*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 4***
***DOI: https://doi.org/10.17762/ijritcc.v11i4.6456***
***Article Received**: 10 February 2023 **Revised**: 18 March 2023 **Accepted**: 26 March 2023*

# Alzheimer’s Disease Diagnosis Using CNN Based Pre-trained Models

**Ragavamsi Davuluri[1], Vyshnavi Mallapragada[2], Uma Maheswara Rao Mamillapalli[3], Manikanta M[4], Sireesha Peeka[5]**
[1]Assistant Professor, Department of CSE
Seshadri Rao Gudlavalleru Engineering College
Gudlavalleru, India
raaga.vamsi@gmail.com
[2]Student, Department of CSE
Seshadri Rao Gudlavalleru Engineering College
Gudlavalleru, India
vyshnavimallapragada2001@gmail.com
[3]Student, Department of CSE
Seshadri Rao Gudlavalleru Engineering College
Gudlavalleru, India
umamahesh1022mamillapalli@gmail.com
[4]Student, Department of CSE
Seshadri Rao Gudlavalleru Engineering College
Gudlavalleru, India
manikanta72888@gmail.com
[5]Student, Department of CSE
Seshadri Rao Gudlavalleru Engineering College
Gudlavalleru, India
sireeshapeeka@gmail.com

**Abstract**— Memory loss and impairment are signs of Alzheimer's disease (AD), which may also cause other issues. It has a significant impact on patients' lives and is incurable, but rapid recognition of Alzheimer's disease can be useful to initiate appropriate therapy to avoid further deterioration to the brain. Previously, Machine Learning methodswere used to detect Alzheimer's disease. In recent times, Deep Learning algorithms have become more popular for pattern recognition. This workconcentrates on the recognition of Alzheimer's disease at a preliminary phase using advanced convolutional neural network models. As the disease advances, they steadily forget everything. It is critical to detect the disease as quickly as possible. The proposed model usespre-trained models that uses magnetic resonance imaging of the brain to determine if a person has very mild, mild, moderate, or non-dementia. The models used for classification are VGG16, VGG19, and ResNet50 architectures and provide performance comparison.

**Keywords**- Alzheimer’s disease, Deep Learning, VGG16, VGG19, ResNet50

## I. INTRODUCTION

The human brain is a magnificent and enthralling organ. It controls and coordinates our respiration, circulation, and pulse rate. Alzheimer's disease is a brain dysfunction that is unrepairable and progressive in nature. Alzheimer's disease, which is the loss of memory, language, decision-making abilities, and other mental abilities. It is a neurological condition that develops with age. It thus implies that it gets worse and improves over time. In the initial phases of the disease, there's commonly a very mild loss of memory or cognitive difficulties. An individual could have trouble recalling their things and where they have put them. In the intermediate phases of the disease, cognitive abilities disintegrate. The individual may now be reluctant to recall individual data such as their own phone number or home address. They could also experience difficulty recalling memories or events from their past. Individuals will necessitate the help of a caretaker to perform their duties as the infection spreads, and caretaking will become more prevalent as the condition worsens. Individuals in the ultimate phases of the disease start losing sense of their surroundings and control over them. At this moment, the individual's mental skills have degraded vastly. Individuals are really no longer willing to speak in lengthy, idiomatic statements but instead in short bits or sentences. Individuals are extremely susceptible to infections at this time. On average, one out of every ten people over the age of 60 is going to be demented, but it might target an earlier age and be diagnosed in some people in their twenties. It is the

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 4***
***DOI: https://doi.org/10.17762/ijritcc.v11i4.6456***
***Article Received**: 10 February 2023 **Revised**: 18 March 2023 **Accepted**: 26 March 2023*

leading cause of dementia in the elderly. Alzheimer's disease causes a decline in the mental skills that are required to make a lot of decisions; sixty to seventy percent of Alzheimer's are cases of dementia. Nevertheless, in Alzheimer's disease, the damage is prevalent, as several neurons halt to function, start losing connections between neurons, and die. Alzheimer's disease interrupts important functions for neurons and their connections, such as information exchange, metabolic activity, and maintenance. Alzheimer's disease commences by demolishing neurons and their networks in memory-related parts of the brain. It also has an impact on the cerebral cortex regions, which are responsible for language, logic, and human behaviour. At last, many of the other components of the brain have been compromised. Alzheimer's disease steadily erodes a person's capacity to survive and act effectively. The disease is potentially life-threatening. The potential danger of this disorder will expand only if the best therapy is initiated. As a result, the elderly are at a greater risk of contracting this disease. Actually, there is no treatment for this disorder at the moment, but early identification can aid in slowing the progression of cognitive impairment. A balanced lifestyle, proper medication, physical exercise, human behaviour, and participating in cognitive activities are used to lower the risk of Alzheimer's disease. These activities are designed to improve brain health and the efficiency of mental functions. Various machine learning algorithms are applied to classify AD. Recently, deep learning methods have been used for pattern classification. With the help of pre-trained models, disease prediction is done. The figure 1 shows the difference between normal and AD brain.

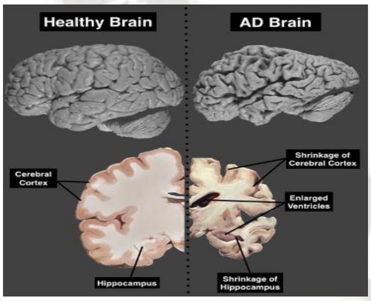


Figure 1. Difference between Healthy brain and Alzheimers brain

## II. LITERATURE SURVEY

Alzheimer's disease is a neurological condition that is defined by a steady deterioration in mental capacity. It accounts for roughly 60–80% of the many kinds of Alzheimer's disease in elderly adults, making it a fairly prevalent condition. Though AD is fairly common, there is presently no treatment for it. From AD's inception to the ultimate diagnosis, a lot of time passes. Mild cognitive impairment (MCI) is a term used to describe adults with initial stage of Alzheimer's disease (AD) [17]. While around 30–40% of MCI patients go on to develop AD, not all MCI patients do. According to research on biomarkers linked to Alzheimer's disease, some areas of the brain have started to shrink. Hence, it is crucial to identify AD as soon as possible. Clinical diagnosis is substantially assisted by digital technologies, which make it easier to categorize clinical data. The most popular methods for identifying neurological disorders are computed tomography (CT) [18], structural magnetic resonance imaging (MRI) [19], and positron emission tomography (PET) [20]. An option to evaluate the mental changes of AD-related intense disorder in vivo is given by magnetic resonance imaging (MRI) [21].

The Machine learning (ML) algorithms are utilized in a number of different tasks, including data mining, image processing, predictive analytics, etc. Even after reviewing the data, we are unable to evaluate or extrapolate the information. We then use machine learning in that situation.ML is used to retrieve significant data. Learning from the data is the goal of machine learning. If the dataset is very large then it is better to go through convolutional neural networks (CNN). Machine Learning algorithms are not much suitable for large datasets and give very less accuracy. It leads to possibility of new error. It is time consuming process.

Neuroimaging data have been constantly employed from over past several decades to explain AD using machine learning (ML) approaches, offering a potential tool for customized identification and interpretation [22]. The inner workings of the patient's body are viewed using MRI scanning. The MRI speed can be increased greatly by using a continuously varying gradient magnetic field, which also improves the resolution of body tissues without harming people from radiation exposure. Feature extraction is a crucial stage in the classification of images. In earlier investigations, the hippocampus, amygdala, and other previously learned places of focus had to be physically extracted to identify AD characteristics. According to Rosen et al. (2002), frontotemporal lobe atrophy helps to identify AD. De Flores, La Joie, and Chetelat (2015) found that hippocampal readings physically monitored also can effectively differentiate between healthy elderly and fairly serious AD patients. These neural images can now be categorized and their characteristics obtained using deep learning methods such as neural networks. [23, 24]. CNN is the most effective deep network used for the identification of disease. On the basis of CNN, appropriate researchers are developing their unique designs. Additionally, so many the fully developed CNN models, including VGGNet and AlexNet, are capable of doing effective AD detection. Several researches recently recommended utilizing CNN techniques to find AD (Aderghal et al., 2018, 2017a, 2017b). Alzheimer's disease is a neurological illness that is defined by a steady deterioration in mental capacity. It is responsible for

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 4***
***DOI: https://doi.org/10.17762/ijritcc.v11i4.6456***
***Article Received**: 10 February 2023 **Revised**: 18 March 2023 **Accepted**: 26 March 2023*

about 60–80% of the many kinds of dementia in elderly adults, making it a fairly prevalent condition. Though AD is fairly common, there is presently no treatment for it.

## III. DATA SET

Dataset is taken from Kaggle website. There are almost 6400 MRI (Magnetic Resonance Imaging) images with 4 classes Non-Dementia, Mild Dementia, Very Mild Dementia and Moderate Dementia. The model uses 80% of the imagees for training and 20% for testing. The datasets should have the same type of distribution so that there is no discrepancy when comparing predictions, such that both models have different types of inputs. Each image is having size of 224 x 224.The figure 2 shows the 4 categories of AD.

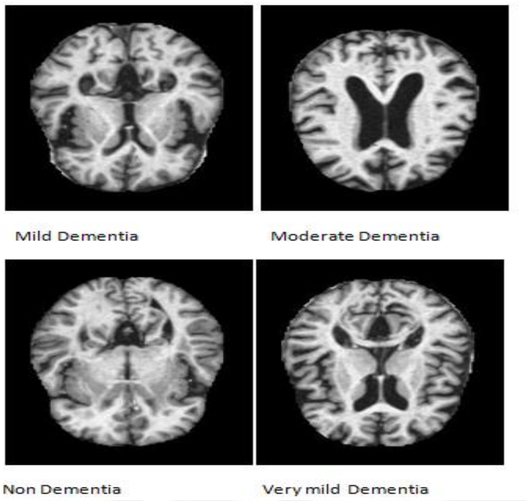


Figure 2. Class labels of brain MRI images

## IV. PROPOSED METHODOLOGY

Deep learning is said to learn a hierarchical set of concepts, so it understands features at three levels: minimum, intermediate, and greater. Neural networks are adaptable to handling increasingly large datasets. Due to its multiple levels, it is suitable for generalizing previously unseen data. For training and testing of the algorithms, various algorithms make use of deep learning's foundational knowledge and different datasets. Similarly, just as humans have sensory neurons, deep learning has layers that assist algorithms in understanding and analysing data. Such layers learn by analysing the information that is offered to them as input and computing the input as they progress through the layers. Lastly, after passing through the final layer, the activation function is used, and the model gets the predicted output. This process improves training accuracy.

Convolutional Neural Networks are a subset of deep learning techniques that are primarily employed for recognition and classification tasks. They are intended to learn spatial feature hierarchies automatically and adaptively from input databases. Overall, CNNs' distinctive architecture has shown to be incredibly efficient for a variety of computer vision applications, and it is still a vital part of many state-of-the-art models in the field. The Proposed model uses transfer learning with VGG16,VGG19 and ResNet50 pretrained models for the early detection of AD. These are the more advanced CNN structures, which are constructed using the basic CNN building blocks.

Transfer learning is an effective method as it saves time and creates accurate models. Transfer learning does not start the learning from very beginning, but few patterns are already discovered on an dataset. Actually, transfer learning is represented using pretrained models. Training such models is therefore computationally expensive, and models imported from the academic research are widely used. The transfer learning model is depicted in figure 3.

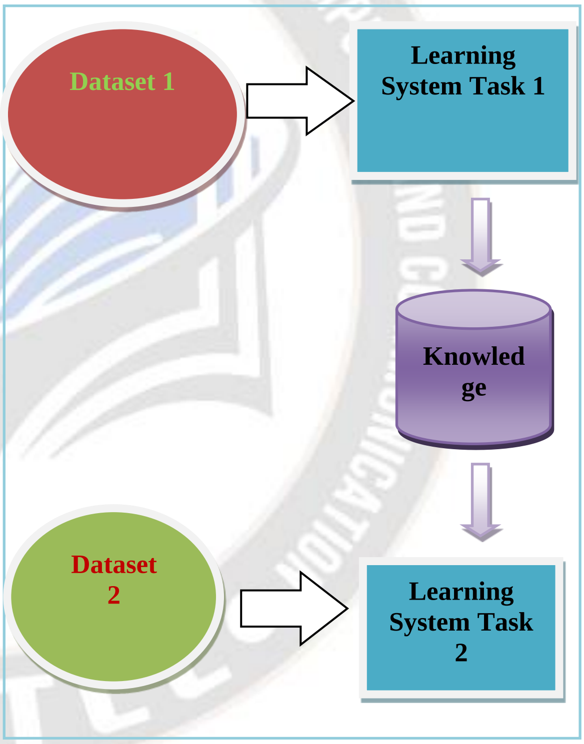


Figure 3. Transfer Learning model

To generate raw data in a format that the model can understand or accept, preprocessing data is a typical preliminary phase in the deep learning approach. To match the size of an image input layer, for instance, you can adjust the input image. Additionally, preprocessing data could be used to improve desired characteristics or minimise abnormalities that could bias the network. For instance, normalise or eliminate noise from the collected data. Different models use smoothening and Rescalingtechniques to preprocess the dataset.

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 4*
*DOI: https://doi.org/10.17762/ijritcc.v11i4.6456*
***Article Received**: 10 February 2023 **Revised**: 18 March 2023 **Accepted**: 26 March 2023*

The Proposed model uses data augmentation inorder to rise the number of images. The procedure of vastly increasing the amount of information by actually producing new points of data from available data is known as "data augmentation”. This involves augmenting data sets by making subtle changes to the information or using deep learning techniques to yield new points of data in the subspace of the original data. It improves the accuracy of the deep learning models. The figure 4 shows few augmented images.

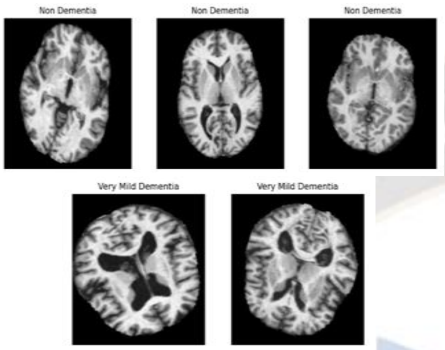


Figure 4. Augmentation of images

## V. PRETRAINED MODELS

A network that has previously undergone training on a huge dataset, generally for a huge image classification problem, is referred to as a pre-trained model. Pre-trained models are advanced convolutional neural networks, which are significantly used for the multiple classification of classes.In comparison to training a model from scratch, pretrained models save time and computing resources and also enhance the model's performance on the new task, particularly when there is an insufficient amount of data

### *A. Diagnosis using VGG16*

K. Simonyan and A. Zisserman developed the deep convolutional neural network model known as VGG16. On Image-Net, a data set of more than 14 hundred thousand images divided into 1,000 classes, this method obtains a top-five accuracy rate of 92.7%. One of the most well-liked models presented to ILSVRC-2014 is this one. NVIDIA Titan Black GPUs were used during the several weeks of training for VGG16. The block of convolutional layers is followed by pooling and three fully connected (FC) layers (with varying depths depending on the architecture) are used. The third conducts 1000 ILSVRC classifications, hence it comprises 1000 channels, while the first two each have 4096 channels (one class each).The soft-max layer is the final one. For all networks, the completely connected layer has the same configuration. In the ILSVRC-2012 and ILSVRC-2013 contests, VGG16 significantly beats the prior generation of models.

The construction of the VGG16 model is depicted in the figure 5. This demonstrates the initial structure of the input and output layers. The filter (3*3) is repeated over the image in between layers of convolution and max pooling, which produces in even finer filters that cover the entire image. In the end, the softmax function gives the likelihood that the image belongs to a specific class. 13 convolutional layers and 3 fully linked layers make up this algorithm. Adam Optimizer is being used to reduce loss and improve accuracy.

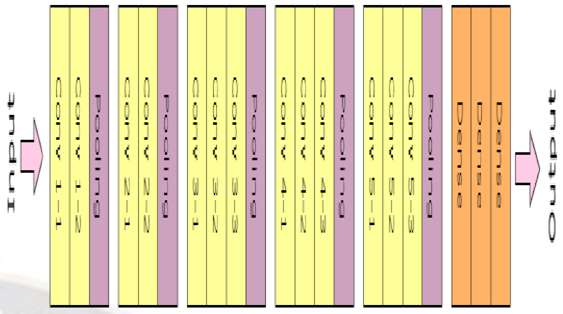


Figure 5. Model of vgg16

### *B. Diagnosis using VGG19*

A deep convolutional neural network with 19 layers is called VGG19. From the Image-Net database, the model has been trained on more than 1,000,000 images. One of the models for convolutional neural networks, with 3 completely connected layers and 16 convolutional layers. In spite of the large number of groups to categorise, it has been demonstrated to yield great results. With 92.7% accuracy, the model was able to categorise a data set with roughly 1000 classifications. The very well-known classification model VGG19, which can categorise diverse classes, has been applied in numerous medical research initiatives. When it comes to forecasting objects like cars and trees, it is extremely accurate.In medical sets of data, the application of VGG19 to predict small groups has increased. Examples include: Breast cancer diagnosis, macular edoema diagnosis, brain tumour diagnosis, etc. It also offers a standardised method for building classifiers. Since it builds the model using only basic convolutional and max pooling layers, it is useful in the majority of research.

The construction of the VGG19 model is depicted in figure 6. This demonstrates the initial structure of the input and output layers. The filter (3*3) is repeated over the image in between layers of convolution and max pooling, which results in even finer filters that cover the entire image. In the end, the softmax function gives the likelihood that the image belongs to a specific class. It has 3 fully linked layers and 16 convolutional layers. Adam Optimizer is being used to reduce loss and improve accuracy.

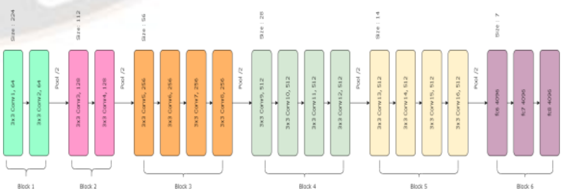

Figure 6. Model of vgg19

### *C. Diagnosis using RESNET50*

Resnet-50 is a well-known Deep Residual Network with 50 layers, which is a subtype of Convolutional Neural Network (CNN). The ResNet comes under artificial neural network that builds a network out of residual blocks that are layered on top of one the other one.

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 4***
***DOI: https://doi.org/10.17762/ijritcc.v11i4.6456***
***Article Received**: 10 February 2023 **Revised**: 18 March 2023 **Accepted**: 26 March 2023*

Kaiming and his collaborators developed ResNet and won the 2015 ImageNet competition. This model demonstrated that any network, regardless of depth, can be trained, and that accuracy increases with network depth. It is made achievable by a decrease in training error rather than overfitting or a large number of parameters. The algorithm's incapacity to backpropagation the gradients is the cause of this. This can be avoided by sending the gradients with a residual block directly to the deeper layers as shown in figure 7.

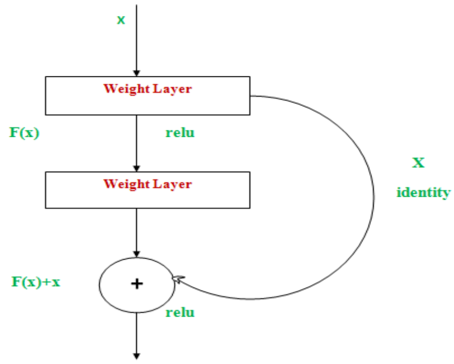


Figure 7. Model ofResNet50

The fact that its ensemble took first place at the ILSVRC 2015 classification competition with a tiny error of just 3.57% indicates that it is a tremendously successful model. More significantly, it won the 2015 ILSVRC & COCO contests for ImageNet detection, ImageNet localization, COCO detection, and COCO segmentation. ResNet has many variations that employ the same basic concept but have varying numbers of layers Resnet-50 is the name given to the form that can operate with 50 neural network layers.

ResNet-50 is a deep convolutional neural network with fifty layers. From the ImageNet database, we can install a pre-trained version of the network that can train on more than 1,000,000 photos. A pre-trained image classification network that has previously mastered extracting potent and useful features from MRI scans can be used as a jumping off point for learning a new task. A portion of the ImageNet database, which is utilized in the ImageNet Large-Scale Visual Recognition Challenge, is used to train the majority of pre-trained networks (ILSVRC) Images may be divided into 1000 different object categories using a pre-trained network. The network can acquire high feature representations for a variety of images as a result. The Resnet 50 blockdiagram is shown in figure 8.

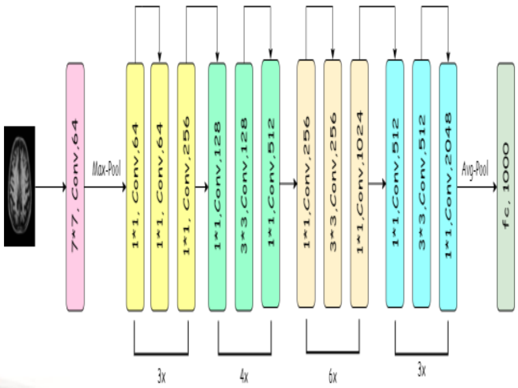


Figure 8. ResNet50 Architecture

## VI. IMPLEMENTATION

This study compares the detection accuracy of deep learning techniques in discovering Alzheimer's disease in magnetic resonance imaging images. Tensorflow and many packages such as Numpy, Pandas, MatplotLib,skimage are imported. Keras is one of the modules of TensorFlow that is used for the VGG16 implementation. The image data generator function was used to collect data, which was then loaded into the VGG16 model. Batch size of 128 is used and an early termination of 10 epochs. Likewise, VGG-19 and Resnet-50 models are implemented by using the Keras module. 4098 MRI images are used for training and 1279 images for testing three models and 1027 images for validation. Adam optimizer is used to reduce the loss. Firstly, all the images are preprocessed and augumentation is done. And then built a model, trained on those models, and evaluated the performance and testing accuracy.

A model with 16 layers and the summary of the model which is built using Keras is given below in figure 9. We have run through 10 epochs and got accuracy of 0.8817, validation accuracy of 0.8147 and model loss which is shown in figure 10.

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 4***
***DOI: https://doi.org/10.17762/ijritcc.v11i4.6456***
***Article Received**: 10 February 2023 **Revised**: 18 March 2023 **Accepted**: 26 March 2023*

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg16 (Functional)          (None, 7, 7, 512)         14714688

 dropout (Dropout)           (None, 7, 7, 512)         0

 flatten (Flatten)           (None, 25088)             0

 batch_normalization (BatchN  (None, 25088)            100352
 ormalization)

 dense (Dense)               (None, 2048)              51382272

 batch_normalization_1 (Batc  (None, 2048)             8192
 hNormalization)

 activation (Activation)     (None, 2048)              0

 dropout_1 (Dropout)         (None, 2048)              0

 dense_1 (Dense)             (None, 1024)              2098176

 batch_normalization_2 (Batc  (None, 1024)             4096
 hNormalization)

 activation_1 (Activation)   (None, 1024)              0

 dropout_2 (Dropout)         (None, 1024)              0

 dense_2 (Dense)             (None, 4)                 4100

=================================================================
Total params: 68,311,876
Trainable params: 53,540,868
Non-trainable params: 14,771,008
_________________________________________________________________
```

Figure 9. Model summary of vgg16

```
Epoch 1/10
33/33 [==============================] - ETA: 0s - loss: 1.2983 - auc: 0.7974
Epoch 1: val_auc improved from -inf to 0.76418, saving model to ./best_weights.hdf5
33/33 [==============================] - 1104s 34s/step - loss: 1.2983 - auc: 0.7974 - val_loss: 1.3682 - val_auc: 0.7642
Epoch 2/10
33/33 [==============================] - ETA: 0s - loss: 1.1348 - auc: 0.8188
Epoch 2: val_auc did not improve from 0.76418
33/33 [==============================] - 64s 2s/step - loss: 1.1348 - auc: 0.8188 - val_loss: 1.2924 - val_auc: 0.6692
Epoch 3/10
33/33 [==============================] - ETA: 0s - loss: 1.0221 - auc: 0.8366
Epoch 3: val_auc improved from 0.76418 to 0.82187, saving model to ./best_weights.hdf5
33/33 [==============================] - 66s 2s/step - loss: 1.0221 - auc: 0.8366 - val_loss: 0.9789 - val_auc: 0.8219
Epoch 4/10
33/33 [==============================] - ETA: 0s - loss: 0.9666 - auc: 0.8475
Epoch 4: val_auc did not improve from 0.82187
33/33 [==============================] - 63s 2s/step - loss: 0.9666 - auc: 0.8475 - val_loss: 1.1313 - val_auc: 0.7401
Epoch 5/10
33/33 [==============================] - ETA: 0s - loss: 0.9172 - auc: 0.8571
Epoch 5: val_auc did not improve from 0.82187
33/33 [==============================] - 62s 2s/step - loss: 0.9172 - auc: 0.8571 - val_loss: 1.0364 - val_auc: 0.7927
Epoch 6/10
33/33 [==============================] - ETA: 0s - loss: 0.8700 - auc: 0.8677
Epoch 6: val_auc did not improve from 0.82187
33/33 [==============================] - 61s 2s/step - loss: 0.8700 - auc: 0.8677 - val_loss: 1.0466 - val_auc: 0.7824
Epoch 7/10
33/33 [==============================] - ETA: 0s - loss: 0.8637 - auc: 0.8683
Epoch 7: val_auc did not improve from 0.82187
33/33 [==============================] - 64s 2s/step - loss: 0.8637 - auc: 0.8683 - val_loss: 0.9998 - val_auc: 0.8099
Epoch 8/10
33/33 [==============================] - ETA: 0s - loss: 0.8370 - auc: 0.8745
Epoch 8: val_auc did not improve from 0.82187
33/33 [==============================] - 61s 2s/step - loss: 0.8370 - auc: 0.8745 - val_loss: 1.0478 - val_auc: 0.7870
Epoch 9/10
33/33 [==============================] - ETA: 0s - loss: 0.7985 - auc: 0.8834
Epoch 9: val_auc did not improve from 0.82187
33/33 [==============================] - 62s 2s/step - loss: 0.7985 - auc: 0.8834 - val_loss: 1.0701 - val_auc: 0.7976
Epoch 10/10
33/33 [==============================] - ETA: 0s - loss: 0.8188 - auc: 0.8817
Epoch 10: val_auc did not improve from 0.82187
33/33 [==============================] - 63s 2s/step - loss: 0.8188 - auc: 0.8817 - val_loss: 1.0020 - val_auc: 0.8147
```

Figure 10. Training iterations of vgg16

A model is built with 19 layers and the summary of the model is given below in figure 11. We have run through 10 epochs and got accuracy of 0.9301, validation accuracy of 0.8709 and model loss which is shown in figure 12.

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg19 (Functional)          (None, 7, 7, 512)         20024384

 dropout_6 (Dropout)         (None, 7, 7, 512)         0

 flatten_2 (Flatten)         (None, 25088)             0

 batch_normalization_6 (Batc  (None, 25088)            100352
 hNormalization)

 dense_6 (Dense)             (None, 4096)              102764544

 batch_normalization_7 (Batc  (None, 4096)             16384
 hNormalization)

 activation_4 (Activation)   (None, 4096)              0

 dropout_7 (Dropout)         (None, 4096)              0

 dense_7 (Dense)             (None, 4096)              16781312

 batch_normalization_8 (Batc  (None, 4096)             16384
 hNormalization)

 activation_5 (Activation)   (None, 4096)              0

 dropout_8 (Dropout)         (None, 4096)              0

 dense_8 (Dense)             (None, 4)                 16388

=================================================================
Total params: 139,719,748
Trainable params: 119,628,804
Non-trainable params: 20,090,944
_________________________________________________________________
```

Figure 11. Model summary of vgg19

```
Epoch 1/10
33/33 [==============================] - ETA: 0s - loss: 0.6391 - auc: 0.9263
Epoch 1: val_auc did not improve from 0.87546
33/33 [==============================] - 58s 2s/step - loss: 0.6391 - auc: 0.9263 - val_loss: 0.9698 - val_auc: 0.8720
Epoch 2/10
33/33 [==============================] - ETA: 0s - loss: 0.6645 - auc: 0.9228
Epoch 2: val_auc did not improve from 0.87546
33/33 [==============================] - 58s 2s/step - loss: 0.6645 - auc: 0.9228 - val_loss: 0.9181 - val_auc: 0.8736
Epoch 3/10
33/33 [==============================] - ETA: 0s - loss: 0.6621 - auc: 0.9230
Epoch 3: val_auc did not improve from 0.87546
33/33 [==============================] - 58s 2s/step - loss: 0.6621 - auc: 0.9230 - val_loss: 1.0940 - val_auc: 0.8475
Epoch 4/10
33/33 [==============================] - ETA: 0s - loss: 0.7081 - auc: 0.9156
Epoch 4: val_auc did not improve from 0.87546
33/33 [==============================] - 58s 2s/step - loss: 0.7081 - auc: 0.9156 - val_loss: 0.9824 - val_auc: 0.8641
Epoch 5/10
33/33 [==============================] - ETA: 0s - loss: 0.6845 - auc: 0.9210
Epoch 5: val_auc did not improve from 0.87546
33/33 [==============================] - 59s 2s/step - loss: 0.6845 - auc: 0.9210 - val_loss: 0.9698 - val_auc: 0.8608
Epoch 6/10
33/33 [==============================] - ETA: 0s - loss: 0.6534 - auc: 0.9239
Epoch 6: val_auc did not improve from 0.87546
33/33 [==============================] - 59s 2s/step - loss: 0.6534 - auc: 0.9239 - val_loss: 0.9418 - val_auc: 0.8617
Epoch 7/10
33/33 [==============================] - ETA: 0s - loss: 0.6666 - auc: 0.9235
Epoch 7: val_auc did not improve from 0.87546
33/33 [==============================] - 59s 2s/step - loss: 0.6666 - auc: 0.9235 - val_loss: 0.9458 - val_auc: 0.8695
Epoch 8/10
33/33 [==============================] - ETA: 0s - loss: 0.6296 - auc: 0.9271
Epoch 8: val_auc did not improve from 0.87546
33/33 [==============================] - 59s 2s/step - loss: 0.6296 - auc: 0.9271 - val_loss: 1.0030 - val_auc: 0.8571
Epoch 9/10
33/33 [==============================] - ETA: 0s - loss: 0.6368 - auc: 0.9281
Epoch 9: val_auc did not improve from 0.87546
33/33 [==============================] - 59s 2s/step - loss: 0.6368 - auc: 0.9281 - val_loss: 0.9637 - val_auc: 0.8729
Epoch 10/10
33/33 [==============================] - ETA: 0s - loss: 0.6286 - auc: 0.9301
Epoch 10: val_auc did not improve from 0.87546
33/33 [==============================] - 59s 2s/step - loss: 0.6286 - auc: 0.9301 - val_loss: 0.9945 - val_auc: 0.8709
```

Figure 12. Training iterations of vgg19

A model is built with 50 layers and the summary of the model is given below in figure 13. We have run through 10 epochs and got accuracy of 0.7885, validation accuracy of 0.7867 and model loss which is shown in figure 14.

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 4***
***DOI: https://doi.org/10.17762/ijritcc.v11i4.6456***
***Article Received**: 10 February 2023 **Revised**: 18 March 2023 **Accepted**: 26 March 2023*

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 resnet50 (Functional)       (None, 7, 7, 2048)        23587712

 dropout_3 (Dropout)         (None, 7, 7, 2048)        0

 flatten_1 (Flatten)         (None, 100352)            0

 batch_normalization_3 (Batc  (None, 100352)           401408
 hNormalization)

 dense_3 (Dense)             (None, 64)                6422592

 batch_normalization_4 (Batc  (None, 64)               256
 hNormalization)

 activation_2 (Activation)   (None, 64)                0

 dropout_4 (Dropout)         (None, 64)                0

 dense_4 (Dense)             (None, 64)                4160

 batch_normalization_5 (Batc  (None, 64)               256
 hNormalization)

 activation_3 (Activation)   (None, 64)                0

 dropout_5 (Dropout)         (None, 64)                0

 dense_5 (Dense)             (None, 64)                4160

 batch_normalization_6 (Batc  (None, 64)               256
 hNormalization)

 activation_4 (Activation)   (None, 64)                0

 dropout_6 (Dropout)         (None, 64)                0

 dense_6 (Dense)             (None, 32)                2080

 batch_normalization_7 (Batc  (None, 32)               128
 hNormalization)

 activation_5 (Activation)   (None, 32)                0

 dropout_7 (Dropout)         (None, 32)                0

 dense_7 (Dense)             (None, 32)                1056

 batch_normalization_8 (Batc  (None, 32)               128
 hNormalization)

 activation_6 (Activation)   (None, 32)                0

 dense_8 (Dense)             (None, 4)                 132
```

Figure 13. Model Summary of ResNet50

```
Epoch 1/10
33/33 [==============================] - ETA: 0s - loss: 1.4224 - auc: 0.6104
Epoch 1: val_auc improved from -inf to 0.63379, saving model to ./best_weights.hdf5
33/33 [==============================] - 928s 28s/step - loss: 1.4224 - auc: 0.6104 - val_loss: 1.3453 - val_auc: 0.6338
Epoch 2/10
33/33 [==============================] - ETA: 0s - loss: 1.2635 - auc: 0.7044
Epoch 2: val_auc improved from 0.63379 to 0.68875, saving model to ./best_weights.hdf5
33/33 [==============================] - 56s 2s/step - loss: 1.2635 - auc: 0.7044 - val_loss: 1.3104 - val_auc: 0.6887
Epoch 3/10
33/33 [==============================] - ETA: 0s - loss: 1.1716 - auc: 0.7511
Epoch 3: val_auc improved from 0.68875 to 0.75988, saving model to ./best_weights.hdf5
33/33 [==============================] - 56s 2s/step - loss: 1.1716 - auc: 0.7511 - val_loss: 1.2389 - val_auc: 0.7599
Epoch 4/10
33/33 [==============================] - ETA: 0s - loss: 1.1342 - auc: 0.7631
Epoch 4: val_auc improved from 0.75988 to 0.77681, saving model to ./best_weights.hdf5
33/33 [==============================] - 55s 2s/step - loss: 1.1342 - auc: 0.7631 - val_loss: 1.1748 - val_auc: 0.7768
Epoch 5/10
33/33 [==============================] - ETA: 0s - loss: 1.1142 - auc: 0.7657
Epoch 5: val_auc did not improve from 0.77681
33/33 [==============================] - 55s 2s/step - loss: 1.1142 - auc: 0.7657 - val_loss: 1.1441 - val_auc: 0.7713
Epoch 6/10
33/33 [==============================] - ETA: 0s - loss: 1.0863 - auc: 0.7717
Epoch 6: val_auc improved from 0.77681 to 0.78714, saving model to ./best_weights.hdf5
33/33 [==============================] - 56s 2s/step - loss: 1.0863 - auc: 0.7717 - val_loss: 1.1103 - val_auc: 0.7871
Epoch 7/10
33/33 [==============================] - ETA: 0s - loss: 1.0804 - auc: 0.7750
Epoch 7: val_auc improved from 0.78714 to 0.78770, saving model to ./best_weights.hdf5
33/33 [==============================] - 54s 2s/step - loss: 1.0804 - auc: 0.7750 - val_loss: 1.0911 - val_auc: 0.7877
Epoch 8/10
33/33 [==============================] - ETA: 0s - loss: 1.0686 - auc: 0.7780
Epoch 8: val_auc did not improve from 0.78770
33/33 [==============================] - 54s 2s/step - loss: 1.0686 - auc: 0.7780 - val_loss: 1.0818 - val_auc: 0.7841
Epoch 9/10
33/33 [==============================] - ETA: 0s - loss: 1.0582 - auc: 0.7801
Epoch 9: val_auc did not improve from 0.78770
33/33 [==============================] - 55s 2s/step - loss: 1.0582 - auc: 0.7801 - val_loss: 1.0726 - val_auc: 0.7843
Epoch 10/10
33/33 [==============================] - ETA: 0s - loss: 1.0407 - auc: 0.7885
Epoch 10: val_auc did not improve from 0.78770
33/33 [==============================] - 55s 2s/step - loss: 1.0407 - auc: 0.7885 - val_loss: 1.0652 - val_auc: 0.7867
```

Figure 14. Some Training iterations of ResNet50

TABLE I. COMPARISION TABLE BETWEEN 3 MODELS

| **Model** | **Type of Images** | **Accuracy** |
|---|---|---|
| VGG16 | MRI | 86% |
| VGG19 | MRI | 87% |
| ResNet50 | MRI | 80% |

## VII. RESULTS

The model requires GPUs to operate with different images, it uses Google Labs as the execution environment and considers data from publicly accessible repositories. The below graphs shows the accuracies and Lossesfor VGG16, 19 and Resnet 50. VGG19 model has produced better accuracy than other models.

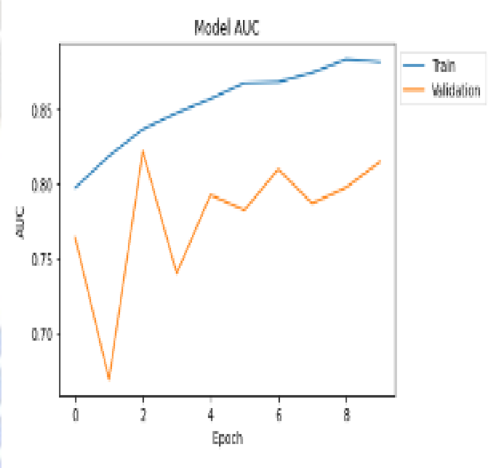


Figure 15. Model accuracy of vgg16

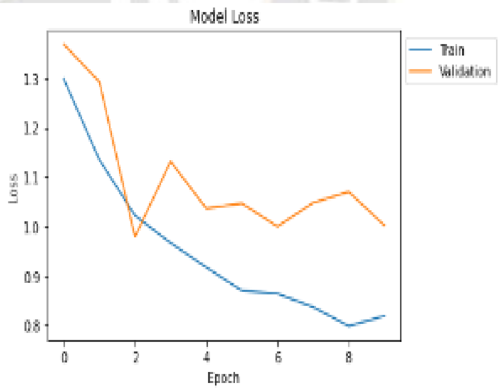


Figure 16. Model loss of vgg16

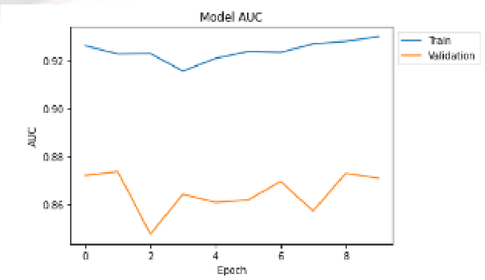


Figure 17. Model accuracy of vgg19

*International Journal on Recent and Innovation Trends in Computing and Communication*
*ISSN: 2321-8169 Volume: 11 Issue: 4*
*DOI: https://doi.org/10.17762/ijritcc.v11i4.6456*
***Article Received**: 10 February 2023 **Revised**: 18 March 2023 **Accepted**: 26 March 2023*

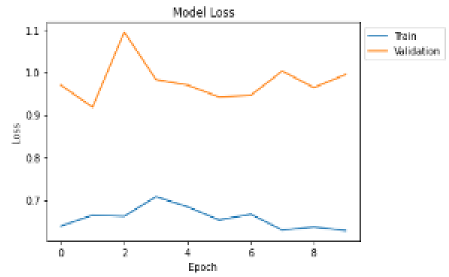


Figure 18. Model loss of vgg19

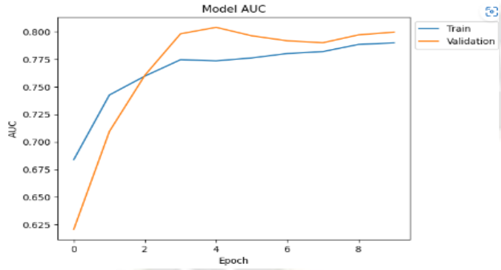


Figure 19. Model accuracy of ResNet50

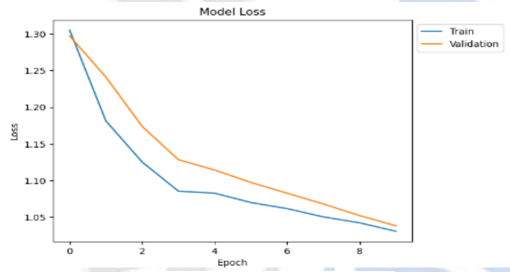


Figure 20. Model loss of ResNet50

## VIII. CONCLUSION AND FUTURESCOPE

Alzheimer's disease is a neurodegenerative brain disease that occurs mostly in older people. This disease damages the main parts of the brain that are responsible for thinking, memory storage, and creativity. Previously, we used Machine Learning algorithms to detect disease, but now we are making use of deep learning techniques. A very basic deep learning method is CNN. Advanced Convolutional Neural Network models are pre-trained models that are used for multiple class classifications. We used the pre-trained models VGG16, VGG19, and ResNet50 for classifying the image dataset. By comparing three models, we discovered that VGG19 has the highest accuracy. On model evaluation, VGG19 got an accuracy of 87%. VGG19 is used in clinical settings to detect Alzheimer's disease because it outperformed VGG16 and ResNet50.The proposed model can be useful for the early diagnosis of Alzheimer's disease by doctors in the future. For better use of this application, this model can be integrated into a website.

## REFERENCES

[1] D. Ragavamsi, and R. Ragupathy, "Neuro–Imaging based Diagnosing System for Alzheimer's Disease using Machine Learning Algorithms," Lecture Notes in Networks and Systems, vol. 385, pp. 501–509, 2022.

[2] D. Ragavamsi, and R. Ragupathy, "Improved Classification Model using CNN for Detection of Alzheimer's Disease," Journal of Computer Science, vol. 18, no 5, pp. 415–425, 2022.

[3] D. Ragavamsi, and R. Ragupathy, "A Pre-Trained Neural network to Predict Alzheimer's Disease at an Early Stage," International Journal of Advanced Computer Science and Applications, vol. 13, no 5, pp. 194–200, 2022.

[4] T. Jo, K. Nho, and A. J. Saykin, "Deep Learning in Alzheimer's Disease: Diagnostic Classification and Prognostic Prediction Using Neuroimaging Data," Frontiers in aging neuroscience, vol. 11, August 2019.

[5] S. Kaur, S. Gupta, S. Singh, and I. Gupta, "Detection of Alzheimer's Disease Using Deep Convolutional Neural Network," International Journal of Image and Graphics, January 2021.

[6] Y. Wang, X. Liu, and C. Yu, "Assisted Diagnosis of Alzheimer's Disease Based on Deep Learning and Multimodal Feature Fusion," Complexity, April 2021.

[7] J. Islam, and Y. Zhang, "Brain MRI analysis for Alzheimer's disease diagnosis using an ensemble system of deep convolutional neural networks," Brain Informatics, May 2018.

[8] R. R. Janghel, "Deep-Learning-Based Classification and Diagnosis of Alzheimer's Disease," Deep Learning and Neural Networks, pp. 1358–1382, 2020.

[9] Y. Zhu, X. Liang, J. Batsis, and R. M. Roth, "Exploring Deep Transfer Learning Techniques for Alzheimer's Dementia Detection," Frontiers in Computer Science, vol. 3, May 2021.

[10] M. S. Syed, Z. S. Syed, E. Pirogova, and M. Lech, "Static vs. Dynamic Modelling of Acoustic Speech Features for Detection of Dementia," International Journal of Advanced Computer Scicence and Applications, vol. 11, October 2020.

[11] A. Mehmood, M. Maqsood, M. Bashir, and Y. Shuyuan, "A deep siamese convolution neural network for multi-class classification of alzheimerdisease," In Brain sciences, 10: 84. DOI: 10.3390/ brainsci10020084.

[12] A. W. Salehi, P. Baglat, B. Sharma, G. Gupta, and A. Upadhya, "A CNN Model: Earlier Diagnosis and Classification of Alzheimer Disease using MRI," International Conference on Smart Electronics and Communication (ICOSEC). IEEE, 156–161, 2020.

[13] C. Feng, A. Elazab, P. Yang, T. Wang, F. Zhou, H. Hu, and B. Lei, "Deep learning framework for Alzheimer's disease diagnosis via 3D-CNN and FSBi-LSTM," IEEE, 7: 63605–63618. DOI: 10.1109/ACCESS.2019.2913847.

[14] S. K. Zhou, H. Greenspan, C. Davatzikos, J. S. Duncan, B. Van Ginneken, A. Madabhushi, and R. M. Summers. "A review of deep learning in medical imaging: Imaging traits, technology trends, case studies with progress highlights, and future promises. Proceedings of the IEEE," vol. 109, pp. 820–838, 2021.

[15] R. Yamashita, M. Nishio, R. K. G.Do and K. Togashi, “Convolutional neural networks: an overview and application in radiology,” Insights into imaging, vol. 9, pp. 611–629. DOI: 10.1007/s13244-018-0639-9.

[16] R. Jain, N. Jain, A. Aggarwal, and D. J. Hemanth, “Convolutional neural network based Alzheimer’s disease classification from magnetic resonance brain images,” Cognitive Systems Research, vol. 57, 2019, pp. 147–159. DOI: 10.1016/j.trci.2019.10.001.

[17] R. C. Petersen , G. E. Smith, S. C. Waring, R. J. Ivnik, E. G. Tangalos, and E Kokmen, “Mild cognitive impairment: clinical characterization and outcome,” National Library of Medicine, vol. 56, 1999, pp. 303–308. DOI: 10.1001/archneur.56.3.303.

[18] K. Kamagata, Z. Andrew, T. Hatano, M. Angelique Di Biase, O. El Samad, S. Saiki, K. Shimoji, K. Kumamaru, K. Kamiya, M. Hori, N. Hattori, S. Aoki, Ch. Pantelis, “Connectome analysis with diffusion MRI in idiopathic Parkinson's disease: Evaluation using multi-shell, multi-tissue, constrained spherical deconvolution,” Pub Med, vol. 17, 2017, pp. :518–529. DOI: 10.1016/j.nicl.2017.11.007.

[19] F. Falahati, E. Westman, A. Simmons, “ Multivariate data analysis and machine learning in Alzheimer's disease with a focus on structural magnetic resonance imaging,” National Library of Medicine, vol. 41, 2019, pp. 685–708. DOI: 10.3233/JAD-131928.

[20] W. E. Klunk, H. Engler, A. Nordberg, Y. Wang, G. Blomqvist, “Imaging brain amyloid in Alzheimer's disease with Pittsburgh Compound-B,” Pub Med, vol. 55, 2004, pp. 306–319. DOI:10.1002/ana.20009.

[21] M. Ewers, R. A. Sperling, W. E. Klunk, Michael W Weiner, H. Hampel, “Neuroimaging markers for the prediction and early diagnosis of Alzheimer's disease dementia,” National Library of Medicine, vol. 34, 2011 pp. 430–442. DOI:10.1016/j.2011.05.005.

[22] F. Falahati, D. Ferreira, J. Sebastian Muehlboeck, M. Eriksdotter, A. Simmons, L. Wahlund, and E. Westman, “Monitoring disease progression in mild cognitive impairment: Associations between atrophy patterns, cognition,” APOE and amyloid. Pub Med Central, vol. 16, pp. 418–428, 2017.

[23] M. Böhle, F. Eitel, M. Weygandt, K. Ritter, and Alzheimer's Disease Neuroimaging Initiative, “Layer-wise relevance propagation for explaining deep neural network decisions in MRI-based Alzheimer's disease classification.” Frontiers in Aging Neuroscience, 11, Article 194, 2019.

[24] L. H. Nguyen, D. A. Drew, M. S. Graham, A. D. Joshi, C. Guo Guo, “Risk of COVID-19 among front-line health-care workers and the general community: a prospective cohort study,” Lancet Public Health, vol. 5, 2020, pp. 475–483. DOI:10.1016/ S2468-2667(20)30164-X.

[25] A. Mallikarjuna Reddy, V. Venkata Krishna, L. Sumalatha, “Efficient Face Recognition by Compact Symmetric Elliptical Texture Matrix (CSETM)”, Jour of Adv Research in Dynamical & Control Systems, Vol. 10, 4-Regular Issue, 2018.

[26] Mallikarjuna Reddy, A., Rupa Kinnera, G., Chandrasekhara Reddy, T., Vishnu Murthy, G., et al., (2019), “Generating cancelable fingerprint template using triangular structures”, Journal of Computational and Theoretical Nanoscience, Volume 16, Numbers 5-6, pp. 1951-1955(5), doi: https://doi.org/10.1166/jctn.2019.7830.

[27] Sudeepthi Govathoti, A Mallikarjuna Reddy, Deepthi Kamidi, G BalaKrishna, Sri Silpa Padmanabhuni and Pradeepini Gera, “Data Augmentation Techniques on Chilly Plants to Classify Healthy and Bacterial Blight Disease Leaves” International Journal of Advanced Computer Science and Applications(IJACSA), 13(6), 2022. http://dx.doi.org/10.14569/IJACSA.2022.0130618.

[28] A. Mallikarjuna Reddy, K. S. Reddy, M. Jayaram, N. Venkata Maha Lakshmi, Rajanikanth Aluvalu, T. R. Mahesh, V. Vinoth Kumar, D. Stalin Alex, "An Efficient Multilevel Thresholding Scheme for Heart Image Segmentation Using a Hybrid Generalized Adversarial Network", Journal of Sensors, vol. 2022, Article ID 4093658, 11 pages, 2022. https://doi.org/10.1155/2022/4093658.

[29] V. NavyaSree, Y. Surarchitha, A. M. Reddy, B. Devi Sree, A. Anuhya and H. Jabeen, "Predicting the Risk Factor of Kidney Disease using Meta Classifiers," *2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon)*, 2022, pp. 1-6, doi: 10.1109/MysuruCon55714.2022.9972392.

[30] B. H. Rao *et al*., "MTESSERACT: An Application for Form Recognition in Courier Services," *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC)*, 2022, pp. 848-853, doi: 10.1109/ICOSEC54921.2022.9952031.

[31] P. S. Silpa *et al*., "Designing of Augmented Breast Cancer Data using Enhanced Firefly Algorithm," *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC)*, 2022, pp. 759-767, doi: 10.1109/ICOSEC54921.2022.9951883.

[32] B. S. Reddy, A. Mallikarjuna Reddy, M. H. D. S. Sradda, T. Mounika, S. Mounika and M. K, "A Comparative Study on Object Detection Using Retinanet," *2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon)*, 2022, pp. 1-6, doi: 10.1109/MysuruCon55714.2022.9972742.