DSNN: A DenseNet-Based SNN for Explainable Brain Disease Classification

Ziquan Zhu[1†], Siyuan Lu[1†], Shui-Hua Wang[1,2]*, Juan Manuel Gorriz[3]* and Yu-Dong Zhang[1,2,4]*

[1]School of Computing and Mathematical Sciences, University of Leicester, East Midlands, United Kingdom, [2]School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo, China, [3]Department of Signal Theory, Networking and Communications, University of Granada, Granada, Spain, [4]Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin, China

Aims: Brain diseases refer to intracranial tissue and organ inflammation, vascular diseases, tumors, degeneration, malformations, genetic diseases, immune diseases, nutritional and metabolic diseases, poisoning, trauma, parasitic diseases, etc. Taking Alzheimer's disease (AD) as an example, the number of patients dramatically increases in developed countries. By 2025, the number of elderly patients with AD aged 65 and over will reach 7.1 million, an increase of nearly 29% over the 5.5 million patients of the same age in 2018. Unless medical breakthroughs are made, AD patients may increase from 5.5 million to 13.8 million by 2050, almost three times the original. Researchers have focused on developing complex machine learning (ML) algorithms, i.e., convolutional neural networks (CNNs), containing millions of parameters. However, CNN models need many training samples. A small number of training samples in CNN models may lead to overfitting problems. With the continuous research of CNN, other networks have been proposed, such as randomized neural networks (RNNs). Schmidt neural network (SNN), random vector functional link (RVFL), and extreme learning machine (ELM) are three types of RNNs.

Methods: We propose three novel models to classify brain diseases to cope with these problems. The proposed models are DenseNet-based SNN (DSNN), DenseNet-based RVFL (DRVFL), and DenseNet-based ELM (DELM). The backbone of the three proposed models is the pre-trained "customize" DenseNet. The modified DenseNet is fine-tuned on the empirical dataset. Finally, the last five layers of the fine-tuned DenseNet are substituted by SNN, ELM, and RVFL, respectively.

Results: Overall, the DSNN gets the best performance among the three proposed models in classification performance. We evaluate the proposed DSNN by five-fold cross-validation. The accuracy, sensitivity, specificity, precision, and F1-score of the proposed DSNN on the test set are 98.46% ± 2.05%, 100.00% ± 0.00%, 85.00% ± 20.00%, 98.36% ± 2.17%, and 99.16% ± 1.11%, respectively. The proposed DSNN is compared with restricted DenseNet, spiking neural network, and other state-of-the-art methods. Finally, our model obtains the best results among all models.

Conclusions: DSNN is an effective model for classifying brain diseases.

Keywords: brain diseases, convolutional neural network, randomized neural network, DenseNet, MRI

INTRODUCTION

Brain diseases refer to intracranial tissue and organ inflammation, vascular diseases, tumors, degeneration, malformations, genetic diseases, immune diseases, nutritional and metabolic diseases, poisoning, trauma, parasitic diseases, etc. Brain diseases often show disorders of consciousness, sensation, movement, or autonomic nerve dysfunction. There may also be fever, headache, vomiting, and other mental symptoms. Taking Alzheimer's disease (AD) as an example, the number of patients dramatically increases in developed countries. By 2025, the number of elderly patients with AD aged 65 and over will reach 7.1 million, increasing nearly 29% over the 5.5 million patients of the same age in 2018 (Lynch, 2018). Unless medical breakthroughs are made, the number of Alzheimer's patients aged 65 and over may increase from 5.5 million to 13.8 million by 2050, almost three times the original.

Now, brain diseases are mainly diagnosed by doctors. However, the manual diagnosis requires much time. At the same time, different doctors may have different views on the same examination results, which has brought a lot of trouble to patients.

More and more researchers use computational methods (Wang et al., 2021) to classify brain diseases. Noreen et al. (2020) introduced a multi-level method using two DensNet201 and Inception-v3 to diagnose early brain tumors. Finally, the accuracy of Inception-v3 and DensNet201 were 99.34% and 99.51%, respectively. Amin et al. (2019a) presented a model using magnetic resonance images to automatically classify brain tumors according to the LSTM model method. What's more, this method obtained 0.97 DSC in practical application. Amin et al. (2019b) used a deep learning model to predict healthy and unhealthy brain tumor slices. Arunkumar et al. (2020) introduced a new model to identify ROI location based on brain tumor MRI. The method finally got 89% sensitivity, 92.14% accuracy, and 94% specificity. Purushottam Gumaste and Bairagi (2020) proposed an algorithm to extract left and right brain features. This article also introduced different statistical feature extraction methods and used a support vector machine to extract tumor regions from statistical features. Chatterjee and Das (2019) proposed a novel method for the segmentation of brain images, which were divided into two categories: benign (low level) and evil (high level). Bhanothu et al. (2020) presented a new method according to R-CNN to detect tumors and mark their location. Finally, the detection and classification accuracy of the three types of brain tumors were 89.45%, 68.18%, and 75.18%. Natekar et al. (2020) compared various technologies for brain tumor segmentation models and visualized the internal concepts to have a deeper understanding of how these technologies segmented with high accuracy. Aboelenein et al. (2020) introduced a novel network (HTTU-Net) for brain tumor cutting. Huang et al. (2020) presented the differential feature neural network (DFNN) method. The method introduced DFM blocks and combined SE blocks. When the DFM block was introduced, the accuracy of the two databases was improved by 1.8% and 1.3%, respectively. Hu and Razmjooy (2020) proposed a meta heuristic-based system to detect tumors. Sadad et al. (2021) introduced a

novel model according to UNET architecture and ResNet50 as the backbone for the detection of brain tumors. Kalaiselvi et al. (2020) proposed a patch-based-updated run-length region growth (PR2G) method to detect and segment tumors. The accuracy of this method was 97%. Kaplan et al. (2020) used two methods to classify the three different types of brain tumors. The two methods were nLBP and αLBP. The highest classification accuracy of brain tumors was 95.56%. Khalil et al. (2020) proposed a new method (DA clustering) to improve the accuracy of extracting initial contour points to detect three-dimensional magnetic resonance brain tumors better. Khan et al. (2020) proposed a new method, partial tree (PART), to detect brain tumors of grade I to grade IV brain tumors. This method used the rule learner of an advanced feature set. Ma and Zhang (2021) proposed a method to intelligently detect brain tumors based on a lightweight neural network. Hollon et al. (2020) introduced a new method for the automatic detection of brain tumors by combining SRH 5–7, CNN, and the label-free optical imaging method. Saba et al. (2020) used a new method to detect brain tumors. The Grasp cut method was used to segment brain tumor symptoms, and VGG-19 was used to obtain features. Sharif et al. (2020) proposed an unsupervised fuzzy set method for brain tumor segmentation. The triangular fuzzy median filter enhanced the image to better detect brain tumors. Xu et al. (2020) presented a new structure for the early detection of brain tumors. The new structure was mainly composed of five parts: tumor segmentation, morphology, denoising, feature extraction, and classification. Hemanth et al. (2011) introduced a novel method (HSBPN) to segment MR brain tumor images. Nayef et al. (2013) introduced a novel structure for the classification of the MRI dataset. Chen et al. (2017) presented an improved method for detecting pathological brains. A new classifier was used in the improved method. Shoeibi et al. (2020) finished a review on the segmentation of the Covid-19 by DL. Shoeibi et al. (2021a) performed a comprehensive survey about the application of DL in the detection of multiple sclerosis. Sadeghi et al. (2021) showed a survey on the automatic diagnosis of the SZ by AI. Shoeibi et al. (2021b) completed a comprehensive review on the application of the various AI techniques in the diagnosis of epileptic seizures. Shoeibi et al. (2021c) completed a review of various methods based on DL for automatic diagnosis of SZ by electroencephalogram (EEG) signals. Shoeibi et al. (2022) proposed a new model to automatically detect Epileptic seizures. The proposed model was based on the DL and the fuzzy theory. Odusami et al. (2022) proposed a method for the recognition of AD. They tested two CNN models (DenseNet201 and ResNet18) to perform this task. This method obtained 98.86% accuracy, 98.94% precision, and 98.89% recall. Razzak et al. (2022) introduced a new network (PartialNet) to detect AD based on MRIs. This network achieved improvements on the AD detection. Ashraf et al. (2021) experimented with different CNN models to detect AD based on transfer learning. Finally, the fine-tuned DenseNet got the highest accuracy (99.05%).

If brain diseases are diagnosed manually, doctors need to spend a lot of time on examination. Sometimes we may encounter the problem that different doctors have different views on the examination results of the same patient. As shown in

**TABLE 1 |** Contributions of state-of-the-art methods.

Method	Contribution
Noreen et al. (2020)	A multi-level method using two DensNet201 and Inception-v3 was proposed to diagnose early brain tumors.
Amin et al. (2019a)	A model according to the LSTM model method using magnetic resonance images was introduced to classify brain tumors automatically.
Amin et al. (2019b)	A deep learning model was used to predict healthy and unhealthy brain tumor slices.
Arunkumar et al. (2020)	A new model was introduced to train MRI brain tumors to identify ROI location.
Purushottam Gumaste and Bairagi (2020)	An algorithm was proposed to extract left and right brain features. This article also introduced different statistical feature extraction methods and used a Support Vector Machine to extract tumor regions from statistical features.
Chatterjee and Das (2019)	A novel method was proposed for the segmentation of brain images.
Bhanothu et al. (2020)	A new method based on R-CNN was presented to detect tumors and mark their location.
Natekar et al. (2020)	Various technologies were compared for brain tumor segmentation models.
Aboelenein et al. (2020)	The HTTU-Net was proposed for brain tumor cutting.
Huang et al. (2020)	The DFNN was proposed. The method introduced DFM blocks and combined SE blocks.
Hu and Razmjooy (2020)	A meta heuristic-based system was presented to detect tumors.
Sadad et al. (2021)	A novel model according to UNET architecture and ResNet50 as the backbone was proposed for the detection of brain tumors.
Kalaiselvi et al. (2020)	The PR2G was proposed to detect and segment tumors.
Kaplan et al. (2020)	Then LBP and αLBP were used to classify the three different types of brain tumors.
Khalil et al. (2020)	The DA clustering was proposed to improve the accuracy of extracting initial contour points to detect three-dimensional magnetic resonance brain tumors better.
Khan et al. (2020)	The PART was introduced to detect brain tumors of grade I to grade IV brain tumors.
Ma and Zhang (2021)	A method was proposed to intelligently detect brain tumors based on a lightweight neural network.
Hollon et al. (2020)	A new method was proposed for the automatic detection of brain tumors by combining SRH 5–7, CNN, and the label-free optical imaging method.
Saba et al. (2020)	A new method was proposed to detect brain tumors. The Grasp cut method was used to segment brain tumor symptoms, and VGG-19 was used to obtain features.
Sharif et al. (2020)	An unsupervised fuzzy set method was introduced for brain tumor segmentation.
Xu et al. (2020)	A new structure was proposed for the early detection of brain tumors. The new structure was mainly composed of five parts: tumor segmentation, morphology, denoising, feature extraction, and classification.
Hemanth et al. (2011)	The HSBPN was proposed to segment MR brain tumor images.
Nayef et al. (2013)	A novel structure was presented for the classification of the MRI dataset.
Chen et al. (2017)	An improved method was introduced for detecting pathological brains.
Shoeibi et al. (2020)	A review was presented on the segmentation of the Covid-19 by DL.
Shoeibi et al. (2021a)	A comprehensive survey about the application of DL in the detection of Multiple Sclerosis
Sadeghi et al. (2021)	A survey was presented on the automatic diagnosis of the SZ by AI.
Shoeibi et al. (2021b)	A comprehensive review was presented on applying the various AI techniques in the diagnosis of Epileptic seizures.
Shoeibi et al. (2021c)	A review of various methods based on DL for automatic diagnosis of SZ by electroencephalogram (EEG) signals was completed.
Shoeibi et al. (2022)	A new model was proposed to detect Epileptic seizures automatically. The proposed model was based on the DL and the fuzzy theory.
Odusami et al. (2022)	A method was proposed for the recognition of AD. They tested two CNN models (DenseNet201 and ResNet18) to perform this task.
Razzak et al. (2022)	A new network (PartialNet) was introduced to detect AD based on MRIs. This network achieved improvements in AD detection.
Ashraf et al. (2021)	Different CNN models were experimented with to detect AD based on transfer learning. Finally, the fine-tuned DenseNet got the highest accuracy (99.05%).

Table 1, most researchers use deep convolution neural networks (DCNNs) to classify and identify brain diseases. However, there will be many parameters and calculations in the training of DCNN, which can lead to a long training time (Zhang et al., 2021). At the same time, DCNNs need a sea number of experimental data for training because a small number of experimental data may lead to overfitting problems (Górriz et al., 2020; Zhang et al., 2020).

To cope with the problems mentioned above, we propose three novel models to classify brain diseases automatically.

They are: DenseNet-based Schmidt neural network (DSNN), DenseNet-based random vector functional link (DRVFL), and DenseNet-based extreme learning machine (DELM). We select DenseNet to extract features and use randomized neural networks (RNNs) for classification.

We modify the pre-trained DenseNet. Then, the modified DenseNet is fine-tuned on the dataset. In the DSNN, the last five layers within the fine-tuned DenseNet are substituted by the Schmidt neural network (SNN). In the DRVFL, we select the RVFL (RVFL) to substitute the last five layers of the

FIGURE 1 | (A) Unhealthy and **(B)** healthy brain images in the dataset.
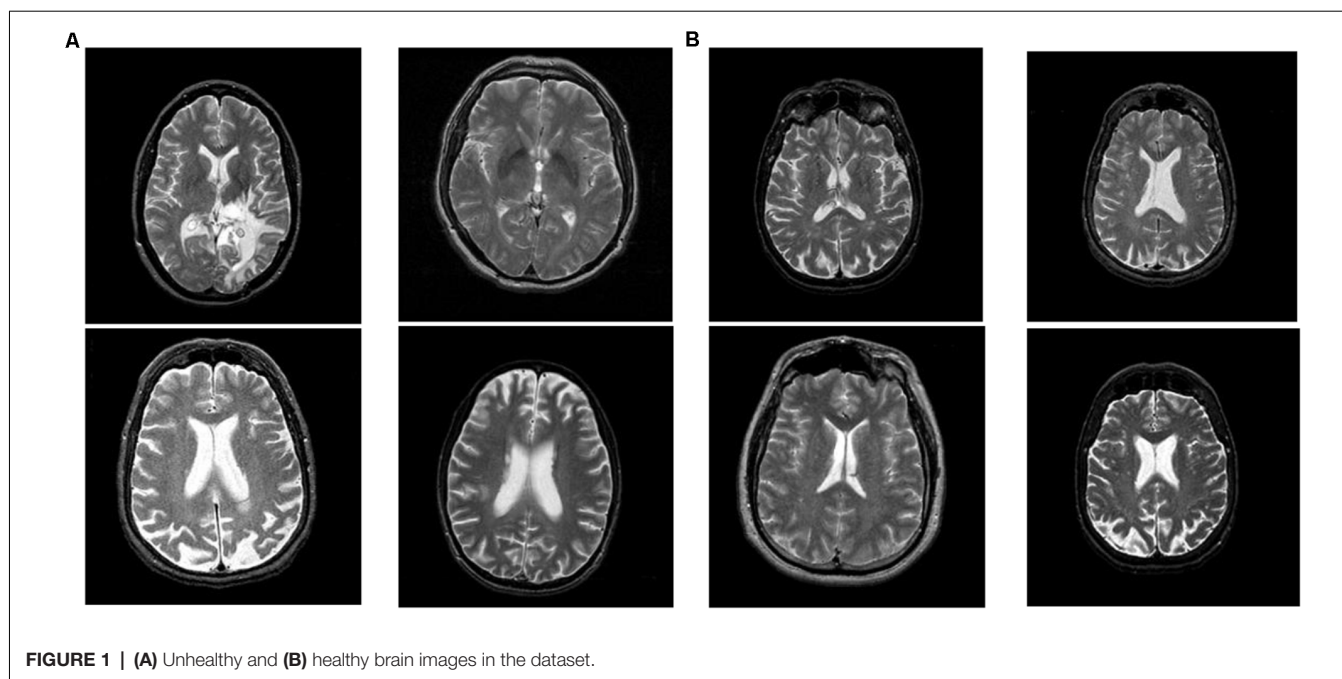
fine-tuned DenseNet. In the DELM, we choose the extreme learning machine (ELM) to replace the end five layers of the fine-tuned DenseNet.Five-fold cross-validation is used to evaluate the proposed three models: DSNN, DRVFL, and DELM, in terms of aspects (Acc, Sen, Spe, Pre, and F1). We finally get thatDSNN gives the best performance among the three proposed models and overperforms the other six state-of-the-art algorithms. The five main innovations of this study are:

(1) DenseNet is validated as the backbone by experiments showing its superiority to AlexNet, ResNet-18, ResNet-50, and VGG.
(2) DSNN, DRVFL, and DELM are proposed by replacing the last five layers within the fine-tuned DenseNet with three randomized neural networks.
(3) The DSNN gets the best performance among the three proposed models.
(4) The DSNN overperforms the restricted DenseNet and spiking neural network by experiments.
(5) The DSNN is compared with six state-of-the-art algorithms and obtains the best results among the list methods.

The rest of this article is as follows. The dataset is given in Section "Materials". Section "Methodology" discusses the methodology. Section "Results and Discussion" is about the experiment results. We conclude this article in Section "Conclusion".

MATERIALS

The dataset is downloaded from the Harvard Medical School website (Johnson and Becker, 2021). There are four types of brain diseases: cerebrovascular disease, neoplastic disease, degenerative disease, and inflammatory or infectious disease. This article classifies all four brain disease images as unhealthy brain images.

A total of 177 unhealthy brain images and 20 healthy brain images are used in this article. The size of all images in this article is 256×256. Some unhealthy and healthy brain images in this article are shown in **Figure 1**. The left four images are the unhealthy brain images, and the right four are the healthy images.

METHODOLOGY

Proposed DSNN

Tables 2, **3** give the acronym definitions and parameter definitions, respectively. More and more researchers devote energy to researching image classification technology (Lu S. et al.,

TABLE 2 | Acronym and full explanation.

Acronym	Full explanation
AD	Alzheimer's disease
Acc	Accuracy
Avr	Average
BN	Batch normalization
CNN	Convolution neural network
DCNN	Deep convolution neural network
DELM	DenseNet-based extreme learning machine
DL	Deep learning
DRVFL	DenseNet-based random vector functional link
DSNN	DenseNet-based Schmidt neural network
ELM	Extreme learning machine
F1	F1-score
FC	Fully connected
ML	Machine learning
Pre	Precision
RVFL	Random vector functional link
RNNs	Randomized neural networks
Sen	Sensitivity
SNN	Schmidt neural network
Spe	Specificity
Std	Standard deviation

TABLE 3 | The definition of the parameter.

Parameter	Definition
O_m	The output of the M-th layer
T_m	The nonlinear transformation
(x_i, y_i)	The given dataset
n	The input dimension
M	The output dimension
w_j	The weights vector
d_j	The bias of the j-th hidden node
P	The final output weights
q	The output biases of SNN
$Y = (y_1, ..., y_N)^\top$	The ground-truth label matrix of the dataset
$X = (x_1, ..., x_N)^\top$	The input matrix
$s()$	The sigmoid function
V	The number of hidden nodes
A	The output matrix of the hidden layer

2021). In image classification, feature extraction is a crucial step. However, the image contains too much messy information, so extracting valuable features is difficult. Decades ago, people usually manually extracted features. However, manual feature extraction takes much time, and the results are usually not ideal. With the continuous progress of computer technology, more and more people use computer models for image feature extraction (Leming et al., 2020). Many computer models are successful (Lu S. Y. et al., 2021), such as CNN models. The convolution layer in the CNN model can significantly reduce the volume of parameters to shorten the training time. Researchers have proposed many great CNN models, such as AlexNet (Lu et al., 2020a), MobileNet (Lu S.-Y. et al., 2020), ResNet (Lu et al., 2020b), and so on. This article proposes three models for the automatic classification of brain diseases: DSNN, DRVFL, and DELM. The DSNN gets the best performance among the three proposed models.

The pseudocode of the proposed DSNN is shown in **Table 4**. The pipeline of our model is given in **Figure 2**. We choose the pre-trained DenseNet as the backbone of the proposed DSNN. We modify the pre-trained DenseNet. Then, the modified DenseNet is fine-tuned on the dataset. The last five layers within the fine-tuned DenseNet are substituted by the Schmidt neural network (SNN). In our model, the fine-tuned DenseNet plays the role of feature extraction. The SNN is trained by the extracted features F from the fine-tuned DenseNet. Five-fold cross-validation is used to evaluate the proposed DSNN.

Backbone of the Proposed DSNN

The CNN models (Albawi et al., 2017) have been researched continuously in recent decades. In 1998, LeCun proposed LeNet (LeCun, 2015) with a five-layer structure. In 2014, the visual geometry group proposed VGG (Simonyan and Zisserman, 2014) with a 19-layer structure. The Highway Networks (Srivastava et al., 2015) were proposed later, with more than 100 layers.

With the increasing number of network layers in CNN models, researchers are troubled by the problem of gradient vanishing. Batch normalization (BN) alleviates the problem of gradient vanishing to some extent. ResNet (He et al., 2016) reduces the gradient vanishing problem by constructing identity

TABLE 4 | Pseudocode of the proposed DSNN.

Step 1: Load the pre-trained DenseNet.
Step 2: Modify the pre-trained DenseNet.
      Step 2.1 Remove softmax and classification layer from the pre-trained DenseNet.
      Step 2.2 Add FC128, ReLU, BN, FC2, softmax, and classification layer.
Step 3: Divide the dataset into five groups of the same size and set $i=1$
      Step 4: Use the i-th group as the test set, and all the other groups form the training set.
      Step 5: Fine-tune the modified DenseNet.
        Step 5.1: Input is the training set.
        Step 5.2: Target is the corresponding label.
      Step 6: Replace the last five layers of the fine-tuned DenseNet with SNN.
      Step 7: Extract features F as the output of the FC128 layer.
      Step 8: Train the classifier of the DSNN on the extracted features F and the labels.
        Step 8.1: Input is the extracted features.
        Step 8.2: The target is the label of the training set.
        Step 8.3: SNN is the classifier of the DSNN.
      Step 9: Test the trained DSNN on the test set.
      Step 10: Report the test classification performance of the trained DSNN.
      Step 11: Set $i= i + 1$, if $i < 6$, go to Step 4.
Step 12: Average test classification performance.

mapping. In 2017, DenseNet (Huang et al., 2017) was proposed to reduce the gradient vanishing problem by establishing dense connectivity between the front and rear layers. Dense connectivity makes more effective use of features than other networks. Thus, DenseNet can achieve better performance. The general view of DenseNet is given in **Figure 3A**.

Dense blocks refer to the specific blocks of DenseNet, as shown in **Figure 3A**. All the front layers are connected with the rear layers. In the same dense block, the height and width of each feature map will not change, but the number of channels will change. In the traditional sequential CNN, if you have M layers, there will be M connections, but DenseNet will introduce $M(M+1)/2$ more connections. Supposing there are M layers, O_M denotes the output of the M-th layer, T_M represents the nonlinear transformation. The comparison of DenseNet with other CNNs is listed below:

Traditional sequential CNN:

$$O_M = T_M(O_{M-1}) \tag{1}$$

ResNet:

$$O_M = T_M(O_{M-1}) + O_{M-1} \tag{2}$$

DenseNet:

$$O_M = T_M([O_0, O_1, ..., O_{M-1}]) \tag{3}$$

where [] is the concatenation.

The transition layer is a module that connects different dense blocks. Its primary function is to integrate the features obtained from the previous dense block and reduce its width and height.

Researchers used the ImageNet dataset to pre-train the DenseNet. There are 1,000 output nodes on the pre-trained DenseNet. However, this article only needs two output nodes. We modify the pre-trained DenseNet. The modifications are shown in **Figure 3B**.

After these modifications, we fine-tune the modified DenseNet by the training set. We remove the last five layers of the
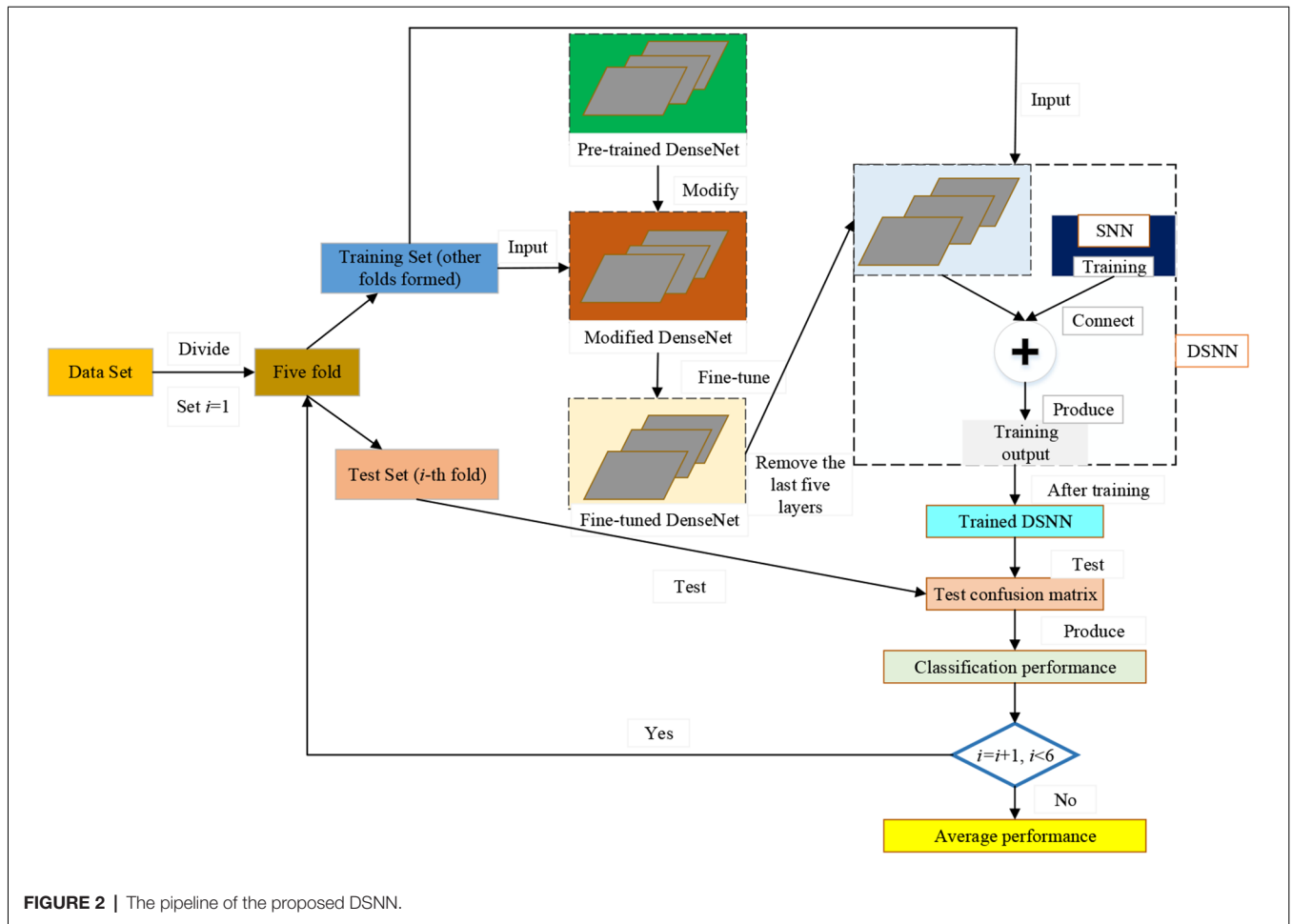
FIGURE 2 | The pipeline of the proposed DSNN.

fine-tuned DenseNet and add SNN to improve the classification performance. In the proposed DSNN, the fine-tuned DenseNet is the feature extraction.

Three Proposed Networks

Compared with the pre-trained DenseNet, randomized neural networks (RNNs) have a much shorter training time. In the DSNN, we replace the end five layers of the fine-tuned DenseNet with the RNN: the Schmidt neural network (SNN; Schmidt et al., 1992). The SNN is trained by extracted features n from FC128. The structure of the SNN is shown in **Figure 4**.

The yellow box is the input, the pink circle represents the hidden nodes, and the green box shows the output. Given N samples and dataset with the i-th sample as (x_i, y_i):

$$x_i = (x_{i1}, ..., x_{in})^T \in R^n, i = 1, ..., N, \quad (4)$$

$$y_i = (y_{i1}, ..., y_{im})^T \in R^m, i = 1, ..., N, \quad (5)$$

where n is the input dimension, m is the output dimension.

The training algorithm of SNN is as follows. The weights vector (w_j) connects the j-th hidden node with input nodes, d_j is the bias of the j-th hidden node. The weights vector (w_j) and the bias (d_j) are assigned with random values and will remain

unchanged during the training process. The output matrix of the hidden layer with V hidden nodes is calculated as follows:

$$\mathbf{A}_{\mathrm{SNN}} = \sum_{j=1}^{V} s(w_j x_i + d_j), i = 1, ..., N, \quad (6)$$

where the sigmoid function is represented as $s()$. Then we use pseudo-inverse to calculate the final output weights (P):

$$(P, q) = \mathbf{A}_{\mathrm{SNN}}^{\dagger} Y, \quad (7)$$

where the output biases of SNN are q, $\mathbf{A}_{\mathrm{SNN}}^{\dagger}$ is the pseudo-inverse matrix of $\mathbf{A}_{\mathrm{SNN}}$, and $Y = (y_1, ..., y_N)^T$ is the ground-truth label matrix of the dataset.

We propose two other models: DRVFL and DELM. The backbone of the two other proposed models is the pre-trained DenseNet. We modify the pre-trained DenseNet in the two proposed models as the "modifications of the pre-trained DenseNet" in the DSNN. We replace the softmax and classification layer of the pre-trained DenseNet with six layers: FC128, ReLU, BN, FC2, softmax, and classification layer. We fine-tune the modified DensNet by the training set. In the DRVFL, we select RVFL (Pao et al., 1994) to substitute the last five layers of the fine-tuned DenseNet. The structure of RVFL is
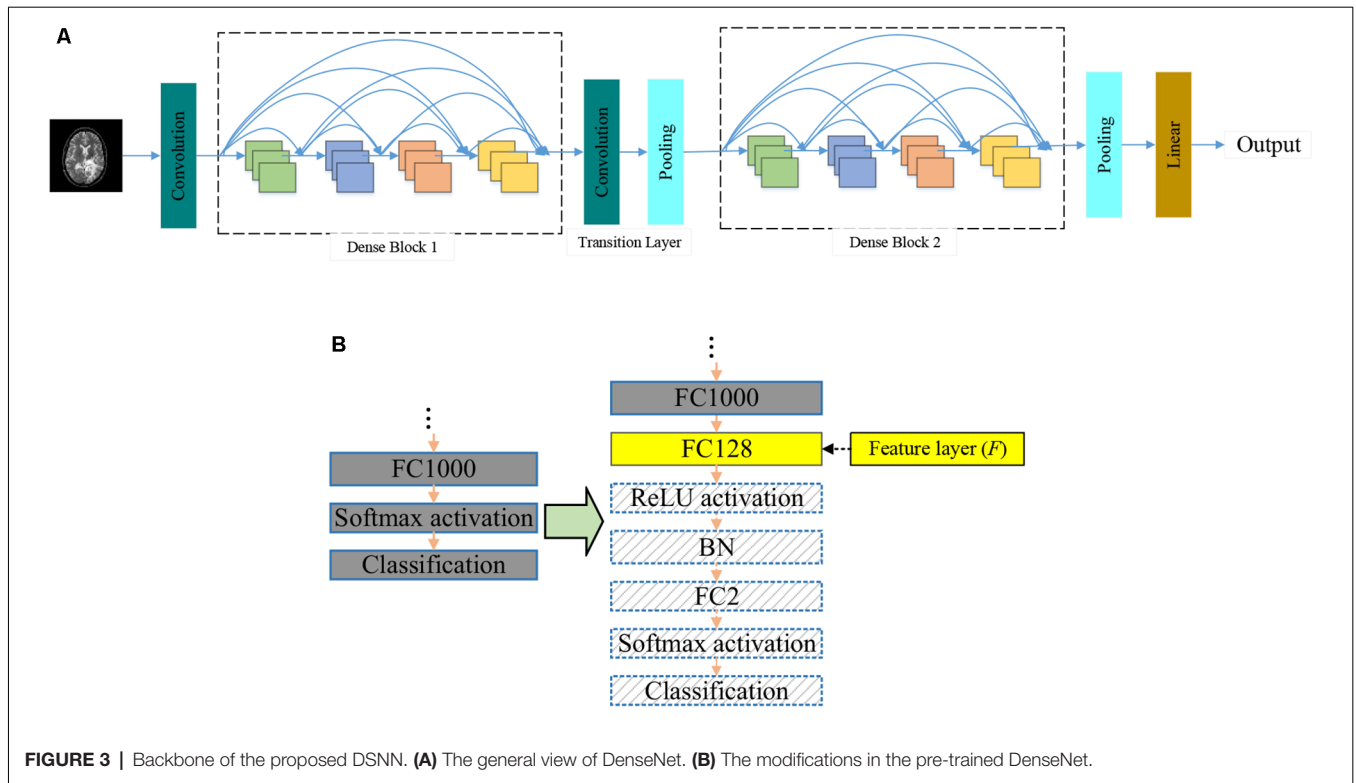
FIGURE 3 | Backbone of the proposed DSNN. **(A)** The general view of DenseNet. **(B)** The modifications in the pre-trained DenseNet.
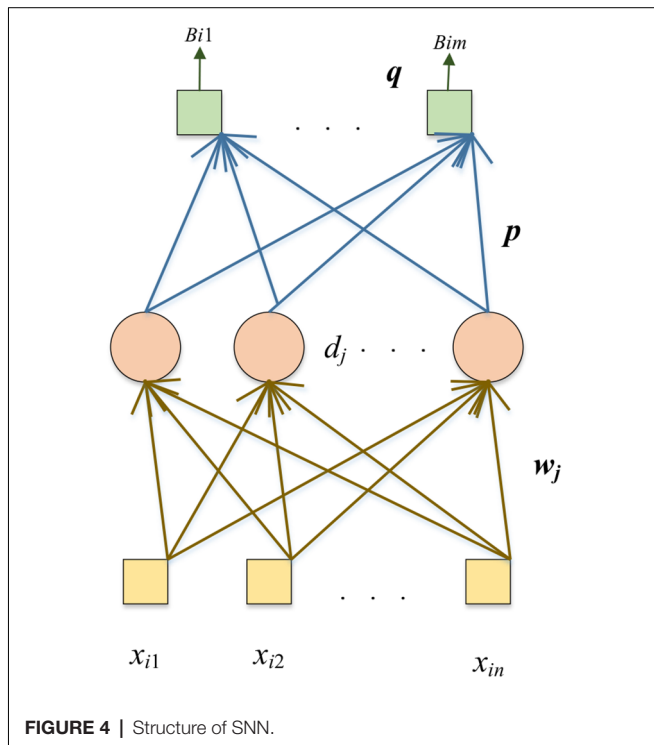
FIGURE 4 | Structure of SNN.

shown in **Figure 5A**. In the DELM, ELM (Huang et al., 2006) is chosen to replace the last five layers of the fine-tuned DenseNet. The structure of ELM is shown in **Figure 5B**. ELM and RVFL are two types of RNNs.

The yellow box represents the input, the pink circle denotes the hidden nodes, and the green box shows the output. The difference between these two RNNs is that there are shortcut connections from the input to the output in RVFL. The calculation steps are similar:

Given N samples and dataset with the i-th sample as (x_i, y_i)

$$x_i = (x_{i1}, ..., x_{in})^{\mathrm{T}} \in R^n, i = 1, ..., N, \quad (8)$$

$$y_i = (y_{i1}, ..., y_{im})^{\mathrm{T}} \in R^m, i = 1, ..., N, \quad (9)$$

where n is the input dimension, m is the output dimension. The training steps of these two RNNs are as follows:

Step 1: w_j is the weight vector, which connects the input nodes with the j-th hidden node. The bias of the j-th hidden node is represented as d_j. We randomly assign w_j and d_j with values. These values will not change in training.

Step 2: The hidden layer's output matrix is calculated as:
For RVFL:

$$\mathbf{A_{RVFL}} = \text{concat}(\mathbf{X}, \mathbf{K}), \quad (10)$$

where $\mathbf{X} = (x_1, ..., x_N)^{\mathrm{T}}$ denotes the input matrix. The $\mathbf{K}$ is calculated as follows:

$$\mathbf{K_{RVFL}} = \sum_{j=1}^{V} s(w_j x_i + d_j), i = 1, ..., N, \quad (11)$$

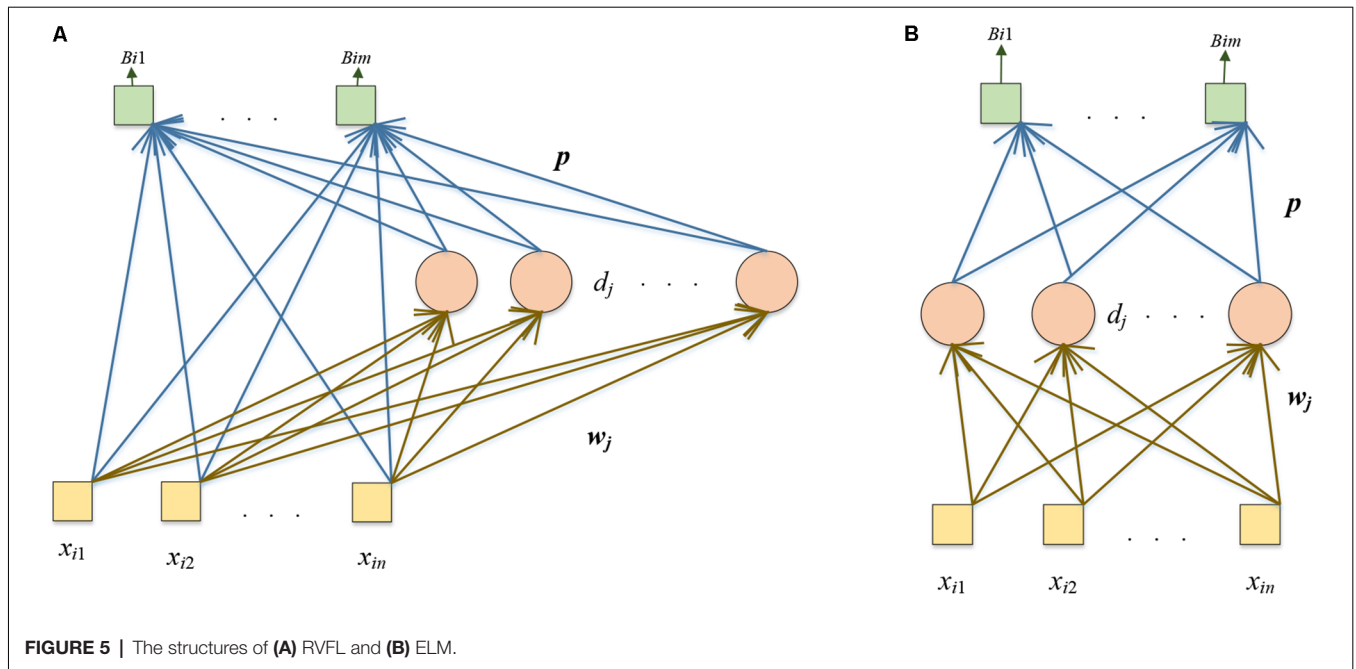where V is the number of the hidden nodes in the hidden layer, $s()$ represents the sigmoid function.

FIGURE 5 | The structures of **(A)** RVFL and **(B)** ELM.

For ELM:

$$\mathbf{A_{ELM}} = \sum_{j=1}^{V} s(\boldsymbol{w_j}\boldsymbol{x_i} + d_j), i = 1, ..., N, \qquad (12)$$

Step 3: The output weights ($\boldsymbol{p}$): can be calculated by pseudo-inverse:

For RVFL:

$$\boldsymbol{p} = \mathbf{A_{RVFL}^{\dagger}}\mathbf{Y}, \qquad (13)$$

where $\mathbf{A_{RVFL}^{\dagger}}$ is the pseudo-inverse matrix of $\mathbf{A_{RVFL}}$, and $\mathbf{Y} = (\boldsymbol{y_1},...,\boldsymbol{y_N})^{\mathrm{T}}$ is the ground-truth label matrix of the dataset.

For ELM:

$$\boldsymbol{p} = \mathbf{A_{ELM}^{\dagger}}\mathbf{Y}, \qquad (14)$$

where $\mathbf{A_{ELM}^{\dagger}}$ is the pseudo-inverse matrix of $\mathbf{A_{ELM}}$.

The backbone of these other two proposed models in this article is the same. The difference is that DRVFL chooses RVFL as its classifier, and DELM selects ELM as its classifier.

Evaluation

We define the unhealthy brain as the positive and the healthy brain as the negative. Five indicators are chosen to verify our model: accuracy (Acc), sensitivity (Sen), specificity (Spe), precision (Pre), and F1-score (F1), respectively. Their formulas are shown below:

$$\begin{cases} \mathrm{Acc} = \frac{\mathrm{TP + TN}}{\mathrm{TP + TN + FP + FN}} \\ \mathrm{Sen} = \frac{\mathrm{TP}}{\mathrm{TP + FN}} \\ \mathrm{Spe} = \frac{\mathrm{TN}}{\mathrm{TN + FP}} \\ \mathrm{Pre} = \frac{\mathrm{TP}}{\mathrm{TP + FP}} \\ \mathrm{F1} = \frac{2 \times \mathrm{TP}}{\mathrm{2TP + FP + FN}} \end{cases} \qquad (15)$$

where the definitions of TP, FN, FP, and TN are the true positive, false negative, false positive, and true negative, respectively.

TABLE 5 | The hyper-parameter settings of the proposed DSNN.

Hyper-parameter	Value
Mini-batch size	10
Max-epoch	4
Learning rate	10^{-4}
Number of the hidden nodes V	400

RESULTS AND DISCUSSIONS

Experiment Settings

We modify the hyper-parameter settings of the proposed DSNN. The max-epoch is set to 4 for reducing overfitting problems. We set our mini-batch size to 10 because the dataset is relatively small. According to the experience, the learning rate is 10^{-4}. A hyper-parameter we set in our model is the number of hidden nodes (V), which is set as 400 based on the input dimension. The hyper-parameter settings of our model are shown in **Table 5**.

Performances of the DSNN

We use five-fold cross-validation to evaluate the proposed DSNN. The classification performance of our model is given in **Table 6**. The Acc, Sen, Spe, Pre, and F1 of the proposed DSNN are 98.46% ± 2.05% , 100.00% ± 0.00% , 85.00% ± 20.00% , 98.36% ± 2.17%, and 99.16% ± 1.11% , respectively. The results of DSNN are higher than 85%. Especially the sensitivity is 100%. The ROC curve is shown in **Figure 6**. The AUC value is 0.9786. It is an effective classifier when the AUC value is greater than 0.95. These results can be concluded that DSNN is an effective model to classify brain diseases.

Comparison of Three Proposed Models

The classification performances of DRVFL and DELM based on the five-fold cross-validation are shown in **Table 6**. For a

TABLE 6 | The classification performance based on five-fold cross-validation (unit: %).

Methods	Fold	Acc	Sen	Spe	Pre	F1
DSNN(Ours)	F 1	100.00	100.00	100.00	100.00	100.00
	F 2	100.00	100.00	100.00	100.00	100.00
	F 3	94.87	100.00	50.00	94.59	97.22
	F 4	100.00	100.00	100.00	100.00	100.00
	F 5	97.44	100.00	75.00	97.22	98.59
	Avr	**98.46**	**100.00**	**85.00**	**98.36**	**99.16**
	Std	±2.05	±0.00	±20.00	±2.17	±1.11
DRVFL(Ours)	F 1	100.00	100.00	100.00	100.00	100.00
	F 2	100.00	100.00	100.00	100.00	100.00
	F 3	89.74	100.00	0.00	89.74	94.59
	F 4	100.00	100.00	100.00	100.00	100.00
	F 5	97.44	100.00	75.00	97.22	98.59
	Avr	97.44	100.00	75.00	97.39	98.64
	Std	±3.97	±0.00	±38.73	±3.97	±2.10
DELM(Ours)	F 1	100.00	100.00	100.00	100.00	100.00
	F 2	100.00	100.00	100.00	100.00	100.00
	F 3	92.31	100.00	25.00	92.11	95.89
	F 4	100.00	100.00	100.00	100.00	100.00
	F 5	97.44	100.00	75.00	97.22	98.59
	Avr	97.95	100.00	80.00	97.87	98.90
	Std	±2.99	±0.00	±29.15	±3.07	±1.60
Fine-tuned DenseNet	F 1	87.50	86.11	100.00	100.00	92.54
	F 2	82.05	80.00	100.00	100.00	88.89
	F 3	89.74	88.57	100.00	100.00	93.94
	F 4	85.00	83.33	100.00	100.00	90.91
	F 5	79.49	77.14	100.00	100.00	87.10
	Avr	84.76	83.03	100.00	100.00	90.67
	Std	±3.67	±4.10	±0.00	±0.00	±2.46
AlexNet-SNN	F 1	89.74	100.00	0.00	89.74	94.59
	F 2	89.74	100.00	0.00	89.74	94.59
	F 3	90.00	97.22	25.00	92.11	94.59
	F 4	90.00	97.22	25.00	92.11	94.59
	F 5	89.74	97.14	25.00	91.89	94.44
	Avr	89.84	98.32	15.00	91.12	94.56
	Std	±0.13	±1.38	±12.25	±1.13	±0.06
ResNet-18-SNN	F 1	100.00	100.00	100.00	100.00	100.00
	F 2	97.50	100.00	75.00	97.30	98.63
	F 3	100.00	100.00	100.00	100.00	100.00
	F 4	94.87	100.00	50.00	94.59	97.22
	F 5	94.87	97.14	75.00	97.14	97.14
	Avr	97.45	99.43	80.00	97.81	98.60
	Std	±2.29	±1.14	±18.71	±2.03	±1.26
ResNet-50-SNN	F 1	95.00	94.44	100.00	100.00	97.14
	F 2	100.00	100.00	100.00	100.00	100.00
	F 3	97.44	97.14	100.00	100.00	98.55
	F 4	95.00	100.00	50.00	94.74	97.30
	F 5	100.00	100.00	100.00	100.00	100.00
	Avr	97.49	98.32	90.00	98.95	98.60
	Std	±2.24	±2.23	±20.00	±2.10	±1.24
VGG-SNN	F 1	97.50	100.00	75.00	97.30	98.63
	F 2	87.50	94.44	25.00	91.89	93.15
	F 3	94.87	97.14	75.00	97.14	97.14
	F 4	89.74	100.00	0.00	89.74	94.59
	F 5	87.18	88.57	75.00	96.88	92.54
	Avr	91.36	96.03	50.00	94.59	95.21
	Std	±4.12	±4.26	±31.62	±3.16	±2.33
Restricted DenseNet-SNN	F 1	94.87	94.29	100.00	100.00	97.06
	F 2	100.00	100.00	100.00	100.00	100.00
	F 3	97.37	97.06	100.00	100.00	98.51
	F 4	94.87	100.00	50.00	94.59	97.22
	F 5	100.00	100.00	100.00	100.00	100.00
	Avr	97.42	98.27	90.00	98.92	98.56
	Std	±2.09	±1.83	±1.97	±2.09	±1.69

The bold values are results of our proposed model.
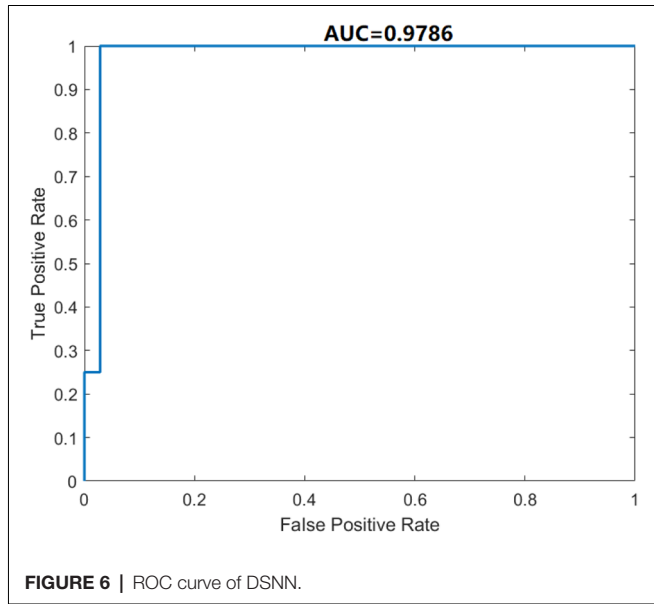
FIGURE 6 | ROC curve of DSNN.

gets the best performance among the three proposed models because there is an output bias in SNN.

Comparison With the Fine-Tuned DenseNet

We compare the proposed DSNN with the fine-tuned DenseNet. The classification performance of the fine-tuned DenseNet is given in **Table 6**. The comparison figure of the proposed DSNN with the fine-tuned DenseNet is given in **Figure 7B**. It can be seen from **Table 6** and **Figure 7B** that the accuracy of the proposed DSNN results is 13.97% greater than that of the fine-tuned DenseNet.

DenseNet has too many layers and parameters and is prone to meet overfitting problems because our dataset is relatively small. The structure of SNN is simple and has only three layers. What's more, there are fewer parameters in SNN, which is not easy to produce overfitting problems. So, our method achieves better accuracy than fine-tuned DenseNet.

Comparison of Different Backbones

We test the performance of the proposed DSNN with different backbones. These backbones are AlexNet, ResNet-18, ResNet-50, and VGG, respectively. The classification performances of the proposed DSNN with different backbones are shown in **Table 6**. For a clear comparison, the comparison of DSNN with different backbones is presented in **Figure 7C**.
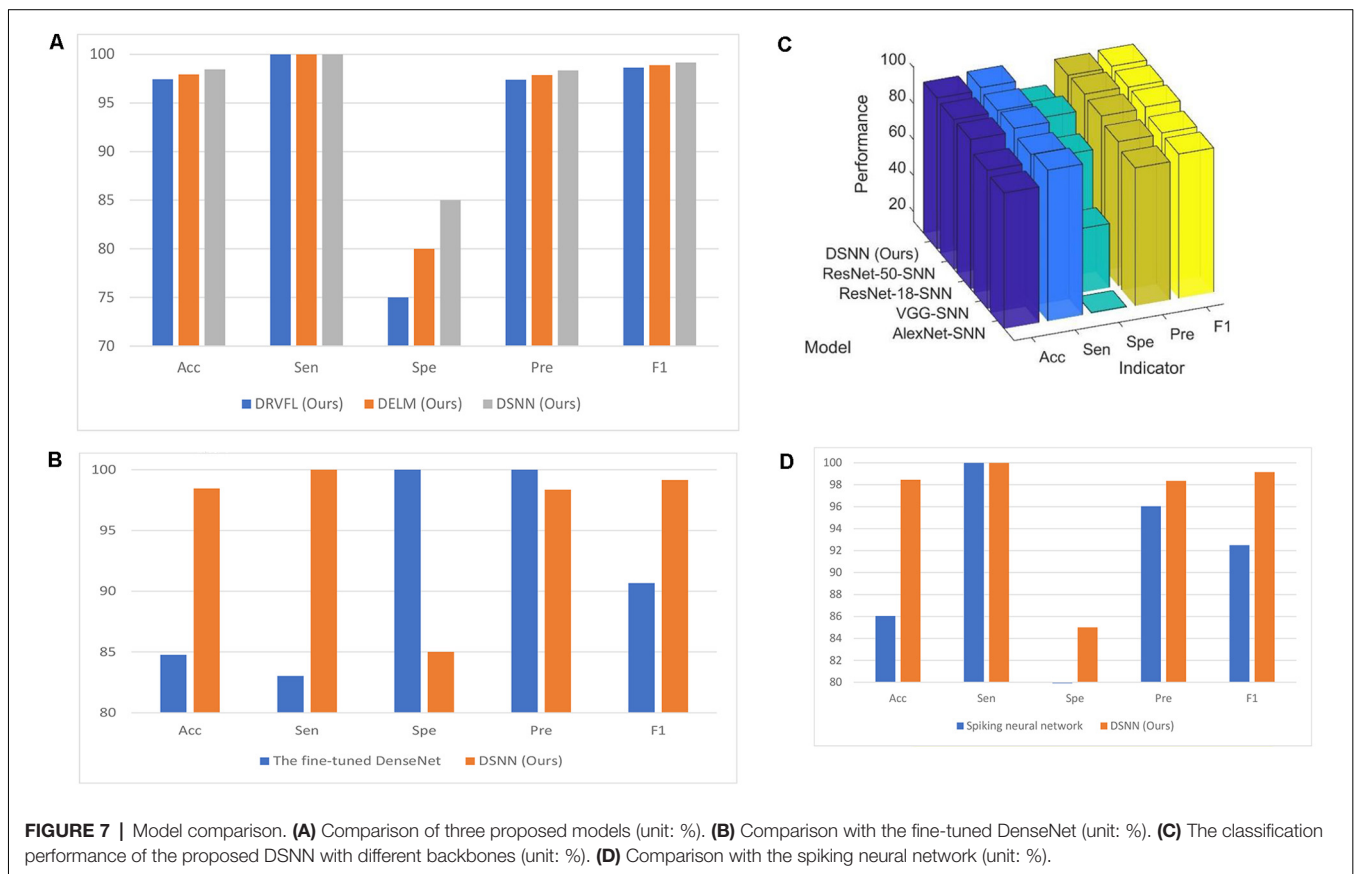
more explicit comparison, the comparison figure of the three proposed models is presented in **Figure 7A**. The proposed DSNN is 1.06% more accurate than the proposed DRVFL and 0.53% more accurate than the proposed DELM. The proposed DSNN

FIGURE 7 | Model comparison. **(A)** Comparison of three proposed models (unit: %). **(B)** Comparison with the fine-tuned DenseNet (unit: %). **(C)** The classification performance of the proposed DSNN with different backbones (unit: %). **(D)** Comparison with the spiking neural network (unit: %).

TABLE 7 | The final result of the spiking neural network (unit: %).

| Model	Acc	Sen	Spe	Pre	F1
Spiking neural network	86.05	100.00	0.00	96.05	92.50
DSNN(Ours)	**98.46**	**100.00**	**85.00**	**98.36**	**99.16**

The bold values are results of our proposed model.
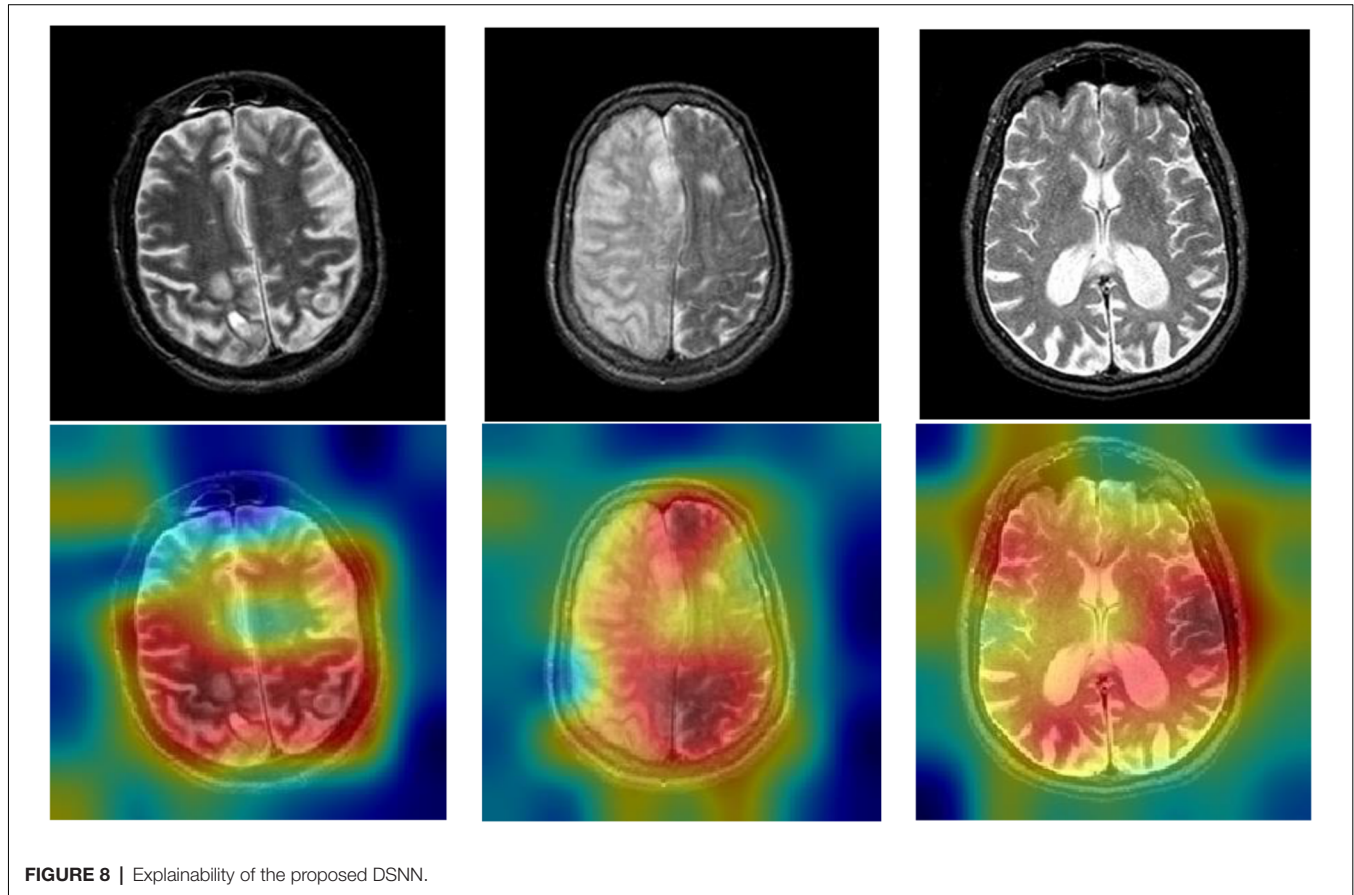
FIGURE 8 | Explainability of the proposed DSNN.

TABLE 8 | Comparison with other state-of-the-art methods (unit: %).

Methods	Sen	Spe	Pre	Acc	F1
ANN (Arunkumar et al., 2020)	89.00	-	-	92.14	-
PR2G (Kalaiselvi et al., 2020)	98.46	-	-	83.90	-
SRH + CNNs (Hollon et al., 2020)	-	-	-	**94.6**	-
BPNN (Hemanth et al., 2011)	57.54	54.50	91.71	57.23	70.72
LVQNN (Nayef et al., 2013)	59.94	61.00	93.08	60.05	72.92
LRC (Chen et al., 2017)	100.00	58.50	95.47	95.74	97.68
DSNN (Ours)	**100.00**	**85.00**	**98.36**	**98.46**	**99.16**

Bold means the best results, - means not available.

DenseNet as the backbone model achieves the best results compared with other backbones. The reason is that DenseNet can reduce gradient vanishing problems better than other CNN models by establishing dense connectivity between all front and rear layers. There are too many parameters in VGG and AlexNet. There are 138M parameters for VGG and 61M parameters for AlexNet. However, there are only 20M parameters in DenseNet. More epoch is needed to converge for VGG and AlexNet. Nevertheless, to prevent overfitting problems, we set the max-epoch to 4. Therefore, DenseNet obtains better performance than VGG and AlexNet. Compared with ResNet, dense connections in the layers can provide more supervision information so that DenseNet can produce better classification performance. DenseNet has also shown its superiority in image
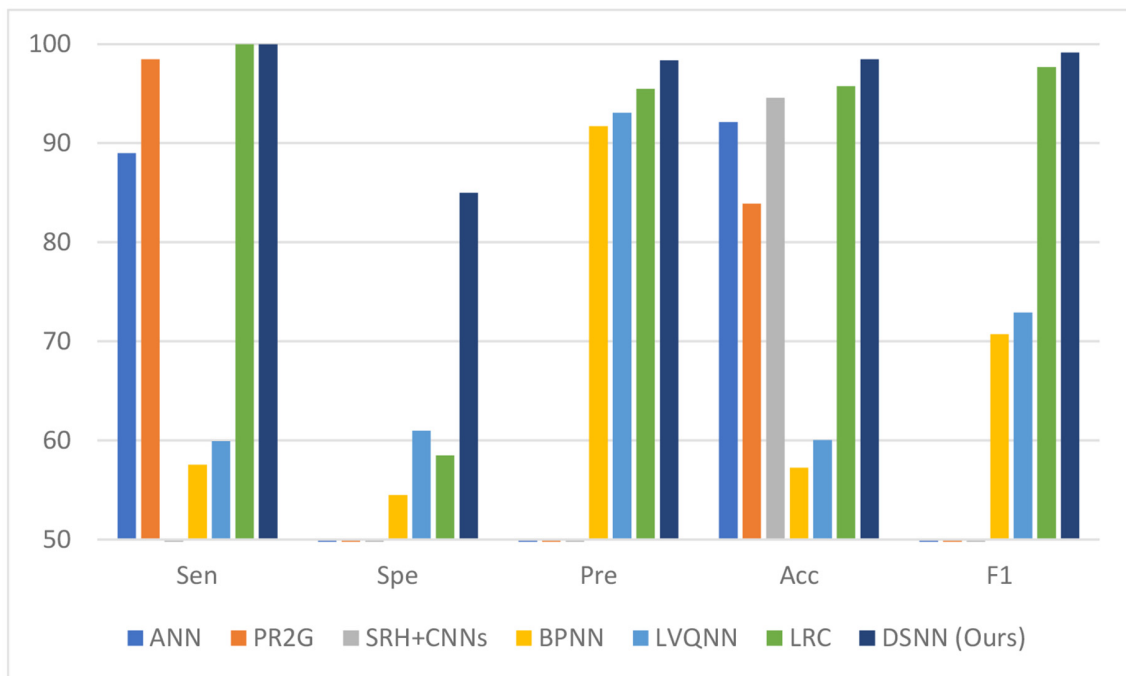
FIGURE 9 | Comparison with other state-of-the-art methods (unit: %).

learning in other studies, such as Ker et al. (2017), Zhang and Patel (2018), and Lundervold and Lundervold (2019).

Comparison With Restricted DenseNet

We limit the number of connections in the DenseNet block. Each layer is only connected to the previous layer in the last block. The results are shown in **Table 6**. Except for the specificity (Spe) value, all other results are not as good as the results of the network we proposed. It is concluded that reducing some dense connections will not improve the classification performance.

Comparison With Spiking Neural Network

We compare the proposed DSNN with the spiking neural network (Yaqoob and Wróbel, 2017). Although the brain inspires spiking and convolutional neural networks, there are still differences. The communication between neurons is completed in the spiking neural network by broadcasting the action sequence (Tavanaei et al., 2019). The final result of the spiking neural network is shown in **Table 7**. The comparison figure is given in **Figure 7D**. In conclusion, the performance of our model is better than the spiking neural network.

Explainability of the Proposed DSNN

It is significant to explain the DCNNs because it is difficult for researchers to figure out how DCNNs make predictions. We can visualize the attention of DCNNs by the Gradient-weighted class activation mapping (Grad-CAM). We present the raw images and heatmap images in **Figure 8**. The brain diseases are within the red region, which is the greatest attention in Grad-CAM.

The blue region is the lowest attention in Grad-CAM. Based on the Grad-CAM, we can conclude that DSNN can classify brain diseases in MRI. Also, some other studies have proven that the Grad-CAM efficiently visualizes the attention of DCNNs, such as Chattopadhay et al. (2018), Woo et al. (2018), Chen et al. (2020), and Panwar et al. (2020).

Comparison With Other State-of-the-Art Methods

We compare the proposed DSNN with other state-of-the-art methods. These state-of-the-art methods are: ANN (Arunkumar et al., 2020), PR2G (Kalaiselvi et al., 2020), SRH + CNNs (Hollon et al., 2020), BPNN (Hemanth et al., 2011), LVQNN (Nayef et al., 2013), and LRC (Chen et al., 2017), respectively. The results are presented in **Table 8**. The comparison chart is given in **Figure 9**. The proposed DSNN gets the best performance among the list methods.

Our model is an effective method to classify brain diseases based on the comparison results. The proposed DSNN can achieve these good results because deep learning is used to extract features, and SNN is used for classification.

CONCLUSION

Three novel models are proposed to automatically classify brain diseases in this article. The proposed models are DSNN, DRVFL, and DELM. The DSNN gets the best performance among the three proposed models in terms of classification performance. The backbone of the proposed DSNN is the pre-trained DenseNet. We modify the pre-trained DenseNet.

Then, the modified DenseNet is fine-tuned on the dataset. The last five layers within the fine-tuned DenseNet are substituted by the Schmidt neural network (SNN). In the proposed DSNN, the fine-tuned DenseNet plays the role of feature extraction. The extracted features train the SNN. We evaluate the proposed DSNN by using five-fold cross-validation. The accuracy, sensitivity, specificity, precision, and F1-score of the proposed DSNN on the test set are $98.46\% \pm 2.05\%$, $100.00\% \pm 0.00\%$, $85.00\% \pm 20.00\%$, $98.36\% \pm 2.17\%$, and $99.16\% \pm 1.11\%$, respectively. The proposed DSNN is compared with other state-of-the-art methods and obtains the best results among the list methods. Our model obtaining the best performance can conclude that DSNN is an effective model for classifying brain diseases.

Although the proposed model gets good results, this article still has some shortcomings. (1) The dataset is relatively small. (2) We divide the datasets into two categories. However, there are many kinds of brain diseases.

We will collect more data to test the proposed model in the future. Then, we will try to classify multiple brain diseases. What's more, we will do more research on brain segmentation. We will try more new deep learning methods, such as VIT, attention learning, etc.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: https://www.med.harvard.edu/aanlib.

REFERENCES

AUTHOR CONTRIBUTIONS

ZZ: conceptualization, software, data curation, writing—original draft, writing—review and editing, visualization. SL: conceptualization, software, data curation, writing—review and editing. S-HW: methodology, software, validation, investigation, resources, writing—review and editing, supervision, and funding acquisition. JG: methodology, validation, formal analysis, resources, writing—original draft, writing—review and editing, supervision. Y-DZ: methodology, formal analysis, investigation, data curation, writing—original draft, writing—review and editing, visualization, supervision, project administration, and funding acquisition.

Aboelenein, N. M., Songhao, P., Koubaa, A., Noor, A., and Afifi, A. (2020). HTTU-net: hybrid two track U-net for automatic brain tumor segmentation. *IEEE Access* 8, 101406–101415. doi: 10.1109/ACCESS.2020. 2998601

Albawi, S., Mohammed, T. A., and Al-Zawi, S. (2017). "Understanding of a convolutional neural network," in *2017 International Conference on Engineering and Technology (ICET)* (Antalya, Turkey). doi: 10.1109/ICEngTechnol.2017.8308186

Amin, J., Sharif, M., Raza, M., Saba, T., Sial, R., and Shad, S. A. (2019a). Brain tumor detection: a long short-term memory (LSTM)-based learning model. *Neural Comput. Appl.* 32, 15965–15973. doi: 10.1007/s00521-019-04650-7

Amin, J., Sharif, M., Gul, N., Raza, M., Anjum, M. A., Nisar, M. W., et al. (2019b). Brain tumor detection by using stacked autoencoders in deep learning. *J. Med. Syst.* 44:32. doi: 10.1007/s10916-019-1483-2

Arunkumar, N., Mohammed, M. A., Mostafa, S. A., Ibrahim, D. A., Rodrigues, J. J., and de Albuquerque, V. H. C. (2020). Fully automatic model-based segmentation and classification approach for MRI brain tumor using artificial neural networks. *Concurrency Comput. Pract. Experience* 32:e4962. doi: 10.1002/cpe.4962

Ashraf, A., Naz, S., Shirazi, S. H., Razzak, I., and Parsad, M. (2021). Deep transfer learning for Alzheimer neurological disorder detection. *Multimed. Tools Appl.* 80, 30117–30142. doi: 10.1007/s11042-020-10331-8

Bhanothu, Y., Kamalakannan, A., and Rajamanickam, G. (2020). "Detection and classification of brain tumor in MRI images using deep convolutional network," in *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)* (Coimbatore, India: IEEE), 248–252. doi: 10.1109/ICACCS48705.2020.9074375

Chatterjee, S., and Das, A. (2019). A novel systematic approach to diagnose brain tumor using integrated type-II fuzzy logic and ANFIS (adaptive neuro-fuzzy inference system) model. *Soft Comput.* 24, 11731–11754. doi: 10.1007/s00500-019-04635-7

Chattopadhay, A., Sarkar, A., Howlader, P., and Balasubramanian, V. N. (2018). "Grad-cam++: generalized gradient-based visual explanations for deep convolutional networks," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)* (Lake Tahoe, NV, USA), 839–847. doi: 10.1109/WACV.2018.00097

Chen, L., Chen, J., Hajimirsadeghi, H., and Mori, G. (2020). "Adapting Grad-CAM for embedding networks," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* (Snowmass, CO, USA), 2794–2803. doi: 10.1109/WACV45572.2020.9093461

Chen, Y., Shao, Y., Yan, J., Yuan, T.-F., Qu, Y., Lee, E., et al. (2017). A feature-free 30-disease pathological brain detection system by linear regression classifier. *CNS Neurol. Disord. Drug Targets* 16, 5–10. doi: 10.2174/1871527314666161124 115531

Górriz, J. M., Ramírez, J., Ortíz, A., Martínez-Murcia, F. J., Segovia, F., Suckling, J., et al. (2020). Artificial intelligence within the interplay between natural and artificial computation: advances in data science, trends and applications. *Neurocomputing* 410, 237–270. doi: 10.1016/j.neucom.2020. 05.078

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Las Vegas, NV, USA), 770–778. doi: 10.1109/CVPR.2016.90

Hemanth, D. J., Vijila, C. K. S., and Anitha, J. (2011). A high speed back propagation neural network for multistage MR brain tumor image segmentation. *Neural Netw. World* 21, 51–56. doi: 10.14311/NNW.2011. 21.004

Hollon, T. C., Pandian, B., Adapa, A. R., Urias, E., Save, A. V., Khalsa, S. S. S., et al. (2020). Near real-time intraoperative brain tumor diagnosis using

stimulated Raman histology and deep neural networks. *Nat. Med.* 26, 52–58. doi: 10.1038/s41591-019-0715-9

Hu, A., and Razmjooy, N. (2020). Brain tumor diagnosis based on metaheuristics and deep learning. *Int. J. Imaging Syst. Technol.* 31, 657–669. doi: 10.1002/ima.22495

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2017). "Densely connected convolutional networks," *Proceedings of the IEEE Conference On Computer Vision And Pattern Recognition (CVPR)* (Honolulu, HI, USA), 4700–4708. doi: 10.1109/CVPR.2017.243

Huang, G.-B., Zhu, Q.-Y., and Siew, C.-K. (2006). Extreme learning machine: theory and applications. *Neurocomputing* 70, 489–501. doi: 10.1016/j.neucom.2005.12.126

Huang, Z., Xu, H., Su, S., Wang, T., Luo, Y., Zhao, X., et al. (2020). A computer-aided diagnosis system for brain magnetic resonance imaging images using a novel differential feature neural network. *Comput. Biol. Med.* 121:103818. doi: 10.1016/j.compbiomed.2020.103818

Johnson, K. A., and Becker, J. A. (2021). Available online at: https://www.med.harvard.edu/aanlib/. Accessed December, 2021.

Kalaiselvi, T., Kumarashankar, P., and Sriramakrishnan, P. (2020). Three-phase automatic brain tumor diagnosis system using patches based updated run length region growing technique. *J. Digit. Imaging* 33, 465–479. doi: 10.1007/s10278-019-00276-2

Kaplan, K., Kaya, Y., Kuncan, M., and Ertunc, H. M. (2020). Brain tumor classification using modified local binary patterns (LBP) feature extraction methods. *Med. Hypotheses* 139:109696. doi: 10.1016/j.mehy.2020.109696

Ker, J., Wang, L., Rao, J., and Lim, T. (2017). Deep learning applications in medical image analysis. *IEEE Access* 6, 9375–9389. doi: 10.1109/ACCESS.2017.2788044

Khalil, H. A., Darwish, S., Ibrahim, Y. M., and Hassan, O. F. (2020). 3D-MRI brain tumor detection model using modified version of level set segmentation based on dragonfly algorithm. *Symmetry* 12:1256. doi: 10.3390/sym12081256

Khan, S. R., Sikandar, M., Almogren, A., Ud Din, I., Guerrieri, A., and Fortino, G. (2020). IoMT-based computational approach for detecting brain tumor. *Future Generation Comput. Sys.* 109, 360–367. doi: 10.1016/j.future.2020.03.054

LeCun, Y. (2015). LeNet-5, convolutional neural networks. Available online at: http://yann.lecun.com/exdb/lenet.

Leming, M., Górriz, J. M., and Suckling, J. (2020). Ensemble deep learning on large, mixed-site fMRI datasets in autism and other tasks. *Int. J. Neural Syst.* 30:2050012. doi: 10.1142/S0129065720500124

Lu, S. Y., Satapathy, S. C., Wang, S. H., and Zhang, Y. D. (2021). PBTNet: a new computer-aided diagnosis system for detecting primary brain tumors. *Front. Cell Dev. Biol.* 9:765654. doi: 10.3389/fcell.2021.765654

Lu, S., Wang, S.-H., and Zhang, Y.-D. (2020a). Detection of abnormal brain in MRI via improved AlexNet and ELM optimized by chaotic bat algorithm. *Neural Comput. Appl.* 33, 10799–10811. doi: 10.1007/s00521-020-05082-4

Lu, S., Wang, S. H., and Zhang, Y. D. (2020b). Detecting pathological brain via ResNet and randomized neural networks. *Heliyon* 6:e05625. doi: 10.1016/j.heliyon.2020.e05625

Lu, S.-Y., Wang, S.-H., and Zhang, Y.-D. (2020). A classification method for brain MRI via mobilenet and feedforward network with random weights. *Pattern Recogn. Lett.* 140, 252–260. doi: 10.1016/j.patrec.2020.10.017

Lu, S., Wu, D., Zhang, Z., and Wang, S.-H. (2021). An explainable framework for diagnosis of COVID-19 Pneumonia via transfer learning and discriminant correlation analysis. *ACM Trans. Multimed. Comput. Commun. Appl.* 17, 1–16. doi: 10.1145/3449785

Lundervold, A. S., and Lundervold, A. (2019). An overview of deep learning in medical imaging focusing on MRI. *Z. Med. Phys.* 29, 102–127. doi: 10.1016/j.zemedi.2018.11.002

Lynch, M. (2018). New Alzheimer's association report reveals sharp increases in Alzheimer's prevalence, deaths, cost of care. *Alzheimer's Dement.* Available online at: www.alz.org/news/2018/new_alzheimer_s_association_report_reveals_sharp_i. Accessed December, 2021.

Ma, L., and Zhang, F. (2021). End-to-end predictive intelligence diagnosis in brain tumor using lightweight neural network. *Appl. Soft Comput.* 111:107666. doi: 10.1016/J.ASOC.2021.107666

Natekar, P., Kori, A., and Krishnamurthi, G. (2020). Demystifying brain tumor segmentation networks: interpretability and uncertainty analysis. *Front. Comput. Neurosci.* 14:6. doi: 10.3389/fncom.2020.00006

Nayef, B. H., Sahran, S., Hussain, R. I., and Abdullah, S. N. H. S. (2013). "Brain imaging classification based on learning vector quantization," *2013 1st International Conference on Communications, Signal Processing and their Applications (ICCSPA)* (Sharjah, United Arab Emirates), 1–6. doi: 10.1109/ICCSPA.2013.6487253

Noreen, N., Palaniappan, S., Qayyum, A., Ahmad, I., Imran, M., and Shoaib, M. (2020). A deep learning model based on concatenation approach for the diagnosis of brain tumor. *IEEE Access* 8, 55135–55144. doi: 10.1109/ACCESS.2020.2978629

Odusami, M., Maskeliūnas, R., and Damaševičius, R. (2022). An intelligent system for early recognition of Alzheimer's disease using neuroimaging. *Sensors (Basel)* 22:740. doi: 10.3390/s22030740

Panwar, H., Gupta, P., Siddiqui, M. K., Morales-Menendez, R., Bhardwaj, P., and Singh, V. (2020). A deep learning and grad-CAM based color visualization approach for fast detection of COVID-19 cases using chest X-ray and CT-Scan images. *Chaos, Solitons Fractals* 140:110190. doi: 10.1016/j.chaos.2020.110190

Pao, Y.-H., Park, G.-H., and Sobajic, D. J. (1994). Learning and generalization characteristics of the random vector functional-link net. *Neurocomputing* 6, 163–180. doi: 10.1016/0925-2312(94)90053-1

Purushottam Gumaste, P., and Bairagi, V. K. (2020). A hybrid method for brain tumor detection using advanced textural feature extraction. *Biomed. Pharmacol. J.* 13, 145–157. doi: 10.13005/bpj/1871

Razzak, I., Naz, S., Ashraf, A., Khalifa, F., Bouadjenek, M. R., and Mumtaz, S. (2022). Mutliresolutional ensemble PartialNet for Alzheimer detection using magnetic resonance imaging data. *Int. J. Intell. Syst.* doi: 10.1002/int.22856

Saba, T., Sameh Mohamed, A., El-Affendi, M., Amin, J., and Sharif, M. (2020). Brain tumor detection using fusion of hand crafted and deep learning features. *Cogn. Syst. Res.* 59, 221–230. doi: 10.1016/j.cogsys.2019.09.007

Sadad, T., Rehman, A., Munir, A., Saba, T., Tariq, U., Ayesha, N., et al. (2021). Brain tumor detection and multi-classification using advanced deep learning techniques. *Microsc. Res. Tech.* 84, 1296–1308. doi: 10.1002/jemt.23688

Sadeghi, D., Shoeibi, A., Ghassemi, N., Moridian, P., Khadem, A., Alizadehsani, R., et al. (2021). An overview on artificial intelligence techniques for diagnosis of schizophrenia based on magnetic resonance imaging modalities: methods, challenges and future works. *arXiv* [Preprint]. doi: 10.48550/arXiv.2103.03081

Schmidt, W. F., Kraaijveld, M. A., and Duin, R. P. (1992). "Feed forward neural networks with random weights," in *11th IAPR International Conference on Pattern Recognition. Vol.II. Conference B: Pattern Recognition Methodology and Systems* (The Hague, Netherlands), 1–4. doi: 10.1109/ICPR.1992.201708

Sharif, M., Amin, J., Raza, M., Anjum, M. A., Afzal, H., and Shad, S. A. (2020). Brain tumor detection based on extreme learning. *Neural Comput. Appl.* 32, 15975–15987. doi: 10.1007/s00521-019-04679-8

Shoeibi, A., Ghassemi, N., Khodatars, M., Moridian, P., Alizadehsani, R., Zare, A., et al. (2022). Detection of epileptic seizures on EEG signals using ANFIS classifier, autoencoders and fuzzy entropies. *Biomed. Signal Process. Control* 73:103417. doi: 10.1016/j.bspc.2021.103417

Shoeibi, A., Khodatars, M., Jafari, M., Moridian, P., Rezaei, M., Alizadehsani, R., et al. (2021a). Applications of deep learning techniques for automated multiple sclerosis detection using magnetic resonance imaging: a review. *arXiv* [Preprint]. doi: 10.48550/arXiv.2105.04881

Shoeibi, A., Ghassemi, N., Khodatars, M., Jafari, M., Moridian, P., Alizadehsani, R., et al. (2021b). Applications of epileptic seizures detection in neuroimaging modalities using deep learning techniques: methods, challenges and future works. *arXiv* [Preprint]. doi: 10.48550/arXiv.2105.14278

Shoeibi, A., Sadeghi, D., Moridian, P., Ghassemi, N., Heras, J., Alizadehsani, R., et al. (2021c). Automatic diagnosis of schizophrenia in EEG signals using CNN-LSTM models. *Front. Neuroinform.* 15:777977. doi: 10.3389/fninf.2021.777977

Shoeibi, A., Khodatars, M., Alizadehsani, R., Ghassemi, N., Jafari, M., Moridian, P., et al. (2020). Automated detection and forecasting of covid-19 using deep learning techniques: a review. *arXiv* [Preprint]. doi: 10.48550/arXiv.2007.10785

Simonyan, K., and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv* [Preprint]. doi: 10.48550/arXiv.1409.1556

Srivastava, R. K., Greff, K., and Schmidhuber, J. (2015). Highway networks. *arXiv* [Preprint]. doi: 10.48550/arXiv.1505.00387

Tavanaei, A., Ghodrati, M., Kheradpisheh, S. R., Masquelier, T., and Maida, A. (2019). Deep learning in spiking neural networks. *Neural Netw.* 111, 47–63. doi: 10.1016/j.neunet.2018.12.002

Wang, S.-H., Govindaraj, V. V., Górriz, J. M., Zhang, X., and Zhang, Y.-D. (2021). Covid-19 classification by FGCNet with deep feature fusion from graph convolutional network and convolutional neural network. *Info. Fusion* 67, 208–229. doi: 10.1016/j.inffus.2020.10.004

Woo, S., Park, J., Lee, J.-Y., and Kweon, I. S. (2018). "CBAM: convolutional block attention module," in *Proceedings of the European Conference On Computer Vision (ECCV)* (Munich, Germany), 3–19. doi: 10.1007/978-3-030-01234-2_1

Xu, L., Gao, Q., and Yousefi, N. (2020). Brain tumor diagnosis based on discrete wavelet transform, gray-level co-occurrence matrix and optimal deep belief network. *Simulation* 96, 867–879. doi: 10.1177/0037549720948595

Yaqoob, M., and Wróbel, B. (2017). "Very small spiking neural networks evolved to recognize a pattern in a continuous input stream," in *2017 IEEE Symposium Series On Computational Intelligence (SSCI)* (Honolulu, HI, USA), 1–8. doi: 10.1109/SSCI.2017.8285420

Zhang, H., and Patel, V. M. (2018). "Densely connected pyramid dehazing network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT, USA), 3194–3203. doi: 10.1109/CVPR.2018.00337

Zhang, Y.-D., Satapathy, S. C., Guttery, D. S., Górriz, J. M., and Wang, S.-H. (2021). Improved breast cancer classification through combining graph convolutional network and convolutional neural network. *Info. Process. Manag.* 58:102439. doi: 10.1016/j.ipm.2020.102439

Zhang, Y.-D., Satapathy, S. C., Zhu, L.-Y., Górriz, J. M., and Wang, S.-H. (2020). A seven-layer convolutional neural network for chest CT based COVID-19 diagnosis using stochastic pooling. *IEEE Sensors J.* 20504–20511. doi: 10.1109/JSEN.2020.3025855