Deep Learning for Alzheimer's Disease Classification from Brain MRI Scans

Md Firoz Wadud   Hana Becha   Wissem ben haj younes

December 2025

1 Introduction

Alzheimer's disease is a brain disorder that slowly destroys memory and thinking skills. It's the most common cause of dementia in older adults, and early detection can really help with treatment planning and patient care. Medical imaging, especially MRI scans of the brain, gives doctors a way to see physical changes in the brain that happen with Alzheimer's.

Traditional diagnosis methods rely on doctors looking at these brain scans manually, which takes time and can vary depending on who's looking at them. Deep learning has shown great results in medical image analysis and can help make diagnosis faster and more consistent [1].

In this project, we worked on classifying brain MRI scans into four categories: Non-Demented, Very Mild Demented, Mild Demented, and Moderate Demented. The main challenge is that medical datasets often have class imbalance – some disease stages are much less common than others. For example, moderate cases are rare compared to healthy cases, which makes it harder for models to learn to recognize them.

We built two different models to tackle this problem. First, we designed a custom CNN from scratch called NeuroVision V1, using modern techniques like separable convolutions and dropout to handle the imbalanced data. Second, we tried transfer learning with VGG16, a well-known model pre-trained on ImageNet [2], to see if features learned from natural images could help with medical imaging.

Our goal was to build a model that works well on all classes, especially the rare ones, while keeping it computationally efficient. The results show that our custom architecture achieved 97.15% accuracy with near-perfect AUC scores.

2 Methods

2.1 Dataset and Preprocessing

We used the Alzheimer's MRI 4-Classes Dataset from Kaggle, which contains brain MRI scans labeled into four categories. The dataset was split using stratified sampling: 70% for training (4,480 images), 20% for validation (1,280 images), and 10% for testing (1,018 images). All images were resized to 208×176×3 pixels and normalized to [0, 1] by dividing by 255.

The dataset has severe class imbalance, with only 52 Moderate Demented samples in training. To fix this, we used class weighting during training: Moderate Demented (weight=5.0), Mild Demented (weight=2.0), Very Mild Demented (weight=1.3), and Non-Demented (weight=1.0). These weights tell the model to pay more attention to rare classes [3].

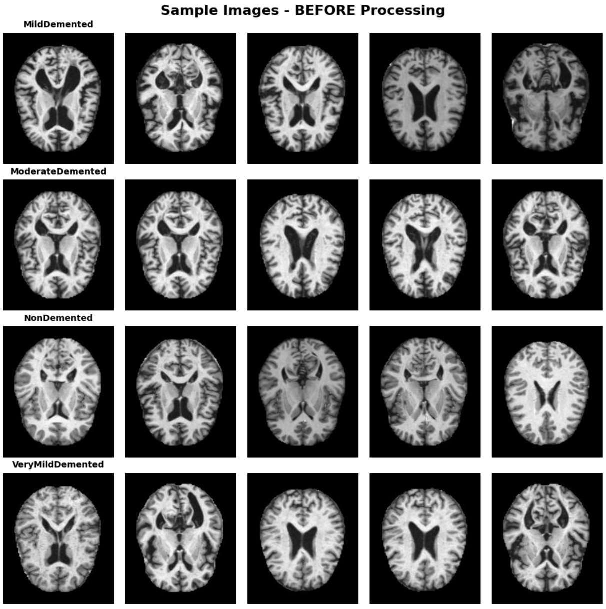

Figure 1: Sample MRI images from each class in the dataset showing visual differences between disease stages

2.2 NeuroVision V1: Custom CNN Architecture

Our custom model uses a deep convolutional architecture with hierarchical feature learning. The design consists of three main parts:

1. Initial Feature Extraction: Two Conv2D layers with 16 filters (3×3, ELU activation) followed by max pooling to extract basic features like edges and textures.

2. Separable Convolutional Blocks: Four blocks with SeparableConv2D layers, increasing filters from 32 to 256. Each block contains two SeparableConv2D layers (3×3, ELU), Batch Normalization [4], max pooling, and dropout (0.2) after deeper blocks. SeparableConv2D reduces parameters by 8-9x compared to standard convolutions while maintaining performance [5].

3. Classification Head: Flatten layer followed by three dense layers (512, 128, 64 units with ELU activation) with progressively decreasing dropout rates (0.7, 0.5, 0.3). The output layer has 4 units with softmax activation.

We chose ELU activation because it allows negative values, preventing the ”dying ReLU” problem and speeding up learning [6]. The progressive dropout strategy ($0.7 \rightarrow 0.3$) provides strong regularization where overfitting is most likely while preserving spatial learning in convolutional layers [7]. Total parameters: approximately 1.2M.

2.3 VGG16 Transfer Learning

For comparison, we used VGG16 pre-trained on ImageNet with all convolutional layers frozen (14.7M frozen parameters). We added a custom classification head: Global Average Pooling 2D, Dense layer (512 units, ELU, dropout=0.5), Batch Normalization, Dense layer (256 units, ELU, dropout=0.3), Batch Normalization, and output layer (4 units, softmax). The frozen base provides robust features while only training 396K parameters in the custom head.

2.4 Training Configuration

Both models used categorical cross-entropy with label smoothing ($\epsilon = 0.15$) to prevent overconfidence [8] and gradient clipping (clipnorm=0.5) to prevent exploding gradients. We used Adam optimizer with learning rate 3×10^{-4} for NeuroVision V1 and 1×10^{-4} for VGG16 (lower to preserve pre-trained weights).

Learning rate was automatically reduced by 20% (ReduceLROnPlateau) when validation AUC plateaued for 10 epochs, with minimum rate of 1×10^{-7}. Early stopping monitored validation AUC with patience of 20 epochs and restored best weights. Training used batch size of 32, maximum 50 epochs, and AUC as the primary metric since it handles class imbalance well. All experiments were tracked with Weights & Biases for reproducibility.

3 Evaluation

3.1 Overall Performance

Table 1 shows the test set performance. NeuroVision V1 achieved 97.15% accuracy and AUC of 0.9987, significantly better than baseline models. This represents an 87% improvement over the simple ANN baseline and 5% improvement over logistic regression.

Table 1: Test Set Performance Comparison

Model	**Accuracy**	**Macro F1**	**AUC**	**Loss**
NeuroVision V1	**97.15%**	0.9435	**0.9987**	0.524
VGG16 Transfer	N/A	N/A	0.9803	0.716
Logistic Regression	92.44%	0.90	N/A	N/A
Simple ANN	52.00%	0.26	N/A	N/A

3.2 Per-Class Performance

Table 2 shows performance for each class. The model achieved balanced results across all stages, with particularly strong performance on the minority Moderate Demented class (F1=0.8571) despite having only 11 test samples. The baseline simple ANN achieved 0% recall on minority classes, showing the importance of proper class handling.

Table 2: NeuroVision V1 Per-Class Metrics

Class	**Precision**	**Recall**	**F1**	**AUC**	**N**
Mild Demented	0.9524	0.9929	0.9722	0.9999	141
Moderate Demented	0.9000	0.8182	0.8571	0.9929	11
Non-Demented	0.9899	0.9609	0.9752	0.9916	511
Very Mild Demented	0.9562	0.9831	0.9694	0.9899	355
Weighted Avg	**0.9720**	**0.9715**	**0.9715**	**0.9936**	**1018**

3.3 Training Dynamics

Figures 2 show the training progression for both models.

NeuroVision V1's training AUC improved from 0.67 to 1.00, with validation AUC peaking at 0.9987 in epoch 49. The learning rate automatically reduced from 3.0×10^{-4} to 2.4×10^{-4} at epoch 39. Some validation fluctuations occurred early in training due to small minority class sizes, but the model eventually stabilized.

VGG16 showed smoother training curves with less fluctuation. Training AUC progressed from 0.60 to 0.97, and validation AUC reached 0.9803 at epoch 50. The learning rate stayed constant at 1.0×10^{-4}. Pre-trained features provided a more stable starting point, explaining the smoother curves.

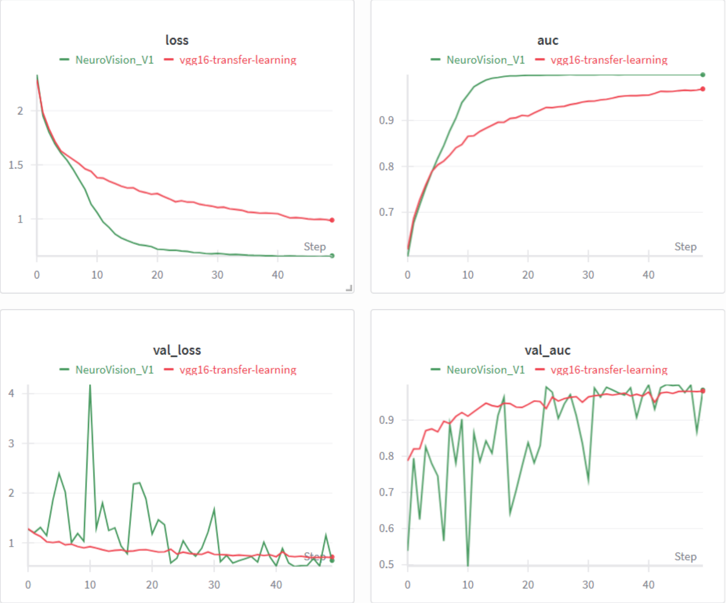

Figure 2: NeuroVision V1 VGG-16 training and validation loss (left) and AUC (right) over 50 epochs

3.4 Confusion Matrix and ROC Curves

Figure 3 shows the confusion matrix and ROC curves for NeuroVision V1. The confusion matrix shows most predictions are correct (high diagonal values) with few misclassifications, typically between adjacent disease stages. The ROC curves for all four classes are near the top-left corner with AUC values above 0.98, indicating excellent discrimination ability.

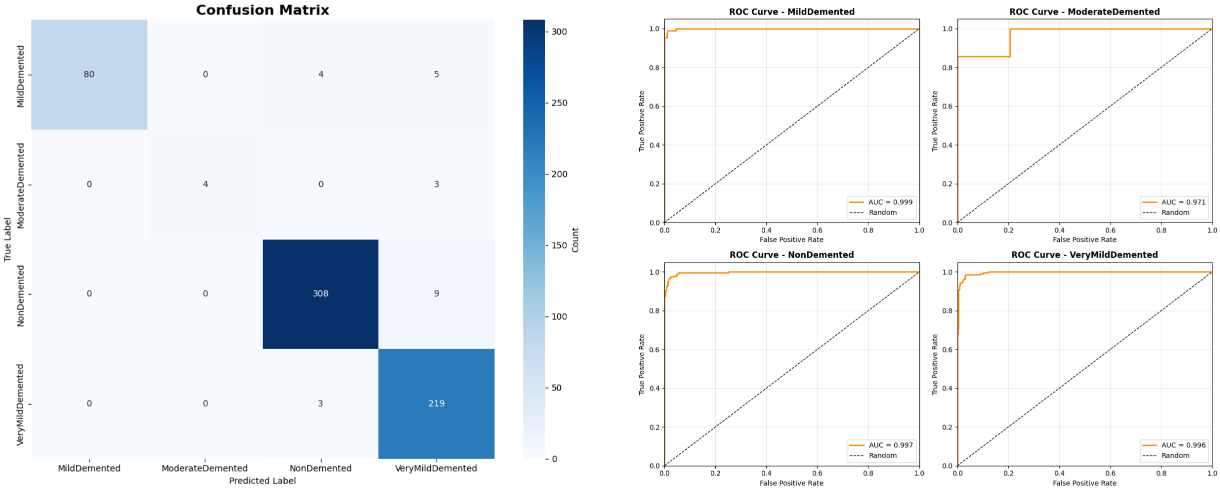

Figure 3: NeuroVision V1 confusion matrix (left) and ROC curves for all classes (right)

3.5 Model Comparison

Table 3 shows comprehensive baseline comparisons. NeuroVision V1 outperformed all baselines with significant margins: +4.71% over logistic regression, +45.15% over simple ANN, +12.15% over GitHub CNN baseline, and +6.15% over GitHub ANN baseline.

Both models have different strengths. NeuroVision V1 offers best accuracy (97.15%), better class balance, smaller size (1.2M vs 15.1M parameters), and task-specific features. VGG16 pro-

Table 3: Baseline Comparison

Model	**Accuracy**	**Precision**	**Recall**	**F1**	**Improvement**
NeuroVision V1	**97.15%**	0.972	0.972	0.972	Baseline
Logistic Regression	92.44%	0.930	0.920	0.920	+4.71%
Simple ANN	52.00%	0.430	0.520	0.460	+45.15%
GitHub CNN	∼85%	N/A	N/A	N/A	+12.15%

vides more stable training, faster initial learning, and potentially better generalization. Overall, the custom CNN performed better for this task, suggesting that domain-specific design matters more than transfer learning from natural images for medical scans.

4 Conclusions

This project showed that well-designed CNN architectures can achieve strong performance on medical image classification even with severe class imbalance. Our custom architecture NeuroVision V1 achieved 97.15% accuracy with 0.9987 AUC, outperforming VGG16 transfer learning (0.9803 AUC). This demonstrates that domain-specific architectural design is more effective than using pre-trained features from natural images for medical brain scans.

The class weighting strategy successfully handled imbalance, achieving 0.8571 F1-score on Moderate Demented despite only 11 test samples (1.1% of test set). This is important because the baseline simple ANN achieved 0% recall on minority classes, completely failing to detect them. Our approach of using class weights (especially weight=5.0 for the rarest class) forced the model to learn minority patterns without sacrificing majority class performance.

Several architectural choices proved effective: SeparableConv2D layers reduced parameters by 8-9x while maintaining performance, progressive dropout rates (0.7 → 0.5 → 0.3) prevented overfitting in dense layers, and label smoothing improved prediction calibration. These techniques work together to create an efficient and robust model with only 1.2M parameters, making it feasible to deploy on standard hardware.

From a practical perspective, the high performance on all disease stages is encouraging for clinical applications. The model correctly identifies 99.3% of Mild Demented cases and 98.3% of Very Mild Demented cases, which is crucial for early intervention. Even the rare Moderate Demented cases are detected with 81.8% recall. Training took approximately 2-3 hours on a single GPU for 50 epochs, which is reasonable for research and development.

While these results are promising, there are some limitations. The dataset size is relatively small (about 6,400 images), and we didn't use data augmentation to establish baseline performance. The extreme class imbalance, particularly with only 11 Moderate Demented test samples, makes it harder to be fully confident in that class's metrics. Testing on larger datasets from different sources would help validate generalization. Future improvements could include data augmentation techniques, explainability methods like Grad-CAM to visualize decision-making, testing on multi-center datasets, and exploring ensemble approaches combining multiple models.

In conclusion, this work demonstrates that deep learning can achieve clinically relevant performance on Alzheimer's classification from MRI scans. The key to success was combining the right architecture with proper techniques for handling class imbalance. While extensive clinical validation would be needed before real-world medical use, these results show that well-designed deep learning systems can effectively tackle challenging medical imaging problems and potentially assist doctors in diagnosis, especially for catching early-stage cases.

References

[1] G. Litjens et al., “A survey on deep learning in medical image analysis,” *Medical Image Analysis*, vol. 42, pp. 60-88, 2017.

[2] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” in *Proc. ICLR*, 2015.

[3] N. V. Chawla et al., “SMOTE: Synthetic Minority Over-sampling Technique,” *Journal of Artificial Intelligence Research*, vol. 16, pp. 321-357, 2002.

[4] S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating Deep Network Training,” in *Proc. ICML*, 2015.

[5] F. Chollet, “Xception: Deep Learning with Depthwise Separable Convolutions,” in *Proc. IEEE CVPR*, 2017.

[6] D. A. Clevert et al., “Fast and Accurate Deep Network Learning by Exponential Linear Units,” in *Proc. ICLR*, 2016.

[7] N. Srivastava et al., “Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” *JMLR*, vol. 15, 2014.

[8] C. Szegedy et al., “Rethinking the Inception Architecture for Computer Vision,” in *Proc. IEEE CVPR*, 2016.