Detection of Alzheimer’s Disease Using Transfer Learning Methods on MRI images

Likitha Santhapalli[#1], Srinikhil Durisetti [*2]

Department of Computer Science

University of Central Florida
Orlando, Florida, United States

[1] li466220@ucf.edu

[2] sr400857@ucf.edu

***Abstract*—Alzheimer’s disease is neurogenerative disorder affected millions of individuals globally. It primarily affects the healthy cells of the brain leading to long term memory loss which is termed as dementia, reduce problem-solving skills, impaired thinking abilities, behavioural inconsistencies and finally death. It is essential to detect and diagnose Alzheimer's disease (AD) accurately at an early stage to ensure timely and effective treatment. This neurodegenerative disease consists of four general stages, which we classified in this research and that includes not demented (No detection), very mild demented (first stage), Mild demented (middle stage), highly demented (last stage). Once the disease gets detected and the pathological load is very high, there is no recovery from that. In this study, we have used MRI images for convolutional neural networks (CNNs) to detect Alzheimer’s disease. Using transfer learning, we have also implemented three pretrained Convolutional Neural Network (CNN) models including VGG16, ReseNet50 and DenseNet-121. Each model is trained on a large dataset of MRI images comprising both healthy and AD-affected. During training, the CNNs automatically learn to recognize patterns and features that distinguish between normal and AD-affected brain regions. To assess the effectiveness of the proposed method, a series of thorough experiments were conducted using publicly available datasets. From the experiments, it is observed that VGG16 approach achieves a good performance compared to other pre-trained models.**

I. Introduction

Alzheimer's disease often called a "devastating neurodegenerative disorder," has affected nearly 50 million people worldwide and is one of the main reasons of dementia, particularly among elder people. It causes irreversible brain damage and impacts thinking and cognitive abilities. The varied symptoms start slowly and early diagnosis can be challenging but important for managing and treating it effectively. The disease is identified by the accumulation of abnormal proteins like beta-amyloid plaques and tau tangles in the brain. This buildup causes a break of the connections between neurons and may lead to loss of them.

Alzheimer's disease can be divided into four stages—Non-Demented, Very Mild Demented, Mild Demented and Severely Demented—based on brain damage and the patient's condition. Analysis using MRI images is the preferred method for diagnosing Alzheimer's disease in clinical research. In recent years, advancements in medical imaging and deep learning have shown the great results in early identification of Alzheimer's disease. Several studies have also used deep learning models like CNNs with MRI scans to improve accuracy. These algorithms automatically analyze and identify subtle indications of the disease, aiding in the early identification of those who might be at risk.

In our study, we are using pre-trained models of CNNs like VGG16, ResNet50 and DenseNet-121 to detect the disease using publicly available Kaggle neuroimaging data. By fine-tuning these models, we aim to find effectiveness of each model.

The next sections of this paper describe: Section II provides an overview of related works in the field of Alzheimer’s disease detection using neuroimaging data and deep learning models. Section III outlines a brief explanation of each proposed model. Section IV presents a detailed description of experimental results. The paper concludes with a summary of findings in Section VII, followed by references.

II. Related work

In this paper [1], Walid and Mohammed worked on pre-trained CNN models with parameter optimization to detect Alzheimer’s disease using neuroimaging data. They have used a transfer learning approach to train the models and conducted various experiments to emphasize the effectiveness of DNNs, especially CNNs in the analysis of medical imaging data.

In [2], Convolutional Neural Networks (CNNs)and Deep Neural Networks (DNNs) have shown promising results in the detection of Alzheimer’s disease (AD). To improve the diagnosis of AD, these algorithms make use of their capacity to extract significant information from medical imaging data. Furthermore, transfer learning has shown to be a useful method that uses pre-trained models to speed up the training of AD-specific models and improve accuracy.

Tissue segmentation and a layer-wise transfer learning method were used in [3] to classify AD. The ADNI database is used in this research. SPM12 was used in the preprocessing step to strip the skull and extract the GM, WM, and CSF. The VGG-19 network is a type of complex pattern recognition system, that was modified by changing its final two decision-making layers. Rather than locking the pre-trained decision layers into place, they have split the model into two parts. They then progressively set certain analysis layers in place so they would remain unchanged. This model was trained using two sets of data: one with augmentation and one without augmentation.

For the augmented dataset, they kept 8 of the analysis layers and 3 of the data-simplification layers static and for the non-augmented dataset, they fixed 12 analysis layers and 4 data-simplification layers. After training with the varied data, the model's ability to correctly identify cases was very high. It accurately identified Alzheimer's disease versus normal conditions 98.73% of the time, early versus later stages of mild cognitive impairment 83.72% of the time, and it had an 80% accuracy rate for other varied conditions.

In this study [4], Hina worked on a computer -aided diagnostic system algorithm that requires real-time diagnosis of the disease. They have suggested categorizing the stages of AD. They have also employed machine learning techniques to create deep feature models and extractions, using classifiers like KNN (K-nearest neighbour), RF (Random Forest), and SVM (Support Vector Machine). To overcome overfitting problems, a large dataset was needed for both classification and the extraction of deep features. In real time, they have proposed how to achieve the maximum accuracy of early Alzheimer's diagnosis by comparing the depth and transmission of learning models with other approaches.

In this study, [5], two deep CNN architectures Inception and VGG16, with the pre-trained and optimised weights of ImageNet data were developed using a transfer learning approach. The last fully linked layer was trained by the researchers with a small set of training MRI scans using a pretrained ImageNet model. MRI images were subjected to image entropy in order to identify the most relevant regions, thereby overcoming the over-fitting of the limited training dataset. An analysis aimed at the binary categorization of AD employed a 416-subject OASIS cross-sectional dataset. For the fully connected layer retraining, five-fold cross-validation was used, with an 80/20 ratio between training and testing. VGG16 was also trained from scratch in order to compare the outcomes.

III. APPROACH

A. Dataset

We have applied the model on Kaggle Dataset consisting of 6400 images in total. The dataset has four categories of files each for Non-Demented, Very Mild Demented, Mild Demented and Moderate Demented.

B. Data Preprocessing

Before training the model, we performed several data preprocessing steps to clean and prepare the dataset. Firstly, we resized the images to ensure uniformity across the dataset. Given the imbalance in the distribution of images across different classes, we employed data augmentation using the ImageDataGenerator to balance the dataset effectively. Additionally, we implemented normalization to scale the pixel values, enhancing the model's ability to converge during training.

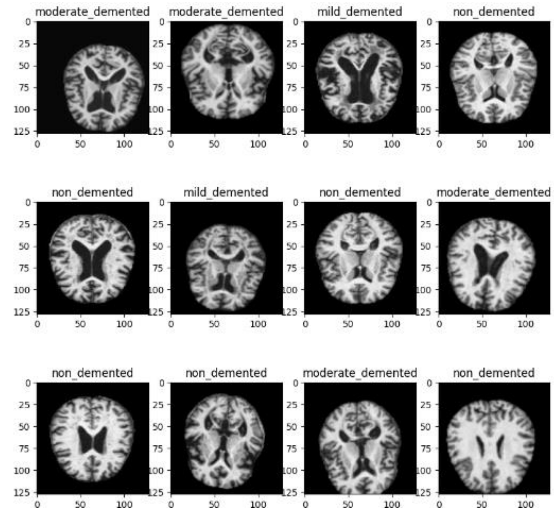


Fig.1. Sample images of the dataset

C. CNN's

ConvNet or Convolutional Neural Networks are commonly used for medical image analysis and have wide application in diagnosis of Alzheimer's disease. These neural network models use a layer of convolutions to identify spatial features of images, pooling layers to reduce the size of the data and fully connected layers that make sense of the detected features for high-level tasks have shown excellent capabilities in classifying AD, segmenting brain regions, and AD-related tasks within medical imaging. The architecture of a CNN typically includes:

1. Convolution layers
2. Pooling layers
3. Fully connected layers
4. A SoftMax layer

In our model, we have used 3 convolution layers each followed by batch normalization and max pooling layers. Following these, there is a flatten layer, which flattens the output for the fully connected layers. Finally, we have three dense layers (dense, dense_1, and dense_2), with the final dense layer being the output layer as shown in Fig 3.

D. Transfer Learning

Pretrained models are neural networks that have already been trained on large datasets. By using these models, transfer learning techniques is used to reduce the training time for each neural network model. When the dataset available is relatively very small to train the model, using pre-trained models works in an effective way.

E. VGG16

VGG16 was developed by the Visual Graphics Group at Oxford. It is known for its simplicity and depth. It is a convolutional neural network has an input layer, output layer and many hidden layers and total of 16 weight layers. These includes 13 convolutional and 3 fully connected layers. The architecture of VGG16 uses a series of convolutional layers followed by max-pooling layers and capped off with fully connected layers. It has a very small convolutional filter of size 3*3 allows it to capture fine details from the images. For our project, we pretrained the model using MRI images and fine-tuned the model to better recognize patters specific to different stage of Alzheimer's disease.

In our VGG16 model, we have VGG16 as a base or functional layer followed by flatten layer and finally dense and dropout layers with each dense layer is followed by dropout layer. These dropout layers help us to reduce the risk of overfitting as shown Fig 6.

F. *ResNet50*

ResNet50 is a Convolutional Neural Network architecture comes from ResNet family. The architecture of ResNet is divided into four main parts: convolutional layers, the identity block, the convolutional block and fully connected layers. The ResNet architecture uses a skip connections or residual connections allowing the network to learn deeper representations. It consists of 50 layers and the main advantage is it has the ability to avoid the vanishing gradient problem allows us the train the models deeply. MRI images consists complex anatomical structures and capture of these complicated details within this image is crucial for accurate diagnosis. ResNet deep architecture helps us to learn wide variety of features at different levels of abstraction. The starting layers might detect simple features like edges and textures, while deeper layers can identify more complex patterns.

G. *DenseNet-121*

DenseNet121 is other form of Dense Convolution Network (DenseNet) architecture. It consists of 121 layers including convolutional and dense layers. It has a unique connectivity pattern where each layer receives inputs from all preceding layers in a feed-forward manner. This creates a very dense connections, which helps in reducing the risk of overfitting on small datasets. This can help in maximum flow of information between layers in the network, improving gradient flow, reducing overfitting and efficiency.

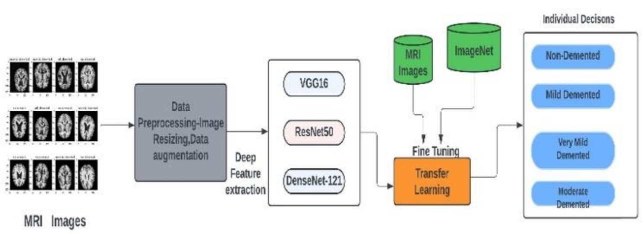


Fig.2. Flow of Framework

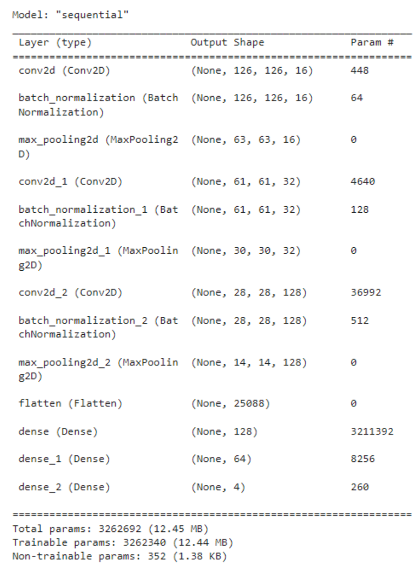
IV. EXPERIMENTS AND RESULTS

In our project, we have used MRI images which are publicly available on Kaggle repositories. This dataset has total of 6400 images including all the four stages. We split the dataset into training, testing and validation dataset as shown in Table 1.

AD stage	Training	Testing	Validation
Non-Demented	2560	320	320
Very Mild Demented	1792	224	224
Mild Demented	716	91	89
Moderate Demented	51	7	6

Tabel.1. Datasets after splitting

As training data is imbalanced, we have balanced the data by generating images using Image Data Generator. The results for all the 4 models are shown from Fig 3 to Fig 18.



```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 126, 126, 16)      448

 batch_normalization (Batch  (None, 126, 126, 16)      64
 Normalization)

 max_pooling2d (MaxPooling2  (None, 63, 63, 16)        0
 D)

 conv2d_1 (Conv2D)           (None, 61, 61, 32)        4640

 batch_normalization_1 (Bat  (None, 61, 61, 32)        128
 chNormalization)

 max_pooling2d_1 (MaxPoolin  (None, 30, 30, 32)        0
 g2D)

 conv2d_2 (Conv2D)           (None, 28, 28, 128)       36992

 batch_normalization_2 (Bat  (None, 28, 28, 128)       512
 chNormalization)

 max_pooling2d_2 (MaxPoolin  (None, 14, 14, 128)       0
 g2D)

 flatten (Flatten)           (None, 25088)             0

 dense (Dense)               (None, 128)               3211392

 dense_1 (Dense)             (None, 64)                8256

 dense_2 (Dense)             (None, 4)                 260

=================================================================
Total params: 3262692 (12.45 MB)
Trainable params: 3262340 (12.44 MB)
Non-trainable params: 352 (1.38 KB)
_________________________________________________________________
```

Fig.3. Summary of CNN model.

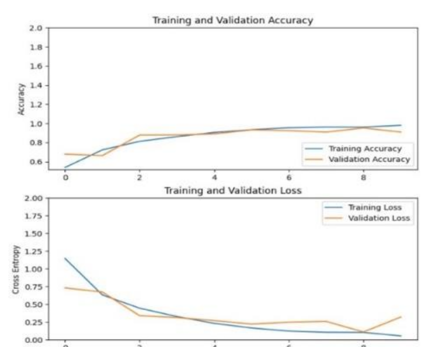


Fig.4. Accuracy and loss for training and validation data for CNN

```
Accuracy of CNN Model: 88.94
Classification report for CNN:
              precision    recall  f1-score   support

         0.0       0.97      0.70      0.82        91
         1.0       0.78      1.00      0.88         7
         2.0       0.93      0.91      0.92       320
         3.0       0.82      0.93      0.87       224

    accuracy                           0.89       642
   macro avg       0.87      0.89      0.87       642
weighted avg       0.90      0.89      0.89       642
```

Fig.5. Classification report for CNN

```
Model: "sequential_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg16 (Functional)          (None, 4, 4, 512)         14714688

 flatten_4 (Flatten)         (None, 8192)              0

 dense_12 (Dense)            (None, 256)               2097408

 dropout_6 (Dropout)         (None, 256)               0

 dense_13 (Dense)            (None, 128)               32896

 dropout_7 (Dropout)         (None, 128)               0

 dense_14 (Dense)            (None, 4)                 516

=================================================================
Total params: 16845508 (64.26 MB)
Trainable params: 9210244 (35.13 MB)
Non-trainable params: 7635264 (29.13 MB)
_________________________________________________________________
```

Fig.6. Summary of VGG16

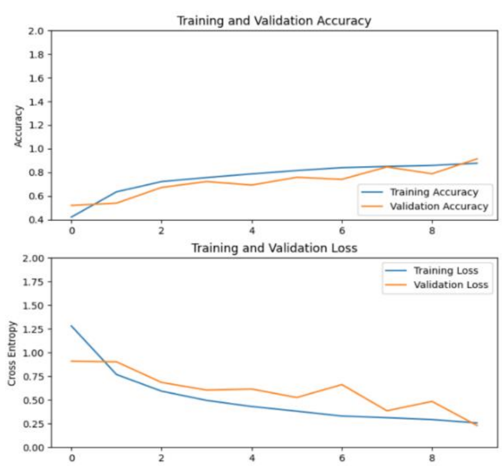


Fig.7. Accuracy and loss of training and validation data for VGG16

```
Accuracy of VGG 16 Model: 90.81
Classification report for VGG 16:
              precision    recall  f1-score   support

         0.0       0.93      0.89      0.91        91
         1.0       1.00      1.00      1.00         7
         2.0       0.94      0.92      0.93       320
         3.0       0.85      0.89      0.87       224

    accuracy                           0.91       642
   macro avg       0.93      0.93      0.93       642
weighted avg       0.91      0.91      0.91       642
```

Fig.8. Classification report for VGG16

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 resnet50 (Functional)       (None, 4, 4, 2048)        23587712

 flatten_2 (Flatten)         (None, 32768)             0

 dense_6 (Dense)             (None, 256)               8388864

 dropout_2 (Dropout)         (None, 256)               0

 dense_7 (Dense)             (None, 128)               32896

 dropout_3 (Dropout)         (None, 128)               0

 dense_8 (Dense)             (None, 4)                 516

=================================================================
Total params: 32009988 (122.11 MB)
Trainable params: 8422276 (32.13 MB)
Non-trainable params: 23587712 (89.98 MB)
_________________________________________________________________
```

Fig.9. Summary of ResNet50

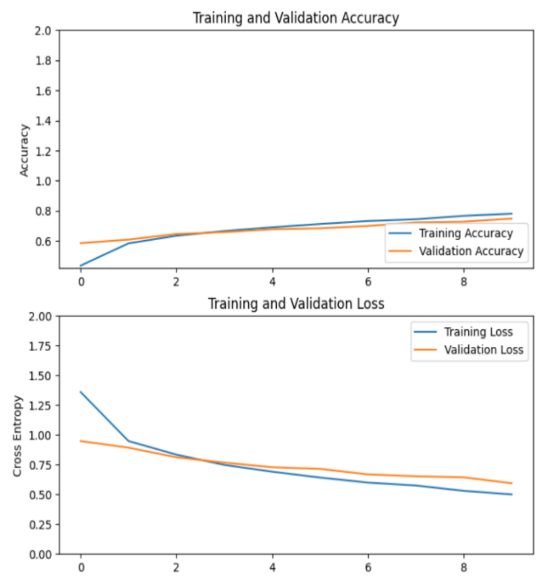


Fig.10. Accuracy and loss of training and validation data for ResNet50

```
Accuracy of ResNet50 Model: 73.05
Classification report for ResNet50:
              precision    recall  f1-score   support

         0.0       0.76      0.46      0.58        91
         1.0       0.88      1.00      0.93         7
         2.0       0.78      0.86      0.82       320
         3.0       0.64      0.65      0.65       224

    accuracy                           0.73       642
   macro avg       0.76      0.74      0.74       642
weighted avg       0.73      0.73      0.72       642
```

Fig.11. Classification report for ResNet50

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 densenet121 (Functional)    (None, 4, 4, 1024)        7037504

 flatten_3 (Flatten)         (None, 16384)             0

 dense_9 (Dense)             (None, 256)               4194560

 dropout_4 (Dropout)         (None, 256)               0

 dense_10 (Dense)            (None, 128)               32896

 dropout_5 (Dropout)         (None, 128)               0

 dense_11 (Dense)            (None, 4)                 516

=================================================================
Total params: 11265476 (42.97 MB)
Trainable params: 4266884 (16.28 MB)
Non-trainable params: 6998592 (26.70 MB)
```

Fig.12. Summary of DenseNet-121

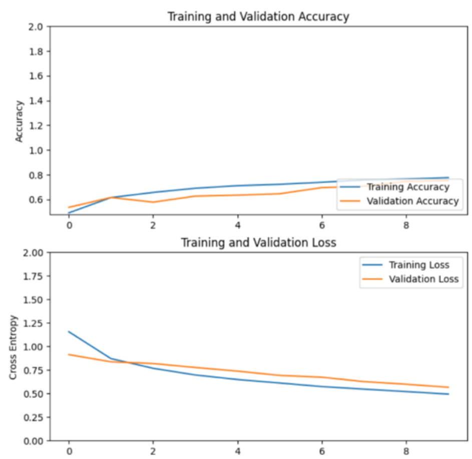


Fig.13. Accuracy and loss of training and validation data for DenseNet121

```
Accuracy of DenseNet121 Model: 75.86
Classification report for DenseNet121:
              precision    recall  f1-score   support

         0.0       0.83      0.69      0.75        91
         1.0       1.00      1.00      1.00         7
         2.0       0.81      0.77      0.79       320
         3.0       0.67      0.76      0.71       224

    accuracy                           0.76       642
   macro avg       0.83      0.81      0.81       642
weighted avg       0.77      0.76      0.76       642
```

Fig.14. Classification report for DenseNet121

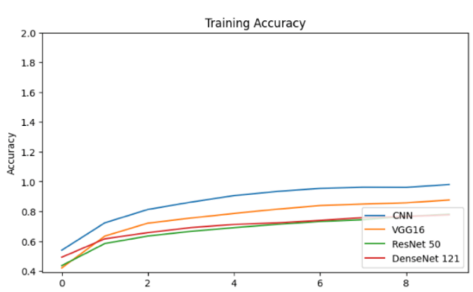


Fig.15. Comparison of Training Accuracy

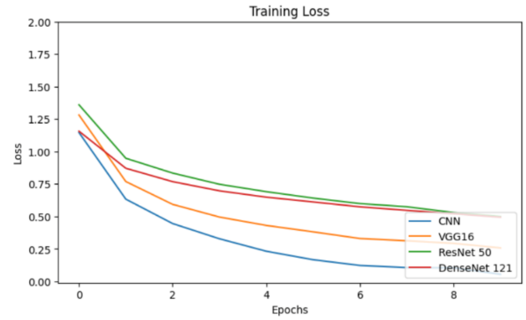


Fig.16. Comparison of Training Loss

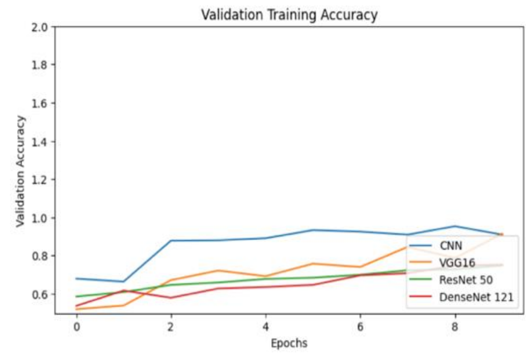


Fig.17. Comparison of Validation Accuracy

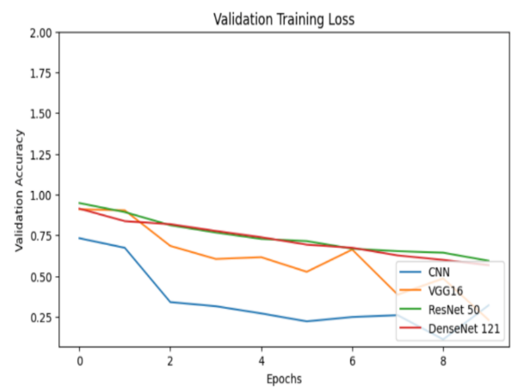


Fig.18. Comparison of Validation loss

From Fig 4, Fig 7, Fig 10, and Fig 13 we can observe the trend of Training accuracy has been improved with epochs and loss has decreased. However, ResNet50 and DenseNet121 training accuracy has increased slightly and got saturated. Due to the computational restrictions of ResNet50 and DenseNet, only a few layers are trained and other layers are frozen with weights as they are trained on ImageNet. Also, if you can Observe Fig 5, Fig 8, Fig 11, and Fig 14, VGG16 has given the highest test accuracy which means it generalized better on the unseen data. Also, CNN has given comparatively better results. We can also observe that VGG16 has shown a balance between all the classes indicating better generalization over unseen data for the precision, recall, and f1-scores as shown in Fig 5, Fig 8, Fig 11, and Fig 14.CNN also performed better and corresponding observations can be seen from the comparison graphs in Fig 15, Fig 16, Fig 17, and Fig 18.

V. CONCLUSION AND FUTURE WORK

In this project we have worked with CNN model and existing pretrained models using transfer learning to detect the Alzheimer's disease from MRI scan images of human brain. We got that our CNN architecture has shown promising results however VGG16 has generalized better and has shown better results on unseen test data as shown from Fig. 14, Fig. 11, Fig. 8 and Fig. 5.

For future work, though other deep learning networks CNN's ResNet50, DenseNet121 are slightly underperforming, they could be fine-tuned better with considering more trainable layers and experimenting with different hyperparameters would enable them to generalize better in real-world large data due to their capability to learn and generalize.

REFERENCES

[1] Boumezaid Mohamed, Walid Hariri Alzheimer Disease Detection using Advanced Transfer Learning Techniques on MRI Images.In Proceedings of 2023 International Conference on Decision Aid Sciences and Applications (DASA) |979-8-3503-4205-5/23/$31.00 ©2023 IEEE.

[2] M. Zaabi, N. Smaoui, W. Hariri, and H. Derbel, "Deep and statistical based methods for alzheimer's disease detection: A survey," Journal of Computing Science and Engineering, vol. 16, no. 1, pp. 113, 2022.

[3] Mehmood, A.; Yang, S.; Feng, Z.;Wang, M.; Ahmad, A.S.; Khan, R.; Maqsood, M.; Yaqub, M. A transfer learning approach for early diagnosis of Alzheimer's disease on MRI images. Neuroscience 2021, 460, 43–52.

[4] H. Nawaz, M. Maqsood, S. Afzal, F. Aadil, I. Mehmood, and S. Rho, "A deep feature-based real-t ime system for Alzheimer disease stage detection," Multimed. Tools Appl., 2020, doi: 10.1007/s11042-020-09087-y.

[5] Hon, M.; Khan, N.M. Towards Alzheimer's disease classification through transfer learning. In Proceedings of the 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Kansas, MO, USA, 13–16 November 2017; pp. 1166–116.

[6] Acharya, U., Fernandes, S., WeiKoh, J., Ciaccio, E., Fabell, M., Tanik, and Yeong, C., 2019. Automated Detection of Alzheimer's Disease Using Brain MRI Images– A Study with Various Feature Extraction Techniques. Journal of Medical Systems, 43(9).

[7] Sarraf, S. and Tofighi, G., 2016. Classification of alzheimer's disease using fmri data and deep learning convolutional neural networks. arXiv preprint arXiv:1603.08631.