SCHOOL OF ARCHITECTURE, COMPUTING & ENGINEERING

Acquiring Alzheimer's Syndrome Detection through the Application of Machine Learning.

Master's Dissertation

In Partial Fulfilment

of the Requirements for the Degree of

Masters in Computer Science

2022-2023

Student Name	:	Viraj Rajendrakumar Shah
Supervisor	:	Prof. Mahmud Ahmed

Abstract

A degenerative brain ailment that has a severe impact on both individuals and their families, Alzheimer's disease is a major public health concern. For early interventions and customised healthcare, Alzheimer's must be accurately and promptly diagnosed. Utilising neuroimaging and clinical data, machine learning techniques have recently produced intriguing findings in helping with the recognition and forecasting of Alzheimer's disease. The most recent and cutting-edge machine learning techniques used in Alzheimer's disease detection are highlighted in this in-depth assessment. The research findings deliver particulars on how a variety neuroimaging techniques, including magnetic resonance imaging (MRI), positron emission tomography (PET), and functional MRI (fMRI), help in acquiring the course of the disease and figuring guaranteed biomarkers.

A comprehensive review of the machine learning techniques and algorithms used to identify Alzheimer's disease is delivered. It explores cutting-edge deep learning models like convolutional neural networks (CNNs) and recurrent neural networks (RNNs), in addition to well-known supervised learning techniques like support vector machines (SVM), logistic regression, and decision trees. The analysis not only evaluates the strengths, flaws, and comprehending of each model individually but also looks at how ensemble approaches and transfer learning tactics might help the models perform better as a whole.

The greatest method for diagnosing Alzheimer's disease is magnetic resonance imaging (MRI), which is one of the technologies now in use. The early stages of AD make it difficult to notice even the smallest changes in the brain. In this paper, a number of MRI image dataset is utilized to create models powered by deep learning for Alzheimer's identifying purposes. Images of brains in a variety of phases of deceit and absurdity make up the dataset. Convolutional Neural Network (CNN), ResNet50V2, and Visual Geometry Group16 (VGG16) and so on are the deep learning models that can be employed in the study. The customized CNN in which the dataset is iterated gives 98.85% and loss of 0.036. While the accuracy obtained from other pre-defined architectures like ResNet50V2 and VGG16 are 64.17% and 88.25% respectively.

Keywords: Alzheimer’s Disease; Neurodegenerative Disorder; Convolutional Neural Network (CNN); Machine Learning Algorithms; MRI – magnetic resonance imaging; Data Augmentation and Data Preprocessing.

Acknowledgements

I wish to sincerely thank the University of East London and recognise them. They welcomed me into the UEL community and gave me the chance to pursue my Master's in Computer Science. My educational and professional abilities have significantly improved as a result of the insights I got from this programme. I owe a huge debt of gratitude to my outstanding professors for their wisdom, support, and direction throughout my academics. The direction and excellence of my studies have been greatly influenced by their extensive knowledge and insightful comments. I am incredibly appreciative of their commitment and constant encouragement in helping me advance academically.

I would especially like to convey my gratitude to **Professor Mahmud Ahmed**, who served as my project adviser, for his essential assistance with my dissertation. Their kind criticism, insightful advice, and methodical approach dramatically improved the quality of my work. They have been steadfast in providing thorough comments and direction at each level of my project, and I appreciate that because it has helped me refine my work and ensure its successful conclusion.

In addition, I want to thank the entire teachers and staff of the University of East London's Computer Science department. Their mentoring, information sharing, and creation of an outstanding scholarly environment have all been greatly valued. My academic career has been greatly enriched by the variety of resources and possibilities offered by the university. Last but not least, I want to express my sincere gratitude to my family, friends, and other loved ones for their unfailing support, inspiration, and compassion over the course of my master's studies. They have continually given me a source of inspiration and drive because of their unfailing faith in me.

In closing, I want to express my gratitude to everyone who has helped shape my academic career and helped me become the skilled individual I am today. I also like to acknowledge the important contributions made by the researchers whose work and viewpoints had a big impact on this study's findings.

Table of Contents

List of Figures

List of Acronyms

AD = Alzheimer's Disease

MRI = Magnetic Resonance Imaging

PET = Positron Emission Tomography

fMRI = functional MRI

ROC = Receiver Operating Characteristic

CNNs = Convolutional Neural Networks

RNNs = Recurrent Neural Networks

ReLU = Rectified Linear Unit

VGG16 = Visual Geometry Group16

ADNI = Alzheimer's Disease Neuroimaging Initiative

RAM = Random Access Memory

SVM = Support Vector Machine

RMSprop = Root Mean Square Propagation

SGD = Stochastic Gradient Descent

GPU = Graphics Processing Units

API = Application Programming Interface

TFX = TensorFlow Extended

GUI = Graphical User Interface

Chapter 1

Introduction:

Innovative healthcare methods have been made possible by improvements in healthcare imaging and machine learning techniques, particularly for the early identification and classification of complicated respiratory diseases. Alzheimer's disease is a neurological ailment that causes memory loss and cognitive deterioration. It presents difficult problems for both patients and medical professionals. Using brain MRI scans and the development of machine learning algorithms has become a potential method for prompt and accurate diagnosis in recent years.

In-depth machine learning techniques are used in this report to investigate the area of Alzheimer's disease identification utilising brain MRI images. Our goal is to investigate how computational analysis and pattern recognition might improve the precision and effectiveness of diagnosis. It is impossible to overestimate the importance of prompt identification because it provides access for prospective preventative measures and therapies that may slow the spread of the disease.

The basic principles of Alzheimer's disease, the function of brain MRI imaging in its diagnosis, and the complicated operation of several machine learning algorithms used in this context will all be covered in detail in this paper. In order to shed light on the possible effects of this interdisciplinary strategy on patient care and broader healthcare systems, the difficulties, possibilities, and accomplishments will be addressed (Kavitha1, et al., 2022).

Hope so that the expanding body of knowledge that exists at the interface of medical science and technology traverse the complexities of Alzheimer's disease detection utilising brain MRI images and machine learning and can be considered. The insights that will advance the invention of precise, effective; and freely available diagnostic instruments for Alzheimer's disease, that will eventually contribute to improved patient outcomes and a better quality of life, through empirical evaluations, case research, and a comprehensive review of the existing literature shall be provided (Herrera, et al., 2013).

1.1 Background:

Due to its far-reaching effects on elderly populations, AD is one of the most important and challenging concerns in contemporary healthcare. Dementia is a progressive neurological condition that progressively impairs a person's capacity to think, remember, and reason, eventually leaving them unable to care for themselves. Abnormal protein aggregates, including beta-amyloid plaques and tau tangles, build up in the brain and are considered to be one of the pathological hallmarks of Alzheimer's disease. The unique cognitive deficits seen in Alzheimer's disease are the direct outcome of synapse dysfunction and neuronal death brought on by these structural alterations. The increasing incidence of AD as the world's population ages has brought into sharp focus the urgent need for efficient early diagnosis and therapeutic options (Li, et al., 2017).

Alzheimer's disease, characterized by cognitive decline, memory impairment, and behavioural changes, has far-reaching consequences for patients, their loved ones, and the healthcare system as a whole. The United Kingdom, like many other developed nations, must deal with two challenges at once: providing for the complex care needs of people with Alzheimer's disease, and developing effective methods for early detection and treatment of the illness.

An unprecedented chance to tackle Alzheimer's disease presents itself as the growing Alzheimer's problem coincides with developments in machine learning. This highlights the critical need to explore alongside employ the strengths of machine learning algorithms to improve the accuracy and efficiency of Alzheimer's disease detection, which would allow for earlier interventions and superior results for affected people. Exploring machine learning-powered AD detection fits well with the UK's commitment to advancing healthcare solutions as the country deals with the economic and sociological ramifications of an aging population (Langerman, 2019). With the goal of improving Alzheimer's disease detection's precision, efficiency, and accessibility, this highlights the urgent need to investigate and devote resources to machine learning methods within the UK's healthcare infrastructure. Such developments may not only change how AD is diagnosed and treated in the United Kingdom, but also provide useful information for the worldwide fight against the disease (Colin L. Masters, 2015).

1.2 Problem Statement:

Patients and medical organizations face major issues as a result of the landscape of chronic neurological conditions, notably Alzheimer's disease (AD). Because of the cognitive effects of Alzheimer's disease, a timely diagnosis is vital; nevertheless, the mild early symptoms and the limits of current diagnostic methods lead to delayed therapies and impaired results. In response to this issue, a number of research have investigated the use of machine learning strategies to improve AD detection. Researchers are investigating the possibility of using machine learning for the early identification of Alzheimer's disease (AD), using data patterns to develop new diagnostic approaches. The effectiveness of deep learning in using brain MRI scans to identify minor alterations associated to Alzheimer's disease is highlighted here (J, et al., 2022)

Convolutional Neural Networks, or CNNs, are being considered for the diagnosis of Alzheimer's disease using MRI data because of their ability to improve detection accuracy. This investigation into neural networks extends to the identification of ailments in a wider variety of scenarios. Collectively, these research provide light on the revolutionary potential of machine learning to revolutionize Alzheimer's disease diagnosis, close the gap in the availability of early intervention, and redefine the treatment of neurodegenerative diseases. This combination of machine learning with medical imaging provides a potentially fruitful way to improve diagnosis accuracy and reconfigure approaches to the treatment of neurodegenerative diseases, including Alzheimer's disease (AD), which is becoming more prevalent in the general population and has significant repercussions for society (Kadhim, et al., 2023).

Accuracy, sensitivity, specificity, and the area under the Receiver Operating Characteristic (ROC) curve are some of the metrics that are the most significant problems that need to be examined in order to deliver the highest degree of accuracy and specificity of the findings that can be determined from the data being analysed of MRI scans. Other crucial metrics include the area of the curve that symbolizes the ROC. In current circumstances, these happen to be complications that need to be addressed and handled.

1.3 Research Question and Objectives:

The foremost question of this research is that how might machine learning techniques can be used to the early diagnosis of Alzheimer's disease, and what implications may this have for reconsidering how neurodegenerative diseases are diagnosed and treated and how the higher amount of accuracy and specificity can be achieved?

AIM: The main motive behind this research is to design & develop an accurate framework or architecture for the classification of Alzheimer’s Disease. The current investigation is focused on researching the adoption of algorithms based on machine learning, specifically deep learning models and Convolutional Neural Networks, for prompt identification of Alzheimer's disease. The overarching goal is to change the present strategy for treating neurodegenerative diseases by boosting the accuracy and minimising the loss, to achieve f1-score and recall of the images for initial diagnoses while establishing unconventional approaches to healthcare.

Objectives:

Considering the interrogation, some of the intents of this research paper are as below.

- Analyse the degree to which deep learning models can spot Alzheimer's disease through evaluation of brain MRI imaging. The purpose of this research is to test the extent that these models can recognize small changes that may precede the first signs of a disease.

- Investigate the efficacy produced by employing a deep-learning technique on Alzheimer's disease determination and categorization. Monitoring the probability for enhanced diagnostic accuracy and what it means for better outcomes for patients is part of the methodologies.

- Examines the viability of using MRI processing based on Convolutional Neural Networks for the rapid identification of Alzheimer's disease. With such an objective, we hope to establish how successfully we are capable to capture microscopic/delicate but crucial characteristics of the brain in scans that have connection to the earliest stages of the disease.

- Determine the possible applications of machine learning software that recognizes images advances for identifying diseases in a variety of medical environments. This goal delves into the ways this sort of tech could possibly be used to enhance the analytic competencies in a wider context.

1.4 Expected Outcomes:

In this paper the outcomes may illustrate the possible ability of neural network models for identifying minor alterations suggesting of Alzheimer's disease will be discovered via exploration into their practicality, most notably in the examination of brain MRI scans. This may indicate the path for an emergence of dependable testing equipment that are able to identify the disorder with greater accuracy, even during its earliest stages.

Improved accuracy in diagnostics for Alzheimer's disease could emanate from implementing a deep-learning strategy to locate and characterize occurrences. This achievement has the opportunity to further improve the lives of patients along with disease management by causing to an accurate distinction of disease instances and facilitating exclusive the intervention methodologies.

In an attempt to attest to the utility of acquiring comprehensive imaging data on brain attributes, MRI analysis based on Convolutional Neural Networks will soon be put underneath microscope examination for initial Alzheimer's disease identification. Evidently, this could potentially open the way for another generation of leading-edge algorithmic diagnostics that use such traits to consistently recognize those with or who are at risk for the illness.

Disease detection is merely one field whereby machine learning-based imaging recognition technology possesses the capacity to improve diagnostic skills, and this will be explored in a closer look of the technology's expanded application. The discoveries of the current study could throw lights concerning how to better implement machine learning-related technologies into conventional medical procedures for the purpose of strengthening the swiftness and accuracy with which diseases may be determined.

Ultimately, favourable outcomes incorporate improvements to the identification and classification of Alzheimer's disorder by employing machine learning. Opportunities for better evaluations, earlier illness detection, and more targeted alternatives to therapy are all examples of these ramifications. The results of this research may have profound effects for the area of neuronal disease treatment through boosting care for patients while encouraging medications at the very beginning stages of the diseases with more accuracy and specificity.

Summary: The first chapter represents a fundamental introduction to Alzheimer's disease within the United Kingdom's healthcare structure. It kicks off with a substantial investigation of the history of Alzheimer's disease, delivering insight on the worldwide and, in particular, UK-centric incidence of this awful neurological ailment. By providing this background, the chapter emphasizes the critical need for committed investigation and creativity in the diagnosis and treatment of Alzheimer's disease. The issue statement is laid out brilliantly against this background. Alzheimer's disease is a massive healthcare concern owing to its complexity and the lack of a definite solution. Accurate and timely diagnosis is critical since it may significantly impact the course of the illness and the person's quality of life. The chapter emphasizes the importance of this topic, stating that early detection of Alzheimer's disease represents a big unmet need. To address this obstacle, the study described in Chapter 1 possesses a defined target and goal: to use a machine learning technique to diagnose Alzheimer's disease. The aim is incredibly ambitious, nonetheless critical for improving the state of healthcare procedures. The goal is to increase diagnostic accuracy and efficiency through maximizing the power of machine learning, allowing for earlier intervention and better patient care. Finally, Chapter 1 analyses the anticipated outcomes of this study project. It envisions the creation and deployment of a machine learning-based detection system as a game changer in the diagnosis of Alzheimer's disease. This method promises to greatly improve diagnostic accuracy and speed, resulting in improved patient outcomes and perhaps opening the door to more efficient treatment techniques. In essence, Chapter 1 lays the groundwork for subsequent research by contextualizing Alzheimer's disease, framing the problem statement, establishing research objectives, and optimistically projecting favourable results in the race to combat this imposing healthcare challenge globally.

Chapter 2

Literature Review

Convolutional Neural Network:

The Convolutional Neural Network, also known as the deep learning model specifically developed for efficiently processing input that exhibits grid-like patterns, such as imagery. The suggested approach derives insight from the organizational structure of the cortex that controls vision in numerous species of animals, as its goal of proactively accumulating knowledge about spatial patterns all over multiple levels of complexities. Convolutional neural network models (CNNs) comprise of three fundamental aspects, namely convolution, pooling, and fully connected layers. The convolution and pooling layers are primarily concerned with obtaining the extraction of features, whilst the fully connected layer serves as accountable for interpreting these extracted distinctive features into an ultimate output, commonly utilized in tasks which include categorisation.

The convolution layer has a crucial spot within the CNN framework. The system incorporates an arrangement of computations which use convolution, a highly sophisticated variant of linear operation. In the context of digital images, which can be considered two-dimensional structures made up of pixel values, a kernel, serving as a feature extractor, is applied to each point throughout the image. This kernel, consisting of a minimal set of parameters, is used to analyse the image data. The distinctive architecture of Convolutional Neural Networks (CNNs) renders them very effective in the realm of image processing, owing to their ability to detect characteristics at any location within an image.

The efficacy of Convolutional Neural Networks (CNNs) stems from their inherent hierarchical structure. As the outputs of each layer are sent to the succeeding layer, the retrieved characteristics undergo a gradual increase in complexity. The process of optimizing parameters, such as kernels, is often referred to as training. The objective of this procedure is to reduce the discrepancy between the outputs generated by the model and the true labels by using optimization methods such as backpropagation and gradient descent.

Convolutional Neural Networks (CNNs) use their specific architectural design to autonomously acquire knowledge about patterns within data grids, particularly in the context of visual information such as photographs. The aforementioned capability arises from the intricate structure of the system, whereby each layer enhances the comprehension of distinctive attributes, and the process of optimization further enhances the model's proficiency in generating precise predictions (Yamashita, et al., 2018).

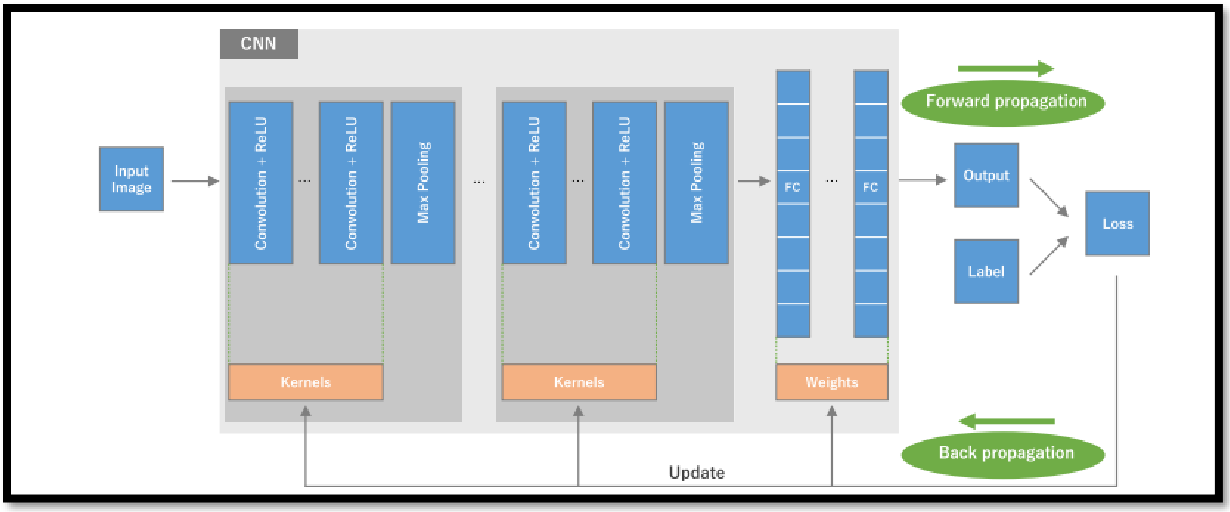


Figure 1 Working of CNN

➔ <u>Layers of CNN:</u>

Convolution Layers -

The convolution procedure improves in the feature extraction process. It analyzes primitive factors like lines and corners in the very first convolution layer. Complex aspects are extracted by the network at higher tiers. The 3D convolution method employed in CNNs can be viewed in the figure below.

The N by N by D input data is convolved using H kernels, each of size k by k by D. Each kernel contains itself and delivers a single discrete output trait. During convolution, the kernel successively examines the input, originating in the top-left corner and concluding in the rightmost part. When it reaches the highest point on the right, it shifting one element down and back to the left. The kernel travels in this manner until it reaches the lowest right corner.

Take the case when N=32 and k=5. The kernel may be placed in one of 28 distinct horizontal and vertical configurations. Thus, the locations the kernel takes up mean that each export feature will have 28 components. Sliding window operations multiply and accumulate element-wise the k x k x D elements of the input by the k x k x D elements of the kernel at each kernel location. A single output feature takes k x k x D multiply-accumulate operations since this accumulation procedure is performed for each kernel point.

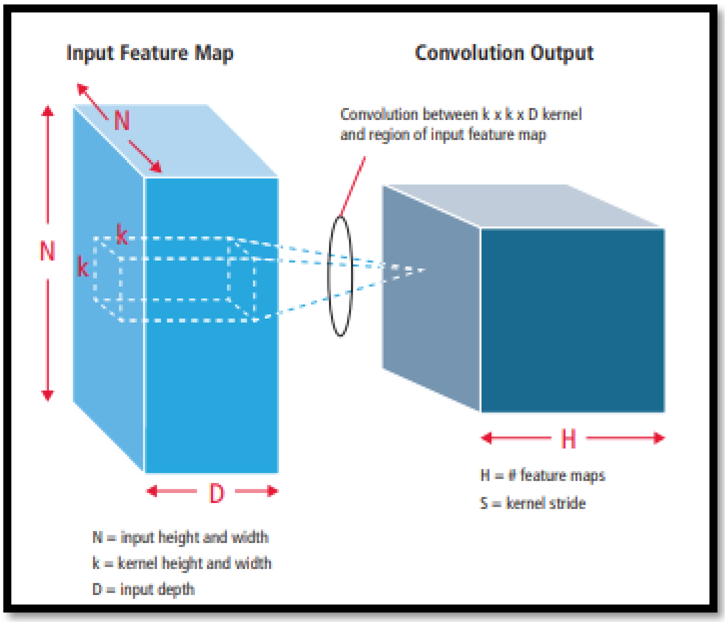


Figure 2 Pictorial representation of convolution process

Pooling Layers -

They are also known as subsampling layers. It helps to coarsen the acquired distinctive features. Features have been rendered more durable versus interference and abnormalities with the assist of this modification. Both optimum and average pooling are often utilized. The data being processed has been split into unique, non-overlapping 2D sections in both scenarios.

The pooling layer, for illustration, is shown as layer 2 in figure below. Each 28x28 input feature is split into 14x14 sub features, each of which is 2x2. With average pooling, an average of the four results for each area can be taken. Contrary to this, max pooling chose the most significant value out of the four as its ultimate result.

In the below figure, the input size is 4 by 4. The 4x4 picture is converted into four separate, non-overlapping 2x2 matrices for subsampling employing 2x2. In max pooling, the output is the value that is the maximum in each 2x2 matrix. Average pooling, on the other hand, determines the output by averaging the four values in each 2x2 matrix.

It is essential to bear in mind that any time averaging delivers a fractional number, as in the case of the final product with index (2,2), its outcome is rounded up to the most adjacent integer.

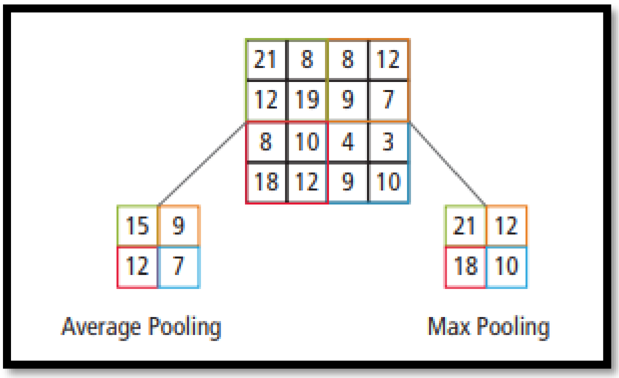


Figure 3 Representation of max pooling and average pooling

Non-Linear Layers -

Non-linear layers are important in neural networks, such as Convolutional Neural Networks (CNNs), given that they incorporate non-linearity into the network's equations. This is vital for accumulating data that has complex interactions and qualities. Here's an inventory of the given text: Non-linear layers are implemented by neural networks, most notably CNNs, to successfully detect complicated relationships in the data they receive. These non-linear "trigger" functions serve as vital for alerting the emergence of distinctive and substantial data at each network's concealed layer. CNNs use a variety of non-linear functions to do this. The use of rectified linear units (ReLUs) is a favoured illustration, as is the use of a continuous trigger (non-linear) function.

ReLU Layers -

In neural networks, the Rectified Linear Unit (ReLU) is a unique kind of non-linear activation function. Where x is an input and y is the output, the formula y = max(x, 0) applies. The size of the input and output are preserved constant by this function, which is unique to ReLU layers.

ReLU activation strengthens the network's non-linear features, which in turn upgrades the network's decision function and its entire capabilities. This succeeds by not sacrificing the responsive fields of the convolutional layers that are accountable for picking out features in the data, which is an essential factor to take into account.

See figure below for an animated illustration of ReLU's behaviour. The transfer function of ReLU is highlighted above the arrow to clarify its functioning. For incorporating non-linearity into the network's computations, this function successfully transforms negative input values to zero and passes positive input values by way of intact.

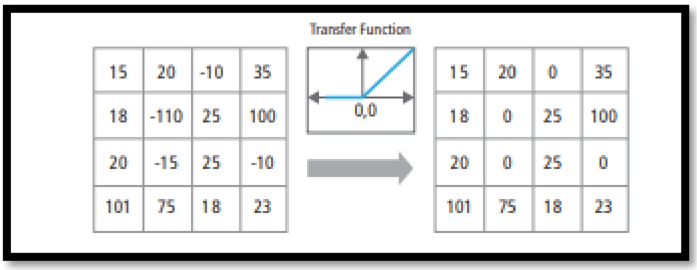


Figure 4 Representation of ReLU functionality

Fully Connected Layers -

The last layers of Convolutional Neural Networks (CNNs) are often fully linked layers, which are also referred to as dense layers. In most CNN concepts the fully linked layers emerge last. These layers conduct a function of mathematics through combining together attributes from the layer below it in a stratified approach. The ultimate sum of these factors is what defines the "mix" of characteristics requisite for attaining the outcome you want. All of the submitted characteristics from the layer beyond get incorporated into the final output features in an entire linked layer. The layers are "fully connected," which means that all output components are impacted by all inputs. This approach emphasizes the principle of interpreting the complex associations and patterns found by earlier steps into a decisive determination or result. The most accurate predictions, classifications, or other desired outputs usually arise by fully linked layers using diverse combinations of data that were learnt earlier in the network.

A fully connected layer is an indispensable component of a convolutional neural network (CNN) since it incorporates the data that comes from lower layers to make more accurate judgments or predictions (Hijazi, et al., 2015)

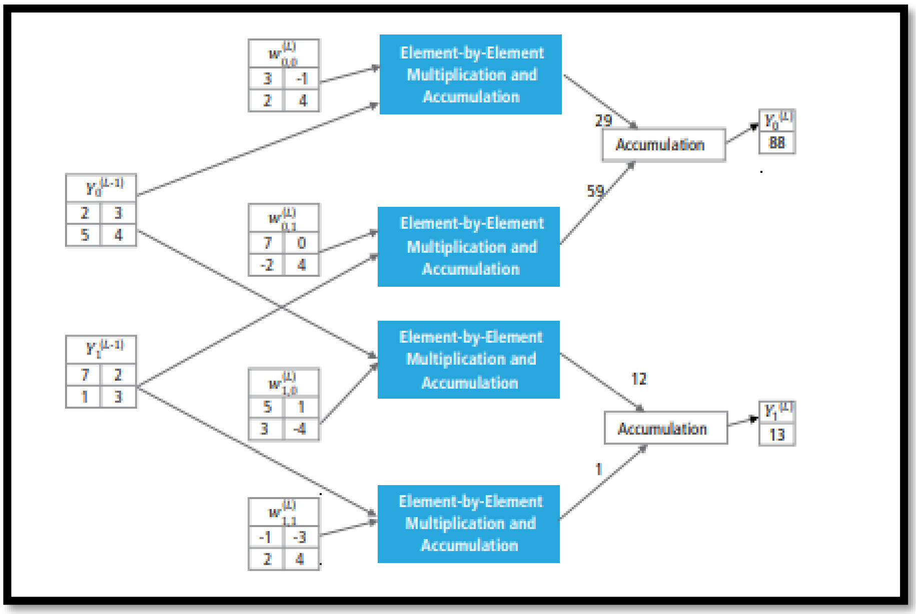


Figure 5 Processing of fully connected layer

Softmax Layer -

For multi-class problems with categorization, deep neural networks (CNNs) frequently utilize a softmax layer as their final concealed layer. The class identification can be rendered quicker via applying the softmax function to a vector of input logits, condensing it into a distribution of likelihood.

In a convolutional neural network (CNN), the softmax layer provides as input a vector of logits, X=[X1,X2,....,Xk], where Xi is a real value signifying a class. Normalized probabilities S = [S1,S2,...,Sk] have been calculated using the softmax function using the following formula:

$$S_i = \frac{e^{x_i}}{\sum_j^{\infty} e^{x_j}}$$

The natural logarithm (Euler's number) adopts e as its foundational value. Total number of classes is denoted by K.

Normalization is achieved by dividing each exponented value by the total amount of exponented value in the softmax function. The above procedure converts the logits to probabilities, with S i signifying the possibility that the input is of class i.

The exponent action ensures that all S Is continues to be positive. Each S i is likely that the data provided will be separated as the ith class, hence that method guarantees that the outcome's statistical distribution will be composed of only positive values. The class estimates from the input logits are productively absorbed by the softmax layer's probability distribution, permitting precision multi-class classification in a wide range of possible instances. Insight into the network's assurance in its predictions for numerous categories based on the information that is entered may be garnered from the probabilities generated via the softmax layer.

In conclusion, multi-class classification in convolutional neural networks leans substantially on the softmax layer and function for translating logits into capable of interpretation probabilities (Gao, et al., 2020).

Why CNN ? -

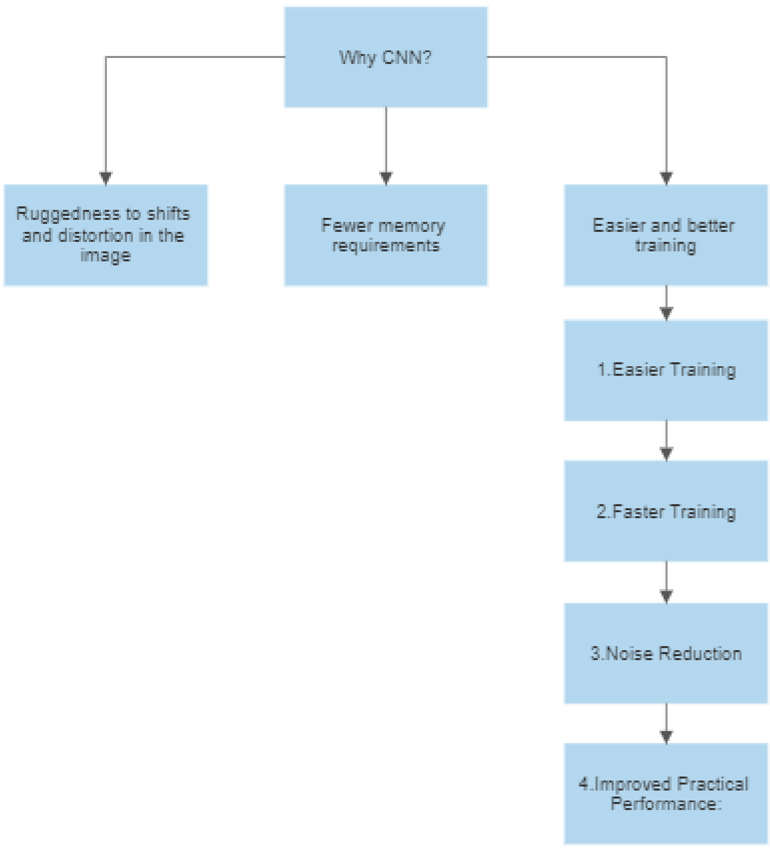


Figure 6 Classification of selecting CNN

1. Ruggedness to shifts and distortion in the Image

Using Convolutional Neural Networks (CNNs) for the identification proved to be tenacious when placed in the face of an assortment of visual distortions and shifts as well as including those induced by photography lens effects, fluctuating enlightenment altering object orientations partially closures, and both upward and downward motion. CNNs, on the contrary conjunction, remain shift consistent since they use the same weight combination no matter where in location they are implemented.

In image classification, shift uniformity serves a purpose due to the fact that allows the network to determine patterns whatever they appear throughout the input image. This can be done by using the identical set of weights across multiple states convolutional techniques applied to multiple segments of the image. That is why CNN algorithms have the ability to recognize aspects even if they've been modified, a bit modified or altered.

While it can be accomplished to validate shift inertia with wholeheartedly linked layers in conceptual terms, executing so in reality represents obstacles. When these networks undergo conditioning using fully linked layers, it is probable that many units throughout the layer at once to demonstrate exactly the same weight patterns albeit being assigned to different positions in the input through the training stage. It would take an immense quantity of training data to effectively memorize these weight combination, since the space of potential permutations of is exceptionally large.

In essence, CNNs have shift invariance embedded in because to weight sharing, but fully connected layers will require a far more expansive dataset for effectively capturing the wide spectrum of spatial modifications needed in order to attain the same degree of shift invariance. This demonstrates the useful advantage provided by CNNs in understanding and broadening features correctly even when imagery have been manipulated.

2. **Fewer memory requirements**

There is a substantial advantage to leveraging convolutional networks over fully connected layers when bandwidth needs of neural network layers are considered into consideration particularly inside the realm of image processing. The fed content is explained below. Let's presume we're in fact extracting features utilising a fully connected layer, assuming that the input picture is 32x32 pixels in size, and thus the hidden layer has 1000 features or neurons. The weights and biases between the input and hidden layers are going to require a large number of coefficients, on the order of 106 (1 million coefficients), in this scenario. Retaining account of all of these settings requires a lot of RAM.

But circumstances transform as we add a convolutional layer. Here, input data from numerous places suffer from the same set of coefficients. This is driven by why convolutional layers redistribute their input weights over many instances of the same kernel or filter. As a consequence, there is an alarming reduction in the number of gigabytes of storage space needed for coefficients. The memory desire dramatically decreases in contrast to using fully linked layers due to the simple reason that the characteristics of the convolutional layer circulate across numerous locations in the image being processed. Convolutional neural networks (CNNs) outperform in this regard when it comes to deciphering immense quantities of picture data.

In conclusion, the use of convolutional layers in neural networks drastically decreases memory requirements, opposed to the greater burdens resulting from fully connected layers, due to the collaborative use of parameters across different geography vacancies.

3. **Easier and better training**

Training neural networks, primarily convolutional neural networks (CNNs), encompasses numerous of advantages that contribute to boosted and more effective training.

I. <u>Easier Training</u>: When evaluating a traditional neural network deemed analogous to a CNN, the sheer amount of factors is an essential consideration. In a traditional neural network comparable to a CNN, the quantity of features would be substantially higher. Due to its growing intricacy, training a network of this type would grow tougher. Due to weight sharing and local connectivity, CNNs have fewer parameters, allowing the procedure for training easier to coordinate and less predisposed to excessive overfitting.

II. Faster Training: When the total amount of characteristics in CNNs shrinks, the training interval drops proportionally. Due to the decrease in parameter space, the procedure for optimizing accelerates and this leads to a quicker approach to the best possible results. This is especially advantageous when juggling enormous data sets and sophisticated architectures.

III. Improved Practical Performance: Theoretically, a traditional neural network ought to be programmed to attain the identical outcome as a CNN, whereas in reality, the variance in parameter count stays relevant. During training, the greater attribute measure of the analogous conventional neural network amplifies its reactivity to noise. As a result, the tangible functionality of a regular neural network can be frequently inadequate to that of a genuine CNN.

IV. Noise Reduction: In the situation for developing neural networks that are comparable to CNNs, a greater variety of parameters rises the potential that the noise will affect the procedure of training. In this regard, noise signifies fluctuations and peculiarities in the data. Minimal CNN parameters attenuate the impacts of noise, empowering the network to accumulate more reliable and adaptable features.

In conclusion, convolutional neural networks (CNNs) receive preference for image-related tasks over ordinary neural networks with identical designs because to their outstanding simplicity during training, more rapid convergence, and decreased reactivity to noise (Hijazi, et al., 2015).

2.1 Comprehensive Overview of Existing Literature:

1. A research paper written by J. Neelaveni and M.S. Geetha Devasana titled "Alzheimer Disease Prediction using Machine Learning Algorithms" is slated to be presented at the 6th International Conference on Advanced Computing & Communication Systems (ICACCS) in 2020, where it will explore potential for earlier identification of Alzheimer's disease using machine learning algorithms. The analysis illustrates the necessity of early identifying attributed to the cumulative history of the ailments and the shortage of curative interventions by targeting significant psychiatric specifications such as age, number of visits, MMSE score, and education. The paper's consideration of "related work" exhibits an in-depth examination of previous initiatives in the area of interest. This entails studies regarding multidisciplinary models utilizing the use of medicinal images, integration of biomarkers and neuroimaging for improved precision, and exploring the possibilities of deep learning algorithms employed with data from clinical trials. In addition, the prospective effects of forecasting techniques on health care and controlling illnesses are addressed and moral challenges for using medical data are brought up. It presents major insights into better the precision of diagnosis and early prevention tactics, and it makes an invaluable contribution to the pursuit of Alzheimer's disease prediction using machine learning (Neelaveni, et al., 2020).

2. In regard to the topic of Alzheimer's disease detection using brain MRI images, a recent paperwork titled "Deep Learning Based Model for Alzheimer's Disease Detection Using Brain MRI Images" worked on by 'Mamun, M., Shawkat, S.B., Ahammed, M.S., Uddin, M.M., Mahmud, M.I., and Islam, A.M.,' and released in the 2022 IEEE 13th Annual Ubiquitous Computing, Electronic Investigations researching the adoption of convolutional neural networks (CNNs) and other deep learning architectures to evaluate brain MRI images may constitute a part of the related work. Existing findings in this area is almost certain to be looked over, underlining the practicality of deep learning models in thoroughly comprehending Alzheimer's disease-related variations and deficiencies in brain imaging information. The following paragraphs could further address the ways in which various initial processing processes, data augmentation draws near, and extracting features tricks are applied in improving the model's accuracy. On top of that, researchers might've taken a glimpse at the difficulties and successes of artificial intelligence-based Alzheimer's disease identification algorithms

when it comes to of comprehension adaptability, and sustainability. In entirety, the data available in the aforementioned field of correlated work likely helps establish greater understanding of the background and current present state of the methods of deep learning employing Alzheimer's disease detection deploying brain MRI scans (Mamun, et al., 2022).

3. The deployment of deep learning methodology has drastically widened the area of preliminary Alzheimer's diagnosis, revealing the efficiency of algorithms developed with machine learning in exact forecasting and surveillance. Kadhima et al.'s thorough study, "Early Diagnosis of Alzheimer's Disease using Convolutional Neural Network-based MRI," delivers an important breakthrough in this particular field. As stated in this paper, the present investigation lends an unusual technique for separating and extracting distinctive characteristics from MRI images extracted from the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset through assessing the effectiveness of convolutional deep neural networks (CNNs). The relevance of swift detection for halting the spread of illness and promoting treatment of patients is one of the study's principal her conclusions. As further proof of the efficacy of their manner, the authors' classification accuracy rates of 97% or superior are attained using their recommended CNN model. This study highlights the ability of deep learning algorithms in analysing sophisticated envision signals for an accurate determination of Alzheimer's disease, therefore would encourage increased research in health IT and enhance the field of medical image analysis as altogether (Kadhim, et al., 2023).

4. The authors of the research study titled "DeepAD: Alzheimer's disease classification via deep convolutional neural networks using MRI and fMRI" by Sarraf et al. provide a novel technique to the diagnosis of Alzheimer's disease. Data obtained from magnetic resonance imaging (MRI) and functional magnetic resonance imaging (fMRI) are fed into deep convolutional neural networks (CNNs), which are then used to categorize Alzheimer's disease. This research attempts to address the urgent need for more precise and earlier diagnostic methods for Alzheimer's disease. The authors investigate the possible applications of several cutting-edge neural network designs in the field of medical picture processing by using the power of deep learning, more especially CNNs. Their work is a contribution to the expanding body of research in the field of Alzheimer's disease diagnosis. It demonstrates the promising prospects of machine

learning techniques in assisting medical personnel in the early identification and management of this neurodegenerative condition, thereby offering hope for improved patient care and outcomes (Sarraf, et al., 2016).

5. The paper titled "A survey on deep learning in medical image analysis" that was written by Litjens et al. (2017) offers a complete overview of the use of deep learning methods in the area of medical image analysis. This article provides an overview of a broad variety of deep learning approaches and their applications in a variety of medical imaging tasks. These activities include, but are not limited to, computer-aided diagnosis, picture segmentation, and disease identification. The authors provide a comprehensive analysis of recent developments in convolutional neural networks (CNNs) and its many offshoots, shedding light on the efficiency with which these networks manage complicated medical imaging data. In addition, the essay addresses the possibilities and problems that are presented by the incorporation of deep learning into clinical practice. In particular, the study emphasizes the possibility for increased diagnosis precision, automated feature extraction, and the creation of robust tools for medical practitioners. This survey is an excellent resource for academics and practitioners who are interested in using the potential of deep learning to improve medical image analysis, which will eventually lead toward more accurate diagnoses and improved patient care (Litjens, et al., 2017).

2.2 Critical Analysis of Existing Studies:

1. The contributors of "Alzheimer Disease Prediction Using Machine Learning Algorithms" by Neelaveni, J. and Devasana, M.G. tried to anticipate Alzheimer's disease using machine learning techniques in the study they published document titled "Alzheimer Disease Prediction Using Machine Learning Algorithms." The researchers utilized the Support Vector Machine (SVM) and Decision Tree algorithms to formulate judgments based on psychological considerations involving age, frequency of visits, MMSE, and education. SVM had an accuracy of 85%, whereas Decision Tree had an accuracy of 83%. The contrast of statistics demonstrated that the SVM executed more effectively. Specifically, the dataset for the study included considerable explanatory aspects such as MMSE score, age, visits, and education. Nevertheless, factors into

account such as collection quantity and relevance should be explored. The study's findings reiterated the possible applications of deep learning in memory loss prediction and outlined the need for ongoing studies deploying larger datasets, include neurological data, for boosting accuracy. This study highlights the positive impact of machine learning in the prompt identification of Alzheimer's disease, therefore assists to better healthcare for patients and an awareness of brain damage.

ALGORITHM	ACCURACY
Support Vector Machine	85%
Decision Tree	83%

Figure 7 Various deep learning models' test outcomes for Alzheimer's disease detection

Technological Limitation:

The author duo of the piece titled "Alzheimer Disease Prediction Using Machine Learning Algorithms" by Neelaveni and Devasana (2020) assessed the possibility of Alzheimer's prediction utilising statistical techniques. The prospective partiality and ability to be generalized of the forecasting algorithms, nonetheless constitutes an unfortunate aspect of the work they do. The exploitation of behavioural traits and data on demographics to foresee ailments formation can disregard inherent nuances in the course of disease. A scientific one disadvantage is the dearth of thorough investigation regarding the choices of neural network algorithms implemented in anticipating, which could jeopardize the system's efficacy in recording not obvious motifs. While their research proposes a glimpse on the potential of machine learning for Alzheimer's prediction, overcoming these barriers may boost predictor model accuracy and endurance (Neelaveni, et al., 2020).

Limitation	Explanation
Limited Focus on Psychological Parameters	The researchers may have ignored a few significant factors underlying Alzheimer's prediction by only considering behavioural specifications include age, number of visits, MMSE score, and education rather than by employing supplementary physiological signs or specialized neuroscience methods.
Potential Lack of Comprehensive Data	The probable insufficient adequate information portraying the complicated processes of Alzheimer's disease evolution is a technological advancements constraint. Recommendations derived from machine learning algorithms might be more reliable and relevant for a variety of contexts once periodic information has been included and any inaccuracies are rectified.

Figure 8 Limitation & it's detail of the Related Work

2. The authors 'Mamun, M., Shawkat, S.B., Ahammed, M.S., Uddin, M.M., Mahmud, M.I. and Islam, A.M.' study's entitled "Deep Learning Based Model for Alzheimer's Disease Detection Using Brain MRI Images" reveals a major breakthrough to anticipate Alzheimer's disease assessment applying sophisticated machine learning techniques. The research study systematically examines deep architectures like CNN, ResNet101, DenseNet121, and VGG16 utilising a dataset of brain MRI images from Kaggle. The recommended tailored CNN model performed well, with accuracy, recall, and AUC all above 93%. Critique, however, opens up factors worth taking careful regard.

Model	Accuracy	AUC	Recall	Loss
1. CNN	97.60%	99.26%	97%	0.091
2. ResNet101	73.85%	83.00%	73.80%	0.556
3. DenseNet121	72.00%	77.63%	71.60%	0.640
4. VGG16	70.20%	77.10%	70.00%	0.583

Figure 9 Results of many tests using various deep learning models to identify Alzheimer's illnesses

Technological Limitation:

The authors of the paper "Deep Learning Based Model for Alzheimer's Disease Detection Using Brain MRI Images" by Mamun et al. (2022) recommend a deep-learning framework to diagnose Alzheimer's disease considering brain MRI inspections. Despite the approach they employ shows covenant, it is possibly handicapped by the deep learning models' lack of comprehensibility which inhibits one from comprehending exactly how an algorithm reaches at its projections. Training deep learning architectures can demand a lot of time as well as finances, especially if you're confronted with immense medical imaging datasets, thus might represent an analytical constraint. The recommendation for a method for Alzheimer's disease determination leveraging brain MRI scans includes multiple limitations the fact that must be taken cautiously whenever assessment conclusions and applying the strategy into execution (Mamun, et al., 2022).

Limitations
Interpretability Challenges: Deep learning models, while achieving high accuracy, often lack transparency in their decision-making process, posing challenges for interpreting and understanding the factors driving predictions. This "black-box" nature could hinder clinical adoption and limit insights into disease mechanisms.
Limited Generalization: The model's effectiveness might be influenced by the specific dataset used for training, potentially leading to challenges in generalizing its performance to different demographics, ethnicities, or imaging protocols. The resulting lack of robustness could impact the model's real-world applicability.
Data Quality and Quantity: The accuracy and reliability of deep learning models heavily rely on the quality and quantity of the training data. Inadequate or biased data could result in suboptimal performance, and collecting diverse and comprehensive datasets can be challenging.
Ethical and Privacy Concerns: Collecting and utilizing medical imaging data, especially in deep learning applications, raises ethical considerations related to patient privacy, informed consent, and data security. Adhering to these considerations is crucial but can be complex.

Figure 10 Limitation & it's detail of the Related Work

3. Predicting Alzheimer's disease employing neural networks is the ultimate objective of the researched document " Early Diagnosis of Alzheimer's Disease using Convolutional Neural Network-based MRI " by "Kadhim, K.A., Mohamed, F., Sakran, A.A., Adnan, M.M., and Salman, G.A., 2023." The analysis utilizes the resources of the widely obtainable and nicely organized Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset, primarily encompasses T1 and T2 MRI images captured on 1.5T and 3T scanners. Under the recommended approach, a convolutional neural network (CNN) architecture is adopted, whose thereafter undergoes training via epidemiological. Using an CNN algorithm that was learned for 15 epochs, this investigation shows favourable findings, with an accurate test result of 0.977 being obtained. Given that the approach employed by CNN has intriguing prognostic capacity, the study's findings might be reinforced by addressing issues which involves the potential occurrence of data biases, problems dealing with a variety dataset, and the need of stretched evaluation on more expansive datasets. Using machine learning algorithms on neuroscience data and population characteristics, this investigation boosts Alzheimer's disease projection and opening up the possibility to more effective early diagnosis alongside potential treatment options (Kadhim, et al., 2023).

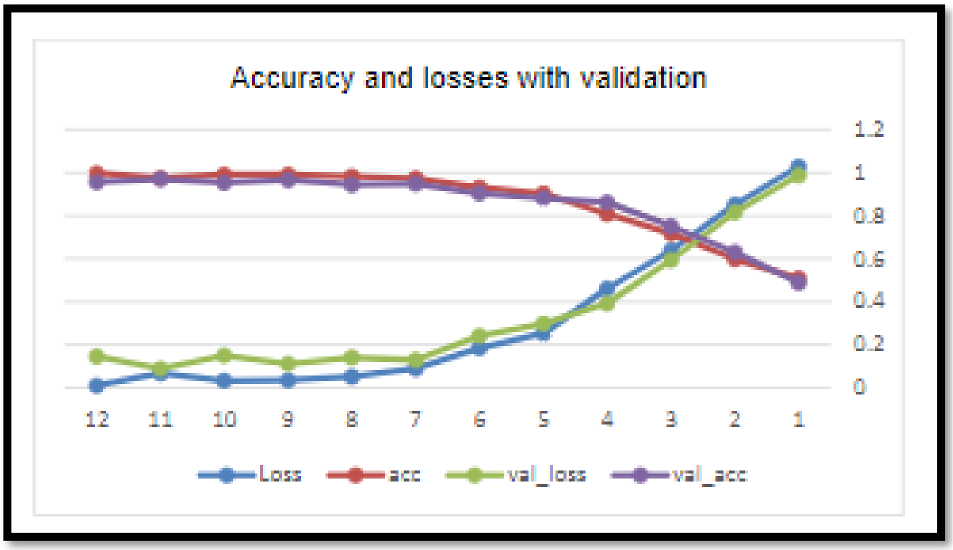


Figure 11 Changes in accuracy and losses as compared to validation accuracy and loss

Technological Limitation:

Research reported in the "Kadhim et al. | Malaysian Journal of Fundamental and Applied Sciences, Vol. 19 (2023) 362-368" outlines the execution of a convolutional neural network (CNN) model developed on the Alzheimer's Disease Neuroimaging Initiative's (ADNI) dataset to detect and categorize Alzheimer's disease across multiple stages. It has to be done to keep in mind multiple states restrictions and technological obstacles, notwithstanding considering that the suggested technique showed favourable results with exceptional precision regarding recognizing Alzheimer's disease levels. As the effectiveness of the model is dependent on the accuracy and variety of the input data, a possible drawback is the possibility of bias or absence of adaptability in the training data. In addition, the importance of neuroimaging data could limited its function to contexts in which it is in fact accessible. Technically, the adoption of vertical filters for adjusting the study's layout may trigger variability, and this might harm the veracity of the conclusion. When presenting the study's results in the classroom, it's important to keep these stipulations and methodological details in mind (Kadhim, et al., 2023).

Limitations
Potential bias or generalizability issues in training data
Reliance on availability of neuroimaging data
Variability introduced by alteration of architecture

Figure 12 Limitation & it's detail of the Related Work

4. The research study titled "DeepAD: Alzheimer's disease classification via deep convolutional neural networks using MRI and fMRI" by Sarraf et al. provides a unique use of deep learning methods to solve the problem of Alzheimer's disease classification. In spite of the fact that the study demonstrates the diagnostic use of deep convolutional neural networks (CNNs) using MRI and fMRI data, it is necessary to do an in-depth analysis of the research. The paper indicates that CNNs may be used for classification, but it also confronts issues relating to the availability of data and generalizability. Neural networks need significant volumes of high-quality data in order to function well. In addition, the work would be strengthened by more research on the interpretability of CNN-based models in the context of Alzheimer's disease diagnosis, as well as by the elimination of any possible biases present in the dataset. Nevertheless, the work that Sarraf and his team have done is an important contribution to the field. It draws attention to the ongoing effort to utilize machine learning in order to achieve a detection of Alzheimer's disease that is both more accurate and more efficient. However, it is acknowledged that additional research and refinement are required in order to translate those advances in technology into beneficial medical uses (Sarraf, et al., 2016).

Technological Limitations:

Because of the complexity of deep convolutional neural networks (CNNs), the application needs a significant amount of processing time and computer resources in order to function properly. This is one of the most noteworthy limitations. It is necessary to have a high-performance computer infrastructure in order to train and fine-tune these deep models, which may not be easily accessible in all types of healthcare settings. Additionally, since this method relies on data obtained from MRIs and fMRIs, expensive and specialized imaging equipment is required. This constrains the scope of this method's potential applications. In addition, despite the fact that the CNN-based technique produces encouraging findings, it poses difficulties in terms of interpretability. This is due to the fact that the internal workings in these deep networks may be difficult to read, which might possibly impede the practical adoption of such models. In conclusion, the performance assessment of the research focuses largely on classification accuracy, which, although very important, does not completely meet the demand for explication and practical application in actual-world healthcare settings (Sarraf, et al., 2016).

5. The writers of the piece titled "A survey on deep learning in medical image analysis" by Litjens et al. (2017) present a thorough overview of the use of deep learning methods in the area of medical image analysis. These techniques include convolutional neural networks, recurrent neural networks, and convolutional neural networks. The research report draws attention to a number of the difficulties and constraints that must be overcome before deep learning can be used to automate a variety of elements of medical picture interpretation. However, it does give some significant insights into the possibilities of deep learning. The adaptability of this technology is shown by the fact that the survey provides a comprehensive coverage of a wide variety of medical imaging modalities and activities to which deep learning has been used. This is one of the most prominent strengths of the study. However, the authors recognize that deep learning algorithms often need huge annotated datasets, which might be a barrier in the medical sector owing to issues around data privacy as well as the requirement for expert annotations. In addition, the research places an emphasis on the interpretability problem of deep learning models in medical contexts. This is because having a comprehension of the decision-making process of these complicated models is essential for their acceptability in clinical settings. In spite of these obstacles, the survey is an invaluable resource for academics and practitioners working in the area of medical image analysis, and it successfully portrays the rising significance of deep learning in improving the field of medical image analysis (Litjens, et al., 2017).

Technological Limitations:

The research conducted by Litjens and colleagues and titled "A survey on deep learning in medical image analysis" draws attention to a number of the technical obstacles that stand in the way of the use of deep learning to medical image analysis. One of the most significant limitations is the need for big datasets that have been annotated, which may be difficult to acquire in the medical field owing to concerns over patients' right to privacy and the long process of annotating data by experts. In addition, the study highlights the problem of interpretability by pointing out that deep learning models typically function as "black boxes," which makes it challenging for physicians to trust and comprehend the conclusions made by these models. The authors also underline the computational needs of deep learning, which require strong hardware and a large amount of processing time. As a result, real-time applications might be hampered by these requirements. In addition, there is still more work to be done to improve the reliability and adaptation of deep learning models spanning a wide variety of imaging techniques in medicine and patient groups. In conclusion, the absence of defined assessment techniques and benchmark datasets makes it impossible to compare and reproduce the findings of several investigations. Because of these technical constraints, continued research is required to find solutions to these problems and make the most of the promise of deep learning in healthcare picture analysis.

Summary:

Chapter 2 serves as a thorough literature review, delving into the realm of Convolutional Neural Networks (CNNs) and their significance. It began with a clarification of what CNNs are and the reason why they are being exploited. With this chapter, readers will be equipped with an intimate understanding of all of the intricate components that jointly make up a CNN, comprising Convolution layers, pooling layers, non-linear layers, ReLU levels, fully connected layers, and softmax layers. It presents an in-depth examination of research conducted on CNNs as they correlate to the diagnosis of Alzheimer's disease, going beyond the introductory material presented in Chapter 1. It delivers a detailed breakdown of these investigations, laying out exactly they technologically dropped short. This just validates the vitality of this study, but also underscores the critical requirement for innovative methods to address the hurdles that are currently hindering the early detection of Alzheimer's disease.

Chapter 3

Methodology

There is a pictorial representation of the whole methodology of this research paper.

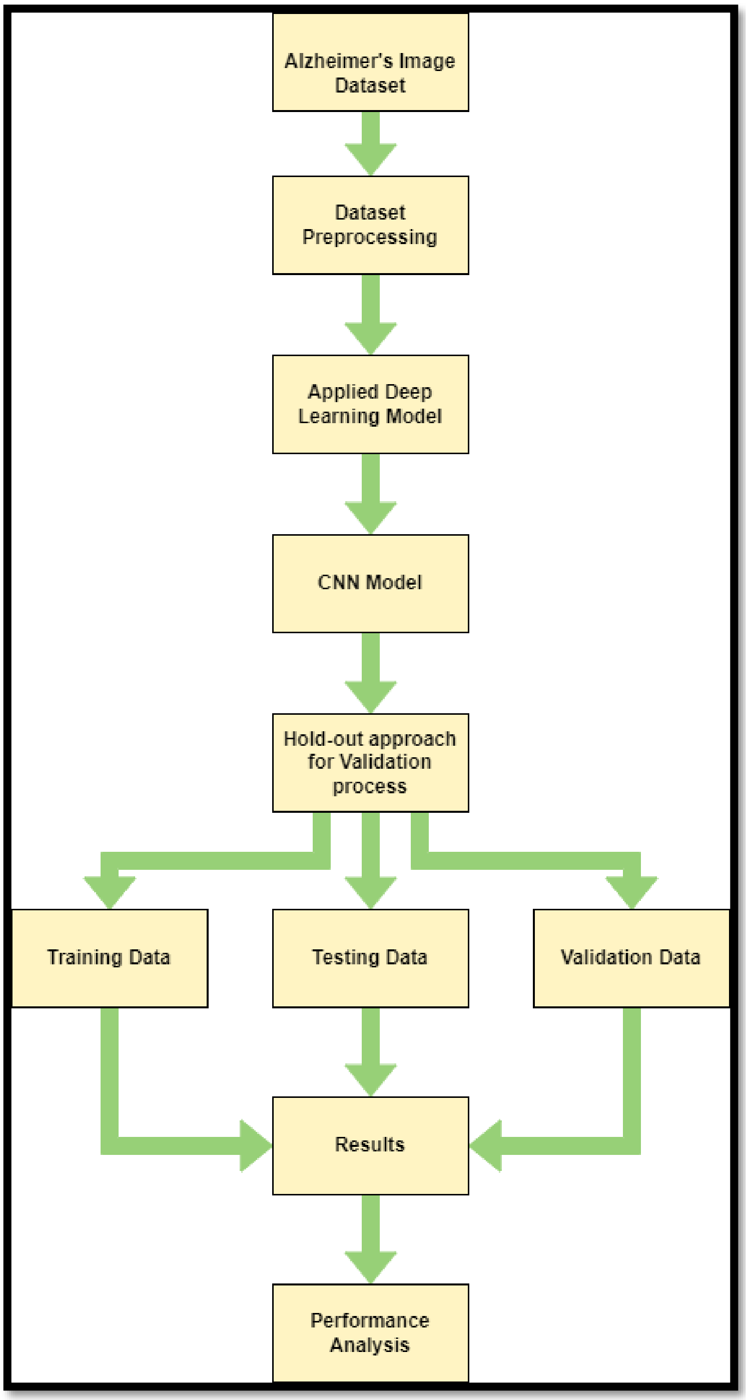


Figure 13 Overview of research

Alzheimer’s Image Dataset:

Within the framework of the investigation, the Alzheimer's Dataset, which consisted of Brain MRI Images, was retrieved from a publicly available online source. The original contributors of the dataset have been recognized for their meticulous collecting of data from a wide variety of online resources, which was followed by demanding label verification methods. The analysis set contains the contents of an extensive collection of 6400 MRI photographs each of which has been annotated in accordance to one of two separate categories. The train dataset is an amalgamation of many different sorts of picture data which is divided into four classes such as Non Demented, Very Mild Demented, Moderately Demented, and Mild Demented, while the test dataset class is also made up of same classes that is employed with particular the objective of spotting cases of Alzheimer's dementia.

Dataset Pre-processing:

When it comes to improving picture data, the dataset pre-processing step plays a vital part in preparing out the foundation for the remainder of deep learning generation of models and subsequent evaluation that will follow. In this step, a bunch of crucial actions that will assist with making the data smoother to work with later on will be done. In order to establish a strong basis and ensuring that all imagery could have been seen, "image reading" needs to be done first. The following action is scaling the images to make sure the dataset has identical dimensions. The resolution of the image has been enhanced through employing reduction of noise methods, such as denoising, to get rid of disturbing distortions. Image segmentation is a method for extracting beneficial insights from data by partitioning photos into separate sections. Transitions that are smoother and less unfavourable adjustments to the data could potentially performed with the use of biomechanical processes, especially edge reduction. Data can be enhanced, established, and precisely organized attributable to this essential pre-processing pipeline. These managed datasets convey a platform for more thorough studies to be undertaken, through the assistance of deep learning models trained exclusively for domains such as image labelling and detection.

CNN Model:

Various kind of CNN architecture will be utilized in the simulation of this paper and will be compared with each other to determine the best optimal solution. Brief explanation has been provided below of the architectures.

1. **ResNet50V2 (Residual Network 50 Version 2)**

   - **Explanation-** ResNet50V2 is a convolutional neural network (CNN) architecture that is classified within the ResNet family, known for its dense structure. The provided architecture is an enhanced iteration of the original ResNet model. The numerical value "50" in the acronyms of the network relates to the quantity of layers integrated into its architecture, specifically encompassing 50 convolutional layers. ResNet50V2 has been specifically developed to prevent the issue of disappearing gradient by using skip connections, also known as residual connections, which facilitate the smoother flow of gradients throughout the training stage (He, et al., 2016).

   - **Key Features-**

     i. **Residual Blocks:** The ResNet50V2 model incorporates the utilization of remaining blocks, which can be regarded as a significant breakthrough in the field. The aforementioned blocks are equipped with skip connections, also known as shortcut connections, which facilitate the network in acquiring knowledge about residual functions. This approach effectively addresses the issue of the vanishing gradient problem, hence facilitating the training of deep neural networks.

     ii. **Bottleneck Architecture:** The ResNet50V2 model employs a bottleneck architecture within its residual blocks. This implies that each block consists of three convolutional layers, specifically 1x1, 3x3, and 1x1 convolutions. The utilization of 1x1 convolutions results in a reduction of processing burden whereas simultaneously maintaining the model's capacity for representing information.

iii. **Pre-Activation:** The ResNet50V2 model possesses pre-activation residual blocks, which involve the use of sequential normalization and ReLU activation prior to the convolutional operation, instead of afterwards it. This phenomenon contributes to the improvement of gradient flow and the maintenance of training consistency.

iv. **Global Average Pooling:** Global average pooling is employed as a means to diminish the number of dimensions to a singular vector at the conclusion of the network, ahead of the ultimate segmentation layer. This phenomenon leads to a decrease in excessive fitting and an enhancement in prediction (He, et al., 2019).

2. VGG16 (Visual Geometry Group 16)

- **Explanation-** The VGG16 architecture is widely recognized in the area of deep learning for its straightforward design and consistent arrangement of layers. The above-mentioned entity achieved the status of being a finalist in the 2014 ImageNet Large Scale Visual Recognition Challenge. The VGG16 architecture is composed of an overall of 16 gradient layers of data, which are further divided into 13 convolutional layers and 3 fully linked layers. The network employs 3x3 convolutional filters with a stride of one and maintains consistent buffering across all layers.

- **Key Features-**

i. **Uniform Architecture:** The VGG16 architecture is renowned for its straightforwardness and consistent design. The network employs 3x3 convolutional kernels with a stride of 1 and consistent padding across all layers. The consistent nature of this uniformity facilitates comprehension and execution.

ii. **Depth:** The VGG16 model might be considered deep in comparison to other models of its period due to its utilization of 16 weight layers. The architecture of the model consists of a total of 13 convolutional layers and 3 fully linked layers, rendering it well-suited for the extraction and representation of intricate visual characteristics within images.

iii. **Small Convolutional Kernels:** The VGG16 model might be considered deeper in comparison to other models of its period due to its utilization of 16 weight layers. The architecture of the model contains a total of 13 convolutional layers and 3 fully linked layers, rendering it well-suited for the extraction and representation of intricate visual characteristics within images.

iv. **Max-Pooling:** The VGG16 architecture incorporates max-pooling layers in order to decrease the spatial dimensions and minimize the feature maps. This technique facilitates the expansion of the receptive field, hence enhancing the network's ability to handle fluctuations in object scales by improving its stability (Simonyan & Zisserman, 2014).

There will be a use of various optimizers with the architecture so an accurate model can be developed. Brief description of optimizers is being conveyed below.

1. Adam (Adaptive Moment Estimation)

- **Explanation**- The Adam optimization technique is a hybrid approach that integrates the advantageous characteristics of two widely used optimizers, namely RMSprop and Momentum. It keeps dynamic learning rates for every parameter, which means it modifies the processing rate throughout training for every parameter based on previous differences and changing averages.

- **Key Features-**
  i. This approach employs both initial-order (gradients) as well as the second-order (moving averages of previous gradients) components.
  ii. It works nicely with sparse gradients.

iii. When contrasted with other optimizers, it usually converges faster and is less sensitive to hyper parameters (Kingma & Ba, 2014).

2. RMSprop (Root Mean Square Propagation)

- **Explanation-** RMSprop has the potential to be classified as a versatile optimization approach for learning rate adjustment. The learning rate for every variable is adapted depending on the magnitude of the previous gradients. Taking advantage of this approach assists in mitigating some constraints associated with the fundamental stochastic gradient descent (SGD) algorithm.

- **Key Features-**
  i. The algorithm calculates and stores the average of the squared gradients for each parameter over a shifting window.
  ii. The learning rates are calculated in a unique manner for every single parameter, taking into account the fluctuating averages.
  iii. One of the benefits of this approach is its ability to address the issue of diminishing or ballooning gradient problem (Haji & Abdulazeez, 2021).

3. SGD (Stochastic Gradient Descent)

- **Explanation-** Stochastic Gradient Descent (SGD) is a fundamental optimization technique frequently used in the training of neural networks for learning purposes. The parameters of the model are altered by utilizing the gradients of its loss functioning relative to the parameters. In its purest state, the algorithm employs a constant learning rate. Stochastic Gradient Descent (SGD) eliminates the mentioned redundancy by executing a single update at each iteration. The stochastic gradient descent (SGD) algorithm is characterized by frequent modifications and high variability, resulting in significant fluctuations in the objective function.

- **Key Features-**
  i. The proposed solution is straightforward and readily executable.

ii. Overall precise modification of the learning rate and learning rate schedule might turn out necessary.
iii. This phenomenon is characterized by a tendency to become trapped in local minima (Ruder, 2016).

There are various libraries that would be utilized in the simulation of this paper. Some of them are:

1. TensorFlow:

- Google is the organization that developed and continues to host the open-source machine learning framework known as TensorFlow. It was first made available to the public in 2015, and since then, it has been receiving an incredible amount of attention in the area of deep learning and artificial intelligence.

- TensorFlow is an open-source framework that offers a versatile and feature-rich environment for the creation and deployment of machine learning and deep learning models. It is geared toward handling an extensive number of activities, including the identification of images and sounds, the processing of natural languages, and many more.

- **Key Features:**

i. Working with massive datasets is simplified with the help of TensorFlow's extensive library of pre-built models and transfer learning tools.
ii. Both high-level APIs like Keras for rapid model construction and low-level APIs for granular control are available in TensorFlow.
iii. Because of its support for distributed computing, machine learning workloads may be scattered over many graphics processing units (GPUs) or even across distributed clusters.
iv. Tools for deploying and maintaining machine learning pipelines in production are available via the TensorFlow Extended (TFX) ecosystem extension (Team, n.d.).

2. scikit-learn:

- scikit-learn is a Python library for machine learning that has been heavily utilized. It also frequently gets referred to as sklearn. It was first made accessible for the public in 2007 and has been built on top of a variety of well-known Python libraries, including NumPy, SciPy, and Matplotlib.

- The library delivers a straightforward and effective tool for data analysis and modelling, and its design has been created in a manner that ensures it can be deployed by both novices and seasoned professionals in the field of machine learning (Cournapeau, 2011).

  - **Key Features:**

    i. Classification, regression, clustering, dimensionality reduction, and more are all achievable using scikit-learn's machine learning techniques.
    ii. It is an entire library for end-to-end machine learning processes, involving tools for data preparation, feature selection, and model validation.
    iii. The consistent and user-friendly API provided by scikit-learn makes it simple to try out new algorithms and methods.
    iv. Utility functions for things like cross-validation, hyper parameter adjustment, and persistent models are included (Thirion, et al., 2011).

Both TensorFlow and scikit-learn are very useful tools in the area of machine learning. They're offering an immense amount of material the fact that can be used for grasping any number of approaches to machine learning and putting those strategies into practice. These collection of libraries lend the tools you require to create effective models while getting insights from the data, irrespective of whether work is being done on deep learning projects with TensorFlow or typical machine learning tasks with scikit-learn. Any of these libraries could be used.

Hold-out Approach for Validation Process:

The hold-out validation approach is a strategy that evaluates a model's efficacy and generalizability; it is constantly used to Convolutional Neural Networks (CNNs). The dataset is partitioned into three parts for this step: the training set, the validation set, and the test set.

The word "hold-out" refers to elements of the dataset that are kept back or reserved for special objectives. As an illustration during training, the validation set can possibly be employed for tracking the model's development and make improvements as essential, whereas the test set could be left unaltered until the end so that it provides an impartial assessment of the model's performance. This strategy delays the model from disregarding the training data and indicates that it is capable of applying competently to new, formerly unidentified data (Kohavi, 1995).

Training Process:

The dataset segment used for instructing the CNN is termed the training set or is called the training process. The model is trained using those instances, gaining into account the common threads and distinctive characteristics underlying them. Feeding samples for training into the CNN, identifying the loss, and tuning the model's parameters (weights and biases) for lowering the loss compose the training process (Caruana & Niculescu-Mizil, 2006).

Testing Process:

When training a model or tweaking its hyper parameters, the test set is put to use alternately. It is an impartial evaluation of the CNN's capacity to interpret newly acquired data. The test set serves a purpose for quantifying the CNN's performance in real-life circumstances and for assessing its ability to be generalized to new locations (He, et al., n.d.).

Validation Process:

The process of validation is of fundamental validity in this investigation, especially for working with large image collections. A hold-out validation methodology is used, a method that has proven effective, to offer proof for the correctness of the results. In this manner, the dataset is split into training, testing, and validation portions ranging from zero to one hundred percent. The efficacy of deep learning models by meticulously examining the dataset in this way can assessed accurately (Hinton, et al., 2012).

Summary:

The methodology chapter, Chapter 3, is a vital component of the research, that clarify the necessary techniques and tactics implemented during the study. It initiates by stating the dataset, which involves MRI scans of brain pictures, underlining its importance as a study's basis. The chapter digs into the delicate data prior to treatment approaches, revealing how the initial data is altered and cleaned up for further examination while promising the dataset's integrity and quality. It unveils the study's core component: Convolutional Neural Network (CNN) models. It delivers an exhaustive outline of CNNs, pointing out their indispensable function in the identification of Alzheimer's disease. On top of that, the part discusses into two particulars CNN designs, ResNet50V2 and VGG16, delivering comprehensive knowledge onto their inner workings and major attributes. This data forms the foundation for later model creation and assessment. Moreover, the chapter focuses into three independent optimizers - Adam, SGD, and RMSprop - whose use is essential to CNN model training. Each optimizer is deeply discussed, accentuating its unique features and incentives, giving a foundation for educated decision-making throughout model construction. In terms of validation, the methodology chapter presents the Hold-out strategy and explains its significance in evaluating model performance. It also describes the complexities of the training, testing, and validation procedures, providing a clear path for evaluating CNN models. Chapter 3 is essentially a complete guide to the research approach, including data preparation, CNN design, optimizer selection, and validation processes. It sets the framework for further research into Alzheimer's disease detection using machine learning.

Chapter 4

Experimental Results

The following paragraphs discusses the outcomes gathered from studies on the significance of magnetic resonance imaging dataset. The current section contains a total of four subsections: The preceding section (4.1), presents an explanation of the dataset's comprehensive explanation. The second section (4.2) includes a brief description of the experimentation arrangement utilized to train the data set and to achieve the outcome. The end result of the investigation has been outlined in third section (4.3), and the side by side comparison with the initial techniques is summed up in last section (4.4). The outcomes were generated on the examination of the data sets acquired from multiple MRI scans and aimed at shedding light on how deep learning may help improve effectiveness and assist in earlier identification of Alzheimer's disease from imagery. The insights have been displayed in tables and figures and are dealt with more thoroughly in the segments that follow.

4.1 Dataset Description:

The dataset is divided into four classes which contains MRI scans of brain namely:

Dataset Brief		
Sr. No.	Class	Total Number of Images
1	NonDemented	3200
2	MildDemented	896
3	VeryMildDemented	2240
4	ModerateDemented	64

1. <u>Non Demented Brain Images</u>:
    - Magnetic resonance imaging (MRI) images of non-demented brain pictures frequently indicate no notable irregularities.
    - Individuals such as these maintain normal brain anatomy and functionality (Buckner, et al., 2000).
    - Their senses serve as an analogy for contrast between brains crippled through dementia.

- This class contains in total 3200 images of very mild demented MRI scans of human brain.

Figure 14 Non demented brain image

2. Mild Demented:
   - Brain scans of people with mild dementia often reveal more obvious brain shrinkage or morphological alterations.
   - Prominent outcomes involve cortical thinning, ventricular enlargement, and hippocampal atrophy (Jack, et al., 2005).
   - Cognitive impairments become more evident but still remain modest.
   - This class contains in total 896 images of very mild demented MRI scans of human brain.

Figure 15 Mild demented brain image

3. Very Mild Demented:
   - Very mild dementia brain imaging may reveal minor alterations, such as minor atrophy or smaller volumes within particular brain areas.
   - Premature indications for hippocampal atrophy or enlarged ventricles could represent amidst these modifications (Pennanen, et al., 2004).
   - At that stage, cognitive losses are minor.

- This class contains in total 2240 images of very mild demented MRI scans of human brain.

Figure 16 Very mild demented brain image

4. <u>Moderate Demented</u>:
    - A large amount of brain shrinkage and harm to the structure can be detected in moderately demented brain visuals.
    - Hippocampal shrinkage, cortical atrophy, and ventricular enlargement have been clearly apparent (Whitwell, et al., 2007).
    - The level of severity of cognitive weaknesses hinders normal daily activities.
    - This class contains in total 64 images of very mild demented MRI scans of human brain.

Figure 17 Moderate demented brain image

There are total 6400 images in the dataset. They will be further divided into three set at the time of training. The dataset was acquired via the publicly accessible Kaggle, a platform for data science competitions and an online community for data scientists and machine learning practitioners run by Google LLC (Google, 2010).

4.2 Experimental Setup and Codebase:

Installing Python on the local machine in order to run the simulation was the first stage in this experiment of the research. The system was set up for further use after installing crucial Python modules like pip and configuring the system's path. Jupyter Lab was afterwards installed by executing the command "**pip install jupyterlab**" in the command prompt. The picture below shows how this installation procedure works.

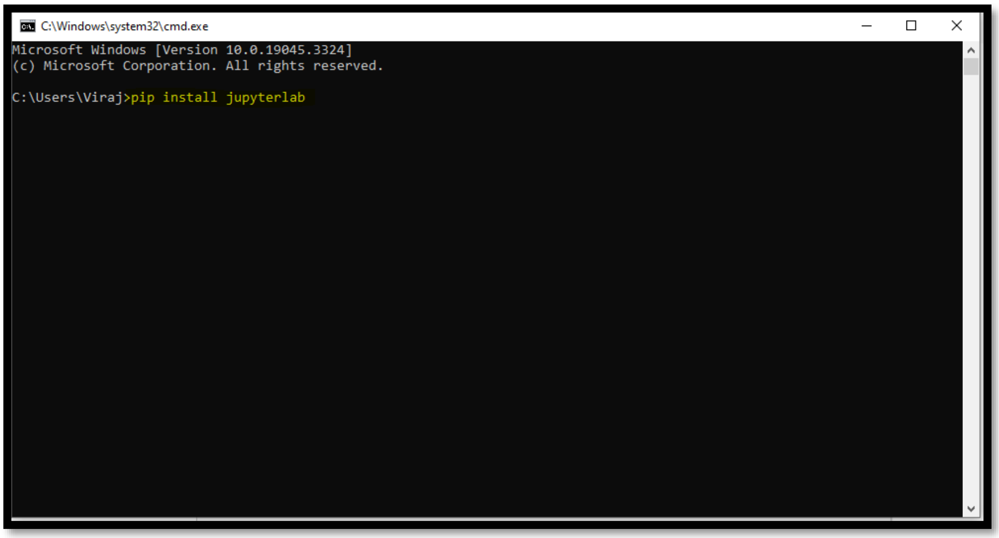


Figure 18 Installation of Jupyter Lab

After successfully installing Jupyter Lab, you can initiate and run it by simply typing the command "jupyter lab" into the command prompt. This procedure is visually represented in the figure below (Jupyter, n.d.).

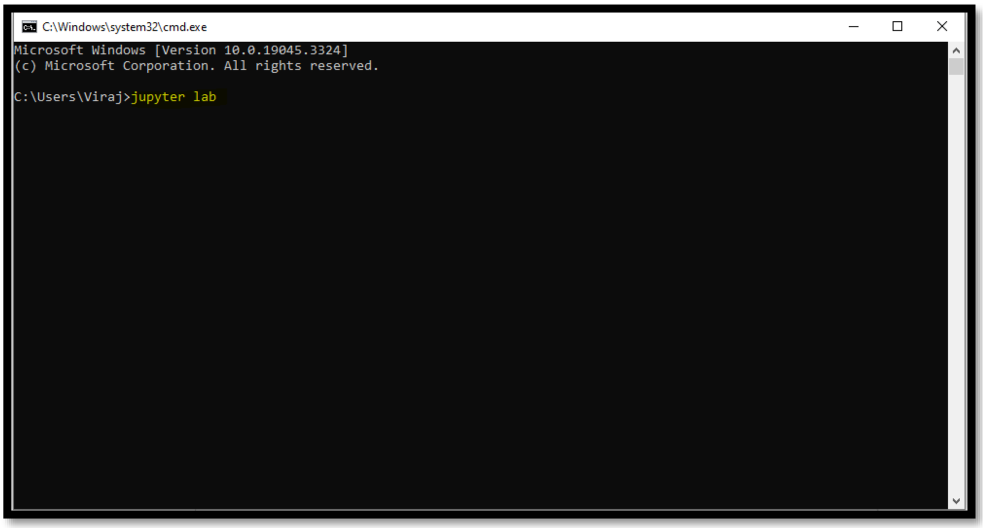


Figure 19 Starting Jupyter Lab

After opening the Jupyter Lab, the dataset should be uploaded to it to start working with it. It can be simply uploaded to it.

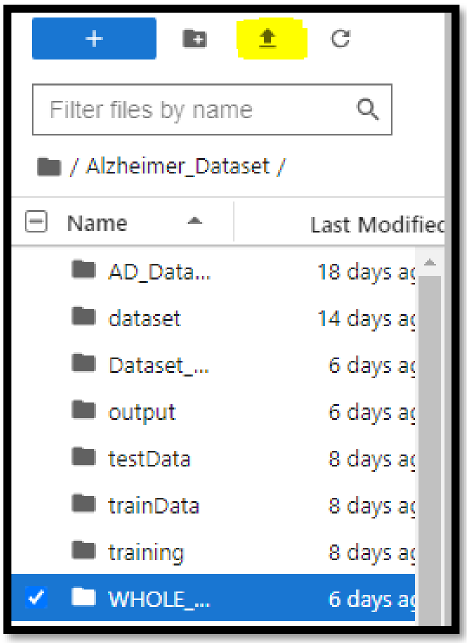
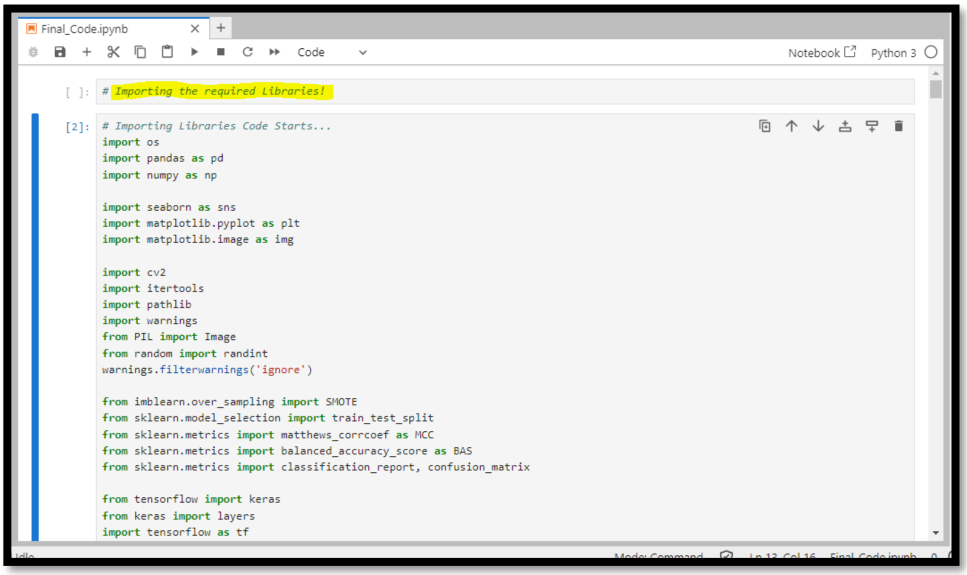


Figure 20 Uploading the dataset to Jupyter Lab

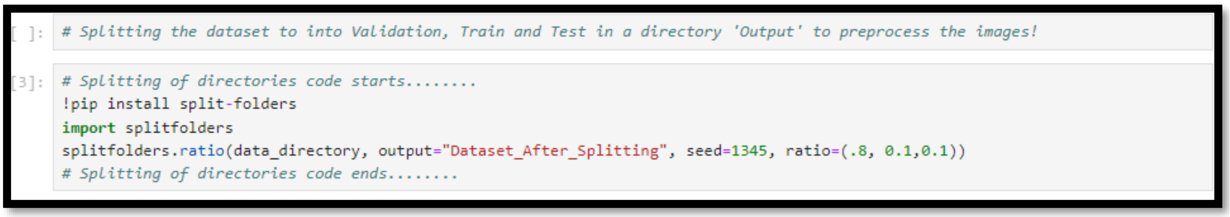
In order to get the simulation up and running, the first thing that needed to be done was to import the Python libraries that were required to execute the code along with any extra libraries that were required, such as TensorFlow, Keras, OpenCV, and numerous other libraries that frequently get utilized for deep learning and data manipulations.



```
[ ]: # Importing the required Libraries!

[2]: # Importing Libraries Code Starts...
import os
import pandas as pd
import numpy as np

import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.image as img

import cv2
import itertools
import pathlib
import warnings
from PIL import Image
from random import randint
warnings.filterwarnings('ignore')

from imblearn.over_sampling import SMOTE
from sklearn.model_selection import train_test_split
from sklearn.metrics import matthews_corrcoef as MCC
from sklearn.metrics import balanced_accuracy_score as BAS
from sklearn.metrics import classification_report, confusion_matrix

from tensorflow import keras
from keras import layers
import tensorflow as tf
```

Figure 21 Importing Libraries in Jupyter Lab

A training set, a validation set, and a test set seemed respectively derived from the primary dataset afterwards it was subdivided even more in a 'Dataset_After_Splitting' directory for pre-processing the images. It is splitted into a ratio of 0.8, 0.1 and 0.1 respectively. It means 80% in the training set, 10% in the validation set and 10% in the test set from the total images.



```
[ ]: # Splitting the dataset to into Validation, Train and Test in a directory 'Output' to preprocess the images!

[3]: # Splitting of directories code starts........
!pip install split-folders
import splitfolders
splitfolders.ratio(data_directory, output="Dataset_After_Splitting", seed=1345, ratio=(.8, 0.1,0.1))
# Splitting of directories code ends........
```

Figure 22 Splitting the Dataset

After that, the data in the dataset went through subjected to some preparatory processing, such as scaling all that of the imaging to the same dimensions is reached.

```
[ ]: # Preprocessing the dataset!

[25]: # Preprocessing dataset code starts..........
IMG_HEIGHT = 128
IMG_WIDTH = 128

training_dataset = tf.keras.preprocessing.image_dataset_from_directory(
    "./output/train",
    seed=123,
    image_size=(IMG_HEIGHT, IMG_WIDTH),
    batch_size=64
)

testing_dataset = tf.keras.preprocessing.image_dataset_from_directory(
    "./output/test",
    seed=123,
    image_size=(IMG_HEIGHT, IMG_WIDTH),
    batch_size=64
)

validation_dataset = tf.keras.preprocessing.image_dataset_from_directory(
    "./output/val",
    seed=123,
    image_size=(IMG_HEIGHT, IMG_WIDTH),
    batch_size=64
)
# Preprocessing dataset code ends
```

Figure 23 Data Preprocessing

The pre-processing of the photographic images further involves a visual representation of several of the sample images acquired from the dataset.

```
[ ]: # To inspect a small sample of the training data to gain a visual understanding of how the images in the dataset are labeled.

[27]: # Code for visualization of sample images starts..........
plt.figure(figsize=(7, 7))
for images, labels in training_dataset.take(1):
    for i in range(9):
        ax = plt.subplot(3, 3, i + 1)
        plt.imshow(images[i].numpy().astype("uint8"))
        plt.title(class_names[labels[i]])
        plt.axis("off")
# Code for visualization of sample images ends..........
```

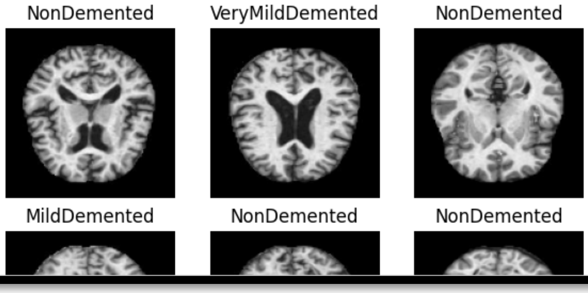


Figure 24 Sample Images Acquired

For the purpose of image classification, a model was successfully implemented, and a specialized convolutional neural network model was developed, the two of those tasks being conducted with the help of Keras. Furthermore, the model incorporates an assortment of pooling layers, dropout layers for the purpose of regularization, and dense layers for the purpose of classification.

```
# Developing the Model!
```

```
# Code for development of the model starts..........
model = keras.models.Sequential()
model.add(keras.layers.experimental.preprocessing.Rescaling(1./255, input_shape=(IMG_HEIGHT,IMG_WIDTH, 3)))
model.add(keras.layers.Conv2D(filters=16,kernel_size=(3,3),padding='same',activation='relu',kernel_initializer="he_normal"))
model.add(keras.layers.MaxPooling2D(pool_size=(2,2)))


model.add(keras.layers.Conv2D(filters=32,kernel_size=(3,3),padding='same',activation='relu',kernel_initializer="he_normal"))
model.add(keras.layers.MaxPooling2D(pool_size=(2,2)))

model.add(keras.layers.Dropout(0.20))

model.add(keras.layers.Conv2D(filters=64,kernel_size=(3,3),padding='same',activation='relu',kernel_initializer="he_normal"))
model.add(keras.layers.MaxPooling2D(pool_size=(2,2)))

model.add(keras.layers.Dropout(0.25))
model.add(keras.layers.Flatten())
model.add(keras.layers.Dense(128,activation="relu",kernel_initializer="he_normal"))
model.add(keras.layers.Dense(64,"relu"))
model.add(keras.layers.Dense(6,"softmax"))
```

Figure 25 Model Development

The model that had to be developed was subsequently optimized through the 'Adam' optimizer, and the loss function and evaluation metric were determined prior it was compiled.

```
# Compiling the Model!
```

```
# Model Compilation code starts......
model.compile(loss="sparse_categorical_crossentropy",
              optimizer = "Adam",
              metrics=["accuracy"]
             )

print("--Model Compiled Successfully--")
# Model Compilation code ends......
```

```
--Model Compiled Successfully--
```

Figure 26 Model Compilation

The model gets trained employing the training dataset for a particular number of epochs while concurrently keeping account of the training metrics and the validation metrics. 20 number of epochs were mentioned in this paper to train the model to achieve the accuracy.

```
[ ]: # Code for training the model!

32]: ModelTraining = model.fit(
         training_dataset,
         validation_data=validation_dataset,
         epochs=20
     )

Epoch 1/20
64/64 [==============================] - 64s 992ms/step - loss: 0.2054 - accuracy: 0.9228 - val_loss: 0.2075 - val_accuracy: 0.
9354
Epoch 2/20
64/64 [==============================] - 61s 944ms/step - loss: 0.1526 - accuracy: 0.9463 - val_loss: 0.1156 - val_accuracy: 0.
9765
Epoch 3/20
64/64 [==============================] - 61s 944ms/step - loss: 0.1288 - accuracy: 0.9516 - val_loss: 0.1070 - val_accuracy: 0.
9706
Epoch 4/20
64/64 [==============================] - 61s 948ms/step - loss: 0.1221 - accuracy: 0.9548 - val_loss: 0.1223 - val_accuracy: 0.
9667
Epoch 5/20
64/64 [==============================] - 61s 945ms/step - loss: 0.1018 - accuracy: 0.9639 - val_loss: 0.1196 - val_accuracy: 0.
9569
Epoch 6/20
64/64 [==============================] - 61s 941ms/step - loss: 0.1038 - accuracy: 0.9624 - val_loss: 0.0940 - val_accuracy: 0.
9824
Epoch 7/20
64/64 [==============================] - 60s 938ms/step - loss: 0.0766 - accuracy: 0.9753 - val_loss: 0.0700 - val_accuracy: 0.
```

Figure 27 Model Training

A graphical visualization is being depicted to observe the accuracy and loss of training and validation dataset. The efficiency of the model on the training dataset, the validation dataset, and the testing dataset, every single of which offered accuracy scores, were utilized as the foundation for an evaluation of the model.

```
[ ]: # Code to evaluate the model for dataset!

57]: # Code for evaluation of model starts.......
     get_los, get_ac == model.evaluate(testing_dataset)
     print(round(get_ac*100,2))
     get_ac, get_ac = model.evaluate(training_dataset)
     print(round(get_ac*100,2))
     get_ac, get_ac = model.evaluate(validation_dataset)
     print(round(get_ac*100,2))
     # Code for evaluation of model ends.......

9/9 [==============================] - 3s 238ms/step - loss: 0.1406 - accuracy: 0.9553
98.04
64/64 [==============================] - 21s 321ms/step - loss: 0.0067 - accuracy: 0.9998
99.98
8/8 [==============================] - 3s 306ms/step - loss: 0.0589 - accuracy: 0.9804
98.04
```

Figure 28 Evaluation of Model

The model that had been developed was trained on an inventory of test visuals in order to generate predictions, and its findings of those predictions could be seen alongside the images' actual labels.

```
[ ]: # Code for visualizing the predictions of a neural network model
     # on a sample batch of test images from a test dataset!

45]: # Code for visualizing the predictions starts.....

     plt.subplots(figsize=(20, 20))
     for images, labels in testing_dataset.take(1):
         for i in range(16):
             ax = plt.subplot(4, 4, i + 1)
             plt.imshow(images[i].numpy().astype("uint8"))
             predictions = model.predict(tf.expand_dims(images[i], 0))
             score = tf.nn.softmax(predictions[0])
             if(class_names[labels[i]]==class_names[np.argmax(score)]):
                 plt.title("Actual: "+class_names[labels[i]])
                 plt.ylabel("Predicted: "+class_names[np.argmax(score)],fontdict={'color':'green'})

             else:
                 plt.title("Actual: "+class_names[labels[i]])
                 plt.ylabel("Predicted: "+class_names[np.argmax(score)],fontdict={'color':'red'})
             plt.gca().axes.yaxis.set_ticklabels([])
             plt.gca().axes.xaxis.set_ticklabels([])

     # Code for visualizing the predictions ends.....

     1/1 [==============================] - 0s 287ms/step
     1/1 [==============================] - 0s 34ms/step
```

Figure 29 Test Visuals

The 'classification_report' function was employed to generate the classification metrics such as precision, f1-score, recall, and support, and it was then utilized to display those measurements for the test dataset.

```
[ ]: # Code to examine the classification performance of the neural network model
     # on the test dataset and detailed metrics provided through the 'classification_report' function.

[47]: # Code for classification performance of the model starts........
      from sklearn.metrics import classification_report,confusion_matrix

      actual_label = []
      pred_label = []

      for img, label in testing_dataset.take(1):
      #     pred = model.predict(tf.expand_dims(img, 0))
          pred = model.predict(img)
          pred = np.argmax(pred, axis=1)

      #     pred_label.append(pred[0])
      #     actual_label.append(label[i])


      #     print(actual_label, pred_label)
          print(classification_report(label,pred))

      # y_test_new = np.argmax(y_test,axis=1)

      # Code for classification performance of the model ends........

      2/2 [==============================] - 0s 145ms/step
                    precision    recall  f1-score   support

                 0       1.00      0.90      0.95        10
                 2       0.94      1.00      0.97        29
```

Figure 30 Measurements

Confusion matrix diagrams are implemented for offering a visual picture of the performance of the model, and heat maps are created to provide a visual depiction of the confusion matrices.

```
[48]: # Code for creation of Heatmap of the confusion matrix starts.......

from matplotlib.colors import LinearSegmentedColormap

label = np.array(label)  # Convert to NumPy array if not already
pred = np.array(pred)    # Convert to NumPy array if not already

# Find indices of empty labels or missing values
empty_label_indices = np.where(label == '')

# Remove empty labels and corresponding predictions
label = np.delete(label, empty_label_indices)
pred = np.delete(pred, empty_label_indices)

# Define your color palette
colors_green = LinearSegmentedColormap.from_list("custom_colormap", [(0, "white"), (1, "green")])
line_color = "#0000FF"

# Create confusion matrix
conf_matrix = confusion_matrix(label, pred)

# Get unique labels for plotting
unique_labels = np.unique(label)

# Create the heatmap
fig, ax = plt.subplots(1, 1, figsize=(14, 7))
sns.heatmap(conf_matrix, ax=ax, xticklabels=unique_labels, yticklabels=unique_labels, annot=True,
            cmap=colors_green, alpha=0.7, linewidths=2, linecolor=line_color)
```

Figure 31 Code for Confusion matrix

4.3 Results:

The discoveries of the simulation purposes, knowing that they correspond to this research, are offered in the present section of this paper.

Using the CNN simulation results lead to a 98.85% accuracy alongside a 3.62% loss. In comparison and for the validation loss as only 5.89% and the validation accuracy equals 98.04%.

```
____________________________________________________________
____________________________________________________________
THE ACCURACY IS: 98.85
________________________
THE LOSS IS: 3.62
________________________
THE Validation Accuracy IS: 98.04
________________________
THE Validation Loss IS: 5.89
```

Figure 32 Results of the Training the Dataset

The graphic illustration about the accuracy and loss of the training dataset is available to witnessed down beneath.

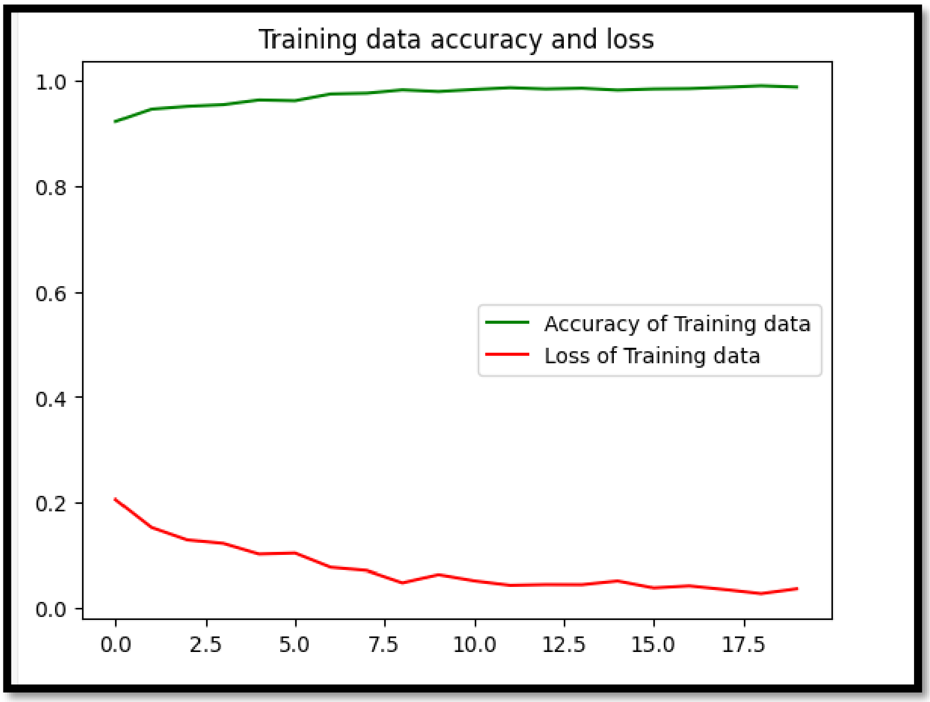


Figure 33 Graphical Representation of Accuracy & Loss of Training Data

The subsequent picture demonstrates a contrast of the plots encompassing the accuracies of the training data and the validation data.

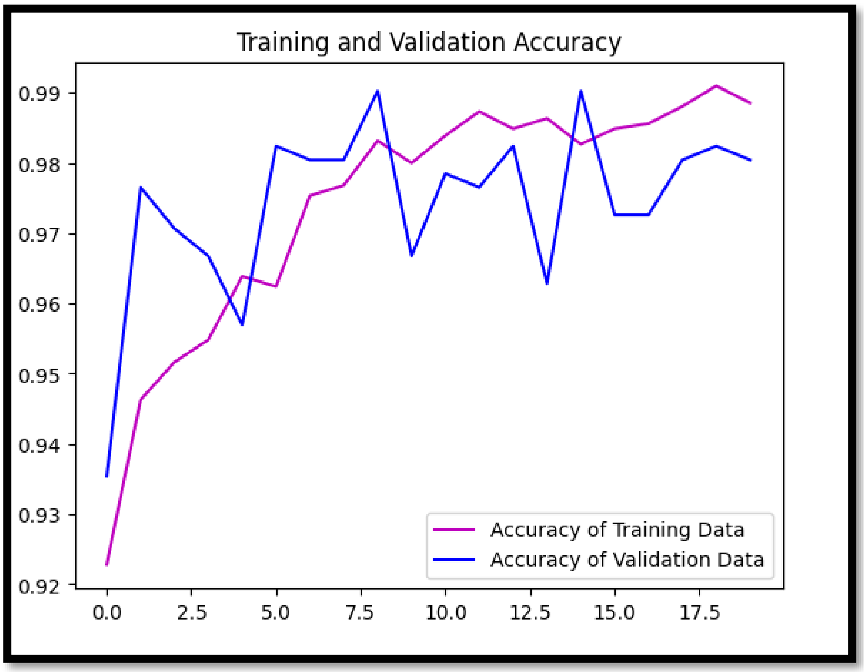


Figure 34 Graphical comparison of Accuracies of training and validation set

The screenshot that exhibits the trajectory for loss of training data and validation data can potentially be viewed down below.

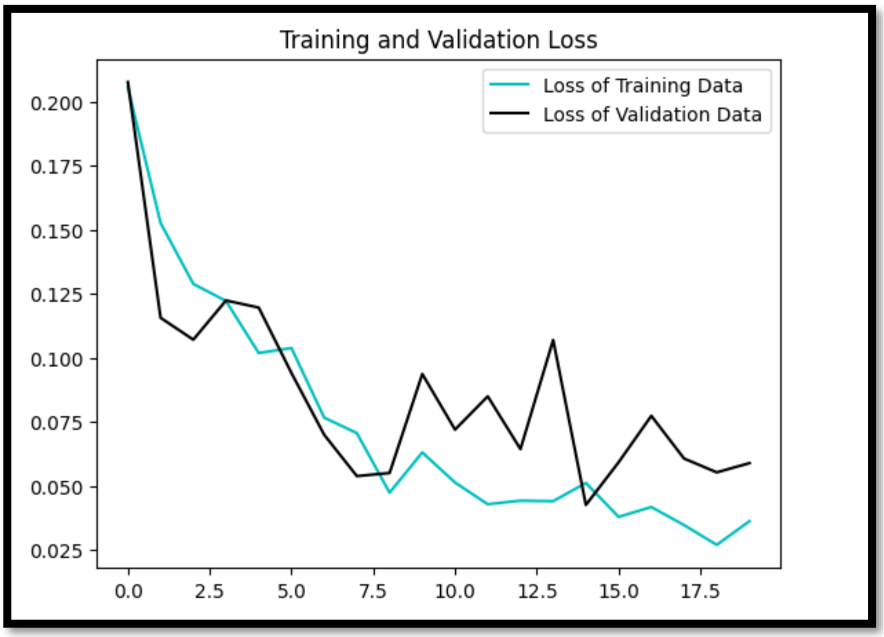


Figure 35 Graphical comparison of Losses of training and validation set

Preceding the finish line of an evaluation of the model independently with each dataset (i.e. training, validation, and testing), the forthcoming diagram exhibits an analysis of accuracy and loss associated with every dataset singularly.

```
64/64 [==============================] - 19s 298ms/step - loss: 0.0067 - accuracy: 0.9998
Accuracy Training Dataset: 99.98
Loss Training Dataset: 0.67
________________________________________
8/8 [==============================] - 3s 298ms/step - loss: 0.0589 - accuracy: 0.9804
Accuracy of Validation Dataset: 98.04
Loss of Validation Dataset: 5.89
________________________________________
9/9 [==============================] - 2s 239ms/step - loss: 0.1406 - accuracy: 0.9553
Accuracy of Testing Dataset: 98.04
Loss of Testing Dataset: 5.89
```

Figure 36 Accuracy and Loss of each dataset after evaluation of the model

A visualization of the predictions was generated in the simulations which means that as consequence, the source code for it is going to gather up some samples of photos coming from the test dataset and iterate on those photos. On the assumption of what was discovered from the model that was trained, predictions are being calculated based on the sample image files. The representation features Actual and Predicted labels, and they both which are completely unique to one another. The actual label reflects which class a photograph does, in fact, belong to, whilst its predicted label specifies what has been forecasted with regard to which class the image is expected to fall into.

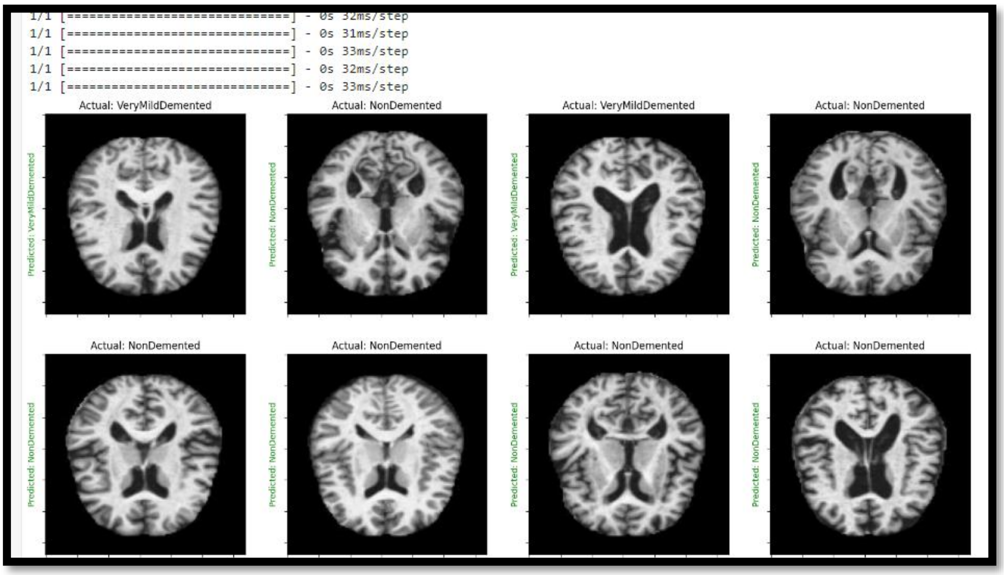


Figure 37 Visualization of the sample images

Evaluation of the classification performance of a machine learning model using TensorFlow and scikit-learn's `classification_report` function. The table shows four results from the sample of images from test dataset. They are: precision, recall, f1-score and support respectively.

2/2 [==============================] - 0s 145ms/step

	precision	recall	f1-score	support
0	1.00	0.90	0.95	10
2	0.94	1.00	0.97	29
3	1.00	0.96	0.98	25
accuracy			0.97	64
macro avg	0.98	0.95	0.96	64
weighted avg	0.97	0.97	0.97	64

Figure 38 Classification Report

Precision:

- Precision is a measure of the frequency that predictions that are favourable are really genuine positives. It estimates the contribution of genuine positives to the overall number of positive prediction (Dennis & Horn, 1966).
- Formula:

Precision = TP / (TP + FP)

Recall (Sensitivity):

- Recall determines the classifier's competence for identifying every significant occurrence in the dataset. It measures the amount of genuine positives to total positives (Davis, et al., 2006).
- Formula:

Recall = TP / (TP + FN)

F1-score:

- The harmonic mean of accuracy and recall is the F1-score. It establishes an acceptable balance amongst accuracy and recall, in particular when the classes in the dataset are unbalanced (Goutte & Gaussier, 2005).
- Formula:

F1-Score = 2 * (Precision * Recall) / (Precision + Recall)

Support:

The total quantity of recurrence of each class in the real dataset can be expressed by support. It is the total amount of genuine indications in the dataset for each class.

Support is not a score, but rather details concerning the dataset. It might come in beneficial when exploring each class's data regarding performance (Chawla, et al., 2003).

A confusion matrix heatmap is a graphical depiction of the confusion matrix which utilizes colours to show various degrees of proficiency in a classification test. A confusion matrix is comprised by combining rows the fact that depict the actual classes and columns that it conveys the predicted classes. Every column in the matrix exhibits the number of samples that are associated with a specific collection of actual and predicted classes (Luque, et al., 2019). The heatmap colours these cells to make it quicker to review and spot trends in a classification model's performance.

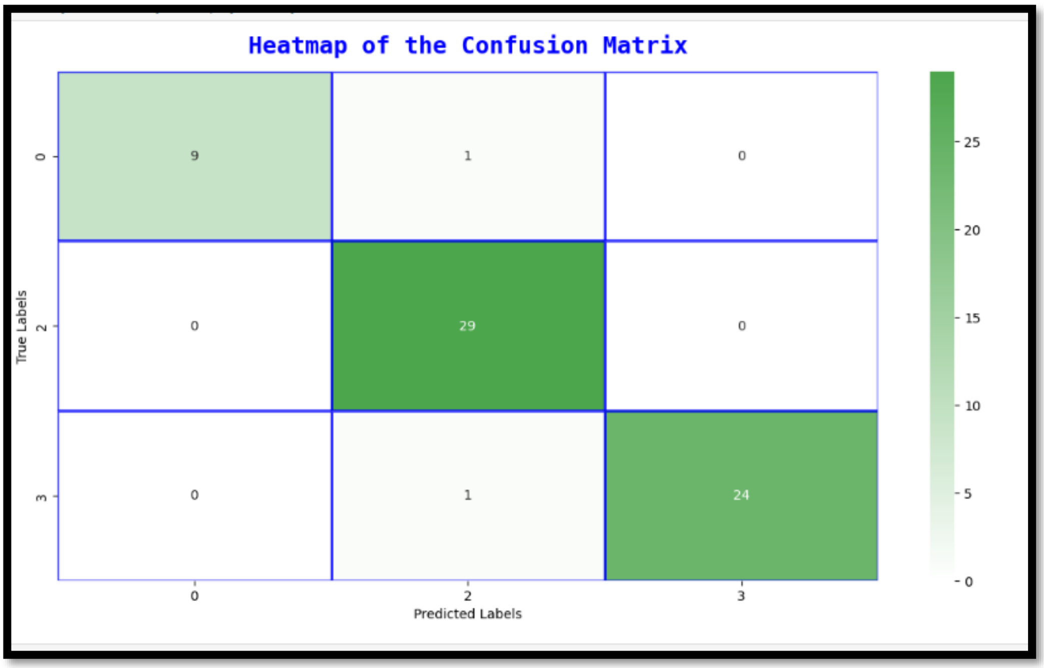


Figure 39 Heatmap of Confusion Matrix

The heatmap of the actual and predicted labels from the results attained during the model's assessment is shown in the image above.

4.4 Comparison with Baseline Methods:

The dataset is gone under training of different architecture of the neural network. The architecture utilized are as under:

[1] CNN (Custom)
[2] ResNet50V2
[3] VGG16

The different types of optimizer are also being used in the model getting the highest accuracy to achieve more and more accuracy.

[1] ADAM
[2] SGD
[3] RMSprop

The below table represents different outcomes when the dataset is trained using different combination of architecture and optimizers.

Comparison of Accuracies and Losses				
Sr No	**Architecture**	**Optimizer**	**Accuracy**	**Loss**
1.	Custom	Adam	98.85%	0.036
2.	Custom	RMSprop	97.36%	0.077
3.	Custom	SGD	64.57%	0.798
4.	ResNet50V2	Adam	64.17%	0.824
5.	VGG16	Adam	88.25%	0.345

By doing a comparative analysis of the five distinct architectures utilizing different optimizers, we have evaluated their performance in terms of accuracy and loss metrics.

Now, let us proceed with the analysis of the obtained results.

- The architecture that exhibits the highest level of accuracy is the **Custom Architecture, which utilizes the Adam Optimizer**. This particular architecture achieves a notable accuracy rate of **98.85%**. This observation implies that the utilization of the customized architecture in conjunction with the Adam optimizer yielded favourable outcomes for the given dataset. The observed low loss value of **0.036** suggests that the training process has been efficient and that the model has converged effectively. In addition, the architecture, **Custom** using the **RMSprop** optimizer, attained a classification accuracy of **97.36%** and a corresponding loss value of **0.077**. The performance of the system was commendable, exhibiting a high level of accuracy and a comparatively minimal degree of loss. RMSprop is renowned for its capacity to effectively accommodate various learning rate schedules, potentially playing a role in the commendable performance seen by this particular model.

- The **VGG16 Architecture with Adam Optimizer**, which obtained an accuracy of **88.25%**, emerged as the second-best model with loss of **0.345**. The VGG16 model has gained recognition for its efficacy in picture classification tasks, and the utilization of the Adam optimizer has further enhanced its overall performance. Although it may not be as precise as the bespoke architecture, it nevertheless yields a robust outcome.

- The architectures that exhibit the poorest performance are the Custom Architecture utilizing the Stochastic Gradient Descent (SGD) Optimizer, and the ResNet50V2 Architecture employing the Adam Optimizer. The Custom Architecture implemented with the **Stochastic Gradient Descent (SGD) Optimizer** exhibits a relatively low accuracy rate of **64.57%** and a correspondingly high loss value of **0.798**. These results indicate that the selection of the SGD Optimizer has had a detrimental impact on the performance of the model. The **ResNet50V2** model, when trained using the Adam optimizer, exhibited subpar performance relative to other models, achieving an accuracy rate of **64.17%** and a very high loss value of **0.824.**

Recommendation on why using Custom Architecture with Adam Optimizer would be more beneficial compared to other architectures and optimizer combination based on the comparison.

1. **Hyper parameter Optimization:** Optimize the hyper parameters of the customized architecture. This entails modifying learning rates, batch sizes, and the architectural components such as the number of layers or neurons within the neural network.

2. **Regularization:** Regularization techniques, such as dropout or L2 regularization, must be implemented to mitigate the issue of overfitting. Although the model exhibits a high level of accuracy, it remains susceptible to overfitting when presented with

novel, unknown data. Regularization techniques have the potential to enhance the generalization capabilities (Srivastava, et al., 2014).

3. **Data Augmentation:** It is advisable to implement data augmentation strategies throughout the training phase. The process involves producing diverse iterations of the training data by the implementation of changes such as rotation, scaling, or cropping. Taking advantage of data augmentation techniques could improve the model's ability to generalize effectively in real-world circumstances (Shorten & Khoshgoftaar, 2019).

Summary: A crucial part of the study, Chapter 4, focuses on the practical implications as well as outcomes of the research. It starts by providing an exhaustive dataset description supplying crucial context for the structure and features of the MRI brain scan data which was used in the study. This contextualization is crucial for understanding the subsequent experimental setup. The experimental setting is meticulously described during the chapter, which likewise offers a clear and replicable approach for training the dataset. This assures flexibility and makes accessible for future research projects. It also provides the code and methodologies employed for the training of Convolutional Neural Network (CNN) model. The outcomes of the experimental phase are laid out in Chapter 4. In the context of Alzheimer's disease detection, it exhibits the performance metrics and findings of the developed CNN model. The results obtained reveal confirmation of the machine learning approach's efficacy and precision in achieving the study's goals. This chapter's relative comparison with baseline approaches is a key factor. It establishes an overview for evaluating the imagination and effect of the study by objectively comparing the customized CNN model to currently used models. The potential improvements and efforts made in the area of diagnosing Alzheimer's disease applying machine learning are brought into focus in this comparative comparison. In essence, Chapter 4 covers the research's technical components, from the description of the dataset and experimental design by means of to the discussion of outcomes and their evaluation against accepted methodologies. It plays a crucial role in confirming the study's contributions to improving medical procedures for detecting Alzheimer's disease.

Chapter 5

Conclusion and Future Work

5.1 Conclusion:

The research project, which consists of four crucial parts, marks a substantial advancement in the diagnosis of Alzheimer's disease in the context of UK healthcare and as well as globally. Chapter 1 underscored the pressing requirement for advancements in research and development by exposing the local and worldwide incidence of Alzheimer's disease. It highlighted the fundamental complexity of a prompt and correct diagnosis, a serious unmet need in healthcare. Convolutional neural networks (CNNs) and their layers were explored in Chapter 2 before a methodological foundation was laid in Chapter 3 with the introduction of the dataset, CNN models (ResNet50V2 and VGG16), and optimizers (Adam, SGD, RMSprop). Finally, Chapter 4 demonstrated the probable effect of our machine learning-based strategy through presenting the real application, results, and a comparison with current methodologies. Based on the obtained findings, it is evident that the utilization of the Custom Architecture with Adam Optimizer is the most optimal selection for this particular dataset, as it has demonstrated the maximum level of accuracy and the lowest degree of loss. The utilization of a customized architecture in conjunction with the Adam optimizer has demonstrated significant potential. However, attaining and sustaining its excellent performance may necessitate meticulous tuning and monitoring. By incorporating these suggested measures, one can effectively augment the precision, resilience, and versatility of the model, hence expanding its utility across a broader spectrum of jobs and datasets. Overall, this study gives hope for improving Alzheimer's disease diagnosis with the aim of earlier intervention and superior results for patients, rendering a major improvement to medical procedures.

5.2 Future Work:

To get started, a Graphical User Interface (GUI) that is pleasant to users may be assembled particularly for doctors or other healthcare professionals. The following is going to guarantee that the prototype is straightforward to use and is adaptable. Throughout the utilization of this user interface, they will be capable to acquire instantaneous outcomes from the images they've chosen to deliver with the simple strike of the upload button. This technique streamlines the whole process for healthcare professionals who may not have substantial technical skills in the field of AI and makes it simpler on them to do their duties.

The graphical user interface can potentially be easily integrated with machine learning methods, which leads to in accuracy levels that are equivalent to those of the trained model. This lessens the possibility that human error will occur. Especially compared to the application in its first iteration, the prototype proposes an alternative that appears to be less time-consuming while being more effective.

The capacity of the prototype to interpret MRI brain scan imagery is perhaps the most significant component of the machine. An MRI scan may be effortlessly uploaded by users, and the system will thereafter evaluate the image through juxtaposing it to the model's training data in the background whereas the upload is being processed. This method delivers instantaneous and legitimate insights, which in turn minimizes significantly the amount of time necessary for diagnosing.

It is possible to design the prototype without adding any unnecessary complexity. Simply said, it will need some basic patient information in order to compile a database for potential future healthcare applications for the same patient. Inputs such as patient details and an MRI scan will provide results such as "Non Demented," "Mild Demented," "Very Mild Demented," or "Moderate Demented," with further labels available as needed.

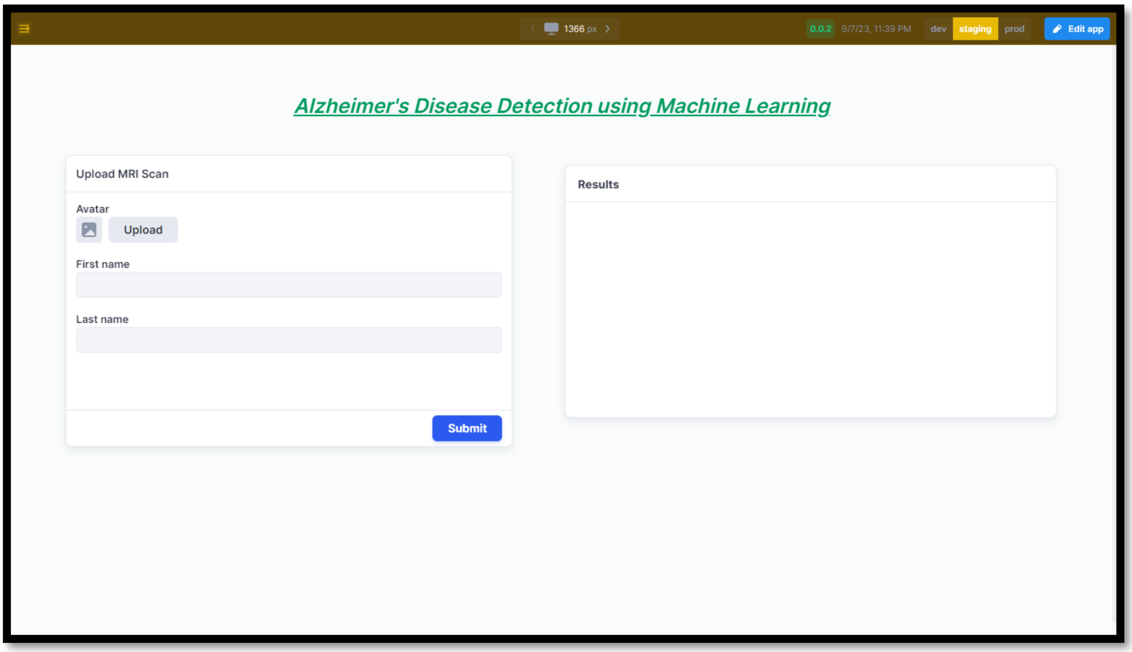


Figure 40 GUI for Prototype

The screenshot that was supplied demonstrates a basic graphical user interface idea along with offering a visual representation of what the finalized interface might look like. This user-friendly techniques strives to enhance the usability of the AI system in a medical setting, with an ultimate objective of expediting the procedure of diagnosing for healthcare practitioners.

References

Buckner, R. L. et al., 2000. Functional Brain Imaging of Young, Nondemented, and Demented Older Adults. *Journal of cognitive neuroscience,* Volume 12, pp. 24-34.

Caruana, R. & Niculescu-Mizil, A., 2006. An empirical comparison of supervised learning algorithms. *Proceedings of the 23rd international conference on Machine learning,* pp. 161-168.

Chawla, N. V., Lazarevic, A., Hall, L. O. & Bowyer, K. W., 2003. SMOTEBoost: Improving Prediction of the Minority Class in Boosting. *Knowledge Discovery in Databases: PKDD 2003. PKDD 2003.,* Volume 2838.

Colin L. Masters, R. B. K. B. C. C. R. R. A. S. &. J. L. C., 2015. Alzheimer's disease.

Cournapeau, D., 2011. *https://scikit-learn.org/stable/modules/classes.html.* [Online]
[Accessed 2023].

Davis, Goadrich, J. a. & Mark, 2006. The relationship between Precision-Recall and ROC curves. pp. 233-240.

Dennis, J. B. & Horn, E. C. V., 1966. Programming semantics for multiprogrammed computations. *Communications of the ACM,* pp. 143-155.

Gao, Y., Liu, W. & Lombardi, F., 2020. Design and Implementation of an Approximate Softmax Layer for Deep Neural Networks. *IEEE International Symposium on Circuits and Systems (ISCAS),* pp. 1-5.

Google, 2010. *https://www.kaggle.com/datasets/tourist55/alzheimers-dataset-4-class-of-images.* [Online]
Available at: https://www.kaggle.com/
[Accessed May 2023].

Goutte, C. & Gaussier, E., 2005. A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation. *Advances in Information Retrieval.,* Volume 3408.

Haji, S. H. & Abdulazeez, A. M., 2021. COMPARISON OF OPTIMIZATION TECHNIQUES BASED ON GRADIENT DESCENT ALGORITHM: A REVIEW. *PalArch's Journal of Archaeology of Egypt/Egyptology,* 18(4), pp. 2715-2743.

He, K., Zhang, X., Ren, S. & Sun, J., 2016. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 770-778.

He, K., Zhang, X., Ren, S. & Sun, J., n.d. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition,* pp. 770-778.

Herrera, L. J. et al., 2013. Classification of MRI Images for Alzheimer's Disease Detection. *International Conference on Social Computing,* pp. 846-851.

He, T. et al., 2019. Bag of Tricks for Image Classification with Convolutional Neural Networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 558-567.

Hijazi, S., Kumar, R. & Rowen, C., 2015. Using Convolutional Neural Networks. pp. 1-12.

Hinton, G. E. et al., 2012. Improving neural networks by preventing co-adaptation of feature detectors.

Jack, C. R. et al., 2005. Brain atrophy rates predict subsequent clinical conversion in normal elderly and amnestic MCI. *Neurology,* 65(8), pp. 1227-1231.

J, H. et al., 2022. Detection of Diseases Using Machine Learning Image Recognition Technology in Artificial Intelligence. *Comput Intell Neurosci,* pp. 2-10.

Jupyter, n.d. *Installing Jupyter.* [Online]
Available at: https://jupyter.org/install
[Accessed 2023].

Kadhim, K. A. et al., 2023. Early Diagnosis of Alzheimer's Disease using Convolutional Neural Network-based MRI. *Malaysian Journal of Fundamental and Applied Sciences,* 19(3), pp. 362-368.

Kavitha1, C. et al., 2022. Early-Stage Alzheimer's Disease Prediction Using Machine Learning Models. *Frontiers in Public Health,* Volume 10, pp. 1-13.

Kingma, D. P. & Ba, J., 2014. Adam: A Method for Stochastic Optimization.

Kohavi, R., 1995. A Study of Cross-Validation and Bootstrap. 14(2), pp. 1137-1145.

Langerman, J. R. a. H., 2019. Alzheimer's Disease – Why We Need Early Diagnosis. *Degenerative Neurological and Neuromuscular Disease,* pp. 123-130.

Li, F., Cheng, D. & Liu, M., 2017. Alzheimer's disease classification based on combination of multi-model convolutional networks. *IEEE International Conference on Imaging Systems and Techniques (IST),* pp. 1-5.

Litjens, G. et al., 2017. A survey on deep learning in medical image analysis. *Medical image analysis,* Volume 42, pp. 60-88.

Luque, A., Carrasco, A., Martín, A. & Heras, A. d. l., 2019. The impact of class imbalance in classification performance metrics based on the binary confusion matrix. *Pattern Recognition,* Volume 91, pp. 216-231.

Mamun, et al., 2022. *Deep Learning Based Model for Alzheimer's Disease Detection Using Brain MRI Images.* New York: IEEE.

Neelaveni, Devasana, J. a. & M.S.Geetha, 2020. Alzheimer Disease Prediction using Machine Learning Algorithms. pp. 101-113.

Pennanen, C. et al., 2004. Hippocampus and entorhinal cortex in mild cognitive impairment and early AD. *Neurobiology of aging,* 25(3), pp. 303-310.

Ruder, S., 2016. An overview of gradient descent optimization algorithms. *arXiv preprint.*

Sarraf, S., DeSouza, D. D., Anderson, J. & Tofighi, G., 2016. DeepAD: Alzheimer's Disease Classification via Deep Convolutional Neural Networks using MRI and fMRI. pp. 2-32.

Shorten, C. & Khoshgoftaar, T. M., 2019. A survey on Image Data Augmentation. *Journal of big data,* 6(1), pp. 1-48.

Simonyan, K. & Zisserman, A., 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint,* pp. 1409-1556.

Srivastava, N. et al., 2014. Dropout: A Simple Way to Prevent Neural Networks from. *Journal of Machine Learning Research 15 (2014),* 15(1), pp. 1929-1958.

Team, G. B., n.d. *https://www.tensorflow.org/.* [Online]
Available at: https://www.tensorflow.org/
[Accessed July 2023].

Thirion, B. et al., 2011. Scikit-learn: Machine Learning in Python. *the Journal of machine Learning research,* Volume 12, pp. 2825-2830.

Whitwell, J. L. et al., 2007. 3D maps from multiple MRI illustrate changing atrophy patterns as subjects progress from mild cognitive impairment to Alzheimer's disease. 130(7), pp. 1777-1786.

Yamashita, et al., 2018. Convolutional neural networks: an overview and application in radiologyYamashita, R., Nishio, M., Do, R.K.G. et al.. pp. 611-629.