



# Detecting the Stages of Alzheimer's Disease with Pre-trained Deep Learning Architectures

Serkan Savaş<sup>1</sup>

Received: 10 June 2021 / Accepted: 18 August 2021  
© King Fahd University of Petroleum & Minerals 2021

## Abstract

Deep learning algorithms have begun to be used in medical image processing studies, especially in the last decade. MRI is used in the diagnosis of Alzheimer's disease, a type of dementia disease, which is the 7th among the diseases that cause death in the world. Alzheimer's disease has no known cure in the literature, so it is important to attempt treatment before starting the irreversible path by diagnosing the pre-illness stages. In this study, the previous stages of Alzheimer's disease were classified as normal, mild cognitive impairment, and Alzheimer's disease through brain MRIs. Different models using CNN architecture were used to classify 2182 image objects obtained from the ADNI database. The study was presented in a very comprehensive comparison framework, and the performances of 29 different pre-trained models on images were evaluated. The accuracy values of each model and the precision, specificity, and sensitivity rates of each class were determined. In the study, the EfficientNetB0 model provided the highest accuracy at the test stage with an accuracy rate of 92.98%. In the comparative evaluation stage with the confusion matrix, the highest rates of precision, sensitivity, and specificity values of the Alzheimer's disease class were achieved by EfficientNetB3 (89.78%), EfficientNetB2 (94.42%), and EfficientNetB3 (97.28%) models, respectively. The results of the study showed that among the pre-trained models, EfficientNet models achieved a high rate of classification performance as the models with the highest performance. This study will contribute to clinical studies in early prevention by detecting Alzheimer's disease before it occurs.

**Keywords** Deep learning · Convolutional neural network · Alzheimer's disease · Mild cognitive impairment · Magnetic resonance image classification · Pre-trained models

## 1 Introduction

Alzheimer's disease (AD) and other forms of dementia ranked as the 7th leading cause of death [1]. AD is the most common type of dementia covering 60%–80% of dementia cases. Dementia is a syndrome in which there is a deterioration in cognitive function beyond what might be expected from normal aging. It causes damage to memory, thinking, orientation, comprehension, calculation, learning capacity, language, and ability to distinguish but consciousness is not affected. The impairment in cognitive function is commonly accompanied and occasionally preceded, by deterioration in emotional control, social behavior, or motivation [2].

Changes in amyloid precursor protein (APP) cleavage and production of the APP fragment betaamyloid ( $A\beta$ ) along with hyperphosphorylated tau protein aggregation coalesce to cause reduction in synaptic strength, synaptic loss, and neurodegeneration. Metabolic, vascular, and inflammatory changes, as well as comorbid pathologies, are key components of the disease process [3].

Dementia is usually chronic or progressive. This progressive process may follow a path starting from the cognitively normal (CN) stage, resulting in mild cognitive impairment (MCI) stage, and AD. For being incurable neurodegenerative disorder [4,5], it is important to detect early stages of AD before it happens. The greatest risk factors for AD are old age, family history, and the presence of the Apolipoprotein e4 (ApoE4) gene in a person's genome [6]. MCI is one of the early stages of AD and causes a slight but noticeable and measurable decline in cognitive abilities, including memory and thinking skills which may cause AD or another dementia.

✉ Serkan Savaş  
serkansavas@karatekin.edu.tr

<sup>1</sup> Department of Computer Engineering, Faculty of Engineering, Çankırı Karatekin University, 18100 Çankırı, Turkey



Early detection of MCI and appropriate treatment methods can delay the development of AD [7].

Due to the irreversible end of cognitive disorders, various studies are continuing to detect in the progressive process. Current medical research has been supported by state-of-the-art analysis algorithms, especially in recent years, so AD researches. Most researches are based on brain images and different graph-based learning [8] algorithms used to identify subjects at different progression stages of AD. Not only brain images used for diagnosis of disease but also different factors such as gender, age, education [9] are being investigated for neurocognitive network connectivity differences between healthy aging and MCI and associations with cognitive status and these factors [10]. In different studies, biological fluid biomarkers such as invasive CSF and non-invasive (blood, saliva, urine, and tears) biomarkers [11,12], dry biomarkers such as structural and functional imaging, and ocular biomarkers of AD [12] have been extensively examined, and the results have been explained. While some research focuses on clinical results to investigate the relationship between progranulin expression in peripheral blood and clinical diagnosis of AD and MCI using microarrays [13] and variant analysis [14], methods such as machine learning (ML) and deep learning (DL) have been used frequently in the context of artificial intelligence (AI) studies [15], especially recently.

Since the day it started with the question “Can machines think?” [16], AI [17] has become to produce high accuracy results for many problems. The neurocognitive researches proposed different ML algorithms such as dual-tree complex wavelet transforms, principal component analysis, linear discriminant analysis, extreme learning machine [18], dynamic connectivity networks learning framework [19], one-way analysis of variance analysis on the multiscale entropy [20], N-fold cross-validation [21], Bayesian latent time joint mixed-effects [22], and ensemble feature selection approach [23] for early detection and classification of AD and for minimizing the mortality rate on brain magnetic resonance images (MRIs). The MRI scan of the brain shows the brain structures, shrinkage of the brain, vascular irregularities, and any other structural changes that might cause cognitive dysfunction [24]. Besides not only MRIs used for detection and classification but also positron emission tomography (PET) images are used with a random forest-robust support vector machine for the identification of MCI [25].

DL algorithms have begun to assist in different applications such as health services, translation services, advertising services, driverless vehicles, film suggestions, chatbot, page suggestions, and many more with successful classification and/or detection solutions [26]. Different DL approaches were also used for AD recognition [27] and to predict MCI-to-AD conversion such as convolutional neural network (CNN) [28]. Classification studies were performed with multiple cluster dense convolutional neural networks [29],

random neural network cluster [30], and diagnosis researches performed with a combination of sparse regression models with deep neural network (DNN) [31]. These DL algorithms used MRIs for analyses besides PET images were used to construct cascaded CNNs to learn the multi-level and multi-modal features [32] and to predict the final diagnosis of AD and MCI [33].

As being widely used in computer identification, autonomous vehicles, natural language processing, handwritten character recognition, signature verification, voice and video recognition, big data [34], image processing, and medical image processing; DL approaches are necessary for automated medical decision-making systems because non-automated processes are more expensive, demand intensive labor and therefore subject to human-induced errors [35]. In this study, performance comparisons and classification successes of pre-trained CNN architectures such as AlexNet [36], ZFNet [37], ResNet 50-101-152-50V2-101V2-152V2 [38], and VGG 16-19 [39], which are successful on the ImageNet competition (ImageNet Large Scale Visual Recognition Challenge - ILSVRC), and some other pre-trained models such as LeNet [40], Xception, MobileNet, MobileNetV2, InceptionV3, InceptionResNetV2, DenseNet 121-169-201, NASNetMobile, NASNetLarge, and EfficientNetB0-B7 from Keras Application Library [41] (29 models in total) were evaluated on CN, MCI, and AD with MRIs data from Alzheimer’s Disease Neuroimaging Initiative [42] study platform.

## 2 Materials and Methods

Data used in the preparation of this article were obtained from the Alzheimer’s Disease Neuroimaging Initiative (ADNI) database ([adni.loni.usc.edu](http://adni.loni.usc.edu)). For experimental tests, *ADNI Complete 3Yr 1.5T* data of ADNI is chosen. This dataset contains T1-weighted sagittal MR images. There are 2182 image items with *.nii* format and size of *44.16 GB* which is provided as Archived, NIFTI, MINC, and Analyze options. After downloading the dataset, an image pre-processing is applied at three steps:

1. Converting images from *.nii* format to *.png* format with Python code: In this step, there are 166 frames exported from each *.nii* image, and two middle images were chosen (number 83 and number 84) from each image item.
2. Splitting the dataset randomly with Python code: In this step, the images are separated as AD, MCI, and CN sub-clusters. From these sub-clusters, train and test images are chosen randomly with the rate of 90% and 10%, respectively.
3. Resizing and splitting train and validation images: In this step, inside the model program before training, the train

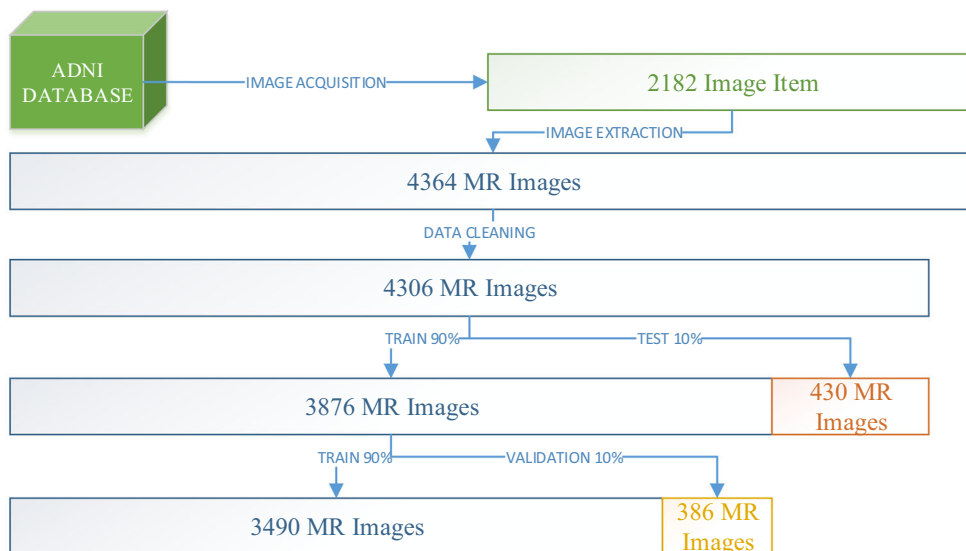


Fig. 1 Image pre-processing steps

Table 1 Summary of the dataset

Type	CN	MCI	AD	Total
Number of patients	135	148	99	382
Male patients	66	105	52	223
Female patients	69	43	47	159
Mean of age	77.24	75.63	75.81	76.23
Mean of visits	4.20	4.30	3.32	3.94
Number of images	1476	1924	906	4306
Train images	1196	1560	734	3490
Validation images	132	172	82	386
Test images	148	192	90	430

images are split into train and validation images randomly with the rate of 90% and 10%, respectively. Besides, all images are resized to 224 × 224 resolution.

The processes of acquiring images, image extraction, data cleaning, and splitting the images into train, validation, and test datasets are shown in Fig. 1.

The summary of the dataset is shown in Table 1 after applying image pre-processing steps to the dataset.

It is seen in Table 1 that there is a total of 382 patients consisting of 223 male and 159 female patients in the dataset. The number of CN, MCI, and AD labeled patients is 135, 148, and 99, respectively. The mean age of patients is 76.2. Data cleaning was performed on 4364 MR images obtained after the image extraction process was performed, and 58 inappropriate images were cleared from the database. As a result, experiments were carried out on 4306 images. 430 (10%) of these images were randomly selected as test data, which

was separated as data that the models would not see during training. Of the remaining 3876 images, 3490 (90%) images were used for training and 386 (10%) images were used for validation. Sample MRIs belonging to all three classes are shown in Fig. 2.

To analyze the data used in the study, DNN architectures created with the CNN algorithm were used. In the structure of the CNN algorithm, there are various layers and functions such as convolution layers, pooling layers, fully connected layers, activation functions, and dropout layers. In models designed using this algorithm, many parameters such as the number of layers, activation functions, dropout rate, and locations, the number of epochs, batch size, learning rate, and optimization method are determined by the programmer who created the model. Researchers can create their models or work on the data using the weights of previously created (pre-trained) models with the transfer learning method. In real-world applications, it is hard or sometimes impossible to achieve or re-collect enough data for training a CNN model. Also, overfitting and convergence issues are the potential problems that may be faced with for training a deep CNN. For this reason, the transfer-learning approach should be preferred to solve these kinds of problems [43]. A sample CNN structure is shown in Fig. 3.

In the convolutional layer, each unit is connected to local patches in the feature maps of the previous layer through a set of weights called a filter bank; thus, an organized feature map units are created. All units in a feature map share the same filter bank. Different feature maps in a layer use different filter banks [44]. The convolution of two functions ( $f * g$ ) in the finite range  $[0, t]$  is defined as Eq. (1) [45].

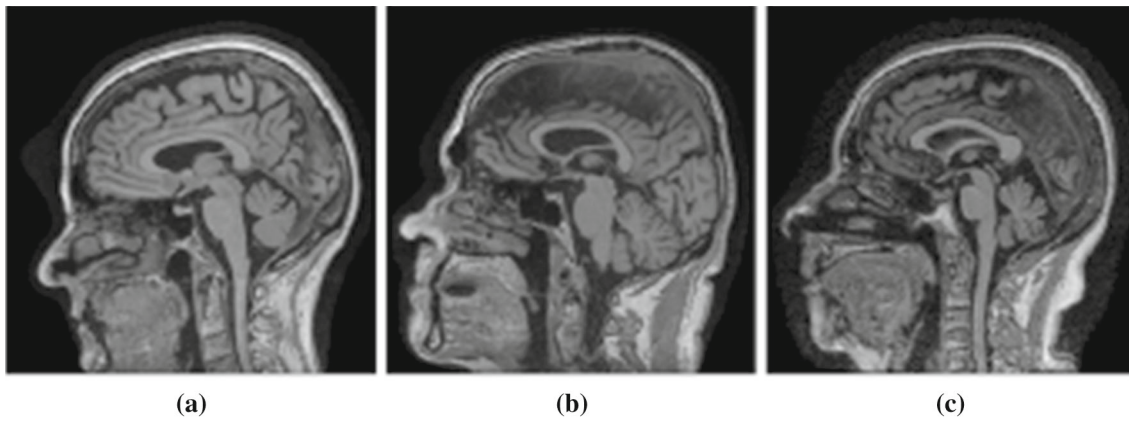


Fig. 2 Sample MRIs of a AD, b CN, c MCI classes

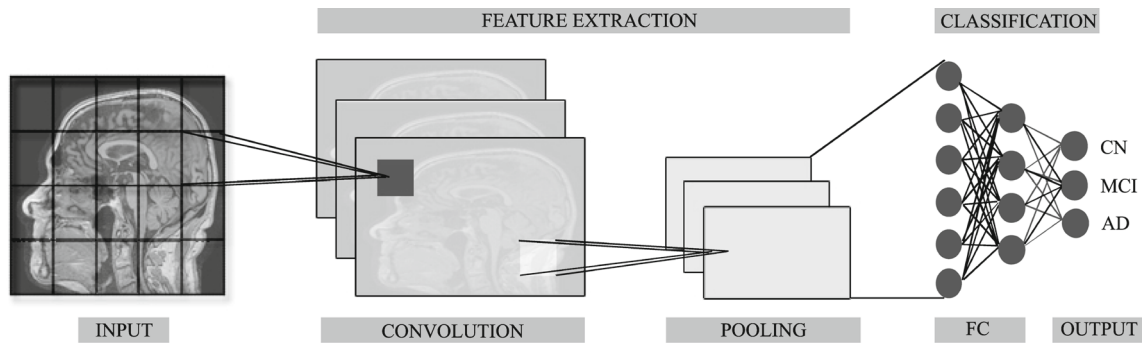


Fig. 3 A sample CNN structure

$$[f * g](t) \equiv \int_0^t f(\tau)g(t - \tau)d\tau \tag{1}$$

In Eq. (1),  $[f * g](t)$  means the convolution of the functions  $f$  and  $g$ . Convolution is taken in an infinite range mostly as calculated Eq. (2) [45]:

$$[f * g] \equiv \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau = \int_{-\infty}^{\infty} g(\tau)f(t - \tau)d\tau \tag{2}$$

Although kernel filter was defined as  $11 \times 11$  in AlexNet and  $7 \times 7$  in ZFNet, nowadays mostly  $3 \times 3$  and  $5 \times 5$  filters are being used in the literature in convolution layers. Angular edges, dark/light transition forms are evaluated and calculated separately as features with different edge detection filters. Here mathematically, the filtering operation performed by a feature map is a discrete convolution, which is giving the name to the algorithm. The result of this locally weighted sum then passes through a nonlinearity such as a sigmoid (Eq. 3), TanH (Eq. 4), and rectified linear unit

(ReLU) (Eq. 5).

$$\text{Sigmoid } f(x) = \frac{1}{1 + e^x} \tag{3}$$

$$\text{TanH } \tanh(x) = \frac{2}{1 + e^{-2x}} - 1 \tag{4}$$

$$\text{ReLU } f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases} \tag{5}$$

Another activation function called softmax is widely used in the last layer of CNNs. The standard softmax function:  $\sigma : \mathbb{R}^K \rightarrow [0, 1]^K$  is defined by Eq. (6) [46,47]:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \text{ for } i = 1, \dots, K \text{ and } z = (z_1, \dots, z_K) \in \mathbb{R}^K \tag{6}$$

and applies the standard exponential function to each element  $z_i$  of the input vector  $z$  and normalizes these values by dividing by the sum of all these exponentials. The normalization ensures that the sum of the components of the output vector  $\sigma(z)$  is 1. These nonlinear (activation) functions are used to control the output value of a neuron to decide whether a neu-

ron will be active or not. In this respect, activation functions are an important feature for DNNs.

The purpose of the pooling layer is to reduce the image size by processing with an  $n \times n$  matrix filter to reduce the computational cost. It is not a mandatory layer for the CNN algorithm, and some models may not use this layer. In this layer, a filter with the determined size is passed over the image according to the determined number of strides. As a result of this process, a reduced new matrix occurs. Maximum pooling and average pooling layers are the most used ones. Besides, there are other pooling methods such as mixed pooling, LP pooling, stochastic pooling, and spatial pyramid pooling [48].

Overfitting is an important problem for ML and DL. The dropout technique reduces overfitting, thus improving the performance of NNs. Normally, learning with backpropagation can adapt the model to the training data, but this learning may not be generalized to test data. Random dropouts inside the model can break these adaptations and improve the test results.

Dropout technique is general for different applications and was found to improve the performance of NNs in a wide variety of application domains such as object classification, digit recognition, speech recognition, document classification, and analysis of computational biology data. Feed-forward NN with dropout is described as [49]:

$$\begin{aligned}
 r_j^{(l)} &\sim \text{Bernoulli}(p), \\
 \tilde{y}^{(l)} &= r^{(l)} * y^{(l)}, \\
 z_i^{(l+1)} &= w_i^{(l+1)} \tilde{y}^l + b_i^{(l+1)}, \\
 y_i^{(l+1)} &= f(z_i^{(l+1)}).
 \end{aligned}
 \tag{7}$$

where  $l \in \{1, \dots, L - 1\}$  index is the hidden layers of the network,  $i$  is hidden unit,  $z^{(l)}$  denotes the vector of inputs into layer  $l$ ,  $y^{(l)}$  denotes the vector of outputs from layer,  $W^{(l)}$  and  $b^{(l)}$  are the weights and biases at layer  $l$ , and  $f$  is an activation function,  $*$  denotes an element-wise product,  $r^{(l)}$  is a vector of independent *Bernoulli* random variables each of which has probability  $p$ .

At the end of the CNN structure, usually, the fully connected (FC) layer operates on a flattened input where each input is connected to all neurons. FC layers can be used to optimize objectives such as class scores [50]. The output of the FC layer is normalized by the softmax activation function, which provides positive numbers that sum to one and is utilized to derive the output probabilities of the classification layer that uses the calculated probabilities to assign the input to one of the mutually exclusive target classes and computes the loss [51].

Although special models that offer efficient solutions for different applications are frequently developed in the literature, the basis of DL lies in the adaptability of a solution to

other problems. For this reason, studies conducted with DL models, especially in recent years, have gradually evolved toward studies conducted on pre-trained models. These studies either directly use the weights of pre-trained models or produce solutions with additional layers by using these weights with the transfer learning method. Pre-trained CNN models were used in the study. The architectural structure and parameter details of the models are given in Table 2.

The hyper-parameters to be used in the algorithm in DL studies are mostly determined by trial and error depending on the intuition of the programmer, previous studies, the structure of the algorithm, the type and size of data, and similar reasons. The hyper-parameters used in this study were determined as follows, by trying many different alternatives with optimum options without disrupting the structure of the pre-trained models. In all models, a standard structure was created. Since the final layers of the selected pre-trained models were designed according to 1000 classifications in ImageNet competition, the output data were gradually reduced to classify the images used in the study as AD, CN, and MCI, and *Dense* layers were added to 512 and 3, respectively. *Dropout* layers were added between these layers to prevent overfitting and its value was determined as 0.5 to avoid inconsistent results. The optimizer is set as *Adam* with a default *learning rate*. *Categorical cross-entropy* was chosen for the *Loss* parameter, and *accuracy* was chosen as metrics. Training processes consisting of standard 250 epochs were carried out for the models. This number of epochs was determined with repetitive tests to stop the training and optimize the hyper-parameters before an overfitting condition occurred. In all models except *NasneNet*, images were processed with  $224 \times 224 \times 3$  image format, only *NasneNet* structure was planned in  $331 \times 331 \times 3$  structure, so the data in this model were trained in the specified dimensions. The final weights of the models were saved during the training and validation phase and tested on data that were reserved for testing which the models had never seen. Thus, the reliability of the test results of the models was ensured. The framework created for the training of models and test matches in the study is shown in Fig. 4.

To use the DL models on MRIs, Kaggle, which is a crowd-sourced platform for data scientists to solve data science, ML, and predictive analytics problems [52], and Google Colaboratory (or Colab), which allows to write and execute arbitrary python code for ML, data analysis, and education with computing resources including graphic processor units (GPUs) [53], were used. The DL models were trained using the *Keras Applications DL libraries* with *Tensorflow* in the *Python* programming language on these platforms. After the operation of the models, graphics and weights were recorded and the accuracy and loss parameters of the models were visualized. To evaluate the performance of the models, *Confusion Matrices* were also created.

**Table 2** Details of the models

N.	Model	Size	Top-1 & Top-5 Acc		Model params	Depth	TPoAD	TDMpT
1	Xception	88 MB	0.790	0.945	22,910,480	126	72,189,227	66.67
2	VGG16	528 MB	0.713	0.901	138,357,544	23	27,561,795	62.50
3	VGG19	549 MB	0.713	0.900	143,667,240	26	32,871,491	66.67
4	ResNet50	98 MB	0.749	0.921	25,636,712	–	74,916,867	62.5
5	ResNet101	171 MB	0.764	0.928	44,707,176	–	93,935,107	83.33
6	ResNet152	232 MB	0.766	0.931	60,419,944	–	109,601,795	116.67
7	ResNet50V2	98 MB	0.760	0.930	25,613,800	–	74,901,635	70.83
8	ResNet101V2	171 MB	0.772	0.938	44,675,560	–	93,911,171	79.17
9	ResNet152V2	232 MB	0.780	0.942	60,380,648	–	109,570,179	104.17
10	InceptionV3	92 MB	0.779	0.937	23,851,784	159	47,984,803	75
11	InceptionResNetV2	215 MB	0.803	0.953	55,873,736	572	73,939,043	104.17
12	MobileNet	16 MB	0.704	0.895	4,253,864	88	28,899,139	58.33
13	MobileNetV2	14 MB	0.713	0.901	3,538,984	88	34,388,563	58.33
14	DenseNet121	33 MB	0.750	0.923	8,062,504	121	32,646,019	70.83
15	DenseNet169	57 MB	0.762	0.932	14,307,880	169	54,232,963	70.83
16	DenseNet201	80 MB	0.773	0.936	20,242,984	201	66,263,939	91.67
17	NASNetMobile	23 MB	0.744	0.919	5,326,716	–	30,727,957	83.33
18	NASNetLarge	343 MB	0.825	0.960	88,949,818	–	334,512,665	362.5
19	EfficientNetB0	29 MB	–	–	5,330,571	–	36,122,239	62.5
20	EfficientNetB1	31 MB	–	–	7,856,239	–	38,627,875	66.67
21	EfficientNetB2	36 MB	–	–	9,177,569	–	43,026,949	75
22	EfficientNetB3	48 MB	–	–	12,320,535	–	49,233,451	91.67
23	EfficientNetB4	75 MB	–	–	19,466,823	–	62,508,363	112.50
24	EfficientNetB5	118 MB	–	–	30,562,527	–	79,723,059	95.83
25	EfficientNetB6	166 MB	–	–	43,265,143	–	98,540,507	116.67
26	EfficientNetB7	256 MB	–	–	66,658,687	–	65,099,731	150
27	AlexNet	298 MB	0.633	0.846	–	8	26,042,435	66.67
28	ZFNet	–	0.64	0.853	–	7	23,650,243	70.83
29	LeNet	539 MB	–	–	–	7	47,172,099	58.33

TPoAD: Total Parameter of ADNI Data  
TDMpT: Training Duration Minutes per Training

### 3 Results

To ensure the accuracy and reliability of the study, fivefold verification training was applied to all models, and data were recorded in separate files during the training, validation, and testing stages. These results were then combined for each model, and the mean values and standard deviations of the models were calculated. The graphics of the results of the models were analyzed comparatively over the average values of the five tests. In the study, a very comprehensive analysis process was carried out by evaluating the performances of 29 different models in total as seen in Table 2. Among these 29 models, ResNet 50V2-101V2-152V2, InceptionV3, InceptionResNetV2, DenseNet 121-169-201, NASNetMobile, NASNetLarge, and ZFNet models are excluded from the

comparison stage because they have an accuracy rate under 50% during the training. Due to the low performance of these 11 models, accuracy and confusion matrix evaluations were continued on the remaining 18 models. The first situation that needs to be explained in the results of these excluded models is that certain model groups do not produce efficient results on brain MRIs. While the performance of the first versions of the ResNet models reached certain rates, the models arranged as V2 could not achieve this performance. Similarly, all DenseNet and NASNet models could not achieve success on these images. Interesting findings have emerged for the ZFNet and Inception models. Although the ZFNet model was established by making minor changes to the AlexNet model, it did not reach the level of performance that AlexNet provided. Interestingly, the Inception model did not achieve

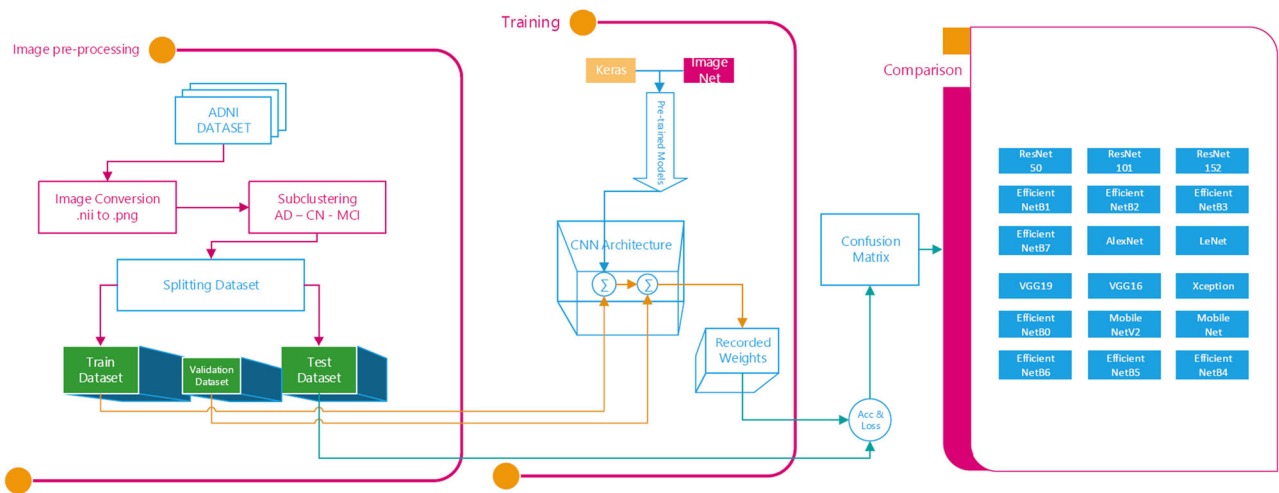


Fig. 4 Train, test, and comparison framework

the same success on MRI, although it previously achieved success in PET images, which are a different type of brain image.

After removing these models from the test and comparison set, the results obtained with the remaining models continue as follows. A single training graph was created for each model by calculating the means and standard deviations of the recorded training accuracies. The training curves of the remaining 18 models are shown in Fig. 5, respectively.

Although almost all of the models shown in Fig. 5 performed a learning leap with vertical acceleration before the first 50 epochs, the number of epochs was determined as 250 since the learning increase of other models except ResNet models continues. This number of epochs was determined as the cutoff point before the models were overfitted the data and applied to all models. Within the training curves, the AlexNet model achieved better accuracy training than the EfficientNet models (Fig. 5a), while the EfficientNet models decreased their performance from B0 to B7 (Fig. 5a–c). While the LeNet model (Fig. 5d) showed the best performance in the training graph of the models, the Xception model (Fig. 5f) showed the lowest performance. ResNet, VGG 16-19, and MobileNet & V2 models achieved similar training performance.

Standard deviation results have also been an indicator of the consistency of the models in training. Standard deviation charts of the models are shown in Fig. 6, respectively.

As seen in Fig. 6, the standard deviations of all models except the Xception model (Fig. 6f) have been realized between 0 & 0.02, proving that the models show consistency in each training. In the Xception model, again a low standard deviation curve was observed, remaining between 0.08 & 0.04. The models that gave the most consistent results from each training process were VGG models (Fig. 6f), ResNet models (Fig. 6e), LeNet, and MobileNet models (Fig. 6d) among the models. While the AlexNet and EfficientNetB0-

B1 models showed more consistent results than the other series of the EfficientNet models, the B6 model (Fig. 6c) showed the most emission among the EfficientNet models.

The loss function is an important indicator because it is used to measure the inconsistency between the predicted value and the actual label. The loss layer of a NN compares the output of the network with the ground truth for the case of image processing [54]. The choice of the loss function is critical for model estimation and evaluation [55]. The loss function used in the study is determined as *categorical cross-entropy* and calculated as in Eq. 8:

$$Loss = - \sum_{i=1}^{OutputSize} y_i \log \tilde{y}_i \tag{8}$$

where  $\tilde{y}_i$  is the  $i$ -th scalar value in the model output,  $y_i$  is the corresponding target value, and the *output size* is the number of scalar values in the model output. In this context,  $y_i$  is the probability that event  $i$  occurs and the sum of all  $y_i$  is 1, meaning that exactly one event may occur. The minus sign ensures that the loss gets smaller when the distributions get closer to each other. Since the Softmax is the only activation function recommended to use with the categorical cross-entropy loss function [56], it is used in these study models. Given the importance of the loss function, it is clear that the estimation and evaluation phases of a model are inextricably linked. If the loss function affects the model specification, estimating a model under one loss and evaluating it under another amount, changing the model specification without allowing the parameter estimates to be adjusted [57]. The robustness of the model increases with decreasing value of the loss function, and the loss value approaching zero in ANN is specified as the desired state. In addition to the accuracy graphs, the loss parameters of the models were also evaluated

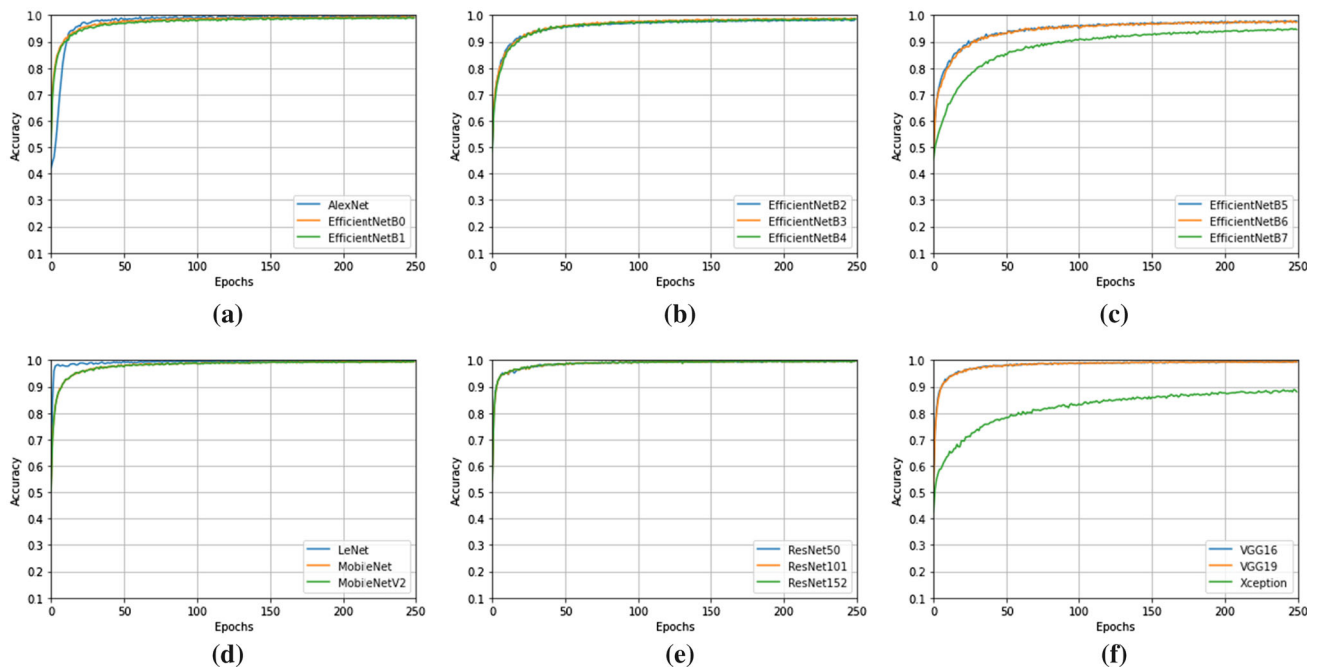


Fig. 5 Train accuracy curves of the models

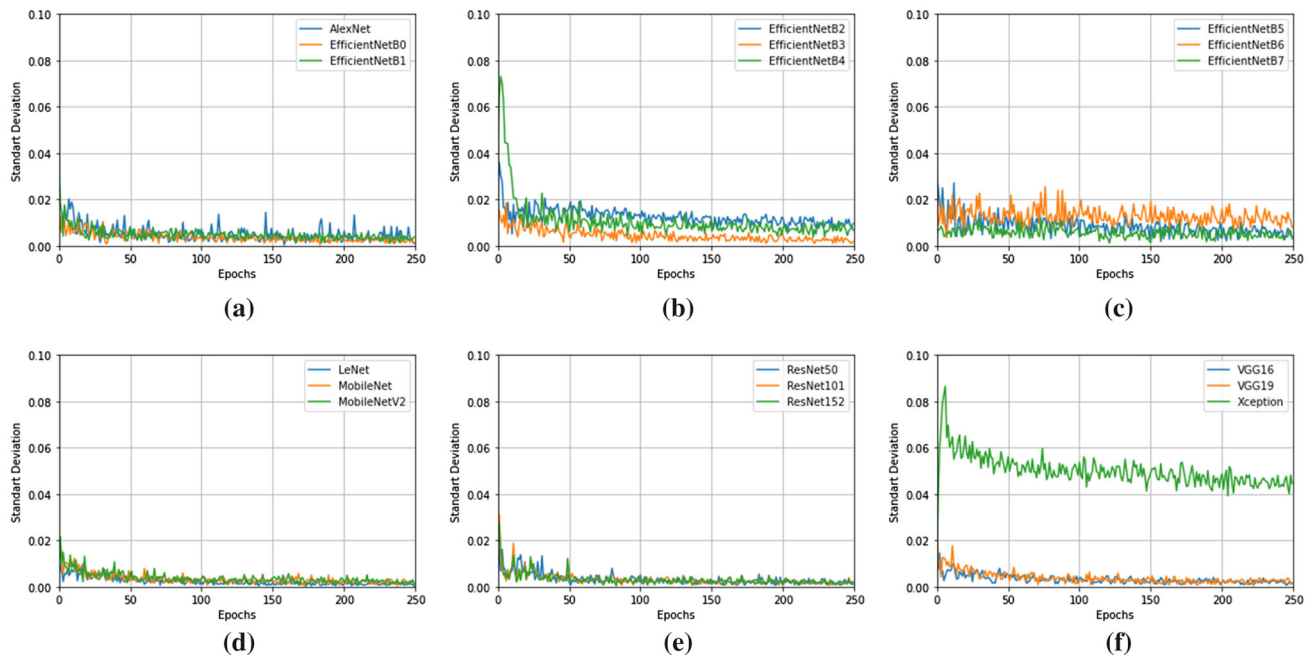


Fig. 6 Train accuracy standard deviation curves of the models

in the study. Loss values of the models are shown in Fig. 7, respectively.

The Xception model produced the lowest performance results in the accuracy charts as seen in Fig. 5 and also produced the highest loss value in loss charts (Fig. 7f). EfficientNetB7 model (Fig. 7c) produced the second highest loss value. An important case has emerged for the LeNet model

that while high accuracy was achieved in the training phase of this model (Fig. 5d), the loss graph showed a fluctuating course (Fig. 7d) and produced a negative result for the consistency of the model accuracy. In loss charts, the lowest values were obtained from ResNet models (Fig. 7e). Afterward, a ranking was formed as MobileNet models, VGG models, AlexNet, and EfficientNetB0-B7 models. The loss



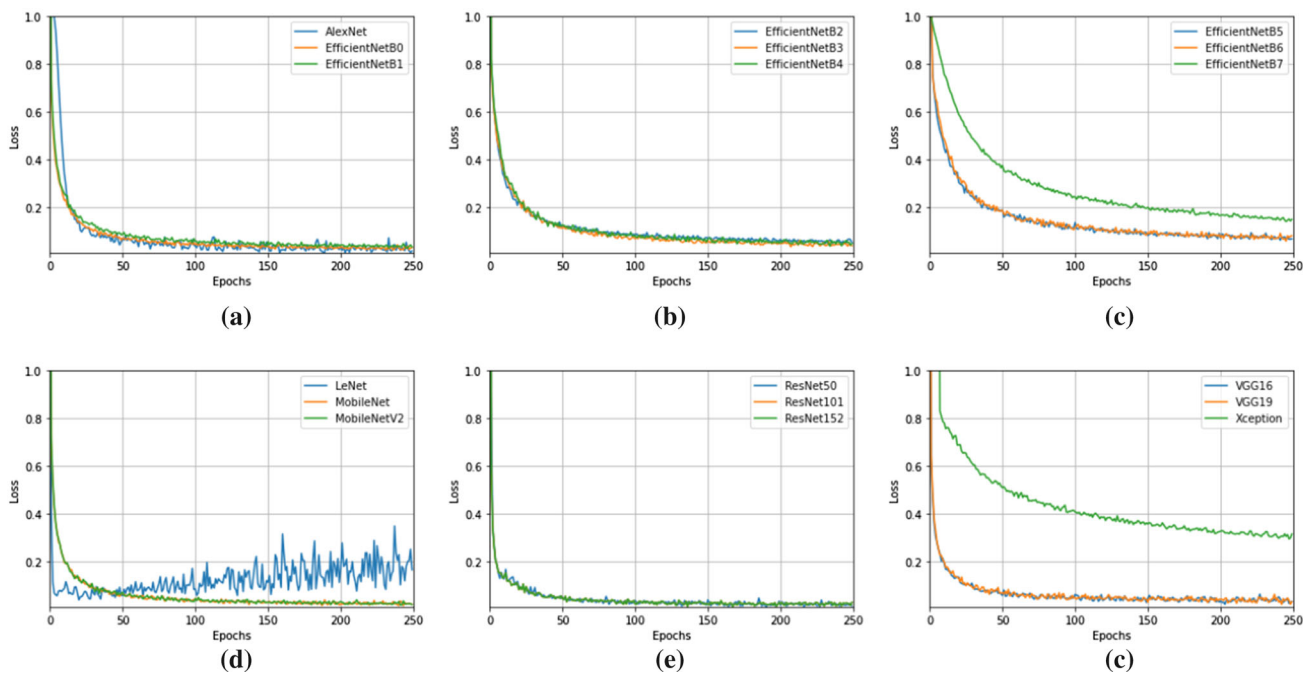


Fig. 7 Loss curves of the models

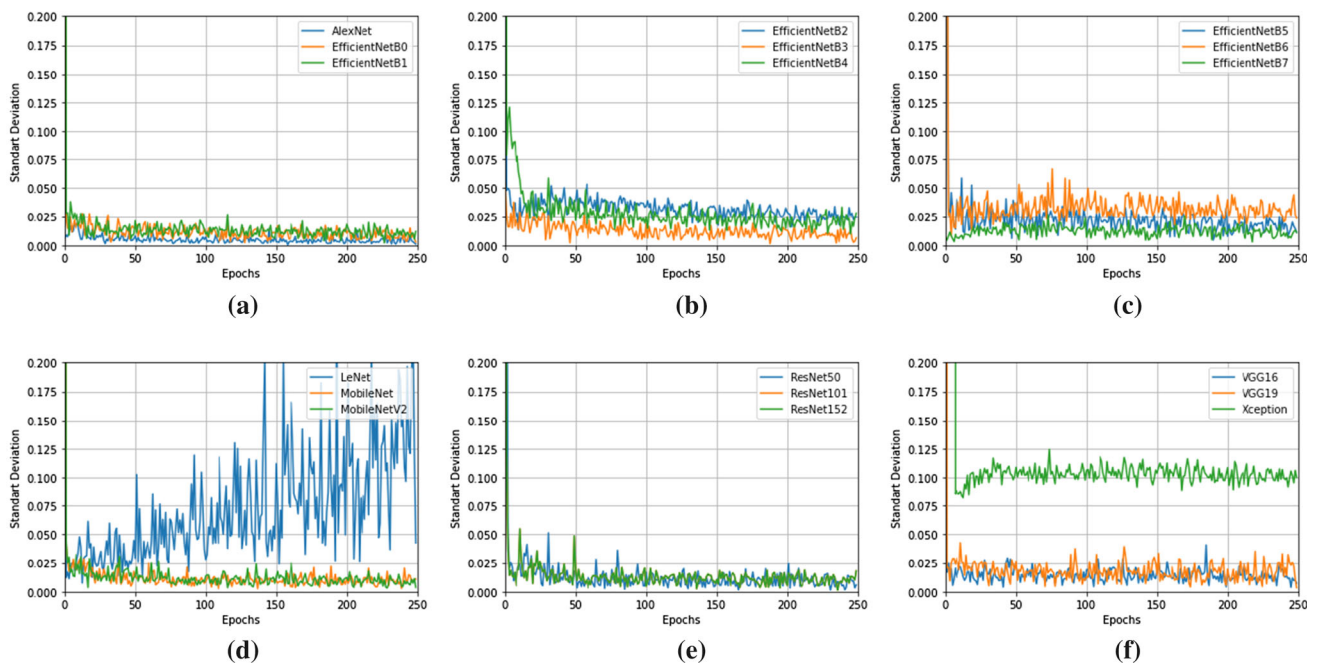


Fig. 8 Standard deviations of loss curves of the models

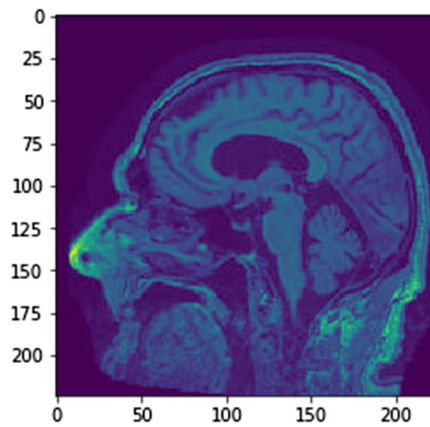
value revealed by AlexNet also continued with oscillation in its graph. Another remarkable result is that the loss values of the EfficientNet models increased from B0 to B7.

The models achieved success by showing low loss rates, and these results were compared in detail during the testing stage. With the loss function, the stability of the models is verified by evaluating the standard deviations of the values

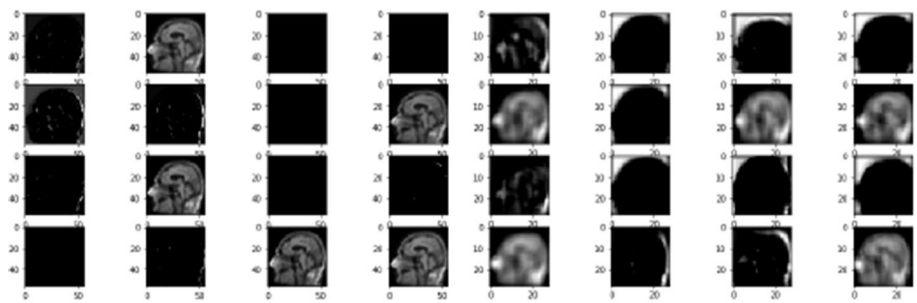
obtained in each epoch in the fivefold verification stage in a similar way to the accuracy values. Standard deviation graphs of loss values are shown in Fig. 8, respectively.

As seen in Fig. 8, the standard deviation values of the loss function of the models in the study showed consistency by remaining between 0 and 0.05 except for the two models. Among the two models, the Xception model showed

**Fig. 9** Feature extraction. **a** Input Image. **b** 1st Layer. **c** 5th Layer. **d** 10th Layer. **e** 15th Layer

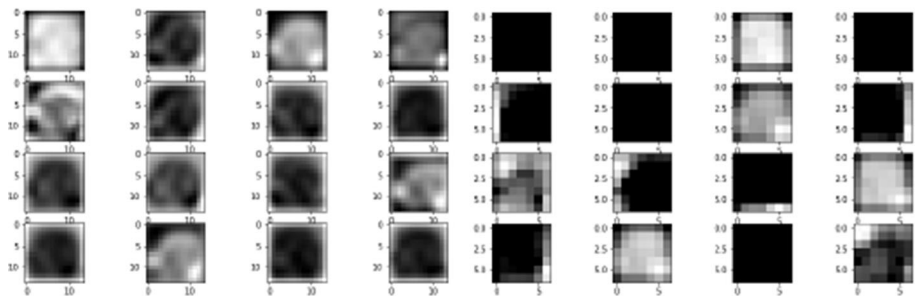


(a)



(b)

(c)



(d)

(e)

consistency by showing a 0.1 standard deviation (Fig. 8f). However, the LeNet model produced a discrepancy with the standard deviation rate increasing of oscillations as the number of epochs increased (Fig. 8d). To decide the performance of this model, the results of the test stage were examined. Apart from the two models, the standard deviation rates of AlexNet, EfficientNetB0-B1 (Fig. 8a), MobileNet models (Fig. 8d), and ResNet models (Fig. 8e) were determined as the most consistent models. After these models, there were VGG models (Fig. 8f) and Efficient B7-B3-B5-B4-B2-B6 models, respectively (Fig. 8b,c).

Testing was carried out on the recorded weights of the models after the training and validation stages. The testing process is important in terms of the accuracy results that the models will produce after training. It is the most important indicator of the performance of the models on the images. During the testing stage, the automatic feature extraction process that takes place in layers of the CNN architecture was also recorded. Sample images were also saved from the feature map created in layers such as the input layer, and the 1st, 5th, 10th, and 15th layers in CNN models by filtering the image in the dataset and are shown in Fig. 9.

**Table 3** Test results of the models

N.	Model	Accuracy	SD	Loss
1	AlexNet	89.95%	0.0132	0.0167
2	EfficientNetB0	<b>92.98%</b>	0.0095	0.0316
3	EfficientNetB1	91.91%	0.0135	0.0336
4	EfficientNetB2	90.88%	0.0076	0.0524
5	EfficientNetB3	90.93%	0.0114	0.0393
6	EfficientNetB4	88.33%	0.0217	0.0481
7	EfficientNetB5	89.91%	0.0145	0.0674
8	EfficientNetB6	89.49%	0.0088	0.0804
9	EfficientNetB7	87.12%	0.0065	0.1497
10	LeNet	87.72%	0.0114	0.1655
11	MobileNet	83.40%	0.0220	0.0208
12	MobileNetV2	80.19%	0.0183	0.0192
13	ResNet50	88.56%	0.0102	<b>0.0145</b>
14	ResNet101	86.98%	0.0097	0.0289
15	ResNet152	87.77%	0.0147	0.0289
16	VGG16	89.72%	0.0058	0.0324
17	VGG19	89.77%	0.0082	0.0307
18	Xception	77.40%	0.0404	0.3151

In the study, the results of the test process separated from the images obtained from the ADNI database and performed on the data that the models have never seen before are shown in Table 3.

When the test accuracy rates of these 18 models are examined in Table 3, the EfficientNetB0 model showed the highest performance with 92.98% accuracy on the test data. This model is followed by EfficientNetB1 (91.91%), EfficientNetB3 (90.93%), and EfficientNetB2 (90.88%), respectively. Thus, EfficientNet models verified the performance curves in the test phase which they showed during the training and produced a reliable and valid result. The models ranked at 89% accuracy rate were AlexNet, EfficientNetB5, VGG19, VGG16, and EfficientNetB6 models, and the accuracy rates of these models were 89.95%, 89.91%, 89.77%, 89.72%, and 89.49%, respectively. Other models provided accuracy ranging from 88% to 77%. Among the models, the lowest performance rate was seen in the Xception model (77.4%).

When the standard deviations of the results obtained by subjecting the models to the fivefold verification test were examined, it was observed that all models showed a standard deviation between 0 and 0.02, except for Xception (0.04), and produced consistent results in each test. In the loss parameter, another performance indicator of the models, the lowest rate with 0.0145 was shown by the ResNet50 model. However, the accuracy rate of this model did not support the model performance to the same extent as 88.56%. Although the EfficientNetB0 model, which has the highest accuracy rate, produces a loss value that cannot be considered lower than

0.0316 other models, this value is considered to be close to zero and a successful value. The accuracy, loss, and standard deviation graphs of the test results of the models are shown in Fig. 10.

Among all models, EfficientNet models achieved promising results with general performance conditions. It is thought that the performance of these models can be further increased through transfer learning methods. Thus, it has been proven that the models prepared for object classification will also be successful in medical image processing.

In the last part of the test stage, the confusion matrixes of all models were created and their precision, recall (sensitivity), and specificity values were calculated. This matrix has contributed to an in-depth study of each model’s performance on images of each class. These values are calculated as Eq. (9):

$$\begin{aligned}
 Precision &= TP / (TP + FP) \\
 Recall(Sensitivity) &= TP / (TP + FN) \\
 Specificity &= TN / (TN + FP)
 \end{aligned}
 \tag{9}$$

where *TP* is true positive, *FP* is false positive, *TN* is true negative, and *FN* is false negative. Precision is a measure of how accurately all classes are predicted. It is also known as positive predictive value. Recall can be defined as the ratio of the total number of correctly classified positive examples divide by the total number of positive examples. Specificity indicates when it is actually negative and how often does it predict negative [35].

The confusion matrix was produced by the weights of the models after each training was recorded, and the average of 5 matrixes was taken after the fivefold verification training. Thus, integrity has been achieved with the previous training and testing phase. The values are not rounded to avoid variation in the number of images used to test the models. This is the justification for the floating number in the number of images belonging to the classes. These floating sections were also taken into account in the calculation of the values. The three values of each model belonging to each class are calculated and shown in Table 4.

Among the confusion matrix values, the models that produced the highest results for all classes were determined. Among these models in the sensitivity field, EfficientNetB2 (94.42%), EfficientNetB1 (94.35%), and EfficientNetB0 (94.34%) models took place, which produce very close rates in the “AD” class. In the “CN” class, B3 (94.87%) and B1 (94.18%) took the first two places among the EfficientNet models, while the VGG16 model (92.91%) took the third place. In the “MCI” class, B0 (94.25%) and B4 (89.25%) of the EfficientNet models took the first and third places, and the second was LeNet (90.65%).

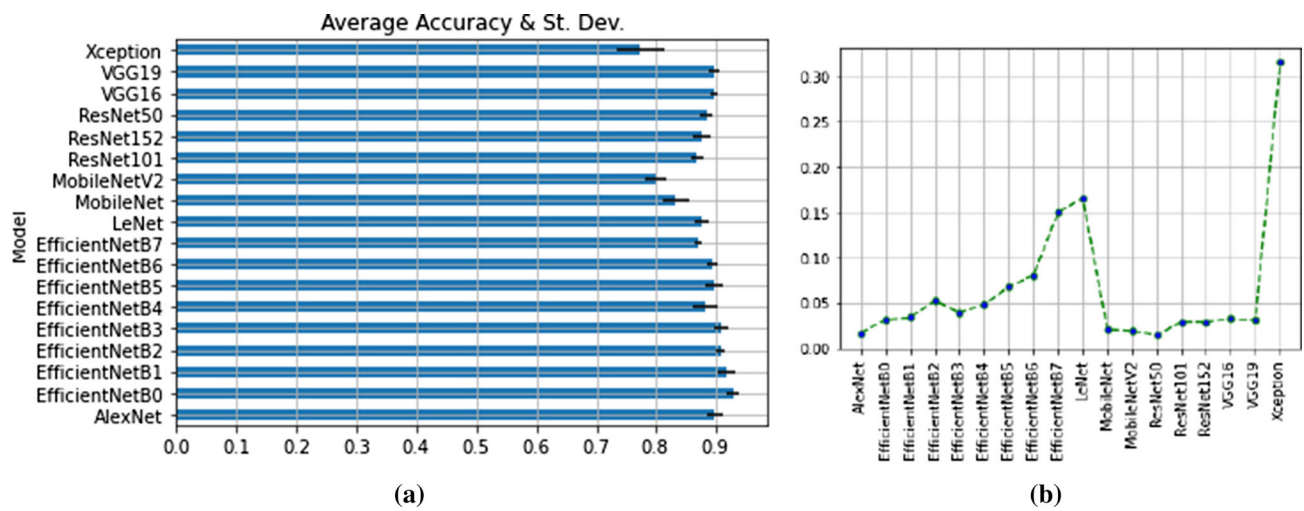


Fig. 10 a Accuracy, standard deviation, and b loss results of the test phase

Table 4 Confusion matrix of the models

Model	Predicted Class	Actual			Recall (Sensitivity)	Precision	Specificity
		AD	CN	MCI			
AlexNet	AD	76	5.6	8.4	91.79%	84.44%	95.97%
	CN	2.4	133.2	12.4	89.52%	90.00%	94.74%
	MCI	4.4	10	177.6	89.52%	92.50%	93.78%
EfficientNetB0	AD	80	13.8	4.2	94.34%	81.63%	94.90%
	CN	1.6	139.6	6.8	86.17%	<b>94.32%</b>	<b>96.96%</b>
	MCI	3.2	8.6	180.2	<b>94.25%</b>	93.85%	95.22%
EfficientNetB1	AD	80.2	1.4	8.4	94.35%	89.11%	97.16%
	CN	1.8	132.8	13.4	94.18%	89.73%	94.74%
	MCI	3	6.8	182.2	89.31%	94.90%	95.66%
EfficientNetB2	AD	77.8	4.4	7.8	<b>94.42%</b>	86.44%	96.49%
	CN	1.6	133.4	13	90.63%	90.14%	94.84%
	MCI	3	9.4	179.6	89.62%	93.54%	94.60%
EfficientNetB3	AD	80.8	2.6	6.6	88.60%	<b>89.78%</b>	<b>97.28%</b>
	CN	3.6	129.4	15	<b>94.87%</b>	87.43%	93.66%
	MCI	6.8	4.4	180.8	89.33%	94.17%	95.08%
EfficientNetB4	AD	74.8	6.2	9	84.81%	83.11%	95.55%
	CN	4.4	133.2	10.4	88.45%	90.00%	94.70%
	MCI	9	11.2	171.8	89.85%	89.48%	91.54%
EfficientNetB5	AD	77.6	5.6	6.8	93.95%	86.22%	96.43%
	CN	3	125.6	19.4	91.15%	84.86%	92.33%
	MCI	2	6.6	183.4	87.50%	<b>95.52%</b>	<b>96.10%</b>
EfficientNetB6	AD	78.6	5	6.4	88.31%	87.33%	96.66%
	CN	3.4	130.4	14.2	90.18%	88.11%	93.83%
	MCI	7	9.2	175.8	89.51%	91.56%	93.07%
EfficientNetB7	AD	77.2	4.2	8.6	87.53%	85.78%	96.26%
	CN	5.2	122.8	20	88.60%	82.97%	91.35%
	MCI	5.8	11.6	174.6	85.93%	90.94%	92.33%

Table 4 continued

Model	Predicted Class	Actual			Recall (Sensitivity)	Precision	Specificity
		AD	CN	MCI			
LeNet	AD	77	5.2	7.8	81.40%	85.56%	96.12%
	CN	8.4	129.6	10	89.38%	87.57%	93.54%
	MCI	9.2	10.2	172.6	90.65%	89.90%	91.90%
MobileNet	AD	67.2	11.6	11.2	84.85%	74.67%	93.50%
	CN	4.4	127.2	16.4	80.00%	85.95%	92.32%
	MCI	7.6	20.2	164.2	85.61%	85.52%	88.33%
MobileNetV2	AD	62.2	13	14.8	77.75%	69.11%	92.06%
	CN	7.8	113	27.2	81.65%	76.35%	88.00%
	MCI	10	12.4	169.6	80.15%	88.33%	89.74%
ResNet50	AD	76.8	5.6	7.6	86.88%	85.33%	96.14%
	CN	6.2	127.6	14.2	88.98%	86.22%	92.88%
	MCI	5.4	10.2	176.4	89.00%	91.88%	93.27%
ResNet101	AD	69.6	10.2	10.2	89.23%	77.33%	94.20%
	CN	2.6	135.2	10.2	83.25%	91.35%	95.22%
	MCI	5.8	17	169.2	89.24%	88.13%	90.52%
ResNet152	AD	73.8	6.6	9.6	89.56%	82.00%	95.34%
	CN	5.2	129.2	13.6	86.13%	87.30%	93.29%
	MCI	3.4	14.2	174.4	88.26%	90.83%	92.43%
VGG16	AD	77.4	3.2	9.4	91.71%	86.00%	96.35%
	CN	4	125.8	18.2	92.91%	85.00%	92.46%
	MCI	3	6.4	182.6	86.87%	95.10%	95.72%
VGG19	AD	72.6	5	12.4	93.80%	80.67%	95.06%
	CN	1.2	133.2	13.2	90.98%	90.24%	94.92%
	MCI	3.6	8.2	180.2	87.56%	93.85%	94.73%
Xception	AD	52.2	18.4	19.4	82.86%	58.00%	89.70%
	CN	4	110.6	33.4	76.70%	74.73%	86.91%
	MCI	6.8	15.2	170	76.30%	88.54%	89.38%

In the precision field, the first three models in the “AD” class were EfficientNetB3 (89.78%), EfficientNetB1 (89.11%), and EfficientNetB6 (87.33%) models, which again produce very close rates. In the “CN” class, EfficientNetB0 (94.32%) took first place, while ResNet101 took second place with 91.35%, and the VGG19 model took third place with 90.24%. In the “MCI” class, B5 (95.52%) and B1 (94.90%) of the EfficientNet models took the first and third places, and the VGG16 (95.10%) model was found in the second place.

Finally, in the field of specificity, the first three models in the “AD” class were EfficientNetB3 (97.28%), EfficientNetB1 (97.16%), and EfficientNetB6 (96.66%) models, which again produce very close rates. In the “CN” class, EfficientNetB0 (96.96%) took first place, while ResNet101 was the second with 95.22%, and the VGG19 model was third with 94.92%. In the “MCI” class, B5 (96.10%) and B1 (95.66%) of the EfficientNet models took the first and third places, and the second was the VGG16 (95.72%) model.

In light of all these results, it is clear that EfficientNet models perform better on brain MRIs than other pre-trained models. In itself, although it shows different rates between classes, there is not much difference between them. Besides, VGG models also produced locally successful results on certain classes of images. Therefore, applying EfficientNet models directly or through transfer learning in future studies will produce promising results in clinical studies.

## 4 Discussion

Due to the acceleration of ML and DL studies in the last decade, these techniques have been emphasized in medical image processing studies as in many other fields. Particularly in recent years, segmentation and classification studies have been carried out using DL techniques. Although medical image processing mainly covers segmentation studies, classification studies have also become widespread. The success



**Table 5** Alzheimer's disease classification studies

Author & Study	Train/Test	Analysis	Results	Data	I-Type
Lu et al. [25]	RF-SVM, LapSVM, kNN-SVM	ROC	RF-RSVM: 92.16% Acc, 88.46% Se, 96% Sp. LapSVM: 70% Acc, 80% Se, 60% Sp. kNN-SVM: 76.47% Acc, 65.38% Se, 88% Sp.	120 MCI, 152 CN	PET
Luo et al. [27]	CNN	CM	93% Sp	47 AD, 34 CN	MRI
Suk et al. [31]	CNN/ DeepESR-Net	CM	MOLR+DeepESRNet: AD vs NC: 90.28% Acc, MC vs NC: 74.20% Acc, JLLR+DeepESRNet: AD vs NC: 91.02% Acc, MCI vs NC: 73.02% Acc	186 AD, 393 MCI, 226 CN	MRI
Ding et al. [33]	CNN/InceptionV3	ROC	82% Sp 100% Se	2149	PET
Li and Liu [29]	K-Means & DenseNet	ROC	AD vs CN: 89.5% Acc 92.4% AUC. MCI vs CN: 73.8% Acc 77.5% AUC.	199 AD, 403 MCI, 229 CN	MRI
Jha et al. [18]	DCWT, PCA, LDA, ELM	CM	90.26% Acc, 90.20% Sp, 90.27% Se on ADNI. 95.72% Acc, 96.59% Sp, 93.03% Se on OASIS.	172 ADNI, 95 OASIS	MRI
Bi et al. [30]	Random NN	Elman NN	92.31% Acc	61	MRI
Lin et al. [28]	CNN/ELM	ROC	79.9% Acc, 86.1% AUC	188 AD, 401 MCI, 229 CN	MRI
Jie et al. [19]	FCN	ROC	IMCI vs eMCI: 78.8% eMCI vs CN: 78.3% AUC: 78.3% and 77.1%	99 MCI, 50 CN	MRI
Liu et al. [32]	CNN	ROC	AD vs NC: 93.26% Acc, pMCI vs NC: 82.95% Acc	93 AD, 204 MCI, 100 CN	MRI+PET
Salvatore et al. (2018) [15]	SVM	Acc, Se, Sp	85% Acc, 83% Se, 87% Sp	50 AD, 50 MCI, 50 CN	MRI
Buyrukoglu [23]	RF & EFS	CM & AUC	91% Acc	188 AD, 402 MCI, 229 CN	MRI

CNN: Convolutional Neural Network, ROC: Receiver Operating Characteristic Curves, Sp: Specificity, Se: Sensitivity, Acc: Accuracy, AUC: Area Under the Curve, CM: Confusion Matrix, NN: Neural Network, EFS: Ensemble Feature Selection.

of AI algorithms in this regard is also seen as the biggest factor. Classification studies about Alzheimer's disease are shown in Table 5.

As seen in Table 5, studies aimed at classifying the stages of Alzheimer's disease have gained momentum, especially in recent years. While different ML and DL techniques are applied in these studies, the results of the studies vary between 70% and 95%. For example, the CNN algorithm achieved 83% and 93% performance in one study [32], while the same algorithm achieved performance results varying between 73% and 91% in another study [31]. While SVM, one of the ML algorithms, performed 85% in one study [15], it gave results varying between 70% and 92% with hybrid methods in another study [25]. Another ML algorithm RF has reached 91%–92% rates with hybrid methods in different studies [23,25]. While InceptionV3, a DL algorithm, achieved sensitivity and specificity rates between 82% and 100% on PET images [33], in this study, it was excluded from the evaluation because it was below 50% accuracy during training. Except for one of the studies shown in Table 5, all of the others worked on less than 1000 images, and most of them worked on a small number of images. Moreover, some of these images (10%–20%) were reserved for the validation phase. In some studies, test images were not included in the training, and in some, validation results during training were given as a result due to a low number of images.

In this study, the accuracy rates of varying between 90% and 93% on the MRIs that the EfficientNet models have never seen, and positive results between 90% and 97% in sensitivity, specificity, and precision values have been indicative of the usability of these models in future clinical studies. Nowadays, the importance of common valid models that can be adapted to more than one dataset, instead of a single model suitable for a single dataset, has increased. For this reason, it has become important to transfer models that have proven their success in different classification competitions to medical image processing. With the very comprehensive and comparative analysis carried out in this study, steps have been taken to prevent the diversity and difference shown in Table 5.

Since the process of learning from the representation of data takes place in DL techniques, studies conducted on more data give more consistent and accurate results. In this study, more images were studied in terms of consistency of the results, and at the same time, a real prediction process was carried out by not using the test data by the model during the training. A step was also taken to avoid inconsistencies in different methods used in the literature. Considering that DL studies have evolved into transfer learning today, it is important to determine the most dominant model and models to be used in biomedical image processing and classification processes among models that have proven their success in different image classifications. Instead of subject-

and situation-specific models, common models should be created and transferred to future studies and used here. Especially considering the small number of images in the field of medicine and the difficulty of image acquisition and processing, it is important to identify models that give common and reliable results in different imaging techniques.

The study has some limitations. The first and most important of these is the need for a large number of images in DL studies, and in addition, the difficulty of obtaining images in the field of biomedical image processing. Moreover, the ethical approval process in the field of individual image acquisition in the field of biomedical image processing is also long. For this reason, a global database was used in the study. However, biomedical images in these databases are often not sufficient for DL studies. Another limitation of the study is the hardware requirement in DL studies. Tools that allow the use of online ML and DL algorithms are often either time-limited or there are problems such as disconnections during use. Therefore, many tests need to be done repeatedly. Finally, it is the problem of viewing images presented in formats such as NIFTI and DICOM by opening them for those who work outside the medical field and do not have certain programs. For this, a solution has been produced by writing a separate program that opens the image sequence and takes the middle two images.

## 5 Conclusion

This study is important in terms of its scope, comparative framework, reliability, and consistency of its results. It has been presented as an exemplary study for the use of DL algorithms, which are becoming increasingly common in every field, in the field of medical image processing. In the study, unlike previous similar studies, not only on a single model or a few models but a comparative and comprehensive analysis process was carried out on all models that have proven their success in different categories in international important competitions and on all models offered to use in DL libraries.

The study has guiding and inspiring contributions to both creating decision support systems in medical research and transfers learning research in the field of DL. Human-intensive studies are carried out to make quick decisions in clinical studies. The lack of expertise causes the processes to be prolonged. In clinical research, getting fast results with the success of DL models, which are one of the sub-disciplines of AI, will both facilitate the work of experts and pave the way for getting more accurate results. Besides, a study was carried out in the field of DL, which will contribute to preventing the emergence of different and data-specific models every day, and foreseeing the dissemination of models from specific to general. Because in the DL discipline, the main purpose is to establish common models that can give reliable

results in different datasets, not just a single model suitable for a single data.

Another contribution of the study to the field was the determination of the use of DL models in biomedical image classification as well as in object identification. Thus, by avoiding the confusion of different models in biomedical image processing, a guiding study was carried out in new studies. Similar studies can be tested with different image sequences such as T1-w, T2-w, and FLAIR, as well as with different planes such as coronal and axial, and the performances of the models can be compared. Thus, comparative options will be presented to researchers for early diagnosis of the disease. Because of the increasing number of different types of biomedical images, the need for standard models that produce common solutions has arisen. This study is presented not only as a comparative study but also as a comprehensive classification study in terms of scope. Classification with high accuracy rates has been carried out to determine the stages of Alzheimer's disease, which is important enough to be in the top 10 in the list of fatal diseases in the world and to take precautions before reaching the point that cannot be reversed.

**Acknowledgements** Data collection and sharing for this project were funded by the Alzheimer's Disease Neuroimaging Initiative (ADNI) (National Institutes of Health Grant U01 AG024904) and DOD ADNI (Department of Defense award number W81XWH-12-2-0012). ADNI is funded by the National Institute on Aging, the National Institute of Biomedical Imaging and Bioengineering, and through generous contributions from the following: AbbVie, Alzheimer's Association; Alzheimer's Drug Discovery Foundation; Araclon Biotech; BioClinica, Inc.; Biogen; Bristol-Myers Squibb Company; CereSpir, Inc.; Cogstate; Eisai Inc.; Elan Pharmaceuticals, Inc.; Eli Lilly and Company; EuroImmun; F. Hoffmann-La Roche Ltd and its affiliated company Genentech, Inc.; Fujirebio; GE Healthcare; IXICO Ltd.; Janssen Alzheimer Immunotherapy Research & Development, LLC.; Johnson & Johnson Pharmaceutical Research & Development LLC.; Lumosity; Lundbeck; Merck & Co., Inc.; Meso Scale Diagnostics, LLC.; NeuroRx Research; Neurotrack Technologies; Novartis Pharmaceuticals Corporation; Pfizer Inc.; Piramal Imaging; Servier; Takeda Pharmaceutical Company; and Transition Therapeutics. The Canadian Institutes of Health Research is providing funds to support ADNI clinical sites in Canada. Private sector contributions are facilitated by the Foundation for the National Institutes of Health ([www.fnih.org](http://www.fnih.org)). The grantee organization is the Northern California Institute for Research and Education, and the study is coordinated by the Alzheimer's Therapeutic Research Institute at the University of Southern California. ADNI data are disseminated by the Laboratory for Neuro Imaging at the University of Southern California.

**Funding** There is no funding for this research.

## Declarations

**Conflicts of interest** The author declares that there is no conflict of interest.

## References

1. WHO. The top 10 causes of death (2020). <https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death>.
2. WHO. Dementia (2020). <https://www.who.int/news-room/fact-sheets/detail/dementia>.
3. Soria Lopez, J.A.; González, H.M.; Léger, Gabriel C.: Chapter 13 - alzheimer's disease. In Steven T. Dekosky and Sanjay Asthana, editors, *Geriatric Neurology*, volume 167 of *Handbook of Clinical Neurology*, pp. 231–255. Elsevier, (2019). <https://doi.org/10.1016/B978-0-12-804766-8.00013-3>. <https://www.sciencedirect.com/science/article/pii/B9780128047668000133>.
4. Penney, J.; Ralvenius, W.T.; Tsai, L.-H.: Modeling Alzheimer's disease with ipsc-derived brain cells. *Mol. Psych.* **25**(1), 148–167 (2020). <https://doi.org/10.1038/s41380-019-0468-3>.
5. Wan, Y.-W.; Al-Ouran, R.; Mangleburg, C.G.; Perumal, T.M.; Lee, T.V.; Allison, K.; Swarup, V.; Funk, C.C.; Gaiteri, C.; Allen, M.; Wang, M.; Neuner, S.M.; Kaczorowski, C.C.; Philip, V.M.; Howell, G.R.; Martini-Stoica, H.; Zheng, H.; Mei, H.; Zhong, X.; Kim, J.W.; Dawson, V.L.; Dawson, T.M.; Pao, P.-C.; Tsai, L.-H.; Haure-Mirande, J.-V.; Ehrlich, M.E.; Chakrabarty, P.; Levites, Y.; Wang, X.; Dammer, E.B.; Srivastava, G.; Mukherjee, S.; Sieberts, S.K.; Omberg, L.; Dang, K.D.; Eddy, J.A.; Snyder, P.; Chae, Y.; Amberkar, S.; Wei, W.; Hide, W.; Preuss, C.; Ergun, A.; Ebert, P.J.; Airey, D.C.; Mostafavi, S.; Yu, L.; Klein, H.-U.; Carter, G.W.; Collier, D.A.; Golde, T.E.; Levey, A.I.; Bennett, D.A.; Estrada, K.; Matthew Townsend, T.; Zhang, B.; Schadt, E.; De Jager, P.L.; Price, N.D.; Ertekin-Taner, N.; Liu, Z.; Shulman, J.M.; Mangravite, L.M.; Logsdon, B.A.: Meta-analysis of the alzheimer's disease human brain transcriptome and functional dissection in mouse models. *Cell Rep.* **32**(2):107908, (2020). ISSN 2211-1247. <https://doi.org/10.1016/j.celrep.2020.107908>. <https://www.sciencedirect.com/science/article/pii/S2211124720308895>.
6. Leandrou, S.; Petroudi, S.; Kyriacou, P.A.; Reyes-Aldasoro, C.C.; Pattichis, C.S.: Quantitative MRI brain studies in mild cognitive impairment and Alzheimer's disease: a methodological review. *IEEE Rev. Biomed. Eng.* **11**, 97–111 (2018). <https://doi.org/10.1109/rbme.2018.2796598>.
7. Alzheimer's Association.: 2015 Alzheimer's disease facts and figures. *Alzheimer's and Dementia* **11**(3):332–384, (2015). <https://doi.org/10.1016/j.jalz.2015.02.003>.
8. Wang, Z.; Zhu, X.; Adeli, E.; Zhu, Y.; Nie, F.; Munsell, B.; Guorong, W.: Multi-modal classification of neurodegenerative disease by progressive graph-based transductive learning. *Med. Image Anal.* **39**, 218–230 (2017). <https://doi.org/10.1016/j.media.2017.05.003>.
9. Hadjichrysanthou, C.; McRae-McKee, K.; Evans, S.; de Wolf, F.; Anderson, R.M.: Initiative for the Alzheimer's Disease Neuroimaging: Potential factors associated with cognitive improvement of individuals diagnosed with mild cognitive impairment or dementia in longitudinal studies. *J. Alzheimer's Dis.* **66**, 587–600 (2018). <https://doi.org/10.3233/JAD-180101>.
10. Sullivan, M.D.; Anderson, J.A.E.; Turner, G.R.; Nathan Spreng, R.: Intrinsic neurocognitive network connectivity differences between normal aging and mild cognitive impairment are associated with cognitive status and age. *Neurobiol. Aging* **73**:219–228 (2019). <https://doi.org/10.1016/j.neurobiolaging.2018.10.001>.
11. Molinuevo, J.L.; Ayton, S.; Batrla, R.; Bednar, M.M.; Bittner, T.; Cummings, J.; Fagan, A.M.; Hampel, H.; Mielke, M.M.; Mikulskis, A.; O'Bryant, S.; Scheltens, P.; Sevigny, J.; Shaw, L.M.; Soares, H.D.; Tong, G.; Trojanowski, J.Q.; Zetterberg, H.; Blennow, K.: Current state of Alzheimer's fluid biomarkers. *Acta Neuropathol.* **136**(6), 821–853 (2018). <https://doi.org/10.1007/s00401-018-1932-x>.



12. Villa, C.; Lavitrano, M.; Salvatore, E.; Combi, R.: Molecular and imaging biomarkers in Alzheimer's disease: A focus on recent insights. *J. Pers. Med.* **10**(3), 1 (2020). <https://doi.org/10.3390/jpm10030061>.
13. Cooper, Yonatan A.; Nachun, Daniel; Dokuru, Deepika; Yang, Zhongan; Karydas, Anna M.; Serrero, Ginette; Yue, Binbin; Initiative, Alzheimer's Disease Neuroimaging.; Boxer, Adam L.; Miller, Bruce L.; Coppola, Giovanni: Progranulin levels in blood in alzheimer's disease and mild cognitive impairment. *Annals of Clinical and Translational Neurology* **5**(5), 616–629 (2018) ISSN 2328-9503. <https://doi.org/10.1002/acn3.560>. <https://onlinelibrary.wiley.com/doi/abs/10.1002/acn3.560>.
14. Goldstein, F.C.; Milloy, A.; Loring, D.W.: Incremental validity of montreal cognitive assessment index scores in mild cognitive impairment and alzheimer disease. *Dement. Geriatr. Cogn. Disord.* **45**(1), 49–55 (2018). <https://doi.org/10.1159/000487131>.
15. Salvatore, C.; Cerasa, A.; Castiglioni, I.: Mri characterizes the progressive course of ad and predicts conversion to Alzheimer's dementia 24 months before probable diagnosis. *Front. Aging Neurosci.* **10**(135), 1 (2018). <https://doi.org/10.3389/fnagi.2018.00135>.
16. Turing, A.M.: I. Computing machinery and intelligence. *Mind* **59**(236), 433–460 (1950). <https://doi.org/10.1093/mind/LIX.236.433>.
17. McCarthy, J.; Minsky, M.L.; Rochester, N.; Shannon, C.E.: A proposal for the Dartmouth summer conference on artificial intelligence. *Rep. Dartmouth Coll.* **31**(08), 1955 (1955)
18. Jha, D.; Alam, S.; Pyun, J.-Y.; Lee, K.; Ho, K., Goo-Rak.: Alzheimer's disease detection using extreme learning machine, complex dual tree wavelet principal coefficients and linear discriminant analysis. *J. Med. Imag. Health Inf.* **8**(5), 881–890 (2018). <https://doi.org/10.1166/jmihi.2018.2381>.
19. Jie, B.; Liu, M.; Shen, D.: Integration of temporal and spatial properties of dynamic connectivity networks for automatic diagnosis of brain disease. *Med. Image Anal.* **47**, 81–94 (2018). <https://doi.org/10.1016/j.media.2018.03.013>.
20. Niu, Yan; Wang, Bin; Zhou, Mengni; Xue, Jiayue; Shapour, Habib; Cao, Rui; Cui, Xiaohong; Jinglong, Wu.; Xiang, Jie: Dynamic complexity of spontaneous bold activity in alzheimer's disease and mild cognitive impairment using multiscale entropy analysis. *Front. Neurosci.* **12**(677), 1 (2018). <https://doi.org/10.3389/fnins.2018.00677>.
21. Sampath, R.; Indumathi, J.: Earlier detection of Alzheimer disease using n-fold cross validation approach. *J. Med. Syst.* **42**(11), 217 (2018). <https://doi.org/10.1007/s10916-018-1068-5>.
22. Li, D.; Iddi, S.; Thompson, W.K.; Rafii, M.S.; Aisen, P.S.; Donohue, M.C.; Initiative, Alzheimer's Disease Neuroimaging.: Bayesian latent time joint mixed-effects model of progression in the Alzheimer's disease neuroimaging initiative. *Alzheimer's and Dement. Diagnos., Assessm. Dis. Monit.* **10**(1), 657–668 (2018). <https://doi.org/10.1016/j.dadm.2018.07.008>.
23. Buyrukoğlu, S.: Early detection of alzheimer's disease using data mining: comparison of ensemble feature selection approaches. *Konya Mühendislik Bilimleri Dergisi* **9**(1), 50–61 (2021). <https://doi.org/10.36306/konjes.731624>.
24. Rajendra Acharya, U.; Fernandes, S.L.; WeiKoh, J.E.; Ciaccio, E.J.; Mohd, F.; Mohd, K.; John, T.U.; Rajinikanth, V.; Yeong, C.H.: Automated detection of alzheimer's disease using brain mri images-a study with various feature extraction techniques. *J. Med. Syst.* **43**(9), 302 (2019). <https://doi.org/10.1007/s10916-019-1428-9>.
25. Lu, S.; Xia, Y.; Cai, W.; Fulham, M.; Feng, D.D.: Early identification of mild cognitive impairment using incomplete random forest-robust support vector machine and fdg-pet imaging. *Comput. Med. Imag. Gr.* **60**, 35–41 (2017). <https://doi.org/10.1016/j.compmedimag.2017.01.001>.
26. Savaş, S.; Topaloğlu, N.; Kazıcı, Ö.; Koşar, P.N.: Performance comparison of carotid artery intima media thickness classification by deep learning methods. In *SETSCI Conference Proceedings*, volume 4(5), pp. 125–131. International Congress on Human-Computer Interaction, Optimization, and Robotic Applications (2019). <https://doi.org/10.36287/setsci.4.5.025>.
27. Luo, S.; Li, X.L.J.: Automatic Alzheimer's disease recognition from mri data using deep learning method. *J. Appl. Math. Phys.* **5**(9), 1892–1898 (2017)
28. Lin, W.; Tong, T.; Gao, Q.; Guo, D.; Du, X.; Yang, Y.; Guo, G.; Xiao, M.; Du, M.; Qu, X.: The Alzheimer's Disease Neuroimaging Initiative. Convolutional neural networks-based MRI image analysis for the Alzheimer's disease prediction from mild cognitive impairment. *Front. Neurosci.* **12**(777), 1 (2018). <https://doi.org/10.3389/fnins.2018.00777>.
29. Li, F.; Liu, M.: Alzheimer's disease diagnosis based on multiple cluster dense convolutional networks. *Comput. Med. Imag. Gr.* **70**, 101–110 (2018). <https://doi.org/10.1016/j.compmedimag.2018.09.009>.
30. Bi, X.; Jiang, Q.; Sun, Q.; Shu, Q.; Liu, Y.: Analysis of Alzheimer's disease based on the random neural network cluster in FMRI. *Front. Neuroinf.* **12**(60), 1 (2018). <https://doi.org/10.3389/fninf.2018.00060>.
31. Suk, H.-I.; Lee, S.-W.; Shen, D.: Deep ensemble learning of sparse regression models for brain disease diagnosis. *Med. Image Anal.* **37**, 101–113 (2017). <https://doi.org/10.1016/j.media.2017.01.008>.
32. Liu, M.; Cheng, D.; Wang, K.; Wang, Y.: Initiative the Alzheimer's Disease Neuroimaging: Multi-modality cascaded convolutional neural networks for Alzheimer's disease diagnosis. *Neuroinformatics* **16**(3):295–308. <https://doi.org/10.1007/s12021-018-9370-4>.
33. Yiming, D.; Jae, H.S.; Kawczynski, M.G.; Trivedi, H.; Harnish, R.; Jenkins, N.W.; Lituiev, D.; Copeland, T.P.; Aboian, M.S.; Aparici, C.M.; Behr, S.C.; Flavell, R.R.; Huang, S.-Y.; Zalocusky, K.A.; Nardo, L.; Seo, Y.; Hawkins, R.A.; Pampaloni, M.H.; Hadley, D.; Franc, B.L.: A deep learning model to predict a diagnosis of Alzheimer disease by using 18f-fdg pet of the brain. *Radiology* **290**(2), 456–464 (2019). <https://doi.org/10.1148/radiol.2018180958>.
34. Yapıcı, M.M.; Tekerek, A.; Topaloğlu, N.: Literature review of deep learning research areas. *Gazi Mühendislik Bilimleri Dergisi (GMBD)* **5**(3), 188–215 (2019)
35. Savaş, S.; Topaloğlu, N.; Kazıcı, Ö.; Koşar, P.N.: Classification of carotid artery intima media thickness ultrasound images with deep learning. *J. Med. Syst.* **43**(8), 273 (2019). <https://doi.org/10.1007/s10916-019-1406-2>.
36. Krizhevsky, A.; Sutskever, I.; Hinton, G.E.: Imagenet classification with deep convolutional neural networks. *Adv. Neural. Inf. Process. Syst.* **25**, 1097–1105 (2012)
37. Zeiler, M.D.; Fergus, R.: Visualizing and understanding convolutional networks. In: *European conference on computer vision*, pp. 818–833. Springer (2014).
38. He, K.; Zhang, X.; Ren, S.; Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778 (2016).
39. Simonyan, K.; Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).
40. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 2278–2324 (1998)
41. Keras. Keras applications (2021). <https://keras.io/api/applications/>.
42. ADNI. Alzheimer's disease neuroimaging initiative (2021). <http://adni.loni.usc.edu/>.



43. Daldal, N.; Cömert, Z.; Polat, K.: Automatic determination of digital modulation types with different noises using convolutional neural network based on time-frequency information. *Appl. Soft Comput.* **86**, 105834 (2020). <https://doi.org/10.1016/j.asoc.2019.105834>
44. LeCun, Y.; Bengio, Y.; Hinton, G.: Deep learning. *Nature* **521**(7553), 436–444 (2015)
45. Weisstein, E.W.: Convolution (2021). <https://mathworld.wolfram.com/Convolution.html>.
46. Gao, B.; Pavel, L.: On the properties of the softmax function with application in game theory and reinforcement learning. arXiv preprint [arXiv:1704.00805](https://arxiv.org/abs/1704.00805) (2017).
47. Wikipedia. Softmax function, (2021). [https://en.wikipedia.org/wiki/Softmax\\_function](https://en.wikipedia.org/wiki/Softmax_function).
48. Gholamalinezhad, H.; Khosravi, H.: Pooling methods in deep neural networks, a review. [arXiv:2009.07485](https://arxiv.org/abs/2009.07485) (2020).
49. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014)
50. Amidi, A.; Amidi, S.: Convolutional neural networks cheat-sheet (2018). <https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks>.
51. Toprak, T.; Selver, M.A.; Secmen, M.; Zoral, E.Y.: Utilizing resonant scattering signal characteristics via deep learning for improved classification of complex targets. *Turk. J. Electr. Eng. Comput. Sci.* **29**(1), 334–348 (2021). <https://doi.org/10.3906/elk-2002-101>.
52. Usmani, Z.: (2018). <https://www.kaggle.com/getting-started/44916>.
53. Colab.: (2021). <https://research.google.com/colaboratory/faq.html>.
54. Zhao, H.; Gallo, O.; Frosio, I.; Kautz, J.: Loss functions for image restoration with neural networks. *IEEE Trans. Comput. Imaging* **3**(1), 47–57 (2017). <https://doi.org/10.1109/TCI.2016.2644865>.
55. Engle, R.F.: On the limitations of comparing mean square forecast errors: Comment. *J. Forecast.* **12**(8), 642–644 (1993). <https://doi.org/10.1002/for.3980120805>.
56. Peltarion.: Categorical crossentropy math, (2021). <https://peltarion.com/knowledge-center/documentation/modeling-view/build-an-ai-model/loss-functions/categorical-crossentropy>.
57. Christoffersen, P.; Jacobs, K.: The importance of the loss function in option valuation. *J. Financ. Econ.* **72**(2), 291–318 (2004). <https://doi.org/10.1016/j.jfineco.2003.02.001>.

