Deep Learning for Alzheimer's Stage Classification: A CNN-Based Approach with Transfer Learning and Data Augmentation

Manuel Hurtado Jr. [1, 2, †]

1 Department of Psychiatry and Behavioral Sciences, Feinberg School of Medicine, Northwestern University

2 https://www.github.com/mhi573

† mhurtado.jr0@gmail.com

Abstract

Alzheimer's dementia (AD), a neurodegenerative disorder, can be diagnosed using Magnetic Resonance Imaging (MRI) scans, but manual analysis of MRI scans is time-consuming and subject to variability among clinicians. Deep learning, specifically Convolutional Neural Networks (CNNs), offers a promising solution for assisting clinicians or automating AD stage classification. This study explores CNN-based classification of AD stages (no dementia, very mild dementia, mild dementia, or moderate dementia) using MRI scan images. The primary goal was to compare the performance of a CNN that was developed and fine-tuned to pre-trained models (InceptionV3, VGG16, and EfficientNetB1). The secondary goal was to explore the impact of data augmentation on performance and generalization. Results showed that transfer learning is a viable option for the utilization of deep learning CNNs as a diagnostic tool for Alzheimer's disease stages. Further, data augmentation may be a technique that improves the performance of pretrained models. Future work should focus on how to implement the use of pre-trained CNNs in medical settings to assist clinicians in their decision-making.

Keywords: CNN, Alzheimer's Disease, MRI, Transfer Learning, Hyperparameter Tuning, Data Augmentation, InceptionV3, VGG16, EfficientNetB1, Classification

Table of Contents

Introduction and Problem Statement

Accurate diagnosis of Alzheimer's is crucial for managing the disease and developing treatment plans - there is no cure. Deep learning, in particular a convolutional neural network (CNN or CNNs), has shown promising results in accurate medical image classification and automating the detection of complex patterns within Magnetic Resonance Imaging (MRI) scans.

Alzheimer's dementia (AD) is a neurodegenerative disorder that affects memory, cognition, and can affect daily functioning (Alzheimer's Association 2025). MRI scans are a widely used tool in Alzheimer's research and for Alzheimer's diagnosis. MRI scans typically enable clinicians the ability to rule out other diseases through the identification of structural brain changes associated with those diseases (Alzheimer's Association 2025). However, manual diagnosis from MRI scans is time-consuming, subjective, and prone to variability and error among trained clinicians (Despotović, Goossens, and Philips 2015). CNNs offer a state-of-the-art, high-performance method for diagnosing Alzheimer's disease, surpassing traditional manual diagnosis. Moreover, exploring the utility of this CNN implementation would benefit clinicians by providing further advice needed to make informed decisions and decide the best next step for their patients.

Convolutional neural networks can be developed using Python programming (Python Software Foundation 2025) and deep learning libraries such as tensorflow, keras, sklearn, and numpy. Existing CNNs should also be leveraged, also known as transfer learning, for this type of task. Using existing architectures may prove easier, cost-effective, or a better diagnostic tool than developing a CNN from scratch. In either case, the role of CNNs must be investigated further.

This study explored the use of CNNs for classifying MRI images into one of four Alzheimer's disease stages (no dementia, mild dementia, very mild dementia, moderate dementia). By comparing different CNN architectures and training strategies, this research aimed to provide insights into the optimal deep learning approach for Alzheimer's stage classification. Findings will contribute to the growing field of AI-assisted medical diagnostics, providing a foundation for further research and real-world applications. More importantly, the goal of this study is to suggest recommendations on the utility of implementing CNNs for Alzheimer's stage classification. Accomplishing these goals was completed with a 2x4 factorial design. CNN model performance (accuracy) was compared across a CNN that was developed to three pre-trained models (i.e., InceptionV3; EfficientNetB1; VGG16). Then, model performance was evaluated following the implementation of data augmentation.

The aims addressed in this study included:

1. Evaluating the performance of a baseline CNN model that was developed and optimized through hyperparameter tuning and architectural modifications
2. Exploring the effectiveness of transfer learning (i.e., InceptionV3; EfficientNetB1; VGG16) and replicating the results of published manuscripts that described implementing CNN transfer learning for the same classification task
3. Determining the impact of data augmentation on the performance of the created model compared to pre-trained CNN models.

Literature Review

A literature review was conducted to understand existing research about MRI classification using CNNs for identifying different stages of Alzheimer's disease. There has been a sizable amount of research investigating CNNs and transfer learning approaches for four-way multiclass classification of Alzheimer's stages. Overall, the literature suggests that transfer learning using pretrained models and data augmentation can achieve high levels of accuracy.

Ali et al. (2024) approached this task using two stages. In the first stage a 26-layer CNN was developed and refined to classify MRIs as either a healthy or dementia patient. Then the frozen weights of the CNN were used in stage 2 that consisted of transfer learning (ResNet50, InceptionV3, GoogleNet, EfficientNet-b0, DenseNet-201). In stage 2, the frozen weights were used "to fine-tune the new transfer-learned model by replacing the last three layers of the developed CNN for dementia subclassification (mild, moderate, and very mild dementia)". This approach achieved 98% accuracy in stage 1 and 99% accuracy in stage 2.

At least two groups of researchers (Kahn et al. 2023; Sharma et al., 2022) implemented transfer learning using VGG models for determining Alzheimer's dementia stages via MRI classification. One group (Kahn et al. 2023) first evaluated the performance of VGG16 and VGG19 followed by evaluating the impact of data augmentation on those models. Results showed performance levels for precision scores, accuracy, and F1-scores of at least 90% across for VGG16, and at least 86% for VGG19. For both pretrained models, nearly all metrics improved after implementing data augmentation. A separate study (Sharma et al., 2022) explored implementing VGG16 (without data augmentation) for four-way classification of MRI scans and also reported approximately 90% for accuracy, precision, recall, AUC, and F1-score.

Farooq and colleagues (2017) also explored applications of transfer learning for a four-way multiclass classification of Alzheimer's dementia stages using GoogLeNet, ResNet-18, and ResNet-152. Results showed high levels of overall accuracy (respectively, 98.88%, 98.01%, 98.14%) and outperformed methods described in other manuscripts.

Data

Data Characteristics

Brain MRI images were downloaded from a publicly available source on Kaggle (Rahman 2023). Two datasets were included, a train dataset and test dataset. The train dataset consisted of 5,120 images and the test dataset consisted of 1,280 images. All images were labeled into one of four categories: 1) no dementia; 2) mild dementia; 3) very mild dementia; and 4) moderate dementia. While there was sizeable representation across categories, both sets were biased toward those without dementia. No other data was provided by the publishing author.

The train dataset consisted of 5,120 images. Out of all the training data (*N*= 5120; see Figure 1) about half of the MRI images were labeled as not having dementia (50.12%, *n*= 2566), about one-third were labeled as 'very mild dementia' (34.78%, *n*= 2566), and less were labeled as 'mild dementia' (14.14%, *n*= 724) or 'moderate dementia' (0.96%, *n*= 49).

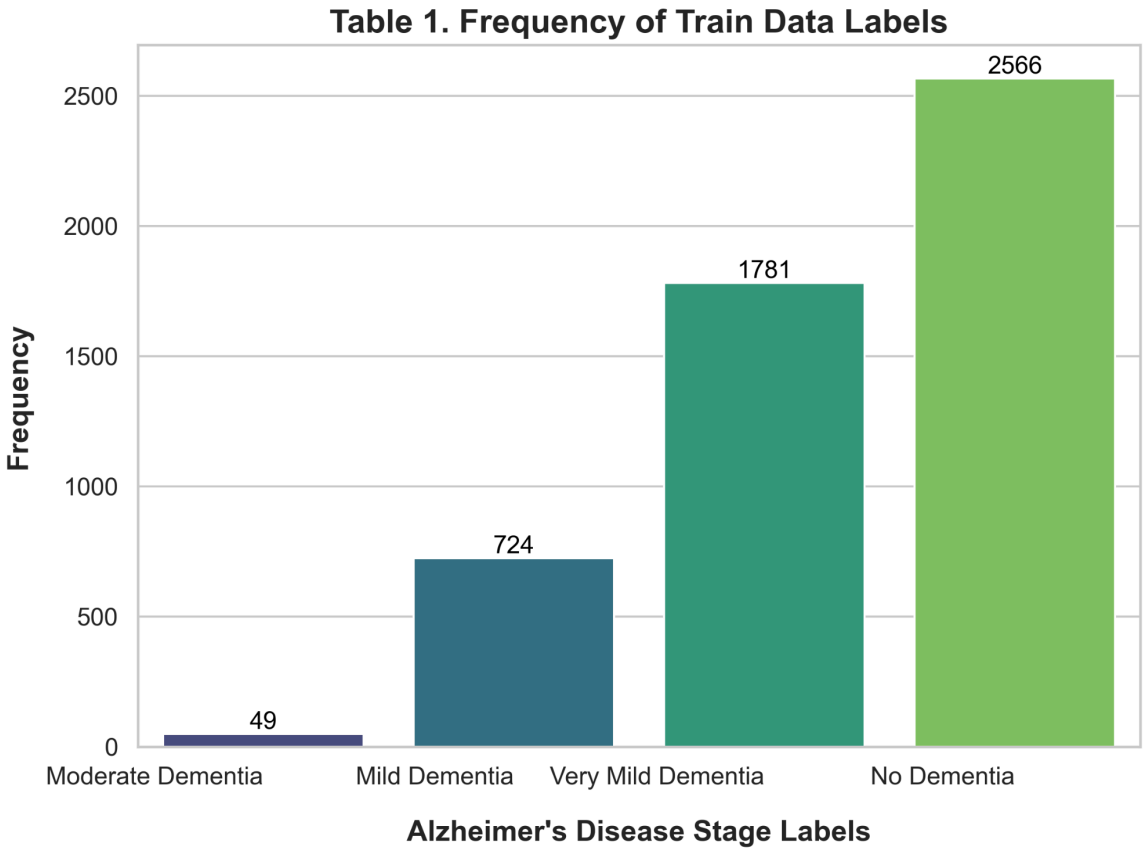

Figure 1: Frequency of Train Data Labels.

The test dataset consisted of 1,280 images. Out of all the testing data (*N*= 1280; see Figure 2) about half of the MRI images were labeled as not having dementia (49.53%, *n*= 634), over one-third were labeled as 'very mild dementia' (35.86%, *n*= 459), and less were labeled as 'mild dementia' (13.44%, *n*= 172) or 'moderate dementia' (1.17%, *n*= 15).

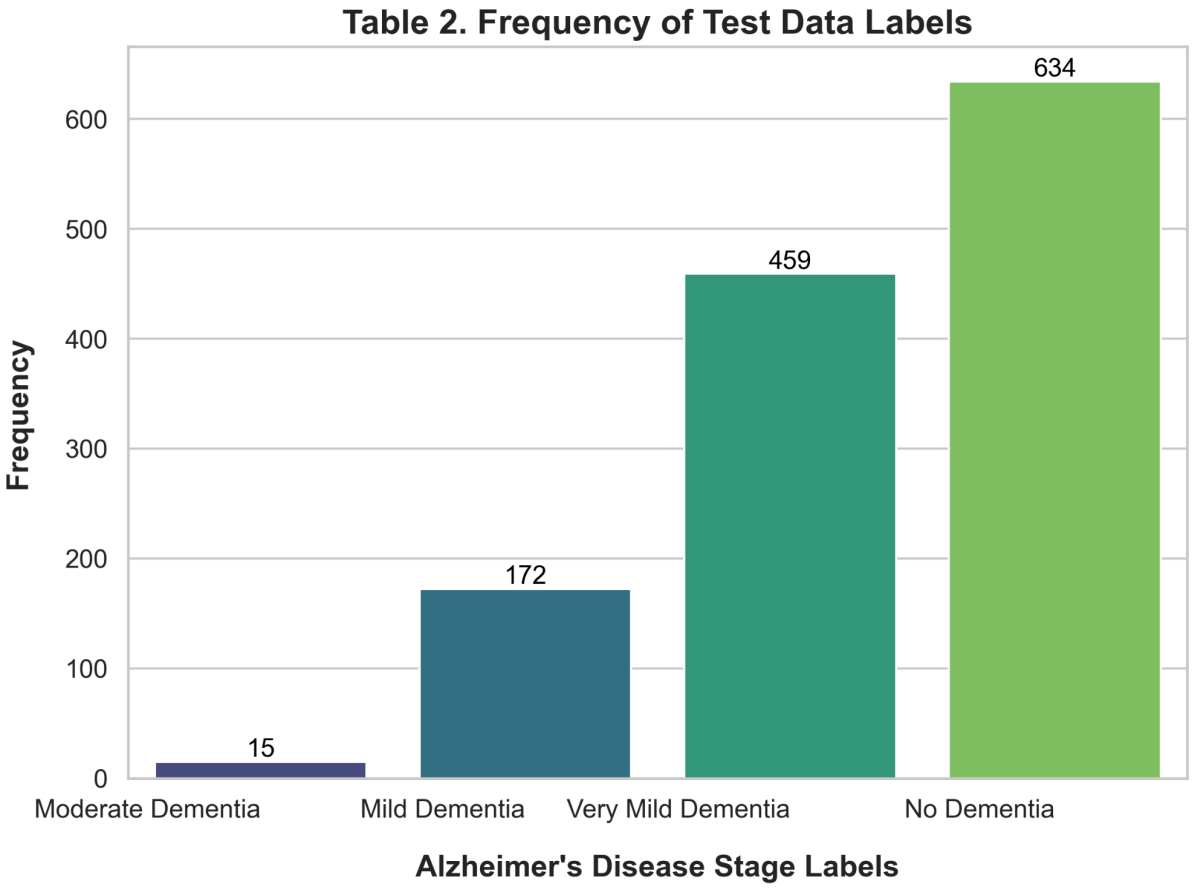

Figure 2: Frequency of Train Data Labels

Data Pre-processing

The downloaded data was provided in a Parquet-formatted file and read into a Jupyter Notebook (Project Jupyter 2025) using Python 3 (Python Software Foundation 2025). This project was conducted entirely in Python. The training dataset and test dataset contained two columns: 1) a classification value and 2) MRI image data. The MRI image data was provided as byte data. Levels of missingness were investigated and no data was missing. The byte data was converted to a NumPy array representing the pixel data of the grayscale MRI images, where pixels ranged from 0 to 255. Images were 128 pixels by 128 pixels in size with a single channel.

For this multiclass classification problem, the labels were one-hot encoded, and image pixel data were normalized for the baseline CNN and standardized for the pre-trained models. For InceptionV3, the grayscale channel value of '1' was converted to '3' to correspond to an RGB channel. Training data were then split into training and validation sets (70% and 30%, respectively). The random_state parameter for data splitting was set to '42' to ensure consistency across this study. The stratify parameter was also set to the one-hot encoded labels to maintain class distributions across training and validation splits.

Method

The study design was a 2x4 factorial design where test accuracy was evaluated for four CNNs (1, created CNN; 2, InceptionV3; 3, VGG16; 4, EfficientNetB1) first without applying data augmentation and then with data augmentation (see Figure 3). Accuracy and loss were the primary metrics that were evaluated to assess success, and precision and recall were also evaluated during training. Greater accuracy on the test data set was considered a success.

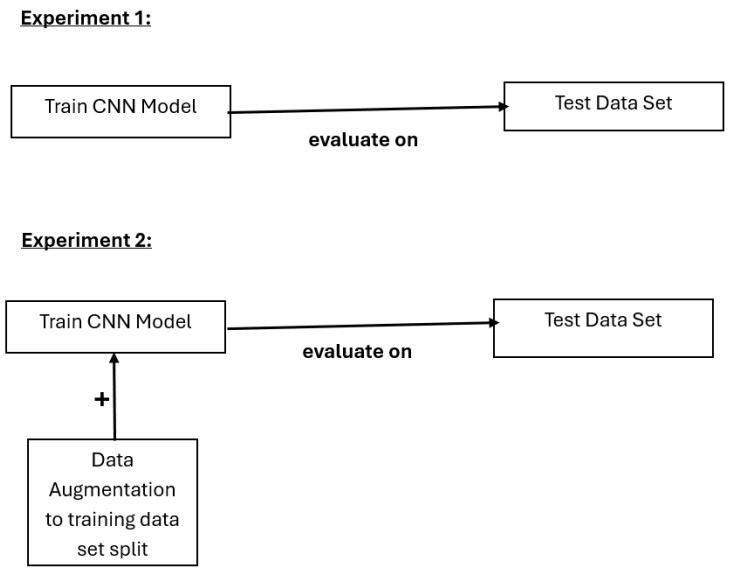

Figure 3: This study evaluated the performance (test accuracy) of four CNNs using a 2x4 factorial study design. One set of experiments were trained without data augmentation and the other with data augmentation.

A baseline CNN model was created, and fine-tuned via hyperparameter tuning, to compare against pre-trained models. The baseline model was optimized in a series of ablation experiments consisting of hyperparameter tuning and architectural modifications.

Hyperparameter tuning was conducted for the baseline model with Keras Tuner's RandomSearch (Chollet and contributors 2025). RandomSearch was chosen over GridSearch due to its lower computational cost. At this point, the hyperparameters that were optimized included the: 1) number of filters across convolutional blocks; 2) number of neurons in fully connected layers; 3) dropout rate; and 4) number of convolutional blocks.

To assess the impact of data augmentation on model training and performance, ImageDataGenerator from Keras (Chollet and contributors 2025) was applied exclusively to the training data, ensuring that the validation and test sets remained unchanged. It was determined that creating synthetic images utilizing a generative adversarial network may be an imprudent addition because a definitive label cannot be synthetically created. While the synthetic data may add to the size of the data, it may be a misrepresentation of the underlying pathology. Data augmentation consisted of the following adjustments: shifting width by 10%, shifting height by 10%, zoom range of 10%, and setting the filling strategy for empty pixels to 'nearest'.

Since this was a four-class classification problem, across all models, the final output layer used the softmax activation function, and the categorical cross-entropy was the ideal loss function. Model evaluation metrics included accuracy, loss, precision, and recall. To prevent overfitting, EarlyStopping and ModelCheckpoint were employed during model training. Early stopping monitored validation loss and training was stopped if no improvement was observed after five epochs. ModelCheckpoint monitored validation accuracy such that only models that improved in this metric were saved as the best model.

Transfer Learning

Transfer Learning was implemented, with and without data augmentation (i.e., separately) using the InceptionV3 CNN (Szegedy et al. 2016), EfficientNetB1 CNN (Tan and Le 2019), and VGG16 CNN (Simonyan and Zisserman 2014). The goal was to determine whether leveraging a pre-trained model enhanced classification performance for MRI-based multiclass classification of Alzheimer's disease stages. After initial implementation of transfer learning, data augmentation was implemented to evaluate its impact on performance.

Results

Baseline CNN Architecture

A baseline CNN model was developed and served as a benchmark for a series of experiments and ablation studies that aimed at identifying optimal choices for the hyperparameters and architectural design. The hyperparameters used for RandomSearch included

the optimization algorithm (Adam or SGD), activation function (Leaky Relu or Relu) learning rate (0.1, 0.01, 0.001, or 0.0001), momentum, batch size (32, 64, or 128), dropout rate (30% to 50%) and number of epochs. The 128 by 128 by 1 grayscale MRI images were processed through multiple convolutional layers to extract features.

RandomSearch was implemented first for a CNN with two convolutional blocks (followed by fully connected layers), then with three convolutional blocks, and with five convolutional blocks. Following RandomSearch, each model was trained on a training and validation set, and then the best model was evaluated on a test set. With regard to the metric of choice (test accuracy), the model with two convolutional blocks outperformed the three block model and five block model.

The architecture of this two block model consisted of two convolutional blocks followed by fully connected layers for four-way multiclass classification (see Figure 4). The first convolutional block applied 32 filters with 3×3 kernels, followed by batch normalization and 2×2 max pooling, reducing the spatial dimensions to 64×64×32. The second convolutional block consisted of two 3×3 convolutional layers with 128 filters each, each followed by batch normalization and 2×2 max pooling, further reducing the dimensions to 32×32×128. The output was then flattened and passed through two fully connected layers with 512 and 192 neurons that utilized the Leaky Relu activation function, each followed by 30% dropout for regularization. Finally, a softmax output layer with 4 neurons produced class probabilities for the four-way multi-class classification. The CNN was compiled with an Adam optimization algorithm, 0.0001 learning rate, categorical cross entropy loss function, 32 batch size, and trained for 8 epochs.

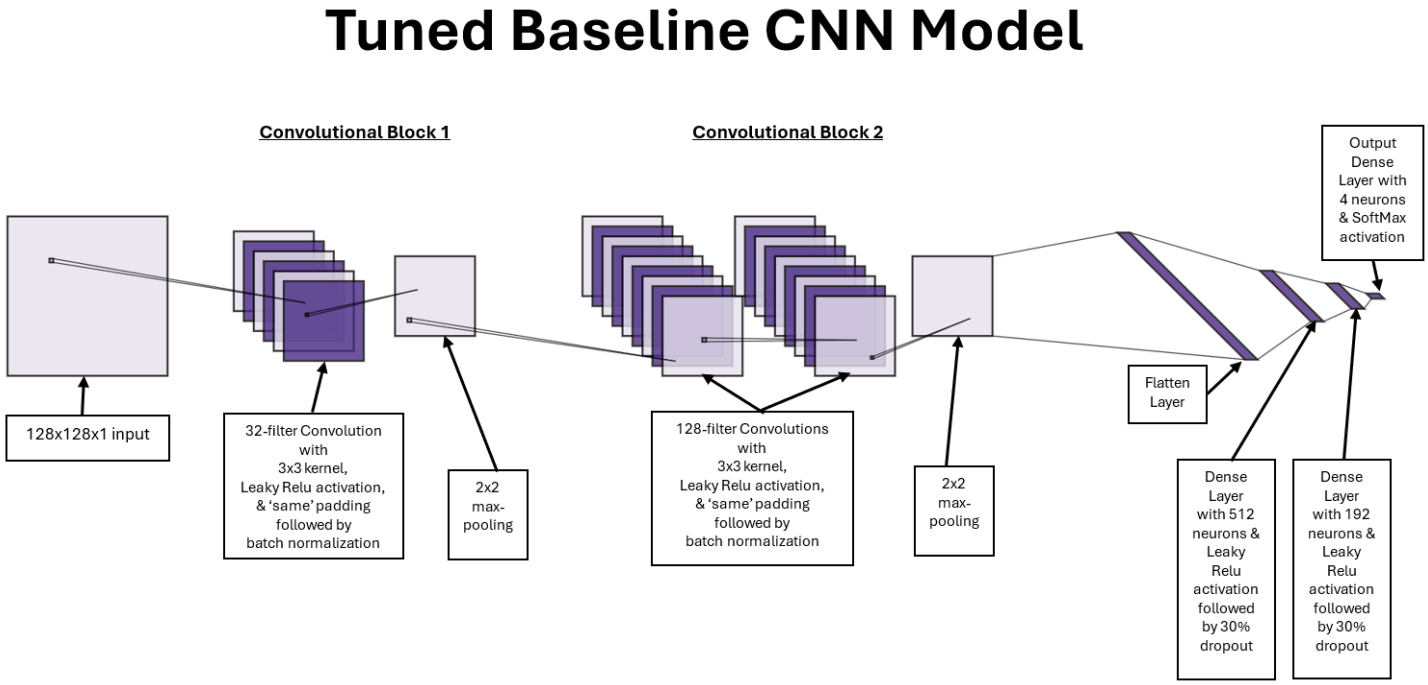

Figure 4: Tuned Baseline CNN Model compiled with an Adam optimization algorithm, 0.0001 learning rate, categorical cross entropy loss function, 32 batch size, and trained for 8 epochs.

Transfer Learning Architecture

Transfer Learning was implemented using the InceptionV3 CNN (Szegedy et al. 2016), EfficientNetB1 CNN (Tan and Le 2019), and VGG16 CNN (Simonyan and Zisserman 2014) architectures. All pre-trained models were implemented without the top layer, with pre-trained weights, and with frozen base model layers. A custom layer for classification was added to the base model (see Figure 5). Global Average Pooling was used for dimensionality reduction instead of flattening, which helped retain spatial information efficiently. The first dense layer had 512 neurons, initialized using He_normal for better weight distribution, followed by Batch Normalization and the Swish activation function to improve gradient flow. A 40% dropout was applied to reduce overfitting. Another dense layer with 256 neurons followed the same pattern—Batch Normalization, Swish activation, and Dropout. Finally, a softmax output layer with 4 neurons was used for multi-class classification. The model was compiled using the Adam optimizer with a low learning rate (0.00001) and categorical cross-entropy loss.

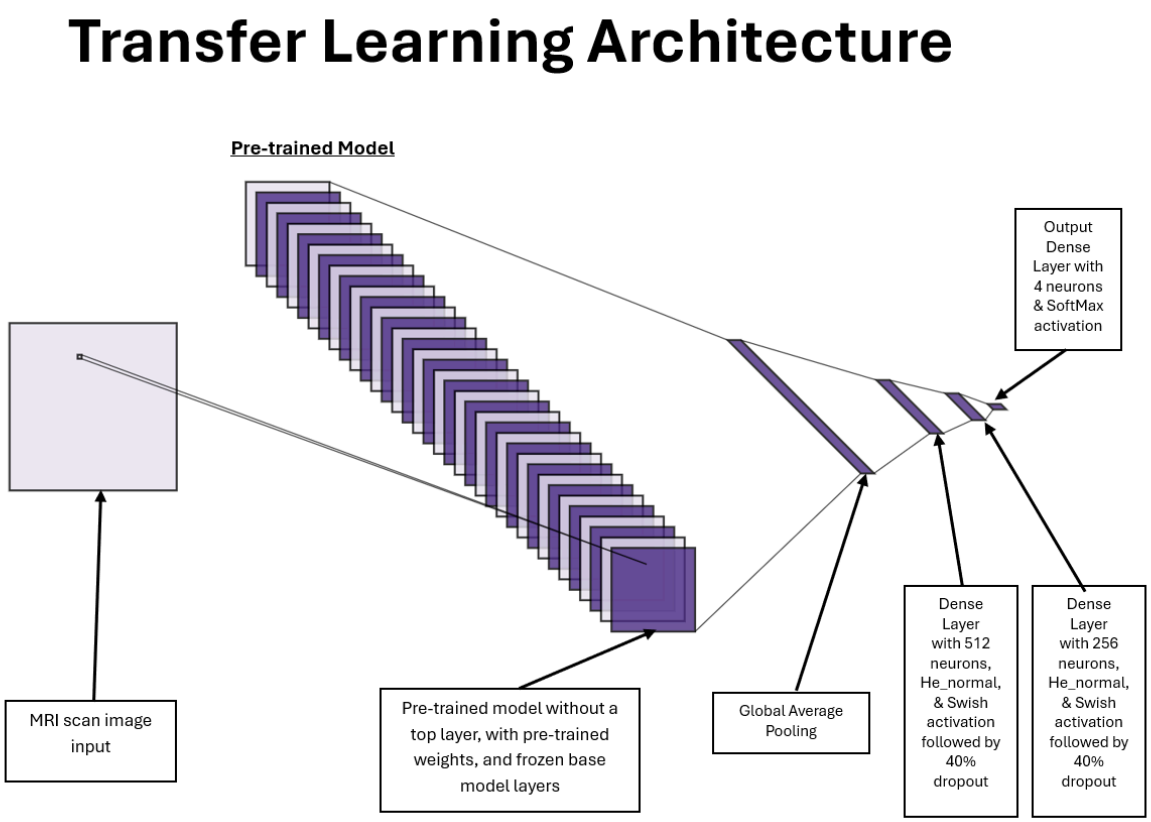

Figure 5: Transfer learning architecture compiled with an Adam optimization algorithm, 0.00001 learning rate, categorical cross entropy loss function, 32 batch size, and trained for 25 epochs.

Model Performance

See Table 1 for a summary of performance based on test accuracy. Four CNNs were evaluated on their test accuracy of a test dataset including one baseline CNN that was created, and three pre-trained models (InceptionV3, VGG16, and EfficientNetB1). The model that outperformed all others was InceptionV3 - before and after data augmentation. Results indicated that InceptionV3 (41.9%) achieved higher test accuracy compared to the baseline CNN (40.9%), VGG16 (39.8%), and EfficientNetB1 (38.3%) - prior to data augmentation. Following data augmentation, InceptionV3 achieved the highest test accuracy (43.9%) followed by VGG16 (40.8%), EffecientNetB1 (38.4%), and the baseline CNN (35.6%). All models exhibited underwhelming performance on test accuracy - regardless of data augmentation.

Data augmentation also resulted in improved metrics for the pretrained models. First, even though all pre-trained models were trained for 25 epochs, InceptionV3 exhibited 41.9% overall accuracy, and approximately 26% precision and recall. Metrics did improve after data augmentation was applied, accuracy rose to 43.9%, precision and recall rose to approximately 27%. Next, VGG16 exhibited 38.9% test accuracy prior to data augmentation with approximately 26% precision and recall. Similarly, these metrics improved after data augmentation, even if marginally, test accuracy rose to 40.8%, precision remained unchanged, and recall increased to 27%. Finally, EfficientNetB1 achieved 38.3% test accuracy prior to data augmentation and effectively remained unchanged at 38.4% test accuracy following data augmentation. Results showed that precision and recall increased from 25% to 27% and 25% to 26%, respectively.

The baseline CNN trained for only eight epochs, and achieved a test accuracy of 40.9%. However, unlike the pretrained models, all metrics dropped following data augmentation. The baseline CNN now achieved 32.6% test accuracy, 63.1% area-under-the-curve, 33.1% precision, 31.0% recall, and 2.73 loss. A sharp decline in performance following the application data augmentation on the training dataset.

TABLE 1. *Test Accuracy across CNN models without and with Data Augmentation*

	Baseline CNN	InceptionV3	VGG16	EfficientNetB1
Without Data Augmentation	40.9%	41.9%	39.8%	38.3%
With Data Augmentation	35.6%	43.9%	40.8%	38.4%

Table 1: Test accuracy scores for four models including one baseline CNN and three pre-trained CNNs. Test accuracy was evaluated first without applying data augmentation to the training dataset and then evaluated with data augmentation.

Discussion

Data augmentation proved to be a technique that added value as accuracy on the test set increased for all the pretrained models. Data augmentation consisted of the following adjustments: shifting width by 10%, shifting height by 10%, zoom range of 10%, and setting the filling strategy for empty pixels to 'nearest'. While only few data augmentation parameters were adjusted, this work shows that data augmentation in this application should be further explored. Evaluating the impact of data augmentation on performance systematically would greatly benefit deep learning programmers.

Pretrained models, specifically InceptionV3 (Szegedy et al. 2016), EfficientNetB1 (Tan and Le 2019), and VGG16 (Simonyan and Zisserman 2014), reinforced findings that suggest that transfer learning can be a good diagnostic tool (Ali et al. 2024; Farooq et al. 2017; Kahn et al. 2023; Sharma et al. 2022). Using pre-trained models can be more cost-effective and time-efficient compared to building proprietary models for Alzheimer's stage classification. Pre-trained models can be called in as little as one line of code eliminating the exhaustive process of building a model from scratch. Reducing the time required for that process alone will shorten the time required to implement this in a real-world setting.

Working with clinicians and researchers about how to best integrate this with existing systems will be difficult. Moreover, the challenging aspect will be working with clinicians who diagnose to integrate the use of MRI scan classification via CNN. Some will inevitably perceive this as a 'replacement' when the reality is that deep learning techniques are meant to aid and

guide the clinician. Saving time is crucial for managing an unforgiving disease, and this study shows that CNNs, in particular pre-trained CNNs, can at least facilitate saving time.

All models delivered underwhelming performance. Yet, pretrained models displayed higher accuracy than the model that was developed as a baseline CNN. All models displayed no greater than 44% accuracy, which stands in contrast with all the identified literature. Those studies reported accuracies upwards of 80% across all metrics. Of note, these studies utilized 3D MRI scans which differ from the 2D MRI scans that were utilized in this study. 3D MRI scans are known to be advantageous over 2D MRI scans. The data used in this study was also downloaded from a publically available data source (Rahman 2023), and data quality may have been compromised. All of the manuscripts in the identified literature explain that their data were provided by organizations.

Conclusion

These experiments showed that for the data used in this analysis, convolutional neural networks are a tool that can aid clinicians in diagnosing patients. CNNs are robust and can achieve high accuracy while being time-saving which is critical for developing a treatment plan for people living with Alzheimer's disease. Deep learning techniques are not meant to replace a trained professional, but are meant to provide an additional measure in order to make an informed decision about what is best for the patient.

This study lends credit to previous work (Kahn et al. 2023) whose researchers suggest that data augmentation increases the performance of pre-trained models (VGG16, VGG19). However, the results of this study stand in stark contrast to the reported accuracies of previous research. All of which were at the lowest in the 80% to 90% range. Nonetheless, pre-trained models showed additional evidence of their utility in diagnosing Alzheimer's disease (Ali et al. 2024; Farooq et al. 2017; Kahn et al. 2023; Sharma et al. 2022).

Directions for Future Work

Next steps include further elucidating the role and impact of data augmentation in medical applications. A resource explaining recommendations and reasons for applying the technique in CNNs can help guide deep learning programmers to build the best models within clinical settings. Future work can also contribute to documenting successful implementation of transfer learning as most published manuscripts contain confusing or altogether missing details about the exact programming that was developed and utilized. Lastly, future work can determine other options with high accuracy is crucial for settings where MRIs are inaccessible or out-of-reach.

Data Availability

Data is available for download from Kaggle (Rahman 2023), and it can be downloaded from my personal Github repository:

https://github.com/mhi573/Deep-Learning-for-Alzheimers-Stage-Classification-with-Transfer-Learning-and-Data-Augmentation/tree/main

Code Availability

The Python code that was written to conduct this project is available, accessible, and can be downloaded from my personal Github repository:

https://github.com/mhi573/Deep-Learning-for-Alzheimers-Stage-Classification-with-Transfer-Learning-and-Data-Augmentation/tree/main

References

Ali, Muhammad Umair, Kwang Su Kim, Majdi Khalid, Majed Farrash, Amad Zafar, and Seung Won Lee. "Enhancing Alzheimer's disease diagnosis and staging: a multistage CNN framework using MRI." *Frontiers in Psychiatry* 15 (2024): 1395563.

Alzheimer's Association. "What Is Alzheimer's?" *Alzheimer's Association*. Accessed February 16, 2025. https://www.alz.org/alzheimers-dementia/what-is-alzheimers.

Chollet, François, and contributors. *Keras: Deep Learning for Humans*. Version 3.8. Accessed February 16, 2025. https://keras.io.

Despotović, Ivana, Bart Goossens, and Wilfred Philips. "MRI Segmentation of the Human Brain: Challenges, Methods, and Applications," *Computational and Mathematical Methods in Medicine (2015):* 450341, https://doi.org/10.1155/2015/450341.

Farooq, Ammarah, Syed Mhuammad Anwar, Muhammad Awais, and Saad Rehman. "A deep CNN based multi-class classification of Alzheimer's disease using MRI", IEE (2017): https://doi.org/10.1109/IST.2017.8261460.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. "Deep Residual Learning for Image Recognition." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. https://arxiv.org/abs/1512.03385.

Khan, Rizwan, Saeed Akbar, Atif Mehmood, Farah Shahid, Khushboo Munir, Naveed Ilyas, M. Asif, and Zhonglong Zheng. "A Transfer Learning Approach for Multiclass Classification of Alzheimer's Disease Using MRI Images." *Frontiers in Neuroscience* 16 (2023): 1050777: https://doi.org/10.3389/fnins.2022.1050777.

Project Jupyter. *Jupyter Notebook: An Open Source Computing Platform*. Version 7.3. Accessed February 16, 2025. https://jupyter.org.

Python Software Foundation. *Python Language Reference, Version 3.1*. Accessed February 16, 2025. https://www.python.org.

Rahman, Abdur. *Alzheimer MRI Disease Classification Dataset*. Accessed February 16, 2025. Kaggle. https://www.kaggle.com/datasets/borhanitrash/alzheimer-mri-disease-classification-dataset

Sharma, Shagun, Kalpna Guleria, Sunita Tiwari, and Sushil Kumar. "A deep learning based convolutional neural network model with VGG16 feature extractor for the detection of Alzheimer Disease using MRI scans." *Measurement: Sensors* 24 (2022): 100506.

Simonyan, Karen, and Andrew Zisserman. 2014. *Very Deep Convolutional Networks for Large-Scale Image Recognition.* arXiv preprint arXiv:1409.1556. https://arxiv.org/abs/1409.1556.

Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. “Rethinking the Inception Architecture for Computer Vision.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)*, 2818–2826: https://doi.org/10.1109/CVPR.2016.30

Tan, Mingxing, and Quoc V. Le. 2019. "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks." *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 6105–6114. https://arxiv.org/abs/1905.11946.