Microsystem Technologies
https://doi.org/10.1007/s00542-020-04888-5

TECHNICAL PAPER

Machine learning technique for early detection of Alzheimer's disease

Rashmi Kumari[1] · Akriti Nigam[1] · Shashank Pushkar[1]

Received: 3 May 2020 / Accepted: 14 May 2020

Abstract
Alzheimer's disease (AD) is non-repairable brain disorder which impacts a person's thinking along with shrinking the size of the brain, ultimately resulting in the death of the patient. It is necessary for the treatment of initial stages in AD so that the further degeneration could be delayed. This diagnosis can be achieved with the application of machine learning techniques which employ various optimization and probabilistic techniques. Hence with an objective of distinguishing people with normal brain ageing from those who would develop Alzheimer's disease, this paper presents an effective machine learning model that successfully diagnosed AD, cMCI, ncMCI and CN which are being detected during pre-stages by itself.

1 Introduction

Alzheimer's disease, one of the most prevalent kinds of dementia, is a neurological brain disorder that is progressive in nature and usually occurs during the later life of human beings. This disease has affected 30 million people all over the world, and this number is expected to increase by three times over the next five decades because of the increase in the population of old people. During this condition, one can observe that the patient's memory and intellectual functions decline (Liu et al. 2014). The clinical precursor for AD is considered to be mild cognitive impairment (MCI). This is a phase of transition from healthy to dementia. Till now, there has been no complete cure. But, there has been the advancement of some medications that delay the progression of AD, specifically when it is was diagnosed in the initial stages. Hence, early diagnosing is significant for treating the AD patients. But the precise as well as initial diagnosis of AD is still difficult for the doctors. The magnetic resonance images (MRI) inclusive of functional MRI (fMRI) and structural MRI (sMRI) are useful imaging tools that aid in understanding and evaluating the anatomical and neural variations of AD (Jack et al. 2011). Numerous efforts are being applied in the recent decade for developing the computer-aided models which use machine learning techniques for decoding the states of the disease using the MRI images (Suk et al. 2015). Machine learning is utilized for interpreting and analyzing the MRI images. Furthermore, it has the ability to classify the model data and patterns. Various techniques of machine learning based on extracting the features of high dimension from various image modalities like positron emission tomography (PET) and MRI have been developed for diagnosing the AD (Hinrichs et al. 2009; Zhang et al. 2011). These methods along with identifying the subjects in AD stage predicting the risk of mild cognitive impairment subjects converting into Alzheimer's disease. Hence, according to the risk of the disease progression, the instances of MCI could be named as MCI converters (cMCI) or else MCI non-converters (ncMCI). Therefore diagnosing the AD at the initial stages can be modelled as a multi-classification problem (Suk et al. 2014). Many of these machine learning approaches achieved promising accuracies, but they were not assessed on pathologically proven datasets which have been obtained from distinct modalities of imaging, making it challenging to compare it in a fair manner. Additionally, various factors like preprocessing the significant characteristics for selecting the features, and class imbalance considerably impact the prediction accuracy. To overcome the constraints and restrictions in previous studies, this paper proposes a generalized and useful machine learning technique for early diagnosis of Alzheimer's disease using magnetic resonance imaging (MRI) images and a convolutional neural network (CNN) classifier. The images are preprocessed, segmented using Gaussian filter and Otsu thresholding algorithm respectively. The image features are

✉ Rashmi Kumari
rksingh210492@gmail.com

[1] Birla Institute of Technology, Mesra, Ranchi, India

Published online: 30 May 2020

obtained using Gray level co-occurrence matrix (GLCM) technique. These features are being classified as CNN classifier.

2 Literature review

With the development of machine learning models, features can be extracted and classified without engaging any of the experts. Hence the researchers are focussed on developing the various models for precisely detecting and classifying the images. Liu et al. (2012) contributed for early diagnosis of AD by computer-aided diagnosis (CAD) by the implementation of an ensemble sparse technique for classifying the images. The high feature dimensionality reduced the classification ability of the standard classification models; hence the authors developed a sparse representation-based classifier (SRC) for generating the localized patches that are fused at a later stage in order to provide precise classification. This method was evaluated on 652 subjects (which included 198 AD patients, 229 normal and 225 MCI) from the Alzheimer's disease Neuroimaging Initiative database of MRI images. From the experimental outcome, it was found that the technique obtained classification accuracy of 90.8% for AD, and 92.90% for MCI. From the study, it could be inferred that the classification accuracy is high when the patches generated are from AD; otherwise, it will be low. Further, the pathologically unproven dataset and imbalance in the class demonstrated the uncertainty in the results. Ramírez et al. (2013) contributed towards for early diagnosis of AD by developing a CAD model for enhancing the early detection of AD. This model was based on selecting the parameters of the image and classification using support vector machine. A study was conducted to determine the Region of Interests (ROIs) and most discriminant metrics of the image. The primary objective of this study was to decrease the dimensionality of the input space and diagnose the AD with higher precision with the aid of the radial basis function (RBF) SVM. The technique achieved a sensitivity of 93.10%, an accuracy of 90.38% and a specificity of 86.96%. However, it is not easy to classify the images into more than two classes in a single setting using SVM classifier. Suk and Shen (2013) developed a feature representation technique based on deep learning with the aid of a stacked auto-encoder (SAE). The study assumes that the complex latent patterns like non-linear relations are implicit within the features at low-level. They combined the initial low-level features with the latent information for developing an effective model for classifying the MCI/AD with high diagnostic accuracy. The developed model was tested on ADNI dataset and the accuracies obtained were 95.9%, 75.8%, and 85% for AD, MCI-converter, and MCI respectively. Even though the SAE model successfully classified the images, there is still a possibility for developing a deep multimodal network for shared representation. Payan and Montana (2015) developed and tested a pattern classification model which was a combination of sparse autoencoders and CNN. The primary objective of the research was to evaluate the accuracy of the developed model on a comparatively larger population of patients and compare the performance of the 2D, 3D architecture of CNN. The model was tested using 2265 scanned images of the brain obtained from the ADNI dataset (O'Shea and Nash 2015). The results indicated that the developed 3D model has the ability of capturing the 3D patterns which ought to ameliorate the performance of classifiers by a small range. There exists some drawbacks to this technique; for example, the convolutional layer is trained initially but not fine-tuned. Fine tuning can enhance the system performance thus providing a higher accuracy of classification. Ortiz et al. (2016) presented an approach for diagnosing early AD and AD by the fusion of a structural and functional image data based on the deep learning technique particularly DBN. The regions of the brain are defined based on automated anatomical labeling (AAL). The grey matter (GM) images from each area of the brain are separated into three-dimensional patches based on the areas that were characterised by AAL and these patches aid the training of the DBN. Two structures based on deep learning and four distinct schemes of voting were implemented. The resulting technique is evaluated using the ADNI database. It can be inferred that this method was not only an ideal one for classifying the images but also performed better for MCI subjects as well. The architecture provided an accuracy of 90% for NC/AD and 84% for MCI/AD classification (Cuingnet et al. 2011). Huang et al. (2017) developed DenseNet to be a new structure for deep CNN that connected several layers to every other layer in a feed-forward manner for capturing and reusing features of various layers to perform better when compared to the CNN. This model introduced a direct connection among the two layers having an identical size of feature maps. This study proved that the DenseNets scaled naturally up to hundreds of layers without the presence of any difficulties of optimization. The use of DenseNets consistently improved the accuracy with the increasing number of parameters. Furthermore, the DenseNets required comparatively lesser parameters and computations to achieve the results which are compared to the state-of-the-art techniques. Motivated by this Li et al. (2018) developed a novel technique for classifying the MR images. This approach was based on the multiple cluster dense CNN. The complete brain was first split into distinct local regions for extracting the 3D patches from them. Then the patches from each region are clustered with the aid of the k-means

clustering technique. Finally, the DenseNet is constructed for learning the features of all the clusters, and these features are then ensembled for the process of classification. This technique was evaluated for the ADNI database of 831 subjects which included 403 MCI, 199 AD, and 229 normal controls. From the results, it was determined that the technique achieved an accuracy of 89.5% for AD, and 77.5% for MCI. Compared to the existing approaches this method had certain advantages (1) it alleviated the issue of a smaller set of images on training the DenseNets. (2) ROI segmentations were not required in processing the images thus simplifying the diagnosis process and cost of computation. However, this technique was limited to ADNI datasets and was not evaluated for other datasets for multimodal analysis of the brain images. These examinations exemplified how the results ought to be approved and depicted, particularly in the visualization and forecast of AD. However, having the capacity to recognize the potential issues in the input information, design of the experiment, validating or implementing is exceptionally critical particularly for the individuals who assess various examinations as well as for those intending to use machine learning.

3 Proposed methodology

This study proposes a novel technique for detecting Alzheimer's disease and is shown in Fig. 1. The dataset consisting of MRI images, collected from the open access database of OASIS-3 (http://www.oasis-brains.org). OASIS is a compilation of brain images for > 1000 patients (Fotenos et al. 2005). The images are preprocessed by applying Gaussian filter to remove any of the unwanted noise. Then the images are segmented using Otsu thresholding algorithm, and the edges are detected using the Prewitt edge detection technique. The images are clustered using the fuzzy clustering technique and then GLCM feature extraction technique is applied for extracting the features. Finally, using convolutional neural networks (CNN) the dataset is classified.

3.1 Image preprocessing

The most commonly utilized approach for preprocessing the images is the Gaussian filtering technique. It is an effective local filtering technique popular for smoothening the images. The Gaussian filter is a low pass filter which suppresses the high-frequency details including the edges and noise, in turn preserving the components of low frequency in the image. In simpler terms, it can be referred to as a filter that blurs out anything that is smaller than the image feature (Perumal and Velmurugan 2018). The operation of blurring the image and removing the noise is done by a 2-D convolution operator referred to as the Gaussian smoothing operator. A kernel is utilized that is a representation of the Gaussian (bell-shape) hump. The Gaussian filter is mathematically expressed using the equation given below.

$$g(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x+y}{2\sigma^2}} \quad (1)$$

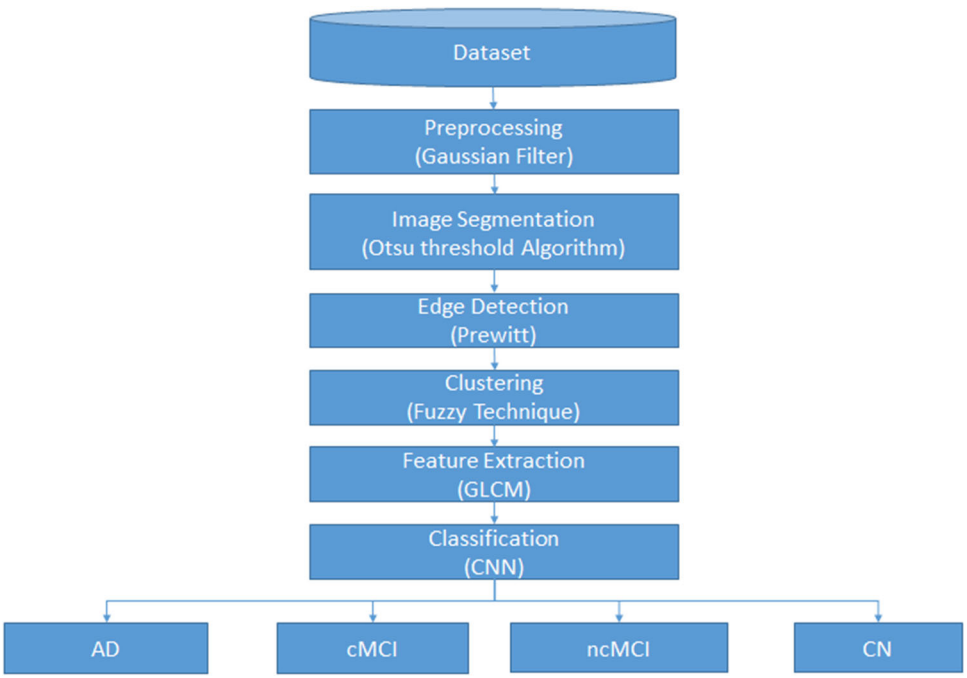


Fig. 1 The flow of the proposed methodology

where y, x = distance between origin and vertical, horizontal axes, σ = standard deviation of the Gaussian function.

Through convolution, the above mentioned 2D distribution is utilized as the point-spread function. The Gaussian function requires a discrete function to perform convolution because the image is stored as a pixels in form of discrete.

3.2 Image segmentation

The developed model utilizes Otsu method that maximizes the variance among the classes for the process of segmenting the image. Reason for using Otsu method is that it is a non-parametric approach popular for its simplicity and effectiveness. The Otsu method involves an exhaustive search for the threshold, which minimizes class variance. The intra-class variance is described as the summation of variances in two classes.

$$\sigma_w^2 = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t) \tag{2}$$

weights ω_i = probability of the classes which are separated by a threshold t, σ_i^2 = variance of the classes. The probability of the classes is formulated with the help of the relations given below.

$$\begin{aligned} \omega_0(t) &= \sum_{i=0}^{t-1} p(i) \\ \omega_1(t) &= \sum_{i=0}^{L-1} p(i) \end{aligned} \tag{3}$$

Otsu demonstrated that minimization of intra-class variance or maximization of inter-class variance which are equal.

$$\begin{aligned} \sigma_b^2(t) &= \sigma^2 - \sigma_w^2(t) = \omega_0(\mu_0 - \mu_r)^2 + \omega_1(\mu_0 - \mu_r)^2 \\ &= \omega_0(t)\omega_1(t)[\mu_0(t) - \mu_t(t)]^2 \end{aligned} \tag{4}$$

which are indicated with respect to class means μ as well as ω. The mean of the classes is given by

$$\begin{aligned} \mu_0(t) &= \frac{\sum_{i=0}^{t-1} ip(i)}{\omega_0(t)} \\ \mu_1(t) &= \frac{\sum_{i=0}^{L-1} ip(i)}{\omega_1(t)} \\ \mu_T(t) &= \sum_{i=0}^{L-1} ip(i) \end{aligned} \tag{5}$$

The steps involved in the Otsu thresholding algorithm are given below (Makkar and Pundir 2014).

- The histogram as well as probabilities of every intensity level is computed.
- The starting values of $\omega_i(0)$ and $\mu_i(0)$ is set up
- The later stage is going through all the thresholds t = 1… Max intensity.
- Updating ω_i, μ_i
- Computing σ_b^2 (t)
- The desired threshold is the maximum value of σ_b^2 (t).

3.3 Edge detection

The developed model utilizes Prewitt edge detection method. The conventional Prewitt operator for detecting the edges comprises of two 3 × 3 matrix groups. This could detect only the vertical and horizontal directions (Yang et al. 2011). Since the edges had more than two directions an eight-direction template Prewitt operator is used. The template of eight directions is illustrated in Fig. 2. The Prewitt algorithm computes the gradient in all

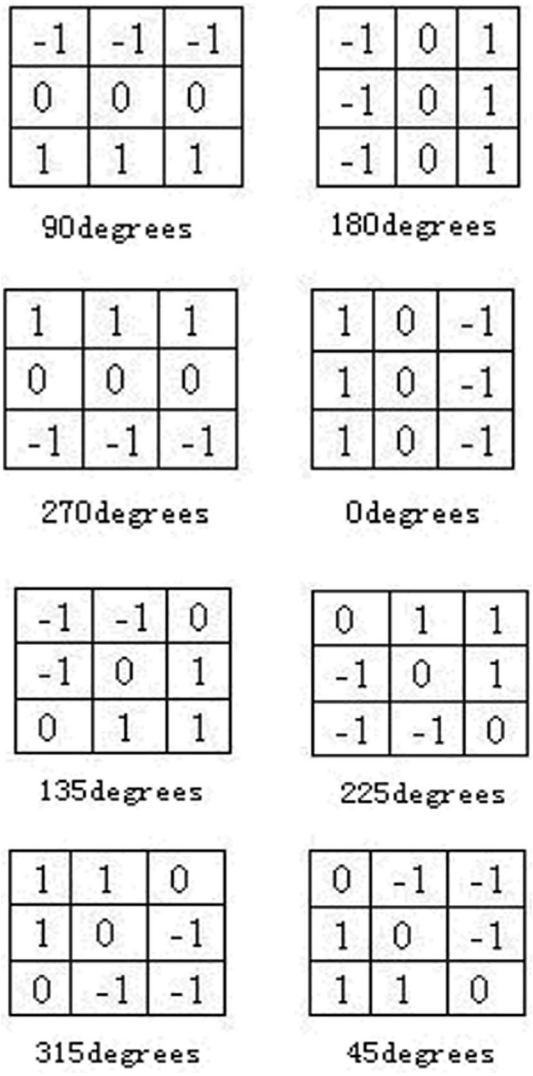


Fig. 2 Template of eight directions

the directions for every point of the image and then considers the maximal value of the magnitude in gradient as well as pixel values for edge images in correspondence point.

3.4 Fuzzy clustering

The images are clustered using Fuzzy C-means (FCM) clustering technique since its performance is high for the images without any noise. The technique of fuzzy clustering will allocate every training vector to the membership value set, one for every cluster instead of allocating the training vectors to just one cluster (Zheng et al. 2015). The objective fuzzy function that requires minimization is given by

$$J_m = \sum_{i=1}^{N}\sum_{j=1}^{J} \mu_{ij}^{m} d_{ij} \tag{6}$$

where y_i is the dataset with i = (1, 2, …, N) for the vector space of D dimension, N = total points of data, J = cluster number, u_{ij} represents the membership degree of y_i within the cluster j, for each function of membership u_{ij}, m represents the weighting exponent, the distance between the centre of the cluster μ_i and y_i is represented by d_{ij} also referred to as distance function. In FCM, the squared Euclidean distance generally applied is given in the form

$$d_{ij} = \|y_i - \mu_i\|^2 \tag{7}$$

Using the function in Eq. (7), through the required conditions the FCM algorithm was iterated to obtain a minimized J_m with the help of the below-mentioned equations.

$$\mu_j = \frac{\sum_{i=1}^{N} \mu_{ij}^{m} y_i}{\sum_{i=1}^{N} \mu_{ij}^{m}} \tag{8}$$

$$\mu_{ij} = \frac{(d_{ij})^{1/(1-m)}}{\sum_{h=1}^{J} (d_{ij})^{1/(1-m)}} \tag{9}$$

With the constraint $\sum_{J=1}^{J} \mu_{ij} = 1$.

3.5 Feature extraction

The proposed model uses GLCM for extracting the features. GLCM is the most popular second order statistical technique utilized for measuring the textural information of the images. It provides adequate information about the textures of the picture which is obtained from two pixels. GLCM was introduced to describe the textures by statistically sampling the occurrence of certain grey levels with respect to other grey levels (Haralick and Shanmugam 1973). Basically, there are two steps, the first step involves the computation of the co-occurrence matrix, whereas in the second step involves calculating the textured based features in the co-occurrence matrix obtained from previous step. The features of the images extracted in this study are contrast, correlation, energy, entropy, homogeneity, standard deviation, skewness, kurtosis, variance and skewness. In GLCM, feature extractions of images are given in the form of different expressions:

Contrast:

$$\sum_{i,j=0}^{N-1} P_{i,j}(i-j)^2 \tag{10}$$

Correlation:

$$\sum_{i,j=0}^{N-1} P_{i,j}\left[\frac{(i-\mu_i)(j-\mu_j)}{\sqrt{(\sigma_i^2)\left(\sigma_j^2\right)}}\right] \tag{11}$$

Energy of image:

$$\sum_{i,j=0}^{N-1} P_{i,j}^2 \tag{12}$$

Entropy of image:

$$-\sum_{b=0}^{b=L-1} P(\mathrm{b})\log 2[\mathrm{P(b)}]$$

where

$$\mathrm{P(b)} = \frac{N(b)}{M} \tag{13}$$

Homogeneity:

$$\sum_{i,j=0}^{N-1} \frac{Pij}{1+(i-j)^2} \tag{14}$$

Standard Deviation:

$$= \sqrt{\frac{\sum_{i=1}^{N}(xi-\bar{x})\hat{2}}{N-1}} \tag{15}$$

xi = the observed values of a sample item
$\bar{x}$ = the S_X mean value of the observations
N = the number of observations

Skewness:

$$= 3 * \frac{(Mean - Median)}{Standard\ Deviation} \tag{16}$$

Kurtosis:

It is defined as fourth standardised moment

$$\mathrm{Kurt[X]} = \mathrm{E}\left(\frac{X-\mu}{\sigma}\right)^4 = \frac{E(X-\mu)^4}{[E(X-\mu)^2]^2} = \frac{\mu_4}{\sigma^4} \tag{17}$$

Variance:

$$\mathrm{Var}(X) = \mathrm{E}\left[(X-\mu)^2\right]. \quad (18)$$

3.6 Image classification

The images are next classified using convolutional neural networks. The CNN is basically applied in convolution of various images in form of kernels as procurement in feature maps. The various kernel weights are benefitted in connection of every unit in feature map to the previous layers. During training of datasets, the weights of the kernels are being utilized in improvement of various input attributes. The number of weights in kernels which have to be trained in the convolutional layers must be lesser when these weights are being compared to the fully connected (FC) layers. This is because of the kernels are common to all the units in one particular feature map. The architecture of CNN is illustrated in Fig. 3.

The functionality of CNN can be bifurcated into four key areas.

- The magnitude of different pixels in an image is obtained from input layer
- The CNN decides output for the neurons that are associated with the input local regions via the computation of scalar product among the regions associated with the volume of the input and the weights of the neurons.
- Then the downsampling of the input is achieved by the pooling layer, hence layer parameters are reduced in a particular activation.
- The fully connected layer then generates scores for the classes (from the activations) which are applied in the process of classification.

4 Results and discussion

The present task considers a total of 200 MRI images of the brain, 100 images for testing and 100 images for training. Figure 4 shows a sample input image and Fig. 5 demonstrates the resized image. Thereafter, above image is then filtered in order to eradicate noise existing in it by using a Gaussian filter (Fig. 6). The contrast of the filter image is then increased (Fig. 7).

The filtered image is then segmented by applying the Otsu thresholding technique. The segmented image is presented in Fig. 8. The performance of segmentation technique is tabulated in Table 1. To this image, Prewitt edge detection method is involved for detecting the edges in the image. The obtained edge detected image is followed in Fig. 9.

From the edge detected image, features are extracted utilizing grey level co-occurrence matrix technique. The extracted features along with their values are tabulated in Table 2. Then fuzzy C-means clustering technique is applied to cluster them into four different groups by reducing the threshold level. The clustered images are shown in Fig. 10. Finally, these clustered images are fed to the CNN classifier for the process of classification for early diagnosis of AD. The classifier classified the images with an accuracy and sensitivity of 90.25% & 85.53% respectively. The classifier output is shown in Fig. 11.

The ROC curve obtained from the proposed technique is shown in Fig. 12, and Fig. 13 provides the parameters of the ROC curve.

The proposed method using CNN classifier is compared to results obtained using KNN classifier implemented on the same dataset. The KNN classifier gave different values of performance metrics for different values of K = 5, 9, 15. The comparison of the KNN and CNN classifier with respect to the performance metrics of accuracy, sensitivity

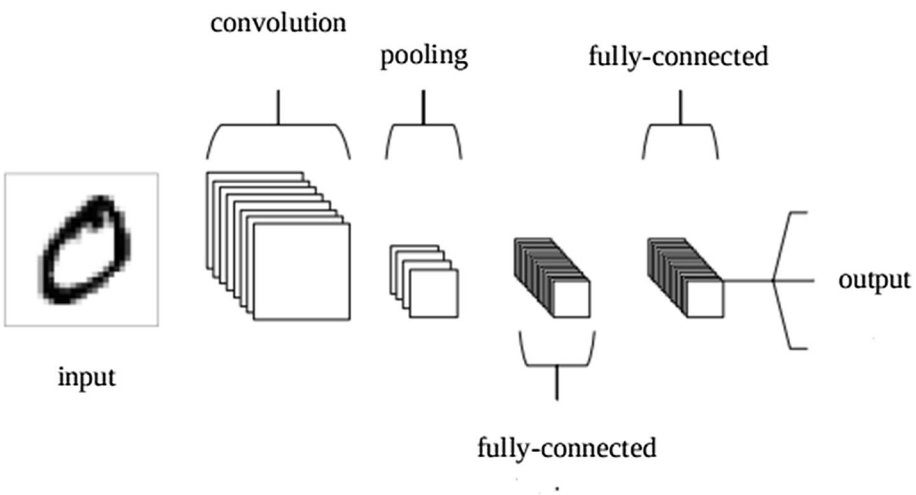


Fig. 3 Architecture of CNN

Fig. 4 Input image

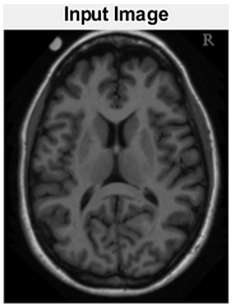


Fig. 5 Resized image

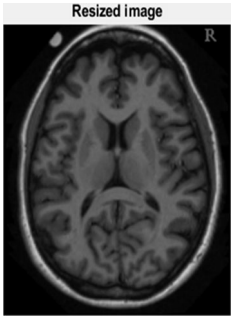


Fig. 6 Filtered image

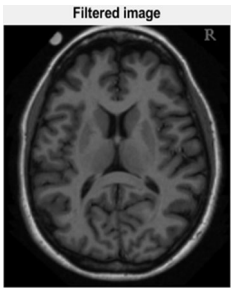
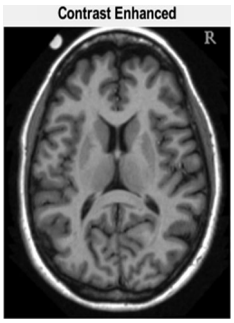


Fig. 7 Contrast-enhanced image



and specificity is tabulated in Table 3. From Table 3, KNN classifier provides an average accuracy of 59.3% while the proposed CNN classifier provides an accuracy of 90.25%.

We correlate our results in proposed method with various techniques available in existing literatures so as to prove the validity of proposed algorithm. The Table 4 demonstrates in comparing with accuracy and sensitivity values accessed by few state-of-the-art methods which are being discussed in literature with our proposed method.

Fig. 8 Segmented image

Table 1 Segmentation results

Performance metric	Result obtained
Accuracy	0.5201
Sensitivity	0.3351
F-measure	0.3831
Precision	0.4471
MCC	0.0035

Fig. 9 Edge detected image

Table 2 Extracted features

Feature extracted	Value
Contrast	1.634
Correlation	0.9521
Energy	0.4816
Homogeneity	0.9792
Mean	0.3333
Standard deviation	0.4376
Entropy	4.2993
RMS	0.4765
Variance	0.1215
Smoothness	1.000
Kurtosis	1.6140
Skewness	0.7163
IDM	220.4068

The graphical representation for the above comparative table is shown in Fig. 14. From the inference of the graph,

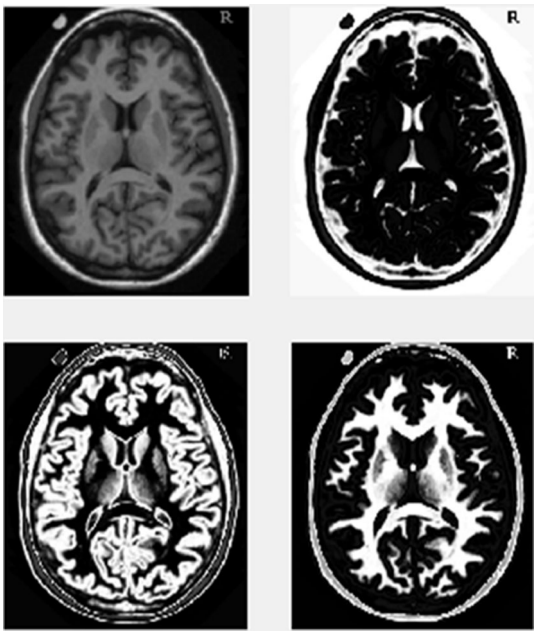


Fig. 10 Clustered image

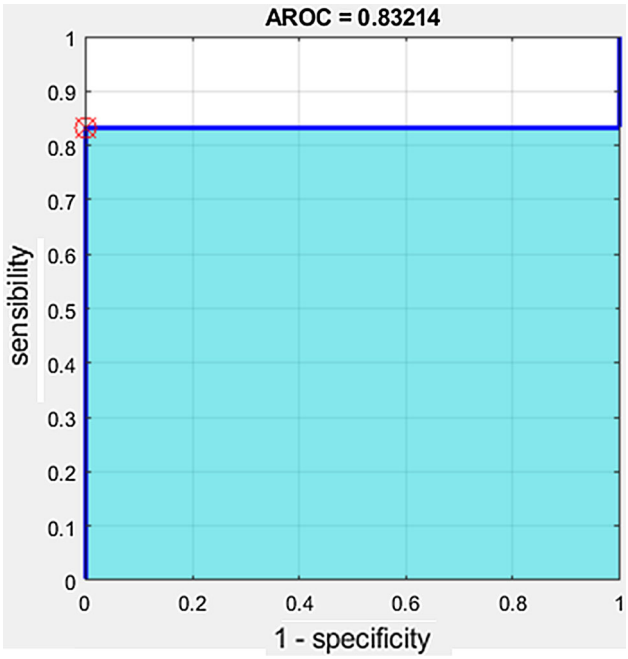


Fig. 12 ROC curve

```
Iter = 61, Objective = 613614532.262792
Training on single CPU.
|=========================================================================================|
|     Epoch     |   Iteration   | Time Elapsed |  Mini-batch  |  Mini-batch  | Base Learning|
|               |               |   (seconds)  |     Loss     |   Accuracy   |     Rate     |
|=========================================================================================|
|             1 |             1 |         0.75 |       0.6933 |        0.00% |       0.0100 |
|             3 |            50 |         1.33 |       0.6247 |      100.00% |       0.0100 |
|             5 |           100 |         1.56 |       0.5597 |      100.00% |       0.0100 |
|             8 |           150 |         1.80 |       0.6058 |      100.00% |       0.0100 |
|            10 |           200 |         2.04 |       0.5623 |      100.00% |       0.0100 |
|            13 |           250 |         2.28 |       0.6056 |      100.00% |       0.0100 |
|            15 |           300 |         2.53 |       0.5623 |      100.00% |       0.0100 |
|            18 |           350 |         2.77 |       0.6056 |      100.00% |       0.0100 |
|            20 |           400 |         3.02 |       0.5623 |      100.00% |       0.0100 |
|            23 |           450 |         3.24 |       0.6056 |      100.00% |       0.0100 |
|            25 |           500 |         3.47 |       0.5623 |      100.00% |       0.0100 |
|            28 |           550 |         3.71 |       0.6056 |      100.00% |       0.0100 |
|            30 |           600 |         3.96 |       0.5623 |      100.00% |       0.0100 |
|            33 |           650 |         4.21 |       0.6056 |      100.00% |       0.0100 |
|            35 |           700 |         4.46 |       0.5623 |      100.00% |       0.0100 |
|            38 |           750 |         4.69 |       0.6056 |      100.00% |       0.0100 |
|            40 |           800 |         4.91 |       0.5623 |      100.00% |       0.0100 |
|            43 |           850 |         5.14 |       0.6055 |      100.00% |       0.0100 |
|            45 |           900 |         5.36 |       0.5623 |      100.00% |       0.0100 |
|            48 |           950 |         5.58 |       0.6055 |      100.00% |       0.0100 |
|            50 |          1000 |         5.81 |       0.5623 |      100.00% |       0.0100 |
|=========================================================================================|

ans =
```

Fig. 11 Classifier output

```
ROC CURVE PARAMETERS
------------------------------
 - Distance:      0.1686
 - Threshold:     118.5000
 - Sensitivity:   0.8314
 - Specificity:   1.0000
 - AROC:          0.8321
 - Accuracy:      0.8317
 - PPV:           1.0000
 - NPV:           0.0084
 - FNR:           0.1686
 - FPR:           0.0000
 - FDR:           0.0000
 - FOR:           0.9916
 - F1 score:      0.9080
 - MCC:           0.0836
 - BM:            0.8314
 - MK:            0.0084
```

Fig. 13 Parameters of ROC curve

proposed technique delivers better accuracy which is 90.25% when compared to other existing algorithms which has the next best value of 88.58%.

5 Conclusion

The present paper proposes an effective machine learning model for detecting the AD in its initial stages. The developed model applied a Gaussian filter for removal of unwanted noise, Otsu thresholding for image segmentation, Prewitt edge detection approach for detecting the edges, GLCM for extracting the features and FCM for clustering and CNN for the final classification of the images. The classifier gave an accuracy of 90.25% and sensitivity of 85.53% in comparison with the KNN classifier that provided an accuracy of just 59.3% and sensitivity of 45.2%. The same results are then compared with various previous works in literature in order to prove the efficiency of our proposed algorithm

Table 3 Comparison of existing (KNN) and proposed (CNN) techniques

Technique	Accuracy	Sensitivity	Specificity
KNN with K = 5	58.63	43.22	64.88
KNN with K = 9	60.64	58.47	NaN
KNN with K = 15	58.63	33.89	59.54
Proposed (CNN)	90.25	85.53	NaN

Table 4 Comparison of proposed approach with state of the art techniques

Methods	Modality	Accuracy	Sensitivity
SK SVM	MRI/PET	84.40	84.64
MK SVM	MRI/PET	86.42	84.98
Liu et al.	MRI/PET	87.76	88.57
Cuingnet et al.	ADNI	88.58	81.00
Hinrichs et al.	MRI/PET	82.00	85.00
Proposed method	MRI	90.25	85.53

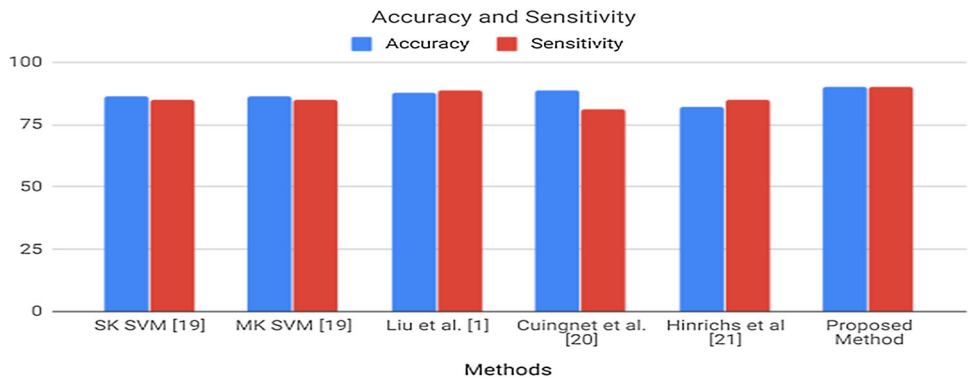


Fig. 14 Graphical representation of proposed approach compared with existing literature

References

Cuingnet R, Gerardin E, Tessieras J, Auzias G, Lehericy S, Habert MO, Chupin M, Benali H, Colliot O (2011) Automatic classification of patients with Alzheimer's disease from structural MRI: a comparison of ten methods using the ADNI database. NeuroImage 56:766–781

Fotenos AF, Snyder AZ, Girton LE, Morris JC, Buckner RL (2005) Normative estimates of cross-sectional and longitudinal brain volume decline in aging and AD. Neurology 64(6):1032–1039

Haralick RM, Shanmugam K (1973) Textural features for image classification. IEEE Trans Syst Man Cybern 6:610–621

Hinrichs C, Singh V, Mukherjee L, Xu G, Chung MK, Johnson SC (2009) Spatially augmented LPboosting for AD classification with evaluations on the ADNI dataset. NeuroImage 48:138–149

Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708

Jack CR Jr, Albert MS, Knopman DS, McKhann GM, Sperling RA, Carrillo MC, Thies B, Phelps CH (2011) Introduction to the recommendations from the National Institute on Aging-Alzheimer's Association workgroups on diagnostic guidelines for Alzheimer's disease. Alzheimer's Dement 7(3):257–262

Li F, Liu M, Initiative Alzheimer's Disease Neuroimaging (2018) Alzheimer's disease diagnosis based on multiple cluster dense convolutional networks. Comput Med Imaging Graph 70:101–110

Liu M, Zhang D, Shen D, Alzheimer's Disease Neuroimaging Initiative (2012) Ensemble sparse classification of Alzheimer's disease. NeuroImage 60(2):1106–1116

Liu S, Liu S, Cai W, Pujol S, Kikinis R, Feng D (2014) Early diagnosis of Alzheimer's disease with deep learning. In: 2014 IEEE 11th international symposium on biomedical imaging (ISBI). IEEE, pp 1015–1018

Makkar H, Pundir A (2014) Image analysis using improved Otsu's thresholding method. Int J Recent Innov Trends Comput Commun 2(8):2122–2126

Ortiz A, Munilla J, Gorriz JM, Ramirez J (2016) Ensembles of deep learning architectures for the early diagnosis of the Alzheimer's disease. Int J Neural Syst 26(07):1650025

O'Shea K, Nash R (2015) An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458

Payan A, Montana G (2015) Predicting Alzheimer's disease: a neuroimaging study with 3D convolutional neural networks. arXiv preprint arXiv:1502.02506

Perumal S, Velmurugan T (2018) Preprocessing by contrast enhancement techniques for medical images. Int J Pure Appl Math 118(18):3681–3688

Ramírez J, Górriz JM, Salas-Gonzalez D, Romero A, López M, Álvarez I, Gómez-Río M (2013) Computer-aided diagnosis of Alzheimer's type dementia combining support vector machines and discriminant set of features. Inf Sci 237:59–72

Suk HI, Shen D (2013) Deep learning-based feature representation for AD/MCI classification. In: International conference on medical image computing and computer-assisted intervention. Springer, Berlin, pp 583–590

Suk HI, Lee SW, Shen D, Alzheimer's Disease Neuroimaging Initiative (2014) Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis. NeuroImage 101:569–582

Suk HI, Lee SW, Shen D, Alzheimer's Disease Neuroimaging Initiative (2015) Latent feature representation with stacked auto-encoder for AD/MCI diagnosis. Brain Struct Funct 220(2):841–859

Yang L, Wu X, Zhao D, Li H, Zhai J (2011). An improved Prewitt algorithm for edge detection based on noised image. In: 2011 4th International congress on image and signal processing, vol 3. IEEE, pp 1197–1200

Zhang D, Wang Y, Zhou L, Yuan H, Shen D (2011) Multimodal classification of Alzheimer's disease and mild cognitive impairment. Neuroimage 55:856–867

Zheng Y, Jeon B, Xu D, Wu QM, Zhang H (2015) Image segmentation by generalized hierarchical fuzzy C-means algorithm. J Intell Fuzzy Syst 28(2):961–973