This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2021.3083274, IEEE Journal of Biomedical and Health Informatics

Classification of Alzheimer's disease using ensemble of deep neural networks trained through transfer learning

M. Tanveer, A.H. Rashid, M.A. Ganaie, M. Reza, Imran Razzak, Kai-Lung Hua, for the Alzheimer's Disease Neuroimaging Initiative[a]

***Abstract*—** **Alzheimer's disease (AD) is one of the deadliest neurodegenerative diseases ailing the elderly population all over the world. Many researchers are using deep learning (DL) techniques to learn highly complicated patterns from MRI scans for the detection of AD. It is also found that an ensemble of predictions from multiple models gives better performance as compared to that of a single model. Two major bottlenecks for developing ensemble of DL models are their high computational complexity and requirement of large sample size for better generalization.In this work, we deal with the aforementioned bottlenecks and propose a computationally efficient, DL-architecture agnostic, ensemble of deep neural networks named 'Deep Transfer Ensemble (DTE)' trained using transfer learning for the classification of AD. The proposed ensemble leverages the diversity introduced by many different locally optimum solutions reached by individual networks through the randomization of hyper-parameters. The proposed ensemble model also introduces further diverse predictions by exploiting complementary feature views. We also test the model vigorously by analysing its performance on a large and a small dataset downloaded from the Alzheimer's Disease Neuroimaging Initiative (ADNI) archive. The DTE utilizes the advantages of random search, transfer learning, and snapshot ensembles in a single ensemble to produce better generalization performance. DTE achieves an accuracy of 99.05% and 85.27% on two independent splits of the large dataset for cognitively normal (NC) vs AD classification task. For the task of mild cognitive impairment (MCI) vs AD classification, DTE achieves 98.71% and 83.11% respectively on the two independent splits. DTE also performed reasonably well on a small dataset consisting of only 50 samples per class. It achieved a maximum accuracy of 85% for NC vs AD on the small dataset. DTE outperformed snapshot ensembles along with several other existing deep models from similar kind of previous works by other researchers.**

***Index Terms*—** **Deep learning, transfer learning, ensemble learning, Alzheimer's disease.**

I. Introduction

ALZHEIMER'S disease (AD) is an incurable, progressive neurodegenerative disease affecting the elderly population. It is expected that by the year 2050, 1 in 85 people worldwide will suffer from AD [1]. The application of machine learning techniques in AD diagnosis has given promising results and currently is a hot topic of research [2]–[7], aided by publicly available data from websites like Alzheimer's Disease Neuroimaging Initiative (ADNI), Australian Imaging, Bio-marker & Lifestyle Flagship Study of Ageing (AIBL) and Open Access Series of Imaging Studies (OASIS). Before progressing to the full-blown AD stage, normal control (NC) subjects experience mild cognitive decline which includes problems with memory, language, judgment and thinking. This onset of cognitive decline is termed as Mild Cognitive Impairment (MCI) [8]. MCI patients have a high chance of advancing to AD with an estimated annual conversion rate of rate 15% [9]. A detailed review of latest state-of-the-art machine learning techniques for the diagnosis of AD can be found in [4].

A drawback of conventional machine learning techniques is the requirement of hand-crafted features, which may lead to sub-optimal performance. Deep learning (DL) techniques, on the other hand, learn important features automatically from the data which makes them extremely efficient. It has been found that the combination of decisions from multiple diverse models produces better results than the decision from a single model [10], [11]. This forms the basis of ensemble models wherein multiple weak learners combine to form a single strong learner. However, there are two major bottlenecks in developing an efficient DL ensemble model that we address in this work:

- Deep learning techniques require a huge amount of training data, sometimes thousands or millions of samples, for successful training owing to their high model complexity

M. Tanveer, A.H. Rashid and M.A. Ganaie are with the Department of Mathematics, Indian Institute of Technology Indore, Simrol, Indore, 453552, India (e-mail (M. Tanveer): mtanveer@iiti.ac.in, email (A.H. Rashid): ashrafrashid@iiti.ac.in, e-mail (M.A. Ganaie): phd1901141006@iiti.ac.in).

M. Reza is with the Department of Mathematics, GITAM University, Hyderabad 502329, India (e-mail: mreza@gitam.edu).

Imran Razzak is with the School of Information Technology, Deakin University, Geelong, Australia (e-mail: imran.razzak@deakin.edu.au).

Kai-Lung Hua is with the Department of Computer Science and Information Engineering National Taiwan University of Science and Technology, Taiwan (e-mail: hua@mail.ntust.edu.tw).

[a] Data used in preparation of this article were obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu). As such, the investigators within the ADNI contributed to the design and implementation of ADNI and/or provided data but did not participate in analysis or writing of this report. A complete listing of ADNI investigators can be found at: http://adni.loni.usc.edu/wp-content/uploads/how_to_apply/ADNI_Acknowledgement_List.pdf

[12]. Such a huge amount of neuroimaging data may not be available in many scenarios.

- High training time is required to train individual DL models due to which developing an ensemble of many DL models becomes very inefficient.

In this work, a computationally efficient ensemble of deep convolutional neural networks (CNNs) trained using transfer learning is proposed for more accurate classification of AD. The major contributions of this paper are:

- We propose a DL-architecture agnostic ensemble strategy leveraging the advantages of random search and snapshot ensembles. The proposed model can be considered as a combination of snapshots of models trained using a random set of hyperparameters.
- Usage of transfer learning helps reduce the computational training complexity, which helps in creation of more computationally efficient DL ensemble model.
- Unlike previous methods, the proposed ensemble leverages diversity obtained through combining predictions from multiple local optimas in the loss surface as well as the diversity obtained by combining multiple feature views with complementary features.
- We rigorously test the proposed model in a large (ADNI baseline) as well as on a small dataset (of only 100 subjects) by using various different experimental settings.

The motivations for the proposed models are as follows:

- Transfer learning computationally aids in faster training of individual models.
- The highly non-convex nature of loss surfaces of deep neural networks posses many different local optimas, that can introduce diverse predictions to make a strong ensemble model.
- Different feature views convey different information and can further introduce diversity in the model to improve the generalization of the ensemble.

The remaining sections of this work are organized as follows:

The section II describes the materials and methods used in this paper. The section III gives details of experiments like the details about train, validation and test splits of the data, hyperparameter tuning and model selection also about the way ensemble of models was performed. The section IV discusses about the results obtained and finally, section V gives the conclusions and mentions some future works.

II. Materials and methods

A. Data acquisition

The experiments were performed on the data acquired from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu). ADNI was launched in the year 2003 with the aim of analysing the efficacy of biological markers, clinical neuropsychological tests and neuroimaging techniques like MRI and PET for diagnosis of AD in early stages. For further details on ADNI, visit `www.adni-info.org`.

In this paper, we use two different datasets : (a) Large dataset (b) Small dataset. In (a), we use the ADNI baseline dataset consisting of 813 3D-MRI scans (187 AD, 228 NC, 398 MCI). In (b), we use a sample dataset for NC vs AD classification on extremely low number of images. We acquired 50 T1-weighted structural MRI (sMRI) scans from the ADNI repository for each of NC and AD categories. Age range of the subjects was between $60-90$ with a mean of 75.83 and a standard deviation 6.07. The range of the Mini-mental state examination (MMSE) score was between $17-30$ with a mean of 26.51 and a standard deviation of 2.88.

B. Image acquisition

For the small dataset, we acquired images of the following specifications: Manufacturer: GE medical systems; Acquisition plane: Sagittal; Angle: 8 degrees; Slice thickness: 1.5 Tesla; Pulse sequence: RM; Acquisition: 3D; Description: MP-RAGE.

C. Image preprocessing

For the large dataset, the 3D-MRI scans were pre-processed using the recon-all pipeline from the FreeSurfer software.

For the small dataset, we used the open source toolbox Statistical Parametric Mapping (SPM-version 12) to preprocess the MRI scans. The preprocessed MRI scans were used for NC vs AD classification. Origin of the raw scans was set manually to Anterior Commissure (AC) before manually registering them with SPM's canonical T1 template image. The registered scans were then passed on to SPM's unified segmentation routine which generated the segmented GM, WM and CSF images. Along with native segmented images, DARTEL [13] imported segmented images were also generated. The DARTEL imported segmented images were used to create a template image that was used to normalize the native segmented images into the MNI space. The DARTEL template is created through a repetitive procedure wherein the parameters required to warp each subject into a standard coordinate space are refined gradually. During creation of the template, a flow field is generated for each image that characterizes the transformation from native image to the template image. Modulation was also performed to preserve the total tissue volume present in native images. Gaussian full width at half maximum (FWHM) 8mm kernel was used for smoothing during the normalization process for noise removal. After normalization, the images of dimension $121 \times 145 \times 121$ and voxel size of $1.5mm^3$ were obtained.

D. Transfer learning

Classical machine learning algorithms assume that the training and testing data are generated by the same probability distribution. This assumption might not hold in scenarios where the size of training data available is very small. Thus, we may be interested in reusing a model already trained on some other (related) data. Transfer learning typically deals with the scenarios wherein we have to transfer the knowledge learned from a source task in a source domain to a target task in a target domain. In this work, we use the VGG16 pre-trained model [14] as the backbone architecture for transferring knowledge

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2021.3083274, IEEE Journal of Biomedical and Health Informatics

to the ADNI dataset. Any other network architecture can also be used as the backbone architecture, since, the proposed ensemble is agnostic to the backbone architecture in use.

E. Random search

Every learning algorithm A has its own "*nuts and bolts*", called hyper-parameters (δ), that often control the efficacy with which it learns. For deep neural networks, δ can be very high dimensional due to the presence of a large number of hyper-parameters. Moreover, the dimension of the hyper-parameter space also increases with increase in depth of the network. Using grid search for optimizing (tuning) such high dimensional hyper-parameter spaces usually is extremely inefficient and computationally intractable. Moreover, grid search suffers from the *curse of dimensionality* as the number of points in the grid to be evaluated rises exponentially in high dimensional spaces. In such cases, the hyper-parameters can be tuned using manual search but reproducing results becomes a bottleneck. To tackle these problems, random search can be used for hyper-parameter tuning [15].

F. Snapshot ensemble

Snapshot ensemble is an efficient ensemble technique for deep neural networks that utilizes multiple locally minimum solutions to boost the model performance. A snapshot of the model is taken after certain number of epochs by using a cyclic annealing cosine learning rate schedule. The predictions from snapshots are then averaged during the test time to generate the final prediction [16].

G. Proposed model

In this paper, we propose a DL-architecture agnostic ensemble strategy for deep neural networks trained through transfer learning. Although we use pretrained models in this paper, an ensemble deep neural networks trained from scratch can also be constructed using our proposed method. Our proposed method combines the advantages of the random search hyperparameter search strategy along with the snapshot ensemble strategy. This combination renders our proposed method more efficient than true snapshot ensembles as it give more robust results within less training time as compared to snapshot ensembles. Our proposed method also avoids the large training time required for conventional ensembles of deep neural networks.

As explained earlier, random search drastically reduces the time in hyperparameter tuning for large hyperparameter spaces, as in case of deep neural networks which have a huge number of hyperparameters. Snapshot ensemble [16] produces a very efficient method for creating ensembles of deep neural networks with the training time of a single model. However, the time required to perform the hyperparameter search is not included when calculating the training time of a snapshot ensemble model. Considering the hyperparameter tuning time, the entire training time of the snapshot ensemble technique also increases by a significant amount. Moreover, using techniques like grid search and manual search is also highly efficient as the former takes exponential amount of time, whereas, the latter is not reproducible and may produce biased results. As mentioned in [15], depending upon the data, the subset of hyperparameters that have a significant impact on network performance differs. Searching for such a set of hyperparameters through grid search or manual search is highly inefficient as the problem may become computationally intractable when the hyperparameter space to be searched is of high dimensions like in the case of deep neural networks. In such a scenario, random search is a default choice. The proposed model combines the advantages of the snapshot ensemble technique along with the random search hyperparameter tuning technique to achieve a more robust and more generalizable ensemble of deep networks.

Contrary to snapshot ensembles, in this paper, we have chosen the ADAM algorithm as the optimizer for the network as it has performed better than SGD for deep networks in many cases [17]. We have also not used the cyclic annealing learning rate (LR) schedule. Further analysis needs to be done to incorporate the cyclic annealing LR schedule and is left as future work. In this work, the hyperparameters we consider for random search hyperparameter tuning are:

- Number of nodes in each fully connected layer.
- Number of fully connected layers.
- Mini-batch size.
- Learning rate.
- Number of epochs.

H. Analysis of the proposed model

The importance of hyperparameters differs with the dataset, as different sets of hyperparameters can be more important for the model performance for different datasets [15]. Let N be the number of random trials to be performed for random search. Let M be the number of models chosen for the ensemble. As we choose the best M models through cross validation, we end up with M models having the lowest expected generalization error among N trails. As each of the M models consist of a randomly chosen set of hyperparameter, it approximates a diverse function as compared to the others. Thus, the individual models satisfy the criteria of (a) low generalization error and (b) diversity for creating a robust ensemble model [16].

On the other hand, to create an ensemble of M models, the snapshot ensemble creates M snapshots of models after a certain number of iterations with a cyclic learning rate schedule. This helps to leverage the predictions of multiple local minima in the final ensemble. However, without an appropriate set of hyperparameters, snapshots with worse generalization error will be created. This will produce a final ensemble with higher generalization error as compared to the proposed model. Algorithm 1 gives the detailed algorithm for the proposed ensemble model. Ensemble of deep neural networks is not popular because of the high training time required to train individual networks. In our case, we do not need to train very deep networks from scratch. Thus, the proposed ensemble strategy is also computationally feasible. Moreover, we use batch normalization [18] and random search [15] during training, both of which further reduce the training time. We term the proposed model as 'Deep Transfer Ensemble (DTE)'.

In DTE, we transfer the convolution layers of VGG16 and only train the fully connected (FC) layers from scratch. The convolution layers act as generic image feature extractors. Whereas, training only the FC layers from scratch ensures the model learns specifically from the given training datasets. Thus, DTE combines the advantages of transfer learning and ensemble learning.

In DTE, we propose two different ensemble strategies - 'Within dataset ensemble' (DTE-W) and 'Across dataset ensemble' (DTE-AC). In DTE-W, we make an ensemble of models trained on a homogeneous set of images. We do an ensemble of n models with randomly chosen hyper-parameter settings (found using cross validation). That is, we do an ensemble of n VGG16 models with randomly chosen hyper-parameter settings trained on GM dataset to get the final classification for a GM image.

A similar procedure is repeated for WM and CSF datasets. The models in DTE-W have a diverse range of hyperparameters which allow them to model seeming distinct predictive functions. DTE-W aims to leverage this diversity to boost the classification performance. It is known that the learning rate is not the only defining factor in learning an accurate predictive function, as depending upon the dataset, different hyperparameters will be of different importance [15]. Thus, we go for model ensemble with many randomly chosen hyperparameters in DTE-W. This ensemble strategy serves two purposes - (1) It reduces the model training time [15] and (2) introduces diversity in the model which can boost the model performance [19].

In DTE-AC, we make an ensemble of n top models (found using cross validation) from each of GM, WM and CSF datasets. That is, we select the n best performing models (found using cross validation) from GM, WM and CSF, respectively and ensemble them to get the final classification result. AD is characterized by atrophy in GM as well as WM and an increase in CSF in many cases [20]. To confirm this hypothesis, we conducted statistical testing on the normalized GM, WM and CSF scans for NC and AD groups. The test was performed using SPM wherein subject age, gender and total intracranial volume (TIV) were added as covariates of no interest. A family-wise error (FWE) corrected p-value of 0.05 was used with an extent threshold of 0 voxels. Table I mentions the contrasts specified and the hypotheses tested for each of GM, WM and CSF images. Figures 1 - 3 shows the results obtained after statistical testing.

The idea for DTE-AC stems from the above mentioned hypothesis about atrophy in GM and WM and increase in CSF. Thus, we use all of GM, WM and CSF images for final classification. The proposed model leverages the diverse information provided by the GM, WM and CSF images by doing an ensemble of models trained on GM, WM and CSF data sets separately.

For the large dataset results, we report slice wise accuracy as done in Hon et al. [21]. However, classifying a subject based on a single slice might provide us with an overly optimistic or an overly pessimistic result. Thus, on the small dataset, we classify a subject to NC or AD by taking maximum voting of predictions of all the slices of a particular subject. Figure 4 gives a detailed view of the proposed ensemble model on the small dataset. Many previous works [22]–[24] only used GM and WM images for classification and outplayed the proficiency of additional information provided by CSF images. Moreover, researchers using DL techniques mostly used unsegmented brain images for AD classification [21], [25]–[28]. To the best of our knowledge, none have ensembled models trained on all three (GM, WM and CSF) tissue images for classification using deep CNNs.

Algorithm 1 Proposed ensemble strategy

```
Inputs:
Hyperparameter search space
    $H = [h_1, h_2, ..., h_t]$.
Number of random trials ($N$).
Number of models to be used in the ensemble ($M$).
Training data
    $X = [x_1, x_2, ..., x_l]$.
Training class labels
    $Y = [y_1, y_2, ..., y_l]$.
Testing data
    $X' = [x'_1, x'_2, ..., x'_p]$.
Training process:
13: Generate the set of $N$ random hyperparameter combinations from $H$.
14: Create $N$ different networks pertaining to $N$ combinations found in step 13.
15: Train each network on $X$ from step 14 using k-fold crossvalidation.
16: Choose best-$M$ networks from step 15.
17: Perform the final training for the best-$M$ networks selected from step 16 using entire training dataset.
Testing process:
19: Input a testing image $x'$ from the $X'$.
20: Generate output predictions from each of the best-$M$ networks from step 17.
21: Perform the final classification by doing an ensemble of predictions from step 20.
Output:
23: Final classification label for a testing image from $X'$.
```

Experiment	Tissue	Contrast	Hypothesis
NC vs AD	GM	[1 -1]	NC >AD
	WM	[1 -1]	NC >AD
	CSF	[-1 1]	NC <AD

TABLE I: Specified SPM contrasts

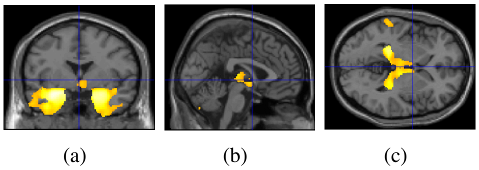
(a) (b) (c)

Fig. 1: Significant regions obtained from GM images

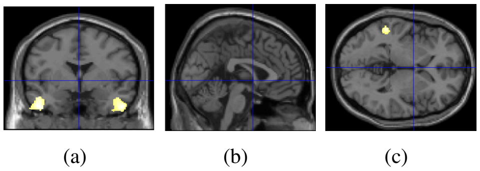

Fig. 2: Significant regions obtained from WM images

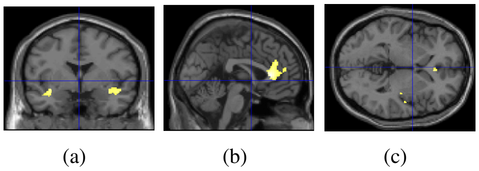

Fig. 3: Significant regions obtained from CSF images

III. Experiments

For the ADNI baseline data, we perform two binary classification tasks: NC vs AD and MCI vs AD. Whereas, for the small dataset, we only perform the NC vs AD classification task. For the large dataset, we choose the 'brain.finalsurfs.mgz' file from each of the subject and extract 32 2-D slices based on top 32 highest image entropy values of the slices. That is, image entropy is calculated for each slice of the 3D scan and were then arranged in descending order of their image entropy values. Then, we choose the top 32 slices from the ordered array of slices [21]. The total number of 2D slices are then considered for classification. We divide the total number of 2D slices into training and testing sets in the ratio $80-20$. That is, we use 80% of the total 2D slices as training set and rest 20% as testing set. The training set is again further divided into 80% – 20% datasets wherein 80% of the training set images are used for training the model during cross validation and the rest 20% of the training set images are used as the validation set during the 5-fold cross validation.

Whereas, for the small dataset, the scans from each of the subject groups (NC and AD) were randomly divided to form a training set of size 80 and a testing set of size 20. That is, from each subject group, 40 MRI scans were selected for training and MRI 10 scans were selected for testing. Then, 32 2-D slices were extracted from each 3-D scan based on highest entropy values and used for the training and testing purposes.

A. Hyper-parameter tuning and model selection

The convolution layers of the VGG-16 pre-trained deep neural network, which is openly available in the Keras library, were used as feature extractors from the MRI scans whereas the fully connected (FC) layers were trained from scratch. A random search of 20 independent trials for each of GM, WM and CSF data was performed for hyper-parameter tuning. We used 5-fold cross validation to obtain an unbiased estimate of the model performance. For each subject, 32 axial slices were extracted for each modality (GM, WM, CSF), based on their entropy values as mentioned in [21].

Thus, the training data for each of GM, WM and CSF images consisted of 2560 2-D slices and the testing data consisted of 640 2-D slices. Each individual slice of an MRI scan was converted to three channels before feeding it to the network. Separate models were trained on each of the GM, WM and CSF datasets independent of each other. The following hyper-parameters were considered for random search:

1) We chose $1, 2, 3$ or 4 FC hidden layers with uniform probability.
2) The number of neurons in first hidden layer was chosen uniformly in the range $[128, 256]$.
3) The number of neurons in the second hidden layer was chosen uniformly in the range $[64, 128]$.
4) The number of neurons in the third hidden layer was chosen uniformly in the range $[32, 64]$.
5) The number of neurons in the fourth hidden layer was chosen uniformly in the range $[10, 32]$.
6) The learning rate was sampled log-uniformly between 10^{-6} to 10^{-1}.
7) A mini-batch size of $8, 16, 32$ or 64 was chosen uniformly for the small dataset. Whereas, a mini-batch size of $8, 16, 32, 64, 128, 256$ was chosen uniformly for the large dataset.
8) Number of epochs between $[30, 100]$ was chosen uniformly.
9) We chose batch normalization for every FC hidden layer before applying the activation function. Bias values were not used in the FC hidden layers due to the use of batch normalization.
10) As batch normalization (BN) also produces a regularization effect, we first experimented without using other regularization techniques like dropout and l_1/l_2-regularization [18].

The range of hyperparameters 1) through 8) were fixed through manual testing on the validation set.

The following hyper-parameters of the model were fixed:

1) Optimizer - Adaptive Moment Estimation (Adam) with default values for β_1, β_2 and ϵ.
2) Activation function - ReLU activation function for hidden FC layers and the Sigmoid activation function for the output layer.
3) Loss function - Binary cross entropy.
4) Weight initializer of neurons - the default Glorot-uniform initializer.

B. Ensemble strategies

For the small dataset, we experimented with two different ensemble methods for the slice-wise ensemble - (a) slice-wise averaging and (b) slice-wise max-voting. In (a), we averaged over the predicted output probabilities for all the 32 slices. Then, the average value was rounded off to two decimal places. Then, the subject was classified as 'AD' if the final value was > 0.5 or as 'NC' if the final value was ≤ 0.5. In (b), we round off the predicted probability value for a slice to two decimal places and assign a slice to the class 'AD' if the final value is > 0.5 or to the class 'NC' if the final values is ≤ 0.5. Then, we take a maximum voting of classes 'AD' and 'NC' from all the 32 slices. The subject is then finally classified

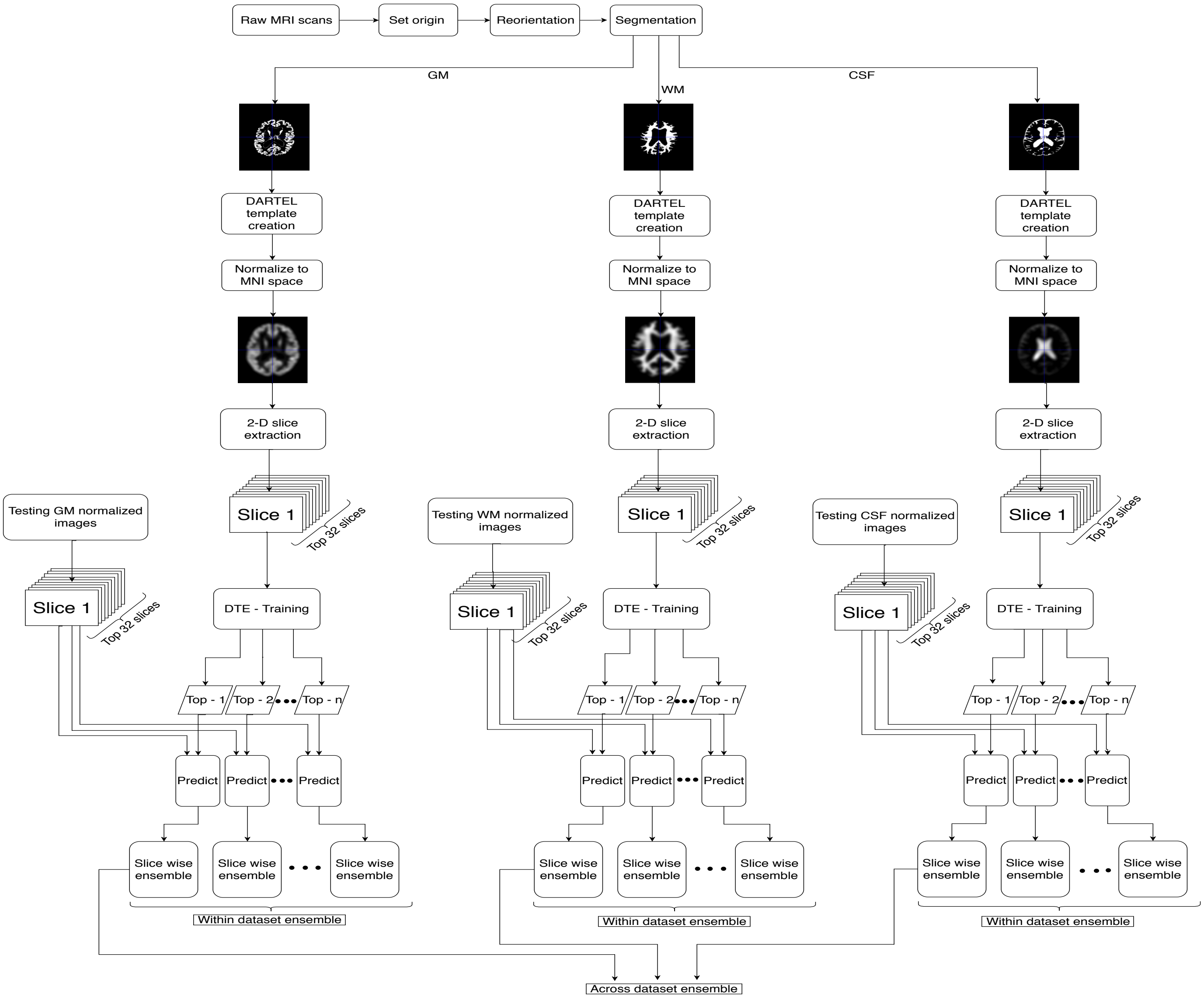

Fig. 4: Proposed model for NC vs AD classification on small dataset

based upon the number of votes received by the classes. It is of utmost importance to note that certain cases in (b), might result in equal number of votes for both the classes (16 votes to ‘NC’ and 16 votes to ‘AD’). The final classification of a subject in such scenarios may need more data for the model to train on, or, an external intervention by an expert medical practitioner. In this work, for such scenarios, we experimented with assigning the subject to the class ‘NC’ (making the model more specific) and also with assigning the subject to the class ‘AD’ (making the model more sensitive).

At the meta-level, the proposed ensemble strategies - DTE-W and DTE-AC are used. We experimented on two different settings - without dropout and with 50% dropout in all the FC hidden layers. Ideally, the number of models to be chosen for the ensemble must be chosen through cross validation. In this work, we have exhaustively experimented with the number of models chosen for doing the ensemble.

IV. Results and Discussions

In this section, we present and discuss the results obtained from the experiments performed. The accuracy shown for the large dataset is slice wise classification accuracy, as done in [21]. This however gives a more optimistic result due to data leakage, as is also the case in many other studies similar to our work [29]. We reserve further study on this issue as a future work. In this work, we have experimented using the model averaging and max voting ensemble schemes.

A. DTE-W

The notation followed in the tables is as follows: Top-n denotes the ensemble of n top models trained on a particular dataset (either GM, WM or CSF). We experimented with $n = 3, 5, 7$ and 9 for DTE-W, as can be seen in the aforementioned tables.

1) Large Dataset: Table II shows the results obtained from the proposed ensemble technique on the large ADNI dataset. We can clearly see the efficiency of the proposed ensemble

	Model averaging						**Max voting**					
	NC vs AD			**MCI vs AD**			**NC vs AD**			**MCI vs AD**		
	Acc	**Sen**	**Spec**	**Acc**	**Sen**	**Spec**	**Acc**	**Sen**	**Spec**	**Acc**	**Sen**	**Spec**
Top-3	98.98	98.57	99.31	98.61	96.57	99.56	99.05	98.66	99.38	90.09	69.92	99.56
Top-5	98.87	98.41	99.24	98.58	96.90	99.37	99.05	98.74	99.31	90.01	70.34	99.48
Top-7	98.79	98.32	99.17	98.58	96.99	99.33	98.84	98.66	99.17	90.01	70.34	99.25
Top-9	98.79	98.24	99.24	98.71	97.32	99.37	98.83	98.32	99.24	90.03	70.34	99.29

TABLE II: Results of DTE-W on ADNI baseline dataset

	Model averaging						**Max voting**					
	NC vs AD			**MCI vs AD**			**NC vs AD**			**MCI vs AD**		
	Acc	**Sen**	**Spec**	**Acc**	**Sen**	**Spec**	**Acc**	**Sen**	**Spec**	**Acc**	**Sen**	**Spec**
Top-3	85.05	87.32	82.28	82.39	90.89	64.32	84.75	86.36	82.79	63.80	56.38	79.61
Top-5	85.27	87.32	82.79	82.13	89.24	67.00	84.14	85.60	82.37	62.76	54.84	79.61
Top-7	84.71	87.04	81.81	82.58	90.73	65.24	85.05	86.77	82.87	63.19	55.55	79.44
Top-9	83.96	90.13	76.44	83.11	91.91	64.41	84.75	87.25	81.70	63.96	57.16	78.44

TABLE III: Results of DTE-W on ADNI baseline data with different train, validation and test split

as it reaches a highest accuracy of 99.05% on NC vs AD classification task and 98.71% on MCI vs AD classification task. From the NC vs AD (model averaging) section, we can observe that the accuracy decreases as we keep adding more models to the ensemble. Whereas, we get the highest accuracy for MCI vs AD (in the model averaging case) on top-9 models. This is mostly due to the fact that the optimal number of models chosen for the ensemble must be chosen through cross validation. However, in this work, we have computed the results on a number of different combinations extensively. From the max voting section we can observe that the accuracy of NC vs AD is highest for top-3 and top-5 models. For MCI vs AD, we get a top accuracy when we combine top-3 models in the ensemble. We also present results of another independent set of experiments (with separate train, validation and test split of the ADNI baseline dataset) performed from scratch in table III.

We can observe from table II, that the proposed model shows reduced difference in sensitivity and specificity values. This pattern can be observed more prominently for the MCI vs AD case. This conveys that the proposed model reduces the biased decisions produced by individual models significantly even on highly imbalanced data.

Table IV shows the comparison between previous studies that were similar to this work. That is, the studies that extract more than one 2-D slice from a 3D brain scan and used the information from the 2D slices for further classification were chosen for comparison. We can observe that the proposed model surpassed other models by a large extent. We must also note that there is a lot of heterogeneity in the datasets as well as the number of subjects in each study, hence, making a clear comparison is highly difficult. Table IV also shows the result of a 3 snapshot ensemble on NC vs AD classification with the hyperparameters: Initial learning rate $= 0.1$, momentum $= 0.9$, Batch size $= 32$, number of hidden FC layers $= 1$, number of nodes in hidden FC layer $= 50$, Epochs $= 30$, number of snapshots $= 3$. We can also observe that the proposed ensemble outperforms the snapshot ensemble by a large extent.

It is also of importance to note that other existing methods [16], [21], [30]–[35] do not deal with subject wise classification. Thus, they cannot directly be applied on our ADNI small dataset. However, the DTE can perform subject-wise classification. This is another advantage that DTE has over these existing methods.

We also present comparative results in table V, wherein all the methods are trained, validated and tested on the same ADNI baseline dataset (processed using freesurfer), which is the same as mentioned in section III but with a different train, validation and test split. We can observe that the proposed method outperforms other methods. We choose to exclude the methods [30]–[34] from this analysis since unlike deep neural networks, these do not learn features from the data explicitly.

2) Small dataset: Tables VI, VII and VIII give the results for DTE-W ensemble for GM, WM and CSF respectively without dropout regularization. We can observe from the table VI that the DTE-W ensemble strategy achieves an accuracy (ACC) of 85%, sensitivity (SEN) of 90% and specificity (SPEC) of 80% for slice-based averaging for GM dataset. A similar result can also be observed in the slice-wise max-voting (more specific) scheme. Most of the individual models trained on this dataset produced an accuracy less than DTE-W. However, individual models 1, 5 and 6 produced a result similar to DTE-W. It is important to notice that these results depict the final classification performance of individual models, and, knowing which individual model will perform best beforehand may not be obvious. Thus, in such scenarios, the DTE-W strategy gives more robust and unbiased results as it leverages predictions from multiple diverse models.

One can also see from table VII that the Top-5 and Top-7 accuracy, sensitivity and specificity are $75\%, 80\%$ and 70% respectively for WM dataset. The ensemble of Top-9 models in this case produces a further enhanced accuracy, sensitivity and specificity of 80% each.

We compared the results of ensemble of Top-n models in GM and WM as mentioned in tables VI and VII respectively. We can see that in case of GM, the ensemble of merely three Top models produces a classification accuracy of 85%. This may be due to the fact that GM is significantly atrophied in AD [36]. This finding is in accordance with previous findings wherein models trained using GM features achieved higher performances [4]. On the other hand, for the WM

Research	Number of subjects	Database	NC vs AD	MCI vs AD
Mahmood et al., 2013 [30]	230 (100 NC, 130 AD)	OASIS	89.22	-
Chyzyk et al., 2014 [31]	98 (49 NC, 49 AD)	OASIS	86	-
Gorji et al., 2014 [32]	500 (148 NC, 172 MCI, 180 AD)	ADNI	97.27	94.88
Wang et al., 2015 [33]	255 (35 NC, 220 AD)	Harvard medical school	100	-
Jha et al., 2017 [34]	126 (98 NC, 28 AD)	OASIS	90.06 ± 0.01	-
Hon and Khan, 2017 [21]	100 (50 NC, 50 AD)	ADNI	96	-
Wang et al., 2018 [35]	196 (98 NC, 98 AD)	OASIS	98	-
Snapshot ensemble - 3	415 (228 NC, 187 AD)	ADNI (baseline)	54.63	-
Proposed	**813 (228 NC, 187 AD, 398 MCI)**	**ADNI (baseline)**	**99.05**	**98.71**
Proposed	**100 (50 NC, 50 AD)**	**ADNI (small dataset)**	**85**	**-**

TABLE IV: Comparison of the proposed method with other related work

Research	Number of subjects	Database	NC vs AD	MCI vs AD
Hon and Khan, 2017 [21]	**813 (228 NC, 187 AD, 398 MCI)**	**ADNI (Baseline)**	84.14	82.26
Wang et al., 2018 [35]			79.93	76.86
Snapshot ensemble - 3			54.93	68.02
Proposed			**85.27**	**83.11**

TABLE V: Comparison of the proposed method with other related work on dataset from table III

	GM								
	Slice Average			Max Voting (More Sensitive)			Max Voting (More Specific)		
	ACC	SEN	SPEC	ACC	SEN	SPEC	ACC	SEN	SPEC
Top-3	85	90	80	80	90	70	85	90	80
Top-5	85	90	80	80	90	70	85	90	80
Top-7	85	90	80	80	90	70	85	90	80
Top-9	80	90	70	80	90	70	80	90	70

TABLE VI: DTE-W on GM images (without dropout)

	WM								
	Slice Average			Max Voting (More Sensitive)			Max Voting (More Specific)		
	ACC	SEN	SPEC	ACC	SEN	SPEC	ACC	SEN	SPEC
Top-3	70	80	60	70	80	60	75	80	70
Top-5	75	80	70	70	80	60	75	80	70
Top-7	75	80	70	75	80	70	75	80	70
Top-9	80	80	80	80	80	80	80	80	80

TABLE VII: DTE-W on WM images (without dropout)

	CSF								
	Slice Average			Max Voting (More Sensitive)			Max Voting (More Specific)		
	ACC	SEN	SPEC	ACC	SEN	SPEC	ACC	SEN	SPEC
Top-3	60	50	70	60	50	70	60	50	70
Top-5	60	50	70	60	50	70	60	50	70
Top-7	60	50	70	60	50	70	60	50	70
Top-9	60	50	70	60	50	70	60	50	70

TABLE VIII: DTE-W on CSF images (without dropout)

	GM + WM + CSF								
	Slice Average			Max Voting (More Sensitive)			Max Voting (More Specific)		
	ACC	SEN	SPEC	ACC	SEN	SPEC	ACC	SEN	SPEC
3-Top-1	75	70	80	85	70	100	85	70	100
3-Top-3	85	90	80	85	90	80	85	90	80
3-Top-5	80	80	80	80	80	80	80	80	80
3-Top-7	80	80	80	80	80	80	85	80	90
3-Top-9	80	80	80	85	90	80	85	80	90

TABLE IX: DTE-AC (without dropout)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2021.3083274, IEEE Journal of Biomedical and Health Informatics

Subject	Group	Top-1	Top-2	Top-3	DTE-W	Top-1	Top-2	Top-3	DTE-W	Top-1	Top-2	Top-3	DTE-W
		SA				MV (SEN)				MV (SPEC)			
1	AD	0.8649	0.8685	0.7447	0.826 → 1	1	1	1	1	1	1	1	1
2	NC	**0.4991**	**0.5247**	**0.3684**	**0.464 → 0**	1	1	0	1	**0**	**1**	**0**	**0**
3	NC	0.2084	0.1944	0.19	0.1976 → 0	0	0	0	0	0	0	0	0
4	NC	0.0039	0.0001	0.0317	0.0119 → 0	0	0	0	0	0	0	0	0
5	NC	0.2948	0.3443	0.4087	0.3493 → 0	0	0	0	0	0	0	0	0
6	NC	0.2336	0.279	0.0975	0.2033 → 0	0	0	0	0	0	0	0	0
7	NC	0.7658	0.8055	0.901	0.8241 → 1	1	1	1	1	1	1	1	1
8	NC	0.6437	0.6194	0.533	0.5987 → 1	1	1	1	1	1	1	1	1
9	NC	0.3916	0.4734	0.3431	0.4027 → 0	0	0	0	0	0	0	0	0
10	NC	0.0669	0.0973	0.125	0.0964 → 0	0	0	0	0	0	0	0	0
11	NC	0.2164	0.1459	0.264	0.2088 → 0	0	0	0	0	0	0	0	0
12	AD	0.9668	0.884	0.9815	0.9441 → 1	1	1	1	1	1	1	1	1
13	AD	0.2479	0.1697	0.1591	0.1922 → 0	0	0	0	0	0	0	0	0
14	AD	0.9683	0.9969	0.9432	0.9695 → 1	1	1	1	1	1	1	1	1
15	AD	0.7269	0.7189	0.7564	0.7341 → 1	1	1	1	1	1	1	1	1
16	AD	0.9694	0.9751	0.9604	0.9683 → 1	1	1	1	1	1	1	1	1
17	AD	0.7668	0.7775	0.7552	0.7665 → 1	1	1	1	1	1	1	1	1
18	AD	0.5113	0.5724	0.5995	0.5611 → 1	1	1	1	1	1	1	1	1
19	AD	0.7641	0.8092	0.7336	0.769 → 1	1	1	1	1	1	1	1	1
20	AD	0.6221	0.6735	0.5698	0.6218 → 1	1	1	1	1	1	1	1	1

TABLE X: Effect of DTE-W trained on GM images on final classification. SA - Slice Average, MV (SEN) - Max Voting (More Sensitive), MV (SPEC) Max Voting (More Specific), 0 - NC, 1 - AD.

Subject	Group	SA	MV (more sensitive)	MV (more specific)
1	AD	0.5033 → 0	0	0
2	NC	**0.367 → 0**	**0**	**0**
3	NC	0.5424 → 1	**0**	**0**
4	NC	**0.0762 → 0**	**0**	**0**
5	NC	**0.3342 → 0**	**0**	**0**
6	NC	**0.4099 → 0**	**0**	**0**
7	NC	0.5583 → 1	**0**	**0**
8	NC	**0.4186 → 0**	**0**	**0**
9	NC	**0.2648 → 0**	**0**	**0**
10	NC	**0.1184 → 0**	**0**	**0**
11	NC	**0.3464 → 0**	**0**	**0**
12	AD	**0.6304 → 1**	**1**	**1**
13	AD	0.4719 → 0	0	0
14	AD	**0.7663 → 1**	**1**	**1**
15	AD	**0.8067 → 1**	**1**	**1**
16	AD	**0.6275 → 1**	**1**	**1**
17	AD	**0.8212 → 1**	**1**	**1**
18	AD	**0.645 → 1**	**1**	**1**
19	AD	**0.6858 → 1**	**1**	**1**
20	AD	**0.5019 → 0**	**0**	**0**

TABLE XI: Effect of DTE-AC on final classification. SA - Slice Average, MV - Max Voting, 0 - NC, 1 - AD.

images, increasing the number of models for ensemble leads to increase in classification performance. This maybe due to the increase in diversity of the model. However, for GM dataset, doing an ensemble of models beyond a certain threshold decreases the accuracy, as can be observed by the results of ensemble of Top-9 models. This maybe due to the introduction of highly non-optimal models which degrade the classification performance. For the case of CSF images, results of DTE-W can be observed from table VIII. One can observe that the ensemble strategy produces no effect on the classification performance. This maybe due to the highly similar nature of CSF images in both the NC and AD groups.

The table X shows the effect of the DTE-W (without dropout) ensemble strategy with $n = 3$ on the final classification results for each of the twenty test subjects. The actual class of the subject is represented by the column 'Group'. The columns Top-1, Top-2 and Top-3 represent the output after slice-wise ensemble for each test subject. SA, MV (SEN) and MV (SPEC) are results for slice averaging, max voting (more sensitive) and max voting (more specific) schemes. The values in bold represent the positive effect (correct classification) produced by the ensemble. The underlined values represent the negative effect (incorrect classification) produced by the ensemble. We exclude the results in which all the models result in similar classification, for example, in table X, all the models for SA result in values > 0.5 for subject 1. In the SA scheme for subject 2 (table X), the Top-2 model produces a probability value of 0.5247 which classifies the subject to the class 1 (AD), since probability values > 0.5 belong to class 1. But, averaging over the probability values of all the three models produce a final result of 0.464 which classifies the subject to class 0 (NC). A similar result can be seen for subject 2 for MV (SPEC) case. On the contrary, for the MV (SEN) for subject 2 the inclusion of miss-classification of Top-1 and Top-2 models along with the correct classification of Top-3 models results in miss-classification of the subject.

B. DTE-AC

Table IX shows the results of DTE-AC without any dropout applied on the small dataset. DTE-AC utilizes the complimentary information provided by GM, WM and CSF images. The notation followed in the tables is as follows: 3-Top-n denotes the ensemble of Top-n models each from GM, WM and CSF datasets. In this work, we experimented with $n = 1, 3, 5, 7$ and 9 for DTE-AC.

We can observe from table IX that for the slice-wise max-voting schemes, DTE-AC classifies with high accuracy, sensitivity and specificity of 85%, 90% and 90% in many cases. One can also notice that ensemble of Top-1 models achieves a high accuracy rate of 85% in the max-voting scheme. The individual models trained on WM and CSF datasets achieved a maximum accuracy of 60%. The individual models did not perform outstandingly, however, they provided enough diversity to DTE-AC along with models trained on GM dataset to boost the classification accuracy up to 85%.

The Table XI shows the effect of the DTE-AC (without dropout) ensemble strategy with $n = 1$. The bold values shows majority of the values get a correct classification due to DTE-AC. The underlined values mark the miss-classified samples.

C. Comparison of DTE-W and DTE-AC

On comparing the results of DTE-W and DTE-AC, one can see that the simple averaging of models produces better results in DTE-W than the max-voting scheme. In DTE-W, taking a max-voting of individual models, which already produce biased results increases the bias in the ensemble, thereby, degrading the performance. Contrary to this, the max-voting scheme produces better results in DTE-AC than the model averaging scheme. This is due to the fact that GM, WM and CSF are significantly distinct feature sets and averaging over models trained on them increases the variance hampering the model performance.

V. Conclusions and future works

In this work, we presented a novel ensemble model (DTE) for the classification of Alzheimer's disease. The DTE utilizes a combination of deep learning, transfer learning and ensemble learning. DTE exploits the diversity of individual models with randomly chosen hyperparameter with low generalization error to produce more accurate and robust results. For the large ADNI baseline dataset, the DTE achieved a maximum classification accuracy of 99.09% for NC vs AD and 98.71% for MCI vs AD classification tasks. For the small dataset chosen from ADNI, the DTE achieved a maximum classification accuracy of 85% for NC vs AD.

By observing the main results from DTE-W, we can summarize that an ensemble of deep models trained with random hyperparameters produces better generalization as compared to that of individual models. This is due to the fact that each model in the ensemble reaches a different local optima in the non-convex loss surface. Models with bad local optima can be avoided through crossvalidation. Similarly, by observing the main results from DTE-AC, we can summarize that along with the diverse nature of models in the ensemble, further diversity can be leveraged by including different feature views with complementary features.

The DTE currently lacks an optimal strategy for choosing the models for ensemble. It also lacks a strategy to provide appropriate weightage to each individual model based on its usefulness in the ensemble. In the future, we plan to address these issues. We also plan to investigate the biological relevance of diverse predictions to further understand their relation with onset and progression of Alzheimer's disease.

Another interesting direction of work with DTE would be to use Bayesian optimization instead of random search. DTE could also be improved by incorporating different cyclical learning rate (LR) routines, wherein, even the parameters of the LR routine could be randomized. Another line of work that would improve DTE further would be to explicitly measure the amount of diversity being added to the model instead of only relying on crossvalidation. This could make DTE more automated in nature and also avoid adding derogatory models. The codes are available on `https://github.com/mtanveer1`

VI. Acknowledgment

This work is supported by Department of Science and Technology, INDIA under Ramanujan fellowship scheme grant no. SB/S2/RJN-001/2016, Council of Scientific & Industrial Research (CSIR), New Delhi, INDIA under Extra Mural Research (EMR) scheme grant no. 22(0751)/17/EMR-II, and Science and Engineering Research Board (SERB), INDIA under Early Career Research Award scheme grant no. ECR/2017/000053. We gratefully acknowledge the Indian Institute of Technology Indore for providing facilities and support.

References

[1] R. Brookmeyer, E. Johnson, K. Ziegler-Graham, and H. M. Arrighi, “Forecasting the global burden of Alzheimer‘s disease,” *Alzheimer‘s & Dementia*, vol. 3, no. 3, pp. 186–191, 2007.

[2] E. Pellegrini, L. Ballerini, M. D. C. V. Hernandez, F. M. Chappell, V. González-Castro, D. Anblagan, S. Danso, S. Muñoz-Maniega, D. Job, and C. Pernet, “Machine learning of neuroimaging for assisted diagnosis of cognitive impairment and dementia: A systematic review,” *Alzheimer's & Dementia: Diagnosis, Assessment & Disease Monitoring*, vol. 10, pp. 519–535, 2018.

[3] C. Zheng, Y. Xia, Y. Pan, and J. Chen, “Automated identification of dementia using medical imaging: a survey from a pattern classification perspective,” *Brain Informatics*, vol. 3, no. 1, pp. 17–27, 2016.

[4] M. Tanveer, B. Richhariya, R. Khan, A. Rashid, P. Khanna, M. Prasad, and C. Lin, “Machine learning techniques for the diagnosis of Alzheimer's disease: A review,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 1s, pp. 1–35, 2020.

[5] B. Richhariya, M. Tanveer, A. Rashid, and A. D. N. Initiative, “Diagnosis of Alzheimer's disease using universum support vector machine based recursive feature elimination (usvm-rfe),” *Biomedical Signal Processing and Control*, vol. 59, p. 101903, 2020.

[6] R. U. Khan, M. Tanveer, R. B. Pachori, and A. D. N. I. (ADNI), “A novel method for the classification of Alzheimer's disease from normal controls using magnetic resonance imaging,” *Expert Systems*, vol. 38, no. 1, p. e12566, 2021.

[7] R. Sharma, T. Goel, M. Tanveer, S. Dwivedi, and R. Murugan, “FAF-DRVFL: fuzzy activation function based deep random vector functional links network for early diagnosis of Alzheimer's disease,” *Applied Soft Computing*, vol. 106, p. 107371, 2021.

[8] D. S. Knopman and R. C. Petersen, “Mild cognitive impairment and mild dementia: a clinical perspective,” in *Mayo Clinic Proceedings*, vol. 89, no. 10. Elsevier, 2014, pp. 1452–1459.

[9] T. L. Michaud, D. Su, M. Siahpush, and D. L. Murman, “The risk of incident mild cognitive impairment and progression to dementia considering mild cognitive impairment subtypes,” *Dementia and Geriatric Cognitive Disorders Extra*, vol. 7, no. 1, pp. 15–29, 2017.

[10] M. A. Ganaie, M. Hu, M. Tanveer, and P. N. Suganthan, “Ensemble deep learning: A review,” *arXiv preprint arXiv:2104.02395*, 2021.

[11] Y. Ren, L. Zhang, and P. N. Suganthan, “Ensemble classification and regression-recent developments, applications and future directions,” *IEEE Computational Intelligence Magazine*, vol. 11, no. 1, pp. 41–53, 2016.

[12] M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. Muharemagic, “Deep learning applications and challenges in big data analytics,” *Journal of Big Data*, vol. 2, no. 1, p. 1, 2015.

[13] J. Ashburner, “A fast diffeomorphic image registration algorithm,” *Neuroimage*, vol. 38, no. 1, pp. 95–113, 2007.

[14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[15] J. Bergstra and Y. Bengio, “Random search for hyper-parameter optimization,” *Journal of Machine Learning Research*, vol. 13, no. Feb, pp. 281–305, 2012.

[16] G. Huang, Y. Li, G. Pleiss, Z. Liu, J. E. Hopcroft, and K. Q. Weinberger, “Snapshot ensembles: Train 1, get m for free,” *arXiv preprint arXiv:1704.00109*, 2017.

[17] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[18] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *arXiv preprint arXiv:1502.03167*, 2015.

[19] G. Brown, J. Wyatt, R. Harris, and X. Yao, “Diversity creation methods: a survey and categorisation,” *Information Fusion*, vol. 6, no. 1, pp. 5–20, 2005.

[20] N. Mattsson, M. Schöll, O. Strandberg, R. Smith, S. Palmqvist, P. S. Insel, D. Hägerström, T. Ohlsson, H. Zetterberg, and J. Jögi, “18f-av-1451 and csf t-tau and p-tau as biomarkers in Alzheimer’s disease,” *EMBO Molecular Medicine*, vol. 9, no. 9, pp. 1212–1223, 2017.

[21] M. Hon and N. M. Khan, “Towards Alzheimer’s disease classification through transfer learning,” in *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2017, pp. 1166–1169.

[22] L. Khedher, J. Ramírez, J. M. Górriz, A. Brahim, F. Segovia, and Alzheimer‘s Disease Neuroimaging Initiative, “Early diagnosis of Alzheimer‘s disease based on partial least squares, principal component analysis and support vector machine using segmented MRI images,” *Neurocomputing*, vol. 151, pp. 139–150, 2015.

[23] F. Segovia, J. Górriz, J. Ramírez, D. Salas-Gonzalez, and I. Álvarez, “Early diagnosis of Alzheimer‘s disease based on partial least squares and support vector machine,” *Expert Systems with Applications*, vol. 40, no. 2, pp. 677–683, 2013.

[24] L. Mesrob, B. Magnin, O. Colliot, M. Sarazin, V. Hahn-Barma, B. Dubois, P. Gallinari, S. Lehéricy, S. Kinkingnéhun, and H. Benali, “Identification of atrophy patterns in Alzheimer‘s disease based on SVM feature selection and anatomical parcellation,” in *International Workshop on Medical Imaging and Virtual Reality*. Springer, 2008, pp. 124–132.

[25] H.-I. Suk and D. Shen, “Deep ensemble sparse regression network for Alzheimer‘s disease diagnosis,” in *International Workshop on Machine Learning in Medical Imaging*. Springer, 2016, pp. 113–121.

[26] H.-I. Suk, S.-W. Lee, D. Shen, and A. D. N. Initiative, “Deep ensemble learning of sparse regression models for brain disease diagnosis,” *Medical Image Analysis*, vol. 37, pp. 101–113, 2017.

[27] R. Jain, N. Jain, A. Aggarwal, and D. J. Hemanth, “Convolutional neural network based Alzheimer‘s disease classification from magnetic resonance brain images,” *Cognitive Systems Research*, 2019.

[28] H. Wang, Y. Shen, S. Wang, T. Xiao, L. Deng, X. Wang, and X. Zhao, “Ensemble of 3D densely connected convolutional network for diagnosis of mild cognitive impairment and Alzheimer‘s disease,” *Neurocomputing*, vol. 333, pp. 145–156, 2019.

[29] J. Wen, E. Thibeau-Sutre, J. Samper-Gonzalez, A. Routier, S. Bottani, S. Durrleman, N. Burgos, and O. Colliot, “Convolutional neural networks for classification of alzheimer’s disease: Overview and reproducible evaluation,” *arXiv preprint arXiv:1904.07773*, 2019.

[30] R. Mahmood and B. Ghimire, “Automatic detection and classification of Alzheimer’s disease from MRI scans using principal component analysis and artificial neural networks,” in *2013 20th International Conference on Systems, Signals and Image Processing (IWSSIP)*. IEEE, 2013, pp. 133–137.

[31] D. Chyzhyk, A. Savio, and M. Graña, “Evolutionary ELM wrapper feature selection for Alzheimer’s disease cad on anatomical brain MRI,” *Neurocomputing*, vol. 128, pp. 73–80, 2014.

[32] H. Gorji and J. Haddadnia, “A novel method for early diagnosis of Alzheimer’s disease based on pseudo Zernike moment from structural MRI,” *Neuroscience*, vol. 305, pp. 361–371, 2015.

[33] S. Wang, Y. Zhang, Z. Dong, S. Du, G. Ji, J. Yan, J. Yang, Q. Wang, C. Feng, and P. Phillips, “Feed-forward neural network optimized by hybridization ofPSO and ABC for abnormal brain detection,” *International Journal of Imaging Systems and Technology*, vol. 25, no. 2, pp. 153–164, 2015.

[34] D. Jha, J.-I. Kim, and G.-R. Kwon, “Diagnosis of Alzheimer’s disease using dual-tree complex wavelet transform, PCA, and feed-forward neural network,” *Journal of Healthcare Engineering*, vol. 2017, 2017.

[35] S.-H. Wang, P. Phillips, Y. Sui, B. Liu, M. Yang, and H. Cheng, “Classification of Alzheimer’s disease based on eight-layer convolutional neural network with leaky rectified linear unit and max pooling,” *Journal of Medical Systems*, vol. 42, no. 5, p. 85, 2018.

[36] V. M. Anderson, J. M. Schott, J. W. Bartlett, K. K. Leung, D. H. Miller, and N. C. Fox, “Gray matter atrophy rate as a marker of disease progression in AD,” *Neurobiology of Aging*, vol. 33, no. 7, pp. 1194–1202, 2012.