using the image 1 a) accuracy b) sensitivity c) specificity



(a) (b) (c)

Fig. 7. Comparative analysis based on cross-fold using the image 1 a) accuracy b) sensitivity c) specificity.

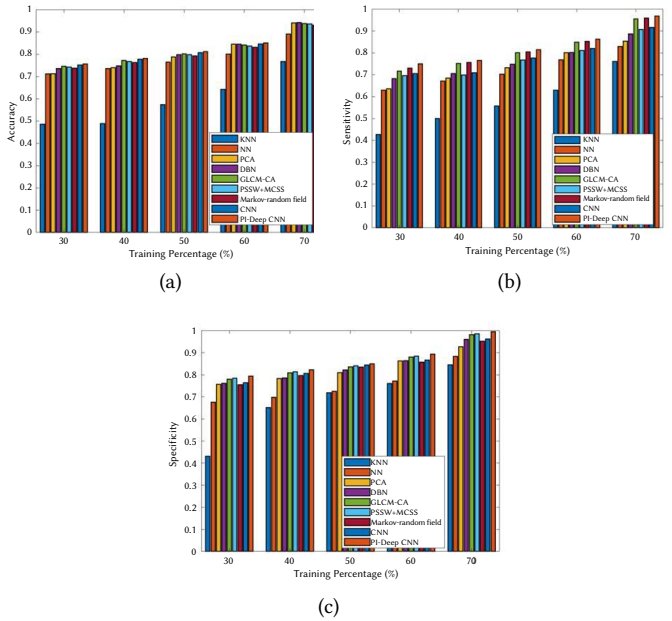


Fig. 8. Comparative analysis based on training percentage using the image 2 a) accuracy b) sensitivity c) specificity.

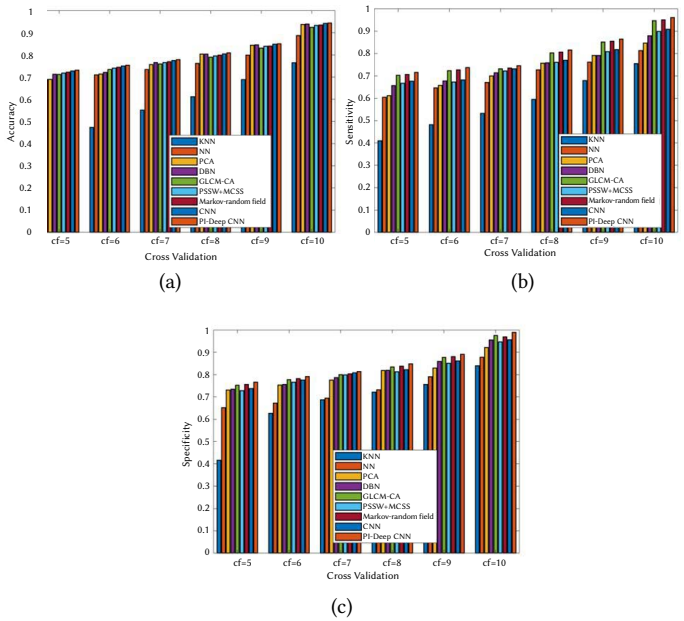


Fig. 9. Comparative analysis based on cross-fold using the image 2 a) accuracy b) sensitivity c) specificity.

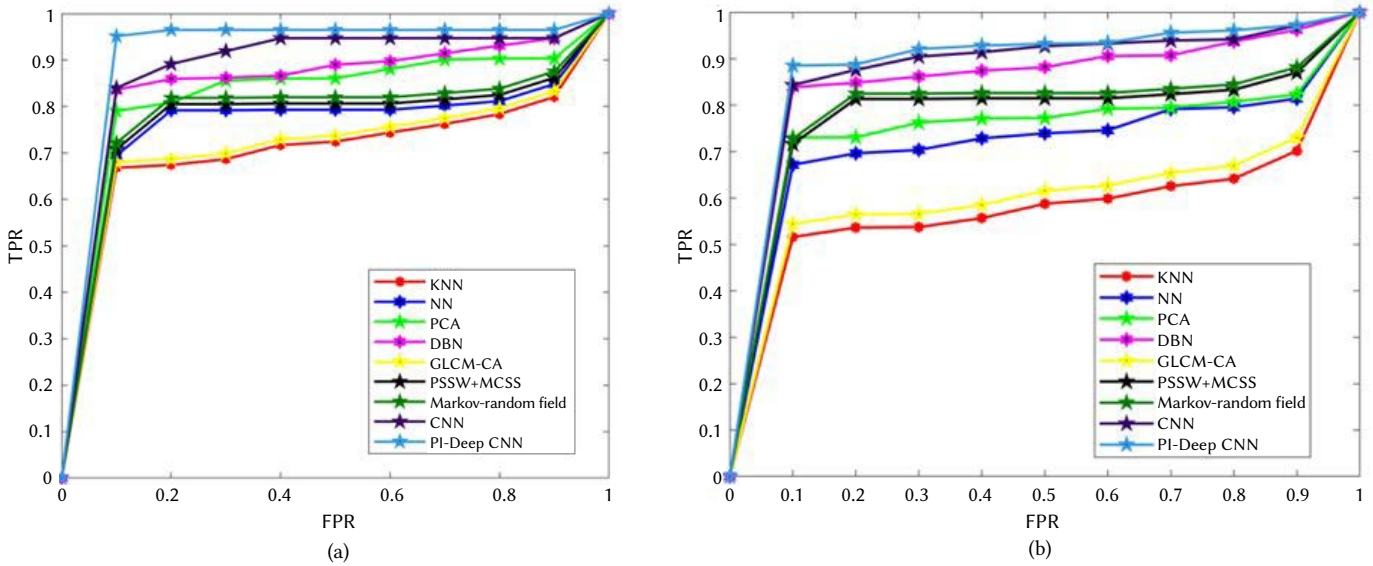


Fig. 10. ROC a) Image 1 b) Image2.

The sensitivity of the comparative methods, KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep CNN, is 0.7612%, 0.8291%, 0.8533%, 0.8866%, 0.9549%, 0.9066%, 0.9589%, 0.9161%, and 0.9684%, respectively, for 70% training data, which proves that the proposed method is superior over the existing methods in terms of sensitivity. The specificity of the comparative methods, KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep CNN, is 0.8448%, 0.8831%, 0.9271%, 0.9604%, 0.9816%, 0.9856%, 0.9523%, 0.9618%, and 0.9951%, respectively, for 70% training data, which proves that the proposed method is superior over the existing methods in terms of sensitivity.

Figures 9.a, 9.b, and 9.c, respectively, display the analysis of accuracy, sensitivity, and specificity utilizing image 2 based on the cross-fold validation. According to the cross-fold validation 10 results, the accuracy of the comparative methods KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep CNN is 0.7647, 0.8873, 0.9371, 0.9395, 0.9246, 0.9326, 0.9346, 0.9421, and 0.9441, respectively. This demonstrates that the proposed method is more accurate than the existing methods. The cross-fold validation 10 shows that the proposed method outperforms the existing methods in terms of sensitivity with sensitivity values of 0.7551, 0.8135, 0.8465, 0.8795, 0.9472, 0.8993, 0.9512, 0.9088, and 0.9607, respectively, for the comparative methods of KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep The KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and PI-Deep CNN comparative methods have specificities of 0.8397, 0.8778, 0.9216, 0.9546, 0.9757, 0.9466, 0.9685, 0.9561, and 0.9892, respectively, for the cross-fold validation 10; this demonstrates that the proposed method is more sensitive than the existing methods.

c) *Analysis of ROC*

The ROC curve is shown using images 1 and 2, respectively, in Figures 10.a and 10.b. The TPR for the minimal value 0.1 of FPR is 0.9514, as can be observed in Fig. 10.a, but for methods like KNN,

NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, and CNN, it is 0.6680, 0.6945, 0.7905, 0.8358, 0.6805, 0.7080, 0.7214, and 0.84. Similar to this, in fig.10.b, the TPR for the proposed approach is 0.8853 when the FPR of the PI-Deep CNN is at a minimum value of 0.1, but it is 0.5158, 0.6716, 0.7302, 0.8388, 0.5441, 0.7162, 0.7277, 0.8435, and 0.8435 for methods like KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, and CNN.

E. Analysis Using the Best Performance of the Comparative Methods

The analysis employing the tumor categorization techniques based on performance indicators is shown in Table I. For 70% training data, the accuracy of the methods KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, CNN, and suggested PI-Deep CNN, respectively, is 0.8871%, 0.89%, 0.945, 0.942%, 0.95%, 0.899%, 0.952%, 0.945%, and 0.965%. The accuracy of PI-Deep CNN is 0.9621 for the cross-fold 10, compared to 0.8845, 0.8873, 0.9371, 0.9391, 0.9471, 0.8945, 0.9491, 0.9421, and 0.9621 for KNN, NN, PCA, DBN, GLCM-CA, PSSW+MCSS, Markov-random field, and CNN, respectively. Table 1 makes it evident that the proposed method, which was based on cross-fold validation and the training percentage, obtained the highest levels of accuracy, sensitivity, and specificity.

TABLE I. Discussion of the Comparative Methods

		Training percentage =70		
		Sensitivity	**Accuracy**	**Specificity**
Based on training percentage	KNN	0.8046	0.8871	0.8448
	NN	0.8533	0.89	0.85
	PCA	0.9214	0.94	0.9413
	DBN	0.947	0.942	0.958
	GLCM-CA	0.889	0.95	0.9025
	PSSW+MCSS	0.901	0.899	0.9131
	Markov-random field	0.9122	0.952	0.9245
	CNN	0.965	0.945	0.9738
	PI-Deep CNN	0.9821	0.965	0.9951
Based on the Crossfold		**Crossfold validation =10**		
	Methods	**Sensitivity**	**Accuracy**	**Specificity**
	KNN	0.7749	0.8845	0.8397
	NN	0.8465	0.8873	0.85
	PCA	0.9140	0.9371	0.9356
	DBN	0.9394	0.9391	0.9522
	GLCM-CA	0.8923	0.9471	0.9125
	PSSW+MCSS	0.9046	0.8945	0.9261
	Markov-random field	0.9149	0.9491	0.9397
	CNN	0.9572	0.9421	0.9679
	PI-Deep CNN	0.9742	0.9621	0.9892

V. Conclusion

Using a novel brain tumor segmentation framework and MRI brain tumor images, the segmentation and classification of brain tumors is advanced. The main step that is necessary for the accurate classification of the level of brain tumor is segmentation. The DBSCAN Clustering Algorithm is used to partition the brain picture, creating clusters that are then subjected to feature extraction. The LNS descriptor and scattering transform are combined in SLNS, which captures textural characteristics for accurate tumor level categorization. The grid-based shape features, which are retrieved from both the tumor and non-tumor regions, make up the feature vector. The Deep Convolutional Neural Network classifier, which was trained using the suggested PI technique, is given the features. In order for the doctor to make an accurate diagnosis, the tumor class is finally determined using the classifier as Normal without tumor, Abnormal, Malignant tumor, and Non-malignant tumor. The proposed method obtained an accuracy of 0.965, a sensitivity of 0.9821, and a specificity of 0.9951, according to the experimentation utilizing the BRATS database.

References

[1] Lubna Farhi, Adeel Yusuf, and Rana Hammad Raza, "Adaptive stochastic segmentation via energy-convergence for brain tumor in MR images," Journal of Visual Communication and Image Representation, vol. 46, pp. 303-311, 2017.

[2] S. Bauer et al., "A survey of MRI-based medical image analysis for brain tumor studies," Phys. Med. Biol., vol. 58, no. 13, pp. 97–129, 2013.

[3] D. N. Louis et al., "The 2007 who classification of tumors of the central nervous system," Acta Neuropathologica, vol. 114, no. 2, pp. 97–109, 2007.

[4]]E. G. Van Meir et al., "Exciting new advances in neuro-oncology: The avenue to a cure for malignant glioma," CA, Cancer J. Clinicians, vol. 60, no. 3, pp. 166–193, 2010.

[5] G. Tabatabai et al., "Molecular diagnostics of gliomas: The clinical perspective," Acta Neuropathologica, vol. 120, no. 5, pp. 585–592, 2010.

[6] Sergio Pereira, Adriano Pinto, Victor Alves, and Carlos 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.

[7] P. John, et al., "Brain tumor classification using wavelet and texture based neural network," International Journal of Scientific & Engineering Research, vol. 3, no.l0, 2012.

[8] S. Charfi, R. Lahmyed, and Rangarajan, "A novel approach for brain tumor detection using neural network," International Journal of Research in Engineering and Technology, vol. 2, pp. 93-104.

[9] Irem Ersoz Kaya, Ayca Cakmak Pehlivanlı, Emine Gezmez Sekizkardes, and Turgay Ibrikci, "PCA based clustering for brain tumor segmentation of T1w MRI images," Computer Methods and Programs in Biomedicine, vol. 140, pp. 19-28, 2017.

[10] Chaiyanan Sompong, and Sartra Wongthanavasu, "An efficient brain tumor segmentation based on cellular automata and improved tumor-cut algorithm," Expert Systems with Applications, vol. 72, pp. 231-244, 2017.

[11] Cristina Utrilla Contreras, Nelson Mauricio, Buitrago Sánchez, Joachim Graessner, Pilar García Raya, and Begoña Marin Aguilera, "Difusion-Weighted MRI: from Brownian Motion to Head&Neck Tumor Characterization," International Journal of Interactive Multimedia and Artificial Intelligence, vol.4, no.5, pp.6-14, 2017.

[12] S. Bauer, R. Weiest, L. P. Nolte, and M. Reyes, "A survey of MRI based medical image analysis for brain tumor studies," Phy.Med. Biology, vol. 58, no. 13, pp. 97–129, 2013.

[13] Afsaneh Abdollahzadeh Rezaie and Ali Habiboghli, "Detection of Lung Nodules on Medical Images by the Use of Fractal Segmentation," International Journal of Interactive Multimedia and Artificial Intelligence, vol.4, no.5, pp.15-19, 2017.

[14] S. K. Warfield, K. H. Zou, and W. M. Wells, "Brain tumor segmentation from multispectral MRIs using sparse representation classification and Markov Random Field regularization," IEEE Trans. Med. Imaging, vol. 23, no. 7, pp. 903–921, 2004.

[15] Yuhong Li, Fucang Jia, and Jing Qin, "Brain tumor segmentation from multimodal magnetic resonance images via sparse representation," Artificial intelligence in medicine, vol. 73, pp. 1-13, 2016.

[16] Elisee Ilunga-Mbuyamba, Jorge Mario Cruz-Duarte, Juan Gabriel Avina-Cervantes, Carlos Rodrigo Correa-Cely, Dirk Lindner, and Claire Chalopin, "Active contours driven by Cuckoo Search strategy for brain tumor images segmentation," Expert Systems with Applications, vol. 56, pp. 59-68, 2016.

[17] Avinash Gopal, "Hybrid classifier: Brain Tumor Classification and Segmentation using Genetic-based Grey Wolf optimization," Multimedia Research, vol 3, No 2, 2020.

[18] Pierfrancesco Fusco, Vincenza Cofini, Emiliano Petrucci, Paolo Scimia, Giuseppe Paladini, Astrid U Behr, Fabio Gobbi, Tullio Pozone, Giorgio Danelli, Mauro Di Marco, Roberto Vicentini, Stefano Necozione, and

Franco Marinangeli, "Unilateral paravertebral block compared with subarachnoid anesthesia for the management of postoperative pain syndrome after inguinal herniorrhaphy a randomized controlled clinical trial," Pain, vol.157, no.5, pp.1105, 1113, 2016.

[19] MD Pierfrancesco Fusco, MD Vincenza Cofini, MD Emiliano Petrucci, MD Paolo Scimia, MD Tullio Pozone, MD Giuseppe Paladini, MD Gaspare Carta, and MD Stefano Necozione, "Transversus abdominis plane block in the management of acute postoperative pain syndrome after caesarean section: a randomized controlled clinical trial," Pain Physician, vol.19, pp.583-591, 2016.

[20] Nitin Deotale,Uttam Kolekar,Anuradha Kondelwar, "Self-adaptive Particle Swarm Optimization for Optimal Transmit Antenna Selection," Journal of Networking and Communication Systems, vol.3, no.1, pp.1-10,2020.

[21] Rupam Gupta Roy,Dibyendu Ghoshal, "Search and Rescue Optimization Algorithm - Second Order Sliding Mode Control: AUV Error Tracking," Journal of Computational Mechanics, Power System and Control, vol.3, no.1, pp.10-20,2020.

[22] T. Wang, I. Cheng, and A. Basu, "Fluid vector flow and application in brain tumor segmentation," IEEE Trans. Biomed. Eng., vol. 56, no .3, pp. 781–789, 2009.

[23] J. Sachdeva, V. Kumar, I. Gupta, N. Khandelwal, and C. K. Ahuja, "A novel content-based active model for brain tumor segmentation," Magn. Reson. Imaging, vol. 30, pp. 694–715, 2012.

[24] A. Hamamci, N. Kucuk, K. Karaman, K. Engin, and G. Unal, "Tumor-cut: segmenation of brain tumors on contrast enhanced MR images for radiosurgery applications," IEEE Trans. Med. Imaging, vol. 31, no. 3, pp. 790–804, 2012.

[25] Mohammad Havaei, Axel Davy, David Warde-Farley, Antoine Biard, Aaron Courville, Yoshua Bengio, Chris Pal, Pierre-Marc Jodoin, and Hugo Larochelle, "Brain tumor segmentation with deep neural networks," Medical image analysis, vol. 35, pp. 18-31, 2017.

[26] F. Heydarpour, E. Abbasi, M. J. Ebadi, and S. M. Karbassi, "Solving an Optimal Control Problem of Cancer Treatment by Artificial Neural Networks," International Journal of Interactive Multimedia and Artificial Intelligence, vol.6, no.4, pp.18-25, 2020.

[27] Elisee Ilunga-Mbuyamba, Juan Gabriel Avina–Cervantes, Arturo Garcia–Perez, Rene de Jesus Romero–Troncoso, Hugo Aguirre–Ramos, Ivan Cruz–Aceves, and Claire Chalopin, "Localized active contour model with background intensity compensation applied on automatic MR brain tumor segmentation," Neuro computing, vol. 220, pp. 84-97, 2017.

[28] J. Shen, X. Hao, Z. Liang, Y. Liu, W. Wang, and L. Shao, "Real-Time Superpixel Segmentation by DBSCAN Clustering Algorithm," in IEEE Transactions on Image Processing, vol. 25, no. 12, pp. 5933-5942, Dec. 2016.

[29] Prateekshit Pandey, Richa Singh, and Mayank Vatsa, " Face recognition using scattering wavelet under Illicit Drug Abuse variations," In International Conference on Biometrics (ICB), pp.1-6, 2016.

[30] J. Shen, X. Hao, Z. Liang, Y. Liu, W. Wang, and L. Shao, "Real-Time Superpixel Segmentation by DBSCAN Clustering Algorithm," in IEEE Transactions on Image Processing, vol. 25, no. 12, pp. 5933-5942, Dec. 2016.

[31] Prateekshit Pandey; Richa Singh; and Mayank Vatsa, "Face Recognition using ScatteringWavelet under Illicit Drug Abuse Variations," International Conference on Biometrics (ICB), 25 August 2016.

[32] Dian Palupi Rini, Siti Mariyam Shamsuddin, and Siti Sophiyati Yuhaniz, "Particle Swarm Optimization: Technique, System and Challenges," International Journal of Computer Applications, vol. 14, no. 1, pp. 0975 – 8887, 2011.

[33] E. Atashpaz-Gargari and C. Lucas, "Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition," in Proceedings of IEEE Congress on Evolutionary Computation, pp. 4661-4667, 2007.

[34] Giduthuri Sateesh Babu, Peilin Zhao, and Xiao-Li Li, " Deep Convolutional Neural Network Based Regression Approach for Estimation of Remaining Useful Life," International Conference on Database Systems for Advanced Applications (DASFAA), pp. 214-228, 2016.

[35] Dong, Chao, Chen Change Loy, Kaiming He, and Xiaoou Tang, "Image super-resolution using deep convolutional networks," IEEE transactions on pattern analysis and machine intelligence, vol. 38, no. 2, pp.295-307, 2016.

[36] Jing Yang and Guanci Yang, "Modified Convolutional Neural Network Based on Dropout and the Stochastic Gradient Descent Optimizer," vol.11, no.3, 2018.

[37] Bjoern H. Menze et al., "The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS)," IEEE Transactions on Medical Imaging, Vol. PP, no. 99, pp.1, 2014.

[38] V.Vani and M. Kalaiselvi Geetha, "Automatic Tumor Classification of Brain MRI Images," International Journal of Computer Sciences and Engineering, vol.4, no.10, 2016.

[39] N. V. S. Natteshan and J. Angel Arul Jothi, "Automatic Classification of Brain MRI Images Using SVM and Neural Network Classifiers," Advances in Intelligent Informatics, vol.320, pp.19-30, 2015.

Maahi Amit Khemchandani

Maahi Amit Khemchandani has completed Bachelor's Degree from North Maharashtra University, Master's Degree from Bharati Vidyapeeth College of Engineering, Mumbai University, Mumbai and Research Scholar at Dr. Babasaheb Ambedkar Technological University, Lonere . She is having more than 12 years of experience in teaching field. Her area of interest is in Networking and Machine Learning.

Shivajirao Manikrao Jadhav

Dr. Shivajirao Manikrao Jadhav has completed his P.hd from Dr. B.A.T.U, Lonere. He is now working as Professor and Head of Department of Information Technology at Dr. Babasaheb Ambedkar Technological University, Lonere. He is having teaching experience over 20 years and providing his valuable guidance to students in many research areas like Soft Computing, Machine Learning, Hybrid Neural Networks and Biomedical Signal Analysis. He has received UGC's Teacher Fellowship award under FIP. He has conducted many workshops. He is Chairman of Board of Studies at Dr. B.A.T.U, Lonere.

B. R. Iyer

Dr. B. R. Iyer has completed his P.hd from Indian Institute of Technology, Roorkee. He is now working as Assistant Professor in Electronics & Telecommunication Engineering Department at Dr. Babasaheb Ambedkar Technological University, Lonere. He is providing his valuable guidance to students in many research areas like NANO Photonics/RF Front End Design Pervasive Healthcare System Design and allied Signal/ Image Processing, Wireless Sensor Network Design / IoT Education & Technology. He has received Indian National Academy of Engineers (INAE) research fellowship-2015-2016. He has honoured with Best research paper award at IEEE MTTs IMaRC-2013 held in New Delhi during 14-16 December 2013. He is Reviewer of IETE Journal of Research, Int. Jr. of RF & Microwave CAD(Willey), IJSAEM (Springer). He is Member, Technical Program Committee of APACE-14, ICACCI-14, ICCME-15, MALSPI-15, CNTIA-15, ICTEC-15, MOBIAPPS-15, I4CT-15, ICCCA-15.