Journal of Advanced Zoology

ISSN: 0253-7214

Volume 44 Special Issue -02 Year 2023 Page 1010:1025

FSCSCOOT: Functional Calculus Competitive Swarm Coot Optimization-based CNN transfer learning for Parkinson's disease classification

Apparna Allada,
Research Scholar, Department of Computer Science and Engineering Faculty of Engineering and Technology
Annamalai University, Chidambaram, Tamilnadu
alladaaparna9@gmail.com
Dr. R. Bhavani,
Professor, CSE Annamalai University, Chidambaram, Tamilnadu
Dr. Kavitha Chaduvula,
Professor & HoD, Information Technology Seshadri Rao Gudlavalleru Engineering College, Gudlavalleru
Dr. R. Priya,
Professor, CSE Annamalai University, Chidambaram, Tamilnadu

Article History
Received: 27Aug 2023
Revised: 28Sept 2023
Accepted: 06Oct 2023

Abstract:
Parkinson's disease (PD) is a neurological disorder of the central nervous system that causes difficulty in movement, often including tremors and rigidity. Early detection of PD can prevent symptoms up to a certain age and increase life expectancy. For this purpose, we have used brain images from magnetic resonance imaging (MRI) technique. Generally dementia can be either classified as Alzheimer's or Parkinson's or sometimes may be due to tumor in brain. Therefore, effectual methods such as Competitive Swarm Coot Optimization_ Convolutional Neural Network (CSCOOT_CNN) with transfer learning and Fractional CSCOOT_ deep neuro-fuzzy network (FCSCOOT_DNFN are newly introduced for classification of brain diseases. At first, input images are acquired from particular datasets, and then input images are given to the pre-processing stage. In a pre-processing module, median filter is utilized for the elimination of noises. Afterward, pre-processed image is then subjected to feature extraction in which CNN features are extracted. In the level of classification, the images are classified into Parkinson by DNFN that is trained utilizing the introduced FCSCOOT algorithm. Furthermore, the FCSCOOT algorithm is newly designed by combination of Fractional Calculus (FC) with CSCOOT algorithm.

Keywords: Deep Neuro Fuzzy Network (DNFN), Competitive Swarm Optimizer (CSO), COOT optimizer, Fractional Calculus (FC).

1. Introduction

Parkinson's disease (PD) has a prevalence rate of 1% in the over-60 age group, and affects about 0–2 per 1000 people. It is the second most common brain disease after Alzheimer's disease [1]. A central nervous system disorder, especially those affecting the brain, causes the neurons to degenerate. A person suffering from this disease will experience tremors at rest, bradykinesia (slow movement), rigidity, sleep disturbances, asymmetry in posture, depression, and other such symptoms. In the advanced stages of the disease, PD dementia becomes coarse and patients have difficulty sleeping or concentrating. People with PD lose the nerve endings that produce dopamine, the prime chemical which controls most of the involuntary functions of the body. This might help explain some of the involuntary symptoms of PD, like tiredness, non-uniform blood pressure, reduced peristalsis, and a sudden drop in blood pressure.

PD appears hereditary in some cases, and certain mutations can be traced to it, but most of the time this disease is random. There is a growing consensus that it is caused by a combination of genetics and environmental factors, such as exposure to toxins. A loss of dopaminergic neurons in the substantianigra region of the brain is one of the leading causes of Parkinson's disease.

The crucial intention of this paper is to introduce effective technique FCSCOOT_DNFN for Parkinson'sclassification. At initial, input images are obtained from dataset collected from kaggle.com. After that, images are fed to pre-processing, where a median filter is utilized. A pre-processed image is then passed to feature extraction module in which89 CNN features are extracted. The extracted features are then fed to classification, where the disease is classified into Parkinson or not using the DNFN [22] [21]. Here, CNN is used with hyper parameters from trained model VGG19. In addition, CNN with transfer learning is tuned utilizing proposed FCSCOOT algorithm. The proposed FCSCOOT algorithm is newly designed by combining FC [25] with CSCOOT algorithm.

An essential contribution of this work is expounded below.

- **Proposed FCSCOOT_DNFN for classification of Parkinson's disease**: An efficacious technique is designed for classification of PD named FCSCOOT_DNFN. In this classification, the tuning of DNFN is performed by FCSCOOT. Moreover, FCSCOOT is newly planned by incorporating FC, CSO and COOT optimizer.

2. Motivation

This part elucidates about an overview of literature using collected research papers, in addition to their benefits and disadvantages. These limitations motive future researchers for designing new approaches for Parkinson's disease classification.

2.1 Literature Survey

The reviews done using the collected papers according to brain disease classification are detailed in this portion.AtifMehmood., *et al*. [1] developed Deep Siamese CNN for predicting the classification stages of Alzheimer's disease. However, the method provided deeper assessment for extracting helpful information from the slices of MRI. but it did not examine the model utilization on computer-aided diagnostic problems. Dr. Rachna Jain., *et al*. [2] presented architecture of CNN for the task of classification, which was capable to extract the helpful

features for classification chore but still failed to improve the total performance by means of fine-tuning. FarheenRamzan., *et al*. [3] developed residual neural networks to perform classification of disease. This technique helped in making decision for earliest diagnosis but only integration of clinical images with deep learning approaches was helpful for uncovering functional alterations patterns in brain corresponding to a development of Alzheimer's disease.SakeenaJavaid., *et al*. [4] introduced DNFN for an effectual cost and load optimization, which proved the optimizer robustness in cost as well as energy efficiency. This method utilized similar group of rules for identical parameters set and hence failed to enhance the performance.
SuriyaMurugan., *et al*. [5] developed DEMentiaNETwork (DEMNET) for detecting the stages of dementia from MRI. This developed model was capable to detect the brain portions related with Alzheimer's disease, but still the utilized data was not adequate and hence failed to manage the computational complications.Hadeer A. Helaly., *et al*. [6] introduced an end-to-end AD early detection and classification (E^2AD^2C) for classification of medical image and detection of Alzheimer's disease, which reduced the computational complications, memory necessities, over-fitting and handling time. This method did not applied MRI segmentation for emphasizing the features of Alzheimer's before classifications.Ravi ChandaranSuganthe., *et al*. [7] constructed Deep Convolution Neural Network (DCNN) and VGG-16 inspired CNN (VCNN) methods for classification of AD from MRI images, but still it did not consider axial as well as coronal views. Ahsan Bin Tufail., *et al*. [8] developed deep 2D convolutional neural networks (2D-CNNs) for classification, which increased the performance bias classification. This developed method did not explore new network frameworks, particularly 3-dimensional architectures.

2.2 Challenges

Some troubles experienced off by reviewed approaches for classification are expounded as follows.

- In [1], the method was developed for predicting the classification of disease did not investigate the classification of training data intelligent splitting and also, failed to deal with count of parameters precisely.
- The CNN in [2] was presented for accurate classification of brain structural MRI slices, even though the data utilized was inadequate and increased the computational complications.
- The DNFN in [4] was introduced for an effectual cost and load optimization, even though it failed to alter the count of error tolerance and epochs to improve the system performance.
- The probe for understanding the fundamental pathology is not still developed novel therapeutic techniques for improving the symptoms or stoppage of disease enhancement.

3. Proposed FCSCOOT_DNFN for Parkinson's Classification

In classification, images are classified into Parkinson and brain tumor employing DNFN, which is tuned using a proposed FCSCOOT method. In addition, FCSCOOT is newly designed by amalgamating FC with CSCOOT technique, as shown in Figure1.

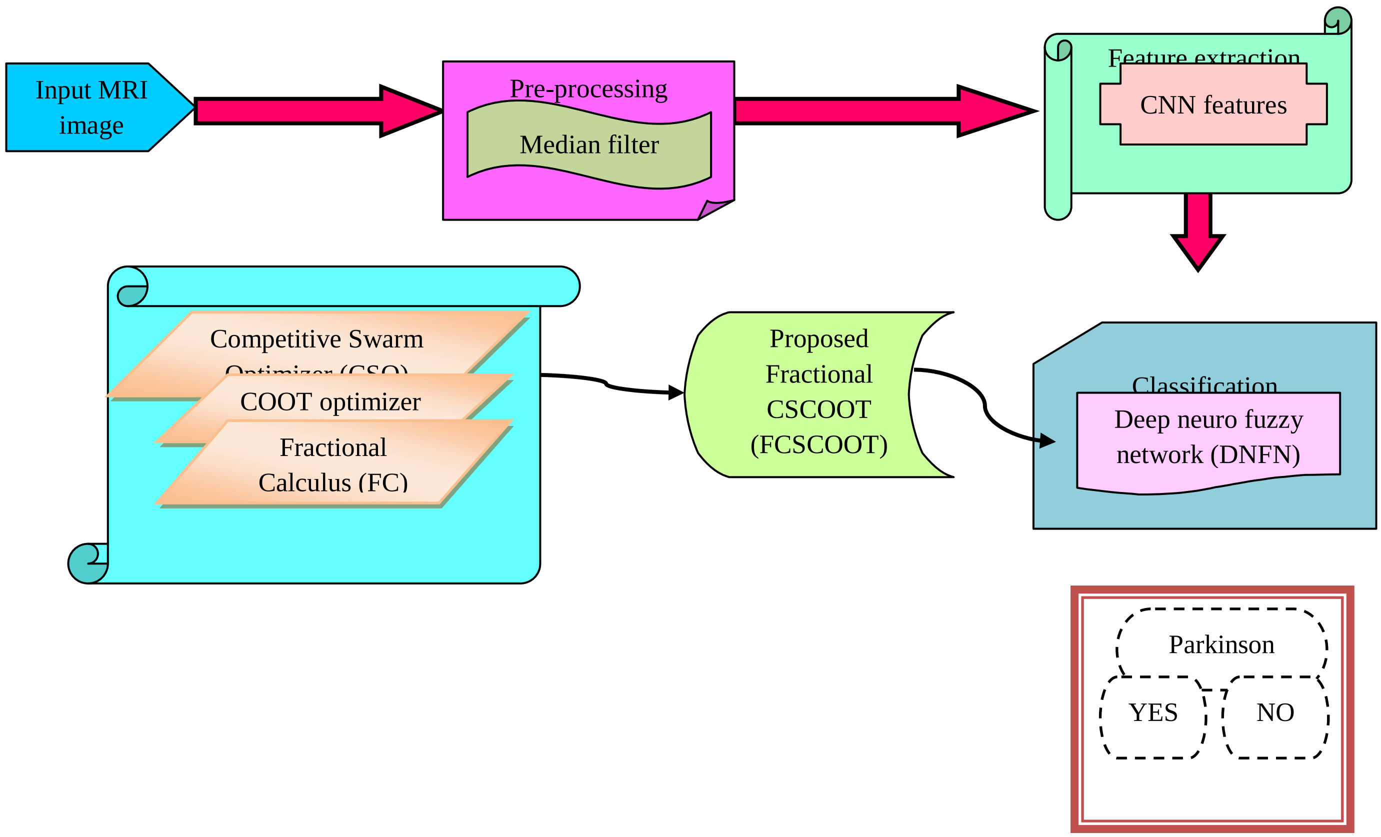


Figure 1. Pictorial illustration of newly introduced techniques for Parkinson's classification

3.1 Proposed technique FCSCOOT_DNFN

If a classified output is Cognitive normal for Alzheimer's classification, then it is subjected to further classification of disease, where the disease is classified as Parkinson. It is performed by DNFN that is trained employing FCSCOOT.

3.1.1 Pre-processing of image using median filter

The median filter [18] is referred as non-linear spatial filter, which is based upon order-statistics theory that is specifically efficient in removing salt and pepper noise. It substitutes a pixel value by grey level median in region of that pixel. A two dimensional median filter is presented by,

$$F(r,s) = median_{(u,v)\in U_{rs}}\{G(u,v)\} \tag{1}$$

Here, U_{rs} indicates the $b \times c$ sub-image of an input noisy image G. It is centred at the coordinates (r,s). $F_{r,s}$ signifies the response of filter at those coordinates. An output achieved from the pre-processing module is T_a.

Experiment outcomes of median filtering for Parkinson's disease are signified in figure 2. Figure 2 a) reveals input image and pre-processed image is shown in figure 2 b).

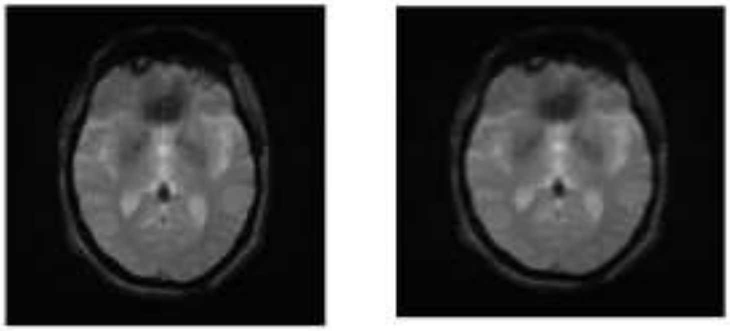


Figure 2. Median filter (a) Original Image (b) Filtered Image

3.1.2 Transfer learning using VGG19

In general, the VGG19 architecture consists of 19 layers, including 16 convolutional layers and 3 fully connected layers.

VGG19 (Visual Geometry Group 19) is a deep convolutional neural network architecture that was proposed by the Visual Geometry Group at the University of Oxford. It is an extension of the earlier VGG16 model, with 19 layers including convolutional and fully connected layers. VGG19 is known for its simplicity and effectiveness in image classification tasks.

Here is an overview of the architecture of VGG19:

Input Layer:

Accepts input images of fixed size (usually 224x224 pixels).

Convolutional Layers:

The network starts with a series of convolutional layers, each followed by a rectified linear unit (ReLU) activation function. The convolutional layers use small receptive fields (3x3) and have a stride of 1. The number of filters gradually increases as we go deeper into the network. There are a total of 16 convolutional layers in VGG19. The number of filters for each convolutional layer is as follows:

Conv1: 64 filters

Conv2: 128 filters

Conv3: 256 filters (Repeated twice)

Conv4: 512 filters (Repeated twice)

Conv5: 512 filters (Repeated twice)

Max Pooling Layers:

After every two convolutional layers, there is a max pooling layer. The max pooling layers downsample the spatial dimensions of the feature maps and reduce computational complexity. Each max pooling layer uses a 2x2 window with a stride of 2.

Fully Connected Layers:

After the convolutional layers, there are three fully connected layers. Each fully connected layer has 4,096 units, which provides a high-capacity feature representation. The fully connected layers are followed by ReLU activation, except for the last layer.

Output Layer:

The final layer is a fully connected layer with the number of units equal to the number of classes in the dataset. It uses a softmax activation function to produce class probabilities.

Overall, VGG19 is a deep and computationally expensive architecture due to its large number of layers and parameters. It has been widely used as a baseline model in various computer vision tasks and has achieved excellent performance on image classification benchmarks, such as the ImageNet dataset.

The convolutional layers are responsible for extracting features from the input images, while the fully connected layers are used for classification. This study used the VGG19 architecture as a feature extractor, where the last fully connected layer is replaced with a new fully connected layer for Parkinson's disease classification. The features are extracted from the output of the last convolutional layer, resulting in 512 features being extracted from each input image. The number of features extracted by VGG19 in Parkinson's disease classification can vary depending on the specific implementation and the layer used for feature extraction. However, in general, VGG19 is capable of extracting a large number of features from input images, which can be useful for accurate classification of Parkinson's disease.

The steps followed by the algorithm for training CNN with transfer learning utilizing newly designed CSCOOT is explained below:

Step 1: Solution initialization

For solving the optimization problems, a solution is initialized firstly, where an initialized population is obtained employing the beneath equation.

$$C = \{C_1, C_2, ..., C_i, ..., C_j\}$$

Where, C_i represents the i^{th} candidate solution whereas j implies the number of variables in a problem and the current solution is given by C.

Step 2: Evaluation of objective function

For attaining finest solution, an objective function (also known as loss function) is utilized, which is calculated by difference among output targeted and acquired output from CNN with transfer learning. The expression is formulated using Eq. (8). The loss function measures the discrepancy between the model's predictions and the true labels of the new task's dataset. For binary classification tasks, where the goal is to classify input into one of two classes (e.g., spam/not spam), the Binary Cross-Entropy (BCE) loss function, also known as Log Loss, is commonly used. So, the loss function used for proposed task is binary cross-entropy.

$$BCE = -\frac{1}{n}\sum_{i=1}^{n}\left(y_{\text{true},i}\cdot\log(y_{\text{pred},i}) + (1-y_{\text{true},i})\cdot\log(1-y_{\text{pred},i})\right)$$

where $y_{true,i}$ is the true binary label for sample i, and $y_{pred,i}$ is the predicted probability of class 1 for sample *I*, is the number of examples in the dataset.

Since the data has 610 normal images and 221 parkinsons images, the calculation of loss function involves more complex calculations, often performed using machine learning libraries like TensorFlow or PyTorch. During training, the model's parameters are adjusted to minimize this loss, thereby improving the model's ability to make accurate predictions. Therefore, Binary Cross-Entropy loss for this dataset is approximately 0.0742.

Step 3: Random selection of two particles

Each of the particles has μ dimensional location and $\mu-$dimensional velocity vector, $Q(g)=Q_1(g),Q_2(g),......Q_\mu(g)$. In each of the generation, the particles in $R(g)$ are allocated randomly into $t/2$ couples, where the size of swarm t is referred as an even number and thereafter, competition is performed among two particles in each of the couple. As an outcome of all competition, the particle with finest fitness is signified as ω is fed to next generation swarm $R(g+1)$ directly whereas the particles, which lose the competition is indicated by ℓ, which update the location as well as velocity by learning from winner. Thereafter, learning from winner, the loser particle is subjected to a swarm $R(g+1)$. It means each of the particles take part in competition at single time. For the size of swarm t, $t/2$ competitions happen and hence all t particles take part in single competition at once and the location as well as velocity of $t/2$ particles is updated.

Step 4: Termination

Above described steps are performed continual manner for achieving an optimal solution.

3.1.3 Architecture of DNFN

The DNFN is referred to hybridization of deep neural network as well as fuzzy logic. For an effective optimization of peak reduction and cost, DNFN is utilized. Figure 3 shows the architecture of DNFN. As it a hybrid technique, where deep neural network is utilized initially followed by fuzzy logic is employed at secondary phase for computation of system intentions. An entire system is comprised of input layer, the hidden layers to learn and validate as well as output layer. An input layer is on basis of input parameters count and the fuzzification values in a system. The count of hidden layers is of three layers namely rule, defuzzification and normalization. The vital parameters of system are premises as well as consequents. The counts of premise parameters include load, time of day and pricing tariff whereas the count of consequent parameters includes peak reduction and cost. In this technique, premise is a foundation for membership operations in fuzzification at an input layer that defines the occurrence level in behaviors of consumers or patterns of energy consumption. The parameters of consequents are corresponded to defuzzification processing. The mathematical expressions of the mentioned parameters are elucidated beneath.

Each of the output or input parameter is mapped to particular entity otherwise node in neuro-fuzzy network for an individual layer whereas each of the input degree is allocated among 0 as well as 1 according to the condition stated by fuzzy model. Each of the entity in an initial layer is go after the value f output. Assume two premises such as ρ and ν as well as single consequent o. Therefore, the equation is represented by,

$$M_{1,\chi} = \eta A_{\chi}(\rho) \text{ or } M_{1,\chi} = \eta Z_{\chi-2}(\nu), \forall \chi = 1,2,3,4 \tag{2}$$

In the above equation, ρ and ν implies the inputs of each χ^{th} entity, ηA and $\eta Z_{\chi-2}$ indicates antecedent membership operations whereas $M_{1,\chi}$ illustrates the membership degree. The membership operations are referred as bell-shaped operations that are allocated with maximal “1” and minimal “0” values.

$$\eta A_{\chi}(\rho) = \frac{1}{1 + \left| \frac{\rho - e_{\chi}}{w_{\chi}} \right|^{2\alpha_{\chi}}} \tag{3}$$

Where, α_{χ}, e_{χ} and w_{χ} reveals a membership operations of premise parameters, which are optimized by means of training.

A second layer is known as rule base layer that is employed for explaining group of rules. All entities in rule base layer multiply values of linguistic variables for fulfilling membership degree. The values of membership variables product illustrates a rule’s firing strength as elucidated in beneath equation.

$$M_{2,\chi} = \phi_{\chi} = \eta A_{\chi}(\rho) \eta Z_{\chi-2}(\nu), \forall \chi = 1,2 \tag{4}$$

A third layer deals with the normalization, wherein all entity assess a ratio of firing strength in χ^{th} rule along with the summation of every rules firing strength. ϕ_{χ} indicates the generic network weight parameter. An outcome of rule is illustrated by,

$$M_{3,\chi} = \overline{\phi_{\chi}} = \frac{\phi_{\chi}}{\phi_1 + \phi_2}, \forall \chi = 1,2 \tag{5}$$

A fourth one is defuzzification layer, where each of consequents rules re assessed for presenting an entire result on output. It can be mathematically modeled as,

$$M_{4,\chi} = \overline{\phi_{\chi}} \gamma_{\chi} = \overline{\phi_{\chi}} (\varsigma_{\chi} \rho + \xi_{\chi} \nu + \zeta_{\chi}), \forall \chi = 1,2 \tag{6}$$

Where, ς, ξ and ζ are the parameters set of consequent. Thereafter, final layer is regarded as summation layer, which evaluates summation of previous layers outcomes. The process of last outcome computation is given beneath,

$$M_{5,\chi} = \sum_{\chi} \overline{\phi_{\chi}} \gamma_{\chi} = \frac{\sum_{\chi} \phi_{\chi} \gamma_{\chi}}{\sum_{\chi} \phi_{\chi}} \tag{7}$$

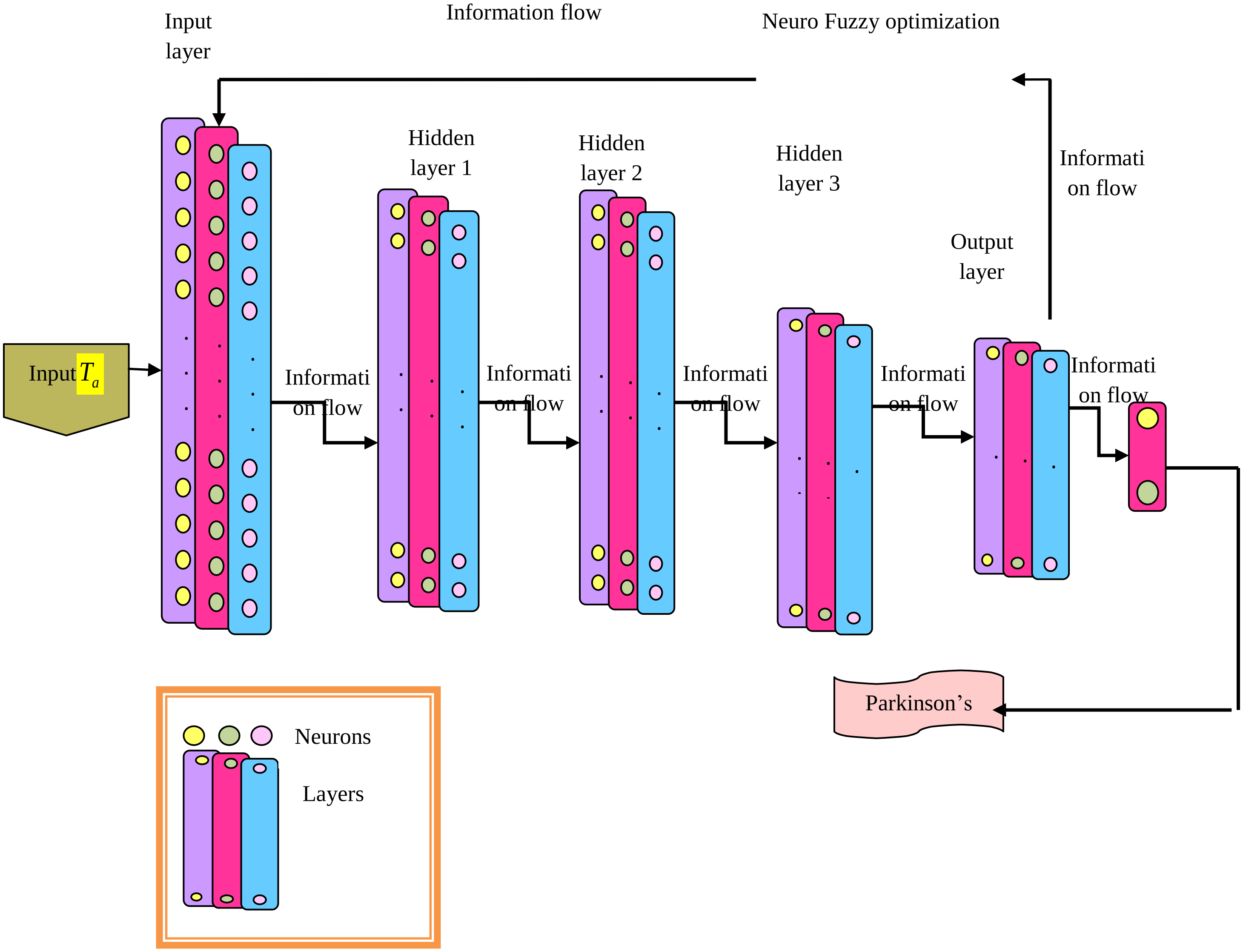


Figure 3. Architecture of DNFN

3.1.4 Proposed FCSCOOT for training DNFN

FC is used for boosting computational performance of technique and hence, DNFN is tuned utilizing newly designed FCSCOOT, which is an amalgamation of FC and CSCOOT as shown in Figure. 1.

The CSO is developed for larger scale optimization that is basically enthused by particle swarm optimization (PSO) however varies in conceptual manner. In CSO, personal finest location of individual particle or global finest location is participated in updating of particles. Alternatively, a pair-wise competition processing is introduced wherein particles lose the competition and update a location by learning from winner. COOT algorithm is inspired from the behavior of birds known as coot. It mimics two diverse methods of bird's movement on water. In an initial phase, the bird's movement is irregular whereas in secondary phase, movement of birds is regular. Here, the two methods are incorporated and named as CSCOOT for training CNN with transfer learning. Therefore, this newly devised technique achieves better performance for solving optimization problems.

Competitive Swarm position encoding

An optimal solution is achieved in search space μ by tuning the learning parameter β, where the expression is given by $\mu = [1 \times \beta]$.

Fitness function

A difference amongst the output targeted and achieved output from CNN with transfer learning is known as fitness function. A fitness function is a mathematical function used in optimization algorithms, such as genetic algorithms, to evaluate how well a solution (individual) performs with respect to a specific task. The fitness function assigns a fitness score to each candidate solution based on its performance. It is formulated by means of,

$$\hbar = \frac{1}{m}\sum_{a=1}^{m}\left[J_a - T_a\right]^2 \tag{8}$$

Here, the overall samples is signified by y whereas J_a and T_a are an output targeted and achieved output from CNN with transfer learning.

4. Results and discussion

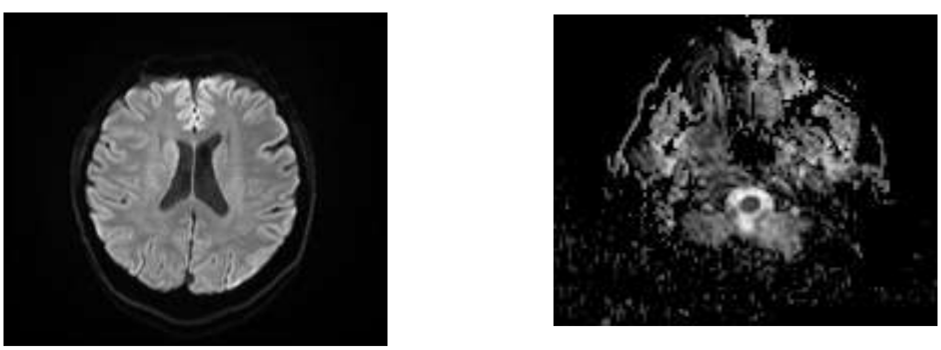
This part reveals results and then the discussions of newly introduced FCSCOOT_DNFN regarding to performance metrics namely, sensitivity, specificity and accuracy. Additionally, experimental setups, dataset description, the metrics for evaluation, outcomes of experiment as well as discussion with existing methods are also expounded below subsections.

4.1 Experimentation setup

An execution of devised techniques for classification of Parkinson's is done in Python tool with Jupyter in PC comprising of Intel core-i3 processor, Windows 10 OS and 8 GB RAM.

4.2 Description of dataset

In this research, Parkinson's dataset is collected from kaggle.com. Parkinson's dataset has 610 normal images and 221 tumor images. Sample images are shown in Figure 4.



(a) **(b)**

Figure 4. Sample input images (a) Normal (b) Parkinson's

4.3 Performance measures

An investigation of devised approaches for PD classification is conducted based upon performance metrics that are explicated below.

(i) Accuracy: Accuracy is referred as a ratio of count of precise predictions to overall count of input samples. It can be formulated by,

$$\delta = \frac{\mathrm{H} + B}{(\mathrm{H} + \mathrm{E} + B + \mathrm{T})}$$

Here, H indicates true positive and B represents true negative whereas E implies false positive and T symbolizes false negative.

(ii) Sensitivity: Sensitivity is termed as the ability for designating the individual person having AD as positive that can be calculated as follows,

$$\varepsilon = \frac{\mathrm{H}}{\mathrm{H} + \mathrm{T}}$$

(iii) Specificity: Specificity is stated as the percentage of peoples having negative results for AD among a group of people having positive results for Alzheimer's disease. It is calculated utilizing the below formula.

$$\partial = \frac{B}{B + \mathrm{E}}$$

4.4 Comparative techniques

In this segment, the proposed techniques like and FCSCOOT_DNFN are compared with some traditional methods namely Deep Siamese CNN [1], CNN [2], Residual neural network [3], DNFN [4] and Competitive Swarm Multi-verse Optimizer (CSMVO)+DNFN for proving the efficacy of newly devised techniques.

4.5 Comparative analysis

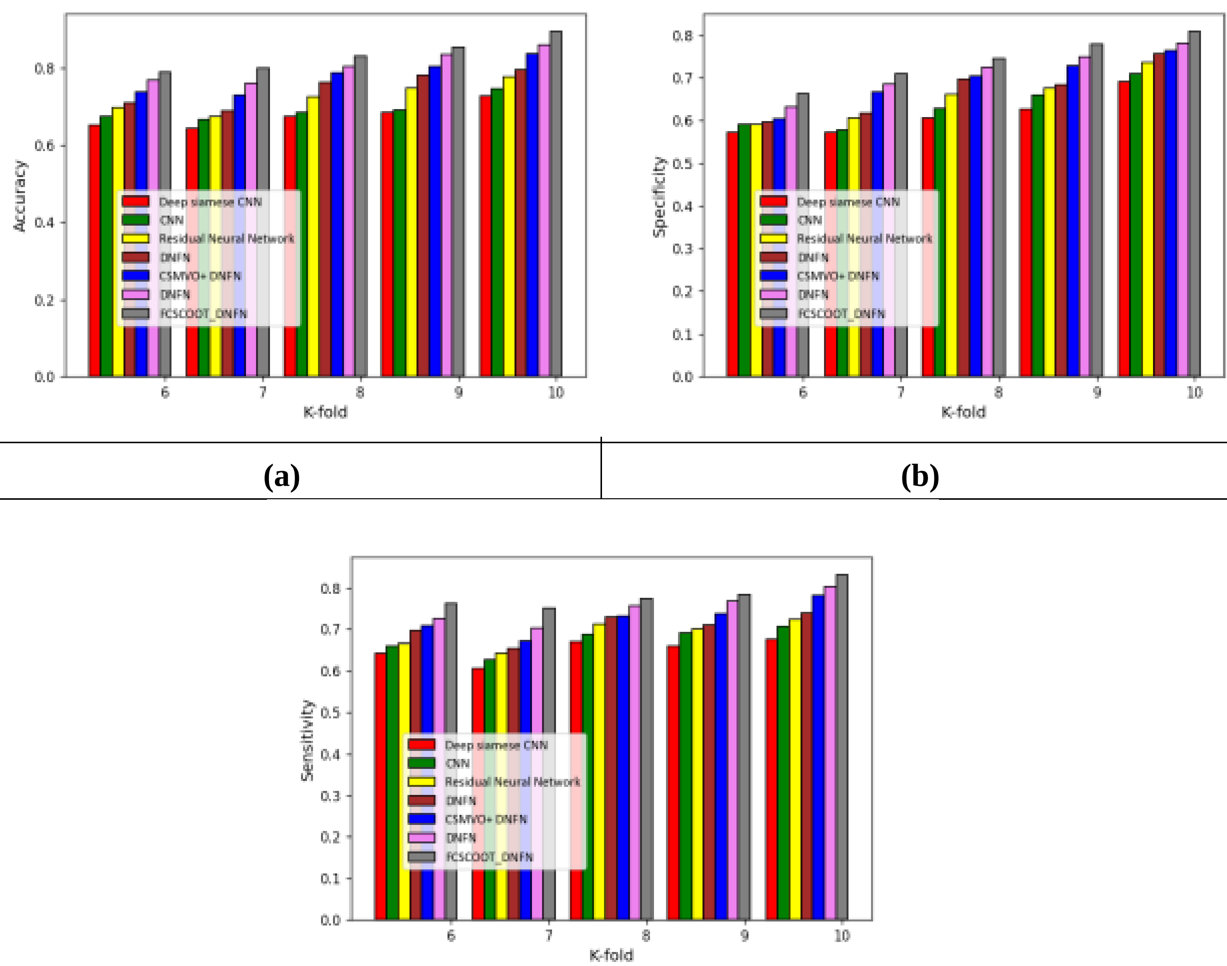
An assessment of comparison between introduced technique and classical techniques regarding the performance measures is elucidated in below subsections.

4.5.1 Analysis based on classification

In classification, the proposed DNFN and FCSCOOT_DNFN are assessed regarding to evaluation metrics by varying k-fold value and percentage of training data.

(i) Analysis based upon k-fold value

A comparing evaluation of newly devised FCSCOOT_DNFN by means of performance metrics by altering k-fold value from 6 to 10 is illustrated in figure 5. An evaluation of FCSCOOT_DNFN with regarding to accuracy is expounded in figure 7 a). An accuracy acquired by proposed DNFN is 0.769 and proposed FCSCOOT_DNFN attained 0.789 whereas existing approaches such as Deep Siamese CNN, CNN, Residual neural network, DNFN and CSMVO+DNFN achieved 0.652, 0.675, 0.698, 0.710 and 0.737 forthe value of k-fold =6. Figure 7 b) shows an evaluation of FCSCOOT_DNFN with respect to specificity. When k-fold value=6, specificity acquired by DNFN and FCSCOOT_DNFN are 0.632 and 0.663 while other classical methods namely Deep Siamese CNN, CNN, Residual neural network, DNFN and CSMVO+DNFN obtained 0.571, 0.592, 0.592, 0.596 and 0.604. An estimation of proposed FCSCOOT_DNFN regarding to sensitivity is explained in figure 7 c). A sensitivity achieved by DNFN is 0.726 and FCSCOOT_DNFN is 0.763 where traditional techniquesnamely Deep Siamese CNN, CNN, Residual neural network, DNFN and CSMVO+DNFN attained 0.642, 0.660, 0.667, 0.697 and 0.708 when the value of k-fold =6.

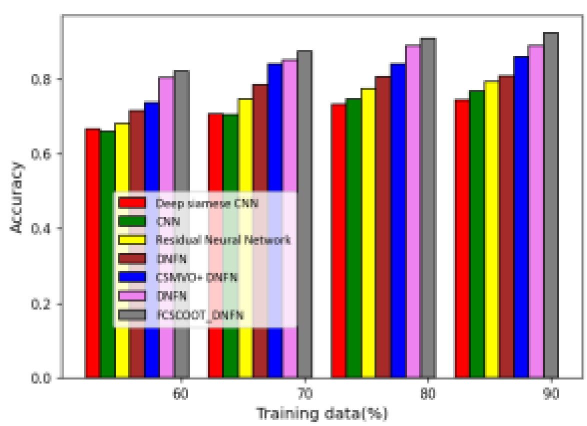


(a) (b) (c)

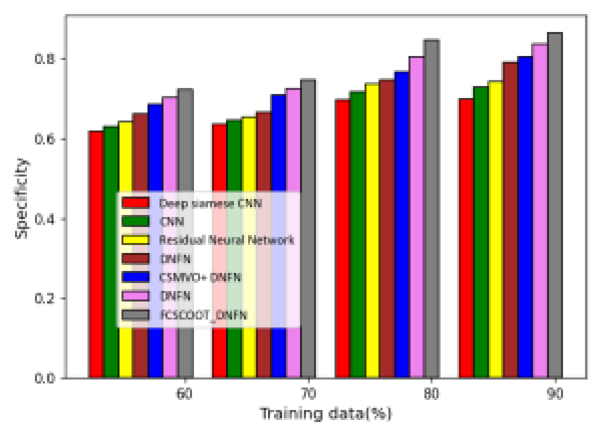
Figure 5. Evaluation based upon k-fold value a) Accuracy, b) Specificity, c) Sensitivity

(ii) Analysis based upon training data

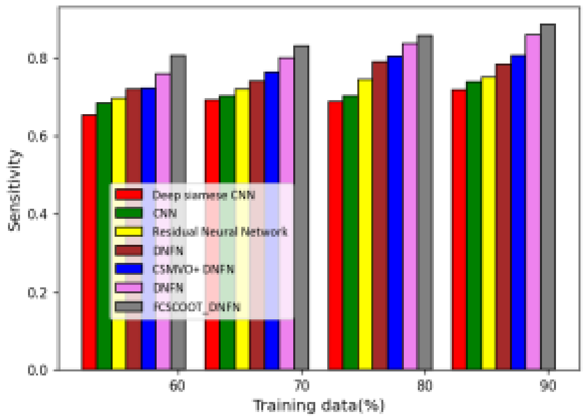
Figure 6 elucidates a comparison evaluation of FCSCOOT_DNFN with regarding to performance measures by varying training data percentage from 60% to 90%. Figure 8 a) reveals an estimation of proposed FCSCOOT_DNFN regarding accuracy. An accuracy achieved by DNFN is 0.805 whereas FCSCOOT_DNFN is 0.820 when other current approaches like Deep Siamese CNN, CNN, Residual neural network, DNFN and CSMVO+DNFN obtained 0.666, 0.660, 0.682, 0.717 and 0.737 for percentage of data=60%. An assessment of proposed FCSCOOT_DNFN on basis of specificity is shown in figure 8 b). While percentage of data=60%, specificity acquired by DNFN and FCSCOOT_DNFN are 0.702 and 0.723 when other methods namely Deep Siamese CNN, CNN, Residual neural network, DNFN and CSMVO+DNFN attained 0.617, 0.629, 0.642, 0.663 and 0.686. Figure 8 c) illustrates an estimation of proposed FCSCOOT_DNFN regarding sensitivity. A sensitivity obtained by DNFN is0.758 whereas sensitivity attained by FCSCOOT_DNFN is 0.806 when existing approaches such as Deep Siamese CNN, CNN, Residual neural network, DNFN and CSMVO+DNFN attained 0.653, 0.683, 0.697, 0.719 and 0.720 for 60% of data.



(a)



(b)



(c)

Figure 6. Analysis based upon training data a) Accuracy, b) Specificity, c) Sensitivity

A comparing deliberation of FCSCOOT_DNFN is reveled in table 2. From the table, it is accepted that FCSCOOT_DNFN has obtained higher accuracy of 0.820, sensitivity of 0.723 and specificity of 0.806 for second level of classification while consideration of 60% of training data.

Table 2. Comparative discussion of proposed FCSCOOT_DNFN

Analysis based upon	***K-fold value=6***			***Training data=60%***		
Metrics/Methods	*Accuracy*	*Specificity*	*Sensitivity*	*Accuracy*	*Specificity*	*Sensitivity*
Deep Siamese CNN' (2020)	0.652	0.571	0.642	0.666	0.617	0.653
CNN (2020)	0.675	0.592	0.660	0.660	0.629	0.683
Residual neural network (2020)	0.698	0.592	0.667	0.682	0.642	0.697
DNFN (2021)	0.710	0.596	0.697	0.717	0.663	0.719
CSMVO + DNFN (2021, 2022)	0.737	0.604	0.708	0.737	0.686	0.720
Proposed FCSCOOT_DNFN (2023)	**0.789**	**0.663**	**0.763**	**0.820**	**0.723**	**0.806**

5. Conclusion

Firstly, the input MRI images are collected Parkinson’s disease form kaggle.com. Then, input images are given to phase of pre-processing for removal of noises, where median filter is used. Thereafter, pre-processed image is passed to feature extraction in which CNN features are extracted. After feature extraction, extracted features are given to first level of classification where disease is classified utilizing CNN with transfer learning, where CNN is employed with hyper parameters from trained model VGG19. Moreover, CNN with transfer learning is tuned byintroducedFCSCOOT technique.In this level of classification, images are classified into Parkinsonutilizing DNFN that is trained by introduced FCSCOOT approach. In addition, FCSCOOT technique is newly designed by combination of FC with CSCOOT algorithm. Furthermore, the success of using VGG19 for Parkinson's disease classification depends on the quality and diversity of the dataset, as well as the availability of relevant labeled images. Additionally, considerations such as data augmentation, cross-validation, and hyperparameter tuning can further enhance the model's performance.

References

[1] AtifMehmood, MuazzamMaqsood, Muzaffar Bashir and Yang Shuyuan, "A deep Siamese convolution neural network for multi-class classification of Alzheimer disease", Brain sciences, vol.10, no.2, pp.84, 2020.

[2] Dr. RachnaJaina, Nikita Jain, AkshayAggarwal and D. Jude Hemanth, "Convolutional neural network based AD classification from magnetic resonance brain images", Cognitive Systems Research, vol.57, pp.147-159, 2019.

[3] FarheenRamzan, Muhammad UsmanGhani Khan, AsimRehmat. SajidIqbal, Tanzila Saba, AmjadRehman and ZahidMehmood, "A deep learning approach for automated diagnosis and multi-class classification of AD stages using resting-state fMRI and residual neural networks", Journal of medical systems, vol.44, no.2, pp.1-16, 2020.

[4] SakeenaJavaid, Muhammad Abdullah, NadeemJavaid, Tanzeela Sultana, Jawad Ahmed and Norin Abdul Sattar, "Towards buildings energy management: Using seasonal schedules under time of use pricing tariff via deep neuro-fuzzy optimizer", In proceedings of 2019 15th international wireless communications & mobile computing conference (IWCMC), pp. 1594-1599, June 2019.

[5] SuriyaMurugan, ChandranVenkatesan, M. G. Sumithra, Xiao-ZhiGao, B. Elakkiya, M. Akila and S. Manoharan, "DEMNET: a deep learning model for early diagnosis of Alzheimer diseases and dementia from MR images", IEEE Access, vol.9, pp.90319-90329, 2021.

[6] Hadeer A. Helaly, Mahmoud Badawy and Amira Y. Haikal, "Deep learning approach for early detection of Alzheimer’s disease", Cognitive computation, pp.1-17, 2021.

[7] Ravi ChandaranSuganthe, RukmaniSevalaiappanLatha, MuthusamyGeetha and GobichettipalayamRamakrishnanSreekanth, "Diagnosis of AD from brain magnetic resonance imaging images using deep learning algorithms", Advances in Electrical and Computer Engineering, vol.20, no.3, pp.57-64, 2020.

[8] Ahsan Bin Tufail, Yong-Kui Ma and Qiu-Na Zhang, "Binary classification of AD using sMRI imaging modality and deep learning", Journal of digital imaging, vol.33, no.5, pp.1073-1090, 2020.

[9] Susanne G. Muellera, Michael W. Weinera, Leon J. Thal, Ronald C. Petersen, Clifford R. Jack, William Jagust, John Q. Trojanowski, Arthur W. Toga and Laurel Beckett, "Ways toward an early diagnosis in Alzheimer’s disease: the AD Neuroimaging Initiative (ADNI)", Alzheimer's & Dementia, vol.1, no.1, pp.55-66, 2005.

[10] J. C. Baron, G. Chetelat, B. Desgranges, G. Perchey, B. Landeau,V. de la Sayette and F. Eustache, "In vivo mapping of gray matter loss with voxel-based morphometry in mild Alzheimer's disease", Neuroimage, vol.14, no.2, pp.298-309, 2001.

[11] ShishirGoswami, AnandSaoji, Navneet Kumar, Vijay Thawani, MeenalTiwari and ManasiThawani, "Effect of Bacopamonnieri on Cognitive functions in AD patients", International Journal of Collaborative Research on Internal Medicine & Public Health, vol.3, no.4, pp.285-293, 2011.

[12] Hiroki Fuse, Kota Oishi, NorihideMaikusa, Tadanori Fukami and Japanese AD Neuroimaging Initiative, "Detection of AD with shape analysis of MRI images", In proceedings of 2018 Joint 10th International Conference on Soft Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems (ISIS), pp. 1031-1034, December 2018.

[13] Ewald Moser, Andreas Stadlbauer, Christian Windischberger, Harald H. Quick and Mark E. Ladd, "Magnetic resonance imaging methodology", European journal of nuclear medicine and molecular imaging, vol.36, no.1, pp.30-41, 2009.

[14] R. Sampath and A. Saradha, "AD Image Segmentation with Self-Organizing Map Network", Journal of Software, vol.10, no.6, pp.670-680, June 2015.

[15] Eman M. Ali, Ahmed F. Seddik and Mohamed H. Haggag, "Automatic detection and classification of AD from MRI using TANNN", International Journal of Computer Applications, vol.148, no.9, 2016.

[16] Ahmad WaleedSalehi, PreetyBaglat and Gaurav Gupta, "AD diagnosis using deep learning techniques", International Journal of Engineering and Advanced Technology (IJEAT), vol.9, no.3, pp.874-880, February 2020.

[17] SamanSarraf, Danielle D. DeSouza, John Anderson, GhassemTofighi, for the AD Neuroimaging Initiative, "DeepAD: AD classification via deep convolutional neural networks using MRI and fMRI", BioRxiv, pp.070441, 2017.

[18] Ning Chun-yu, Liu Shu-fen and Qu Ming, "Research on removing noise in medical image based on median filter method", In proceedings of 2009 IEEE International Symposium on IT in Medicine & Education, vol.1, pp. 384-388, August 2009.

[19] Ruizhe Yao, Ning Wang, Zhihui Liu, Peng Chen and Xianjun Sheng, "Intrusion detection system in the advanced metering infrastructure: a cross-layer feature-fusion CNN-LSTM-based approach", Sensors, vol.21, no.2, pp.626, 2021.

[20] Muhammad Mateen, Junhao Wen, Nasrullah, Sun Song and Zhouping Huang, "Fundus image classification using VGG-19 architecture with PCA and SVD", Symmetry, vol.11, no.1, pp.1, 2018.

[21] TawsifurRahman, Muhammad E. H. Chowdhury, AmithKhandakar, Khandaker R. Islam, Khandaker F. Islam, Zaid B. Mahbub, Muhammad A. Kadir and SaadKashem, "Transfer learning with deep convolutional neural network (CNN) for pneumonia detection using chest X-ray", Applied Sciences, vol.10, no.9, pp.3233, 2020.

[22] OmolbaninYazdanbakhsh and Scott Dick, "A deep neuro-fuzzy network for image classification", arXiv preprint arXiv:2001.01686, 2019.

[23] Ran Cheng and Yaochu Jin, "A competitive swarm optimizer for large scale optimization", IEEE transactions on cybernetics, vol.45, no.2, pp.191-204, 2014.

[24] IrajNaruei and FarshidKeynia, "A new optimization method based on COOT bird natural life model", Expert Systems with Applications, vol.183, pp.115352, 2021.

[25] Pawan R. Bhaladhare and Devesh C. Jinwala, "A clustering approach for the l-diversity model in privacy preserving data mining using fractional calculus-bacterial foraging optimization algorithm", Advances in Computer Engineering, vol.2014, 2014.

[26] AvinashGopal, "Hybrid classifier: Brain Tumor Classification and Segmentation using Genetic-based Grey Wolf optimization", Multimedia Research, vol 3, no 2, 2020.

[27]G.Gokulkumari, "Classification of Brain tumor using Manta Ray Foraging Optimization-based DeepCNN classifier", Multimedia Research, vol 3, no 4, 2020.