A Project for

Advanced-Data Science

(UCS622)

Submitted by:

Tanisha Maheshwary 102153037

Pranav Malhotra 102103817

Submitted to:

Dr. Arun Pundir Singh

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Thapar Institute of Engineering and Technology, Patiala

Session: January – May 2024

INDEX

ABSTRACT

The objective of this study is to analyze the efficacy and efficiency of convolutional neural networks (CNNs) in the detection of Alzheimer's disease. The study also involves a comparison of CPU and GPU architectures. Through an examination of Alzheimer's image datasets, we conduct a comprehensive analysis of training speed, accuracy, and resource utilization. In addition, we will be conducting a comparative analysis between NASNet, a contemporary neural network architecture renowned for its exceptional performance, conventional Convolutional Neural Networks (CNNs), and the VGG model on a Graphics Processing Unit (GPU). The results of our study indicate that utilizing GPU acceleration has a substantial positive impact on the performance and scalability of the model. Specifically, our findings suggest that NASNet shows great promise in improving the diagnosis and treatment of Alzheimer's disease.

INTRODUCTION

Alzheimer's disease (AD) is a progressive neurodegenerative disorder marked by the gradual deterioration of cognitive function and memory. It is the most prevalent cause of dementia, making up around 60-70% of all cases. Alzheimer's Disease primarily impacts older adults, usually manifesting after the age of 65, although there are cases of early-onset forms occurring in younger individuals.

More than 25 million people in the world today are affected by dementia, most suffering from Alzheimer's disease. In both developed and developing nations, Alzheimer's disease has had tremendous impact on the affected individuals, caregivers, and society. In the year 2019, India was estimated to have ~3.69 million active cases of AD and other dementias.

Alzheimer's disease is characterised by the accumulation of abnormal protein aggregates in the brain, specifically beta-amyloid plaques and tau tangles, as observed in medical research. This degeneration tends to occur predominantly in areas of the brain that are crucial for memory, reasoning, and language.

In the context of Alzheimer's disease, the term "demented" encompasses the gradual deterioration of cognitive function and the resulting impact on daily activities and overall well-being. It is crucial to acknowledge that Alzheimer's disease impacts individuals in unique ways, with the speed of advancement and the intensity of symptoms differing greatly among patients.

The progression of Alzheimer's disease is commonly divided into four stages (very mild demented, mild demented, moderate demented and non demented), which are determined by the extent of symptoms and functional decline.

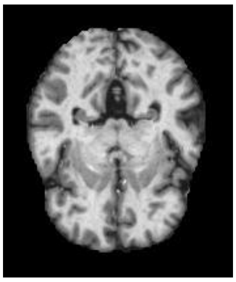


Fig 1: Very Mild Demented

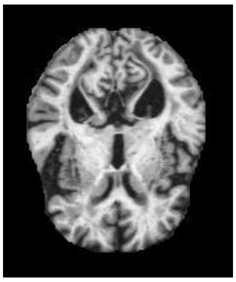


Fig 2: Mild Demented

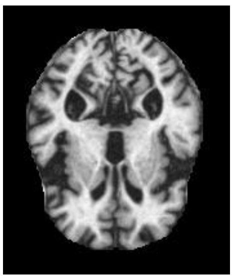


Fig 3: Moderate Demented

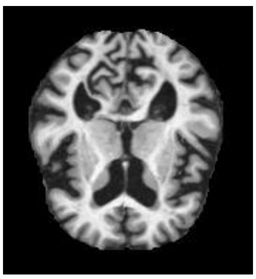


Fig 4: Non-Demented

In the initial phase of Alzheimer's disease, individuals may undergo subtle cognitive changes that are frequently disregarded or attributed to typical ageing. Common symptoms can manifest as sporadic memory lapses, challenges in word retrieval, and mild instances of forgetfulness.

In the mild stage of Alzheimer's disease, it is observed that some individuals may encounter challenges in areas such as memory, language, and problem-solving abilities.

In the moderate stage of Alzheimer's disease, there is a notable decline in cognitive function, with individuals facing significant impairments in memory, reasoning, and judgement. Individuals experiencing cognitive decline may struggle with recognising familiar faces and places, display wandering behaviour, and rely more heavily on others for their daily care.

In the advanced stage of Alzheimer's disease, cognitive decline becomes severe, leading to a loss of effective communication and the inability to perform basic tasks independently. Individuals may experience a decline in mobility, loss of bodily function control, and necessitate continuous care.

Problem Description: It is evident that accurately diagnosing Alzheimer's disease poses a significant challenge, highlighting the urgent need for the development of more advanced and reliable diagnostic techniques. Recent findings suggest that machine learning, particularly convolutional neural networks (CNNs), holds promise in automating the detection of Alzheimer's disease (AD) through the analysis of medical imaging data. Convolutional neural networks (CNNs) possess the capability to enhance diagnostic precision and enable prompt intervention through the analysis of patterns and characteristics in brain images. One possible approach is to utilise Convolutional Neural Networks (CNNs).

Obstacles and difficulties: However, the detection of AD using CNNs presents several challenges. One of the major challenges that needs to be addressed is the computational complexity associated with training and testing deep learning models on large-scale medical imaging datasets. Moreover, for CNN-based models to be applicable in clinical settings, it is crucial to ensure their robustness and interpretability. Furthermore, the inclusion of computational resources, such as graphics processing units (GPUs), and the careful optimisation of model designs are crucial aspects to consider when developing effective diagnostic systems for Alzheimer's disease.

Novelty: It refers to the uniqueness of the idea presented in a research paper. In this research paper, we touch on 2 topics, 1st image classification for Alzheimer’s prediction via CNN models one done with CPU and one done with GPU, 2nd a comparison of accuracy in a normal CNN model, a VGG, and a NASNet model in terms of accuracy, loss, val_loss, val_accuracy, all in GPU.

Another goal of this research paper was to detect Alzheimer's before it properly appears in a person’s brain and takes hold of it.

Not many papers have been published for image classification of Alzheimer’s classification using CNN and other various models.

LITERATURE SURVEY

S.NO	Author Name	Paper Name	Algorithm Name	Result Output
1	Karen Simonyan, Andrew Zisserman	Very Deep Convolutional Networks for Large-Scale Image Recognition	Visual Geometry Group (VGG)	This is an approach to handling large-scale image data
2	Google Brain	NASNet — Neural Architecture Search Network (Image Classification)	NASNet	This is an approach for designing automated neural network architectures. It is advisable to refrain from relying on intuition or prior knowledge when it comes to manual architecture design.
3	Aryan Ganesh and Ganesh Vanamu	A novel approach for early detection of Alzheimer's disease using deep neural networks with magnetic resonance imaging	Resnet-50, VGG-16, DenseNet-169	This study has conducted an examination of various algorithmic base models and evaluated their respective performances. The results indicate that VGG-16 exhibited superior performance in this context.
4	C H. S. C. A. Rama Ganesh, G. Sri Nithin, S. Akshay, T. Venkat Narayana Rao	Multi class Alzheimer disease detection using deep learning techniques	VGG-16, Inception-V3, Xception	This study has conducted an examination of various algorithmic base models and evaluated their respective performances. The results indicate that VGG-16 exhibited superior performance in this context.
5	Gongbo Liang; Xin Xing; Liangliang Liu; Yu Zhang; Qi Ying; Ai-Ling Lin; Nathan Jacobs	Alzheimer's Disease Classification Using 2D Convolutional Neural Networks	2D CNN Models	This study has proposed to use 2D CNN models combined with different temporal pooling strategies for the Alzheimer's disease diagnosis. Compared with the ResNet-based 3D CNN approach, the proposed method is able to improve the classification performance by 8.33%. In addition, the proposed methods reduce the training time up to 89%, from 65 hours to 7 hours.

METHODOLOGY

The 1st and 2nd notebook uses the TensorFlow library to import and utilize keras.layers, keras.models, keras.optimizers.Adam, keras.losses, and keras.preprocessing.image.ImageDataGenerator. train_set and test_set are generated using the train_dir and test_dir and TensorFlow. Keras library.

Then the basic model in both the above-mentioned notebooks is a Sequential model with 3 CNN layers 1 flatten layer, and 2 Dense layers.

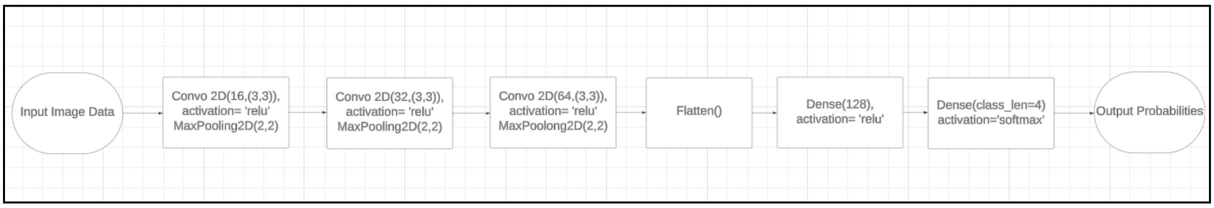


Fig 5: Data Flow Diagram (DFD)

The CNN layers include 2 functions Convo2D and MaxPooling2D.

The 1st CNN layer has 16 kernels, an activation function of 'ReLU' and kernel dimensions 3x3, which are then reduced to 2x2 with MaxPooliing2D, similarly the other 2 layers contain 32 and 64 kernels with dimensions 3x3 and an activation function of 'ReLU', the dimensions are reduced from 3x3 to 2x2 using MaxPooling2D

The next layer is the Flattten layer which further reduces the dimensions by flattening the 2D Matrix formed after the 3 CNN layers.

The 2 Dense layers are also used in the model, the 1st dense layer has 128 output classes and the 2nd dense layer has class len(4 according to our datasets) amount of layers.

```
model = Sequential([
  # Input layer for the image (replace channels with your data's format)
  keras.layers.Input(shape=(img_height, img_width, 3)),

  # Feature extraction with CNN blocks
  Conv2D(16, (3, 3), activation='relu', padding='same'),
  MaxPooling2D(pool_size=(2, 2)),
  Conv2D(32, (3, 3), activation='relu', padding='same'),
  MaxPooling2D(pool_size=(2, 2)),
  Conv2D(64, (3, 3), activation='relu', padding='same'),
  MaxPooling2D(pool_size=(2, 2)),

  # Flatten the features before dense layers
  Flatten(),

  # Dense layers with activation for classification (adapt class_len to your data)
  Dense(128, activation='relu'),
  Dense(class_len, activation='softmax')
])
```

Fig 6: Model built for simple CNN

GPU_MODELS (NASNET):

NASNet is a type of CNN model discovered through neural architecture search. The building blocks consist of normal and reduction cells.

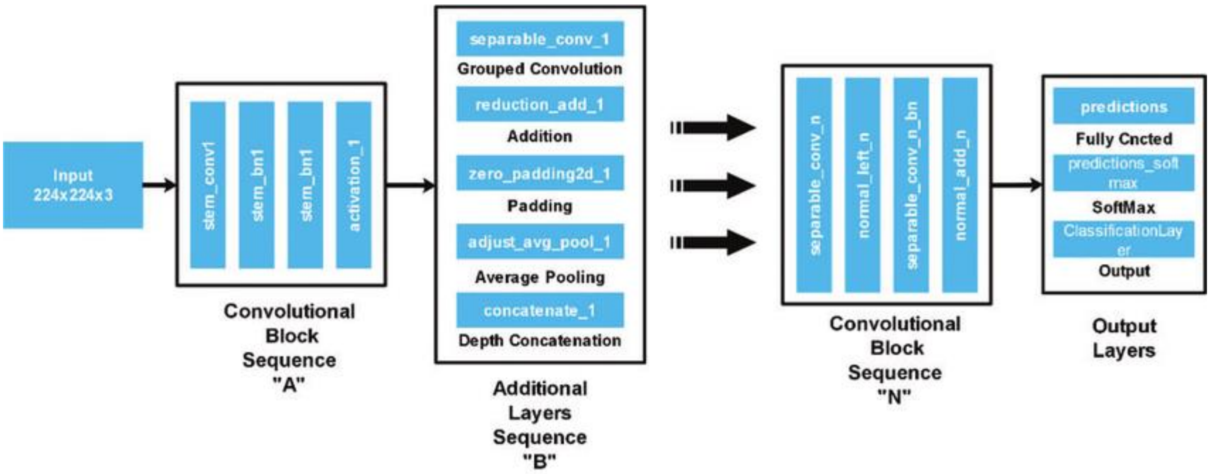


Fig 7: NASNet model architecture

- It is a method used to automate the process of designing of neural network architecture. NAS automatically searches for architectures that perform well on a specific task or dataset, instead of manually designing architectures based on intuition or prior knowledge.
- In NASNet, the architecture design space consists of a set of building blocks (e.g., convolutional layers, pooling layers, skip connections) and rules governing how these blocks can be connected.
- NASNet typically employs reinforcement learning, evolutionary algorithms, or gradient-based methods to search for optimal architectures.
- During the search process, architectures are evaluated based on their performance on a validation dataset. The results are used to update the search strategy and help the exploration towards more better architectures.
- NASNet adopts a cell-based architecture, where each cell represents a repeating pattern of operations. The architecture search focuses on finding optimal cell structures that can be stacked together to form a complete network architecture.

```python
nasnet_model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
accuracy_list_n = []
val_accuracy_list_n = []
loss_list_n = []
val_loss_list_n = []
time_list_n=[]

for epoch in range(2):
    start_time_n  = time.time()
    nasnet_history = nasnet_model.fit(train_generator, epochs=10, validation_data=test_generator)
    end_time_n = time.time()
    epoch_time_n =  end_time_n - start_time_n
    time_list_n.append(epoch_time_n)
    accuracy_list_n.append(nasnet_history.history['accuracy'][0])
    val_accuracy_list_n.append(nasnet_history.history['val_accuracy'][0])
    loss_list_n.append(nasnet_history.history['loss'][0])
    val_loss_list_n.append(nasnet_history.history['val_loss'][0])
```

Fig 8: NASNet model

The model of 'nasnet' that was used in the training of the Alzheimer's image dataset, uses Adam as an optimizer, categorical_crossentropy as a loss function, and accuracy as the measuring metric.

```python
# Define NASNet model
base_nasnet = NASNetLarge(include_top=False, weights='imagenet', input_shape=(img_height, img_width, 3))
x = GlobalAveragePooling2D()(base_nasnet.output)
output_nasnet = Dense(4, activation='softmax')(x)
nasnet_model = Model(inputs=base_nasnet.input, outputs=output_nasnet)
```

Fig 9: Building NASNet model

This NASNet model has a layer of GlobalAveragePooling2D, a Dense layer of 4 output classes for Alzheimer's Dataset.

GPU_MODELS(VGG):

The VGG model is a convolutional neural network architecture proposed by the Visual Geometry Group at Oxford University.

- The VGG architecture is characterized by its simplicity and depth. It consists of multiple convolutional layers stacked on top of each other, followed by fully connected layers.
- The key idea behind VGG is to use a series of smaller convolutional filters (e.g., 3x3) with a small stride and the same padding, rather than larger filters, to capture features at different scales and maintain a high-resolution feature map.
- The architecture of the basic VGG model is much like the model that we built on our own.
- After several convolutional and pooling layers, the VGG architecture typically includes one or more fully connected layers.
- These fully connected layers aggregate the information from the convolutional layers and perform classification based on the learned features.
- The final layer usually employs a softmax activation function to produce probability scores for each class in a classification task.
- The VGG model primarily uses the ReLU activation function after each convolutional and fully connected layer, except for the output layer where softmax is used for classification tasks.
- We use a pre-trained VGG model named ImageNet for our Alzheimer's classification.

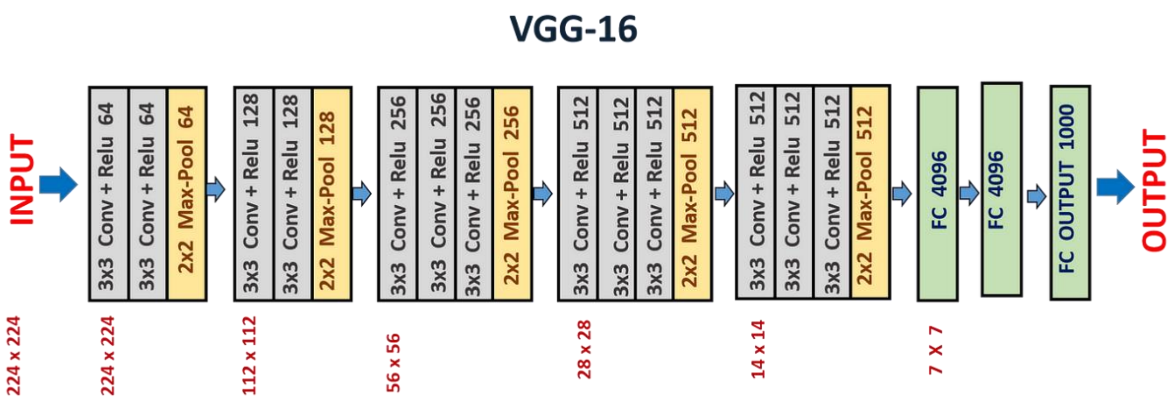


Fig 10: VGG Layers

```python
from tensorflow.keras.applications import VGG16
```

```python
# Freeze the pre-trained layers
for layer in base_model.layers:
    layer.trainable = False

# Add new classification layers on top
model_g = Sequential([
    base_model,
    Flatten(),
    Dense(512, activation='relu'),
    Dense(4, activation='softmax')  # Assuming 4 output classes, change accordingly
])

# Compile the model
model_g.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

```python
time_list_g = []
accuracy_list_g = []
val_accuracy_list_g = []
loss_list_g = []
val_loss_list_g = []
# Train the model
for epoch in range(20):
    start_time_g = time.time()
    history_g = model_g.fit(train_generator, steps_per_epoch=len(train_generator), epochs=1,
                          validation_data=test_generator, validation_steps=len(test_generator))
    end_time_g = time.time()
    epoch_time_g = end_time_g - start_time_g    # Append epoch time to the list
    time_list_g.append(epoch_time_g)
    accuracy_list_g.append(history_g.history['accuracy'][0])
    val_accuracy_list_g.append(history_g.history['val_accuracy'][0])
    loss_list_g.append(history_g.history['loss'][0])
    val_loss_list_g.append(history_g.history['val_loss'][0])
```

Fig 11: VGG model

RESULTS AND ANALYSIS

Dataset Analysis:

The MRI image dataset has been collected from the OASIS dataset and has been preprocessed beforehand. It has 2 folders – test and train each with 4 categories as the follows:

Dataset images	veryMildDemented	MildDemented	ModerateDemented	NonDemented	Data size
Train	1792 files	912 files	67 files	2350 files	5121 files (26.8 MB)
Test	410 files	215 files	14 files	640 files	1279 files (6.19 MB)

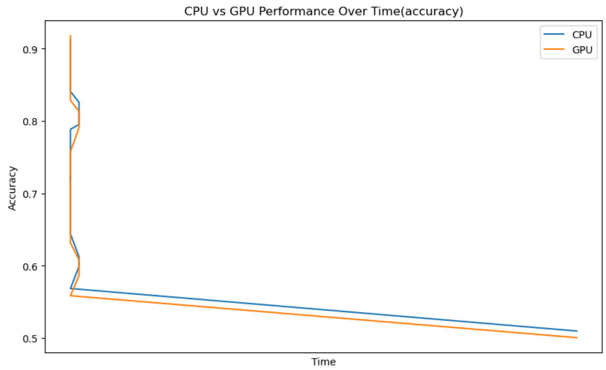
1. *Statistical Analysis:*
    - Train Data Size: 26.8 MB
    - Test Data Size: 6.19 MB

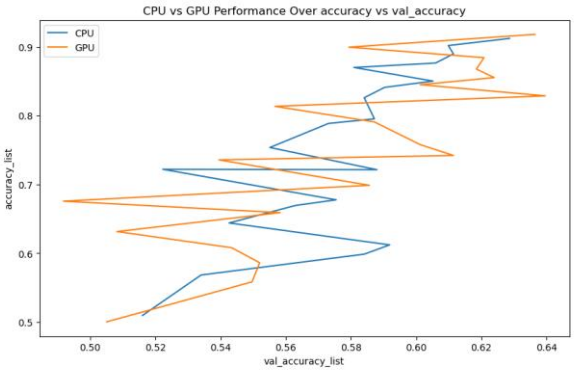
2. *Average File Size (Train):*
    - veryMildDemented: Approximately 5.27 KB per file
    - MildDemented: Approximately 4.17 KB per file
    - ModerateDemented: Approximately 4.18 KB per file
    - NonDemented: Approximately 5.67 KB per file

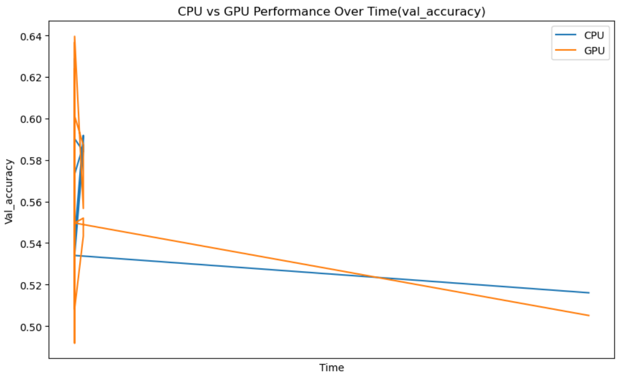
3. *Average File Size (Test):*
    - veryMildDemented: Approximately 5.29 KB per file
    - MildDemented: Approximately 4.16 KB per file
    - ModerateDemented: Approximately 4.27 KB per file
    - NonDemented: Approximately 4.83 KB per file

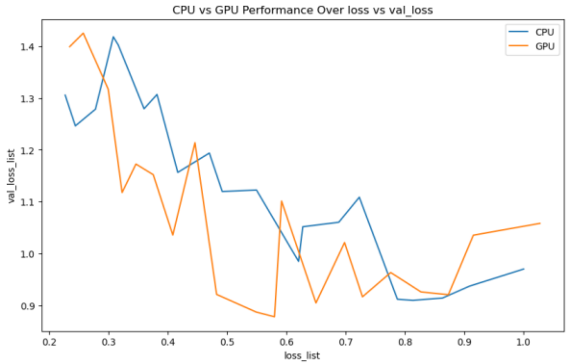
Results:

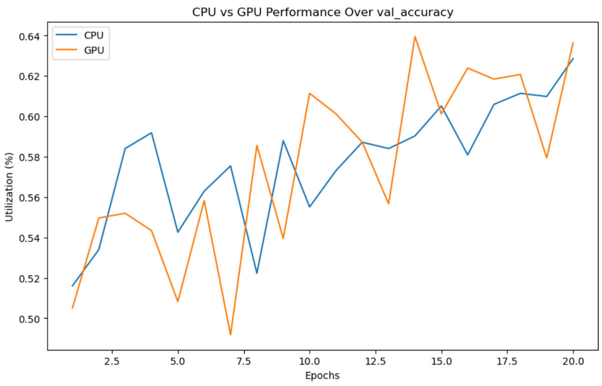
Using TensorFlow's image generator, we efficiently organised our dataset into separate test and train directories, assigning a unique number to each image. Our study entailed a thorough examination of the performance metrics of a CNN model on both CPU and GPU architectures, specifically focusing on accuracy. It is worth mentioning that GPU computation consistently outperformed CPU in multiple parameters, showcasing its superior efficiency and effectiveness.

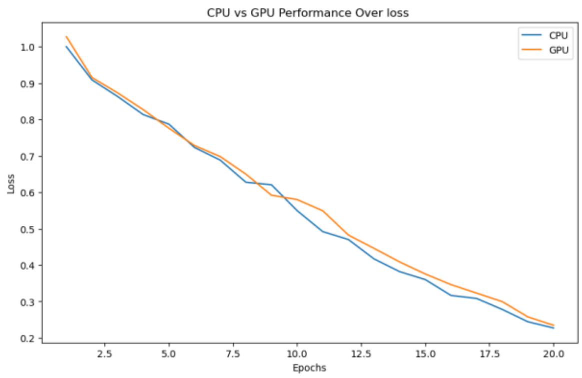


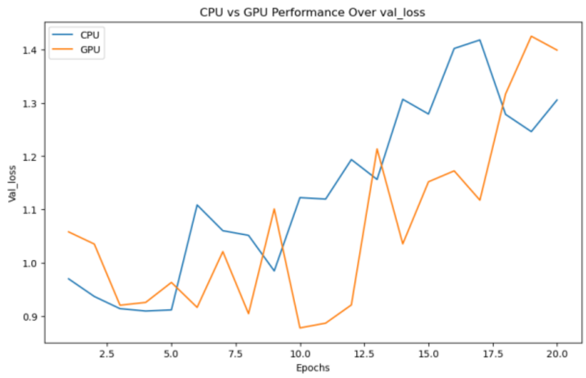


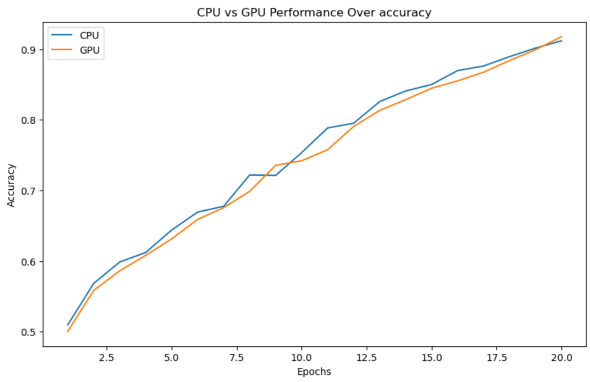












Model Name	CNN on CPU	CNN on GPU
Accuracy	0.91193	**0.91783**
Loss	**0.22706**	0.23466
Val_Accuracy	0.62861	**0.63643**
Val_Loss	0.90983	**0.87813**
Total time required for 20 epochs	19.31 min	**12.15 min**

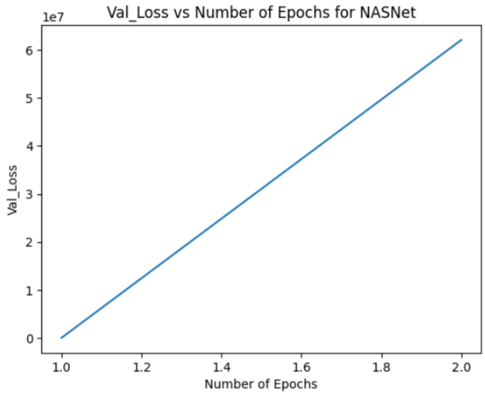
Accuracy - Greater precision values are indicative of superior performance. *Loss* - Also referred to as the error or cost, serves as an indicator of your model's performance during training. Lower loss values indicate a stronger alignment between predictions and ground truth labels.

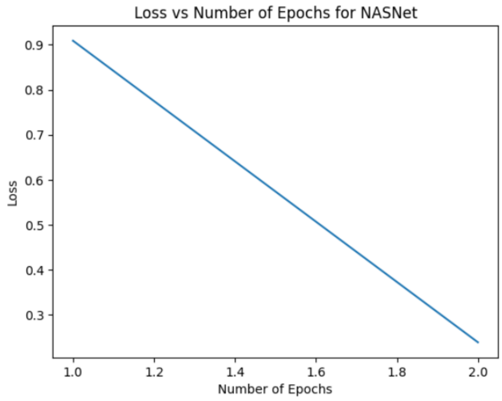
Validation accuracy - also known as val_accuracy, refers to the accuracy of your model when evaluated on a separate validation dataset.

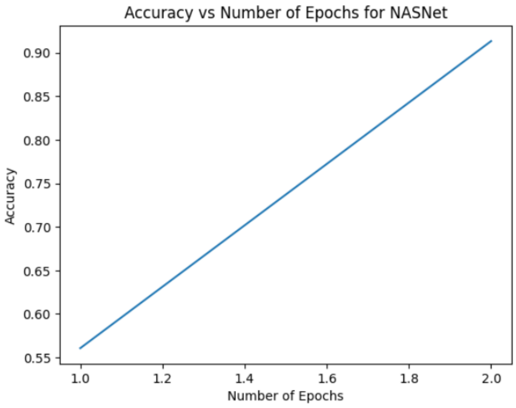
Validation loss - also known as val_loss, represents the loss of your model when evaluated on the validation dataset. Smaller values of val_loss are indicative of superior generalisation performance.

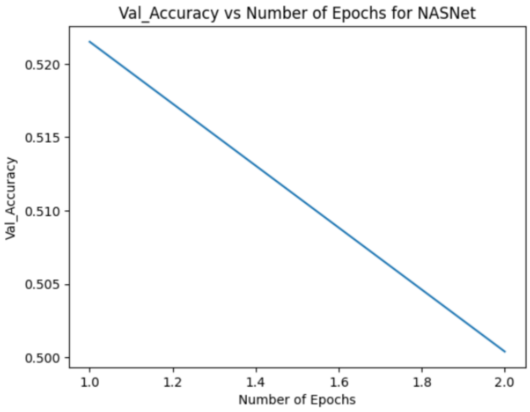
Furthermore, our investigation was expanded through the utilisation of two cutting-edge architectures, VGG and NASNet, with the added advantage of GPU acceleration, enhancing the training of our dataset. It is worth noting that NASNet demonstrated exceptional performance, displaying a consistent and superior trajectory when compared to the other models that were assessed.

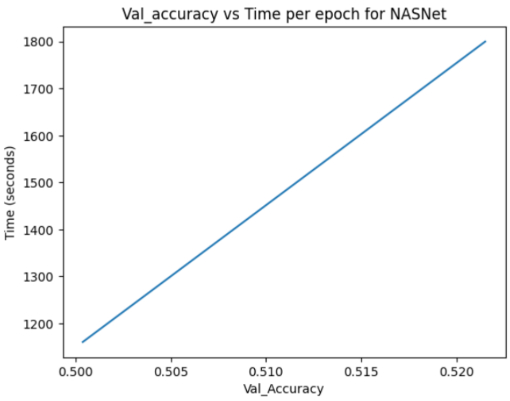
Graphs of NASNet model:

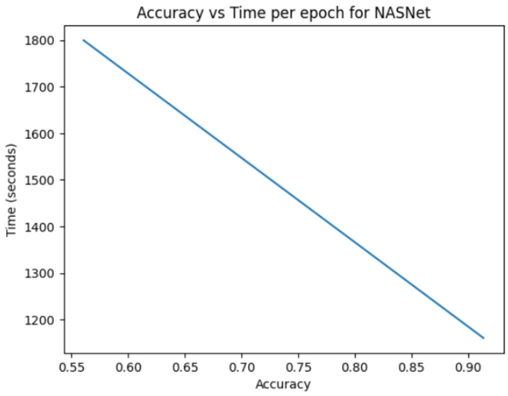


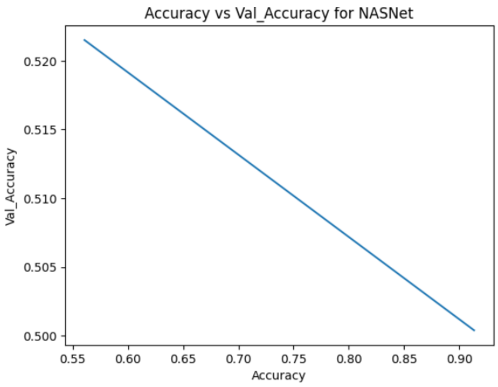


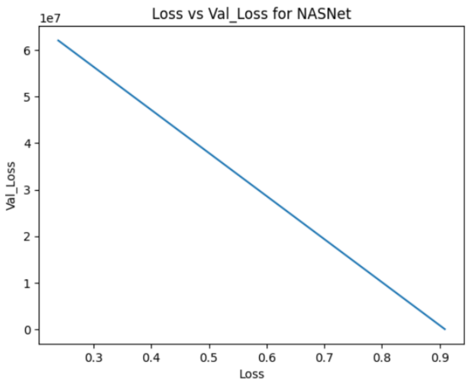




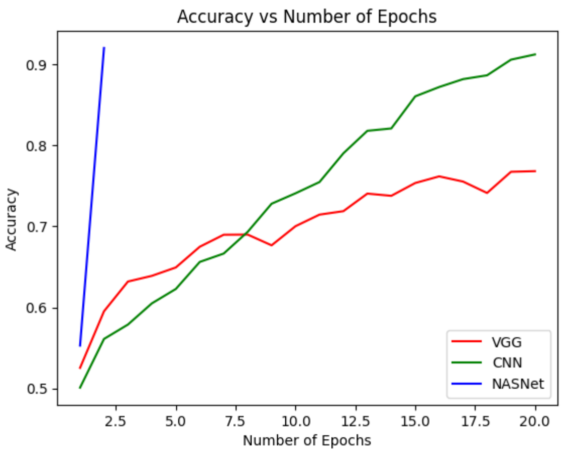


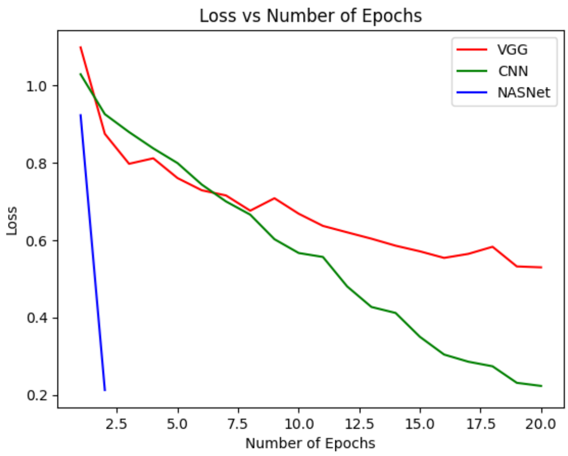


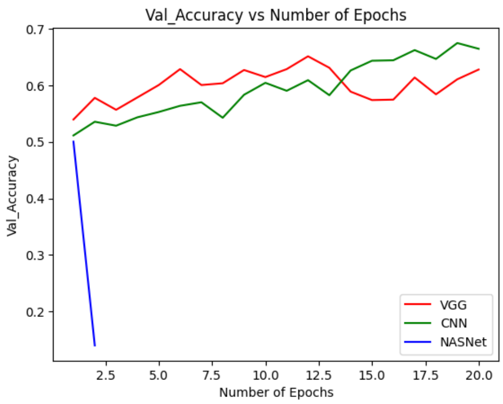




Comparative Study of all models:







The *results of this study emphasise* the potential of NASNet as a reliable solution for our task, demonstrating its effectiveness in improving the diagnostic capabilities for detecting Alzheimer's disease with an accuracy of 96 % surpassing VGG and traditional CNN.

Model Name	**Accuracy**	**Loss**	**Val_Accuracy**	**Val_Loss**	**Total time (for 20 epochs)**
CNN	0.9153	0.2166	**0.6607**	0.8843	14.1 min
VGG	0.7340	0.6083	0.6263	**0.8750**	14.6 min
NASNet	**0.9683**	**0.0950**	0.5376	204.7396	49.2 min

Error Analysis - Errors have been encountered by our model as a result of the intricate nature of the VGG and NASNet models in relation to our dataset. As a result of the scarcity of GPUs and the need to work within a tight schedule, we were only able to train the models for a limited number of epochs.

CONCLUSION

This study has examined the effectiveness and efficiency of convolutional neural networks (CNNs) in Alzheimer's disease detection. A comparison of CPU and GPU architectures is an additional component of the study. A comprehensive analysis is performed on training speed, accuracy, and resource utilisation by means of an examination of Alzheimer's image datasets. Furthermore, we have performed a thorough comparison between NASNet, a cutting-edge neural network architecture known for its outstanding performance, traditional Convolutional Neural Networks (CNNs), and the VGG model on a Graphics Processing Unit (GPU). Our study findings suggest that the utilisation of GPU acceleration significantly enhances the performance and scalability of the model.

The study highlights NASNet's potential as a dependable solution for our purpose by showcasing its efficacy in enhancing diagnostic capacities, outperforming VGG and conventional CNN in diagnosing Alzheimer's disease with an accuracy of 96%.

FUTURE SCOPE

Exploring Transfer Learning: This study aims to examine the effectiveness of transfer learning techniques in adapting pre-trained models for improved detection of Alzheimer's disease. The focus will be on scenarios where there is a scarcity of labelled data.
Multi-Modal Data Integration: The incorporation of additional data modalities, such as genetic markers and cognitive assessments, alongside medical imaging, can enhance diagnostic accuracy and deepen our understanding of Alzheimer's disease pathology.
Scalability and Resource Optimisation: Enhance model architectures and training procedures to achieve scalability and resource efficiency, facilitating deployment on edge devices for point-of-care diagnosis. This can be accomplished through techniques such as model compression and distributed training.

REFERENCES

1. https://ieeexplore.ieee.org/document/9765267
2. https://niscpr.res.in/includes/images/bulletin/bulletin-2023-03-15.pdf
3. https://emerginginvestigators.org/articles/21-053/pdf
4. https://huggingface.co/docs/timm/en/models/nasnet
5. https://sh-tsang.medium.com/review-nasnet-neural-architecture-search-network-image-classification-23139ea0425d
6. https://arxiv.org/abs/1409.1556
7. https://alz-journals.onlinelibrary.wiley.com/doi/10.1002/alz.13016
8. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3181909/#:~:text=More%20than%2025%20million%20people,individuals%2C%20caregivers%2C%20and%20society.
9. https://www.youtube.com/watch?v=2vNsoQQ_aOs
10. https://www.youtube.com/watch?v=X8WsCy6m_Bg
11. https://www.youtube.com/watch?v=Cym0uDqzigs
12. https://www.youtube.com/watch?v=VMuUTqDZq-0
13. https://www.youtube.com/watch?v=mjk4vDYOwq0
14. https://www.youtube.com/watch?v=j05JpWO85_o
15. https://www.youtube.com/watch?v=P9XkNpP3ikU