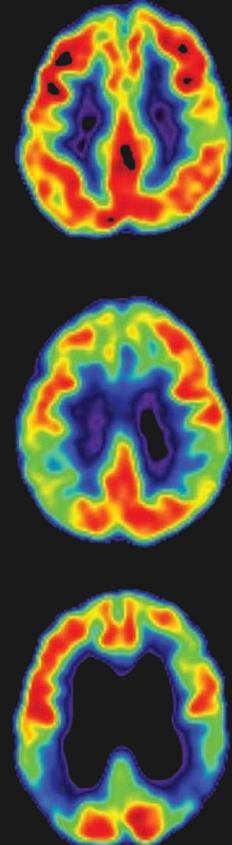


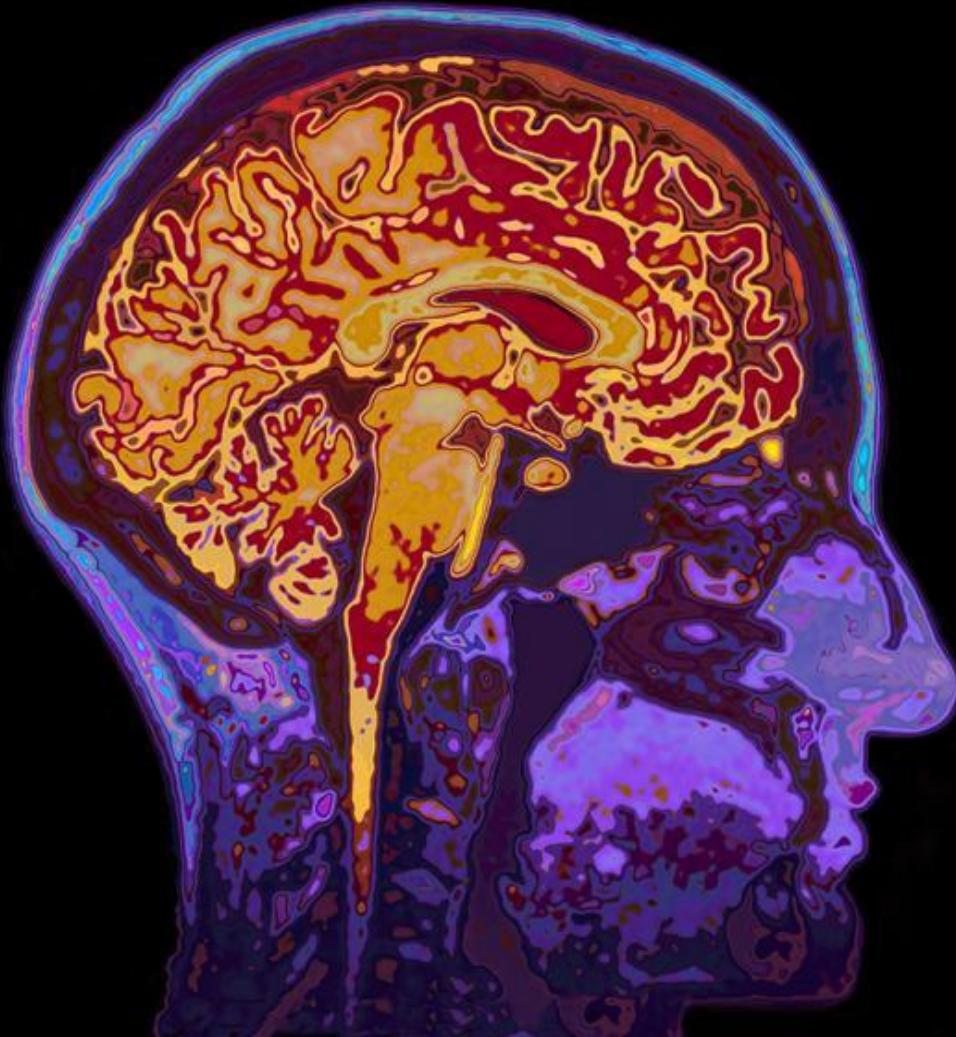
MOX, Politecnico di Milano

Early Alzheimer's disease detection from structural MRIs through Deep Learning Models

July 29, 2025

Alberto Taddei
Tommaso Crippa
Vittorio Pio Remigio Cozzoli



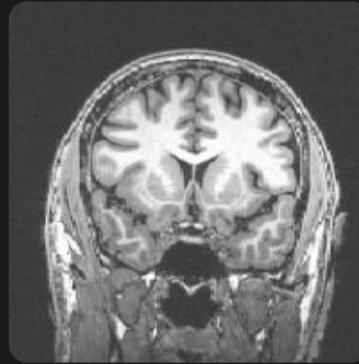


INTRODUCTION

Early diagnosis of Alzheimer's Disease (AD) is critical for maximizing the impact of therapeutic interventions. Traditional diagnosis relies heavily on clinical assessments and invasive procedures, which may delay detection.

- Alzheimer's Disease is a progressive neurodegenerative condition.
- Timely diagnosis can significantly improve patient outcomes.
- There is a growing demand for non-invasive, automated diagnostic tools.

We explore how deep learning, particularly 3D convolutional neural networks (3D CNNs), can be applied to structural MRI (sMRI) scans to detect early signs of AD.



WHY sMRI?

Structural MRI is a powerful tool that captures detailed brain anatomy without the need for radiation or invasive procedures.

- Offers high-resolution anatomical information.
- Non-invasive, radiation-free, widely available.
- Enables measurement of brain volume, cortical thickness, and atrophy.

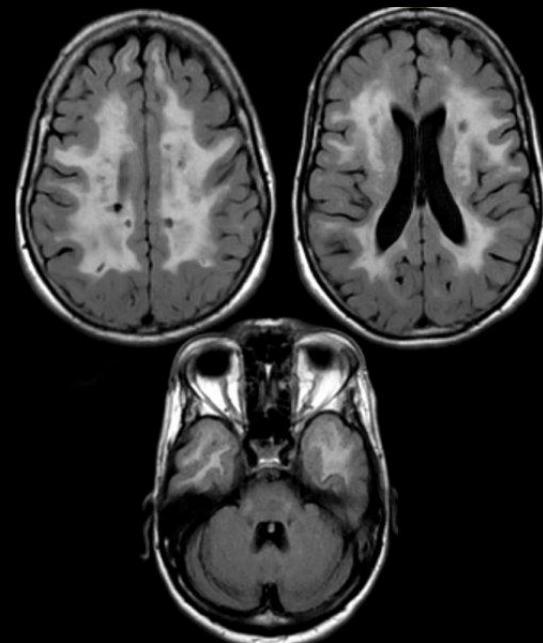
Because of its accessibility and resolution, sMRI is ideal for longitudinal studies and early screening, especially in elderly populations.

sMRI FOR AD DETECTION

sMRI reveals early structural changes in regions like the hippocampus and entorhinal cortex, often before symptoms appear.

It captures brain atrophy patterns that differentiate cognitively normal (CN), mild cognitive impairment (MCI), and AD stages.

These structural features correlate strongly with clinical outcomes, making sMRI ideal for deep learning-based diagnosis.



RELATED WORK

Liu et al.

Researchers

2022

Year

87.59%

AUC-ROC
(CN vs the rest)

Much of the early literature on automated AD diagnosis relied on manually defined regions of interest (ROIs).

- ROI-based models focus on hippocampal volume and cortical thickness.
- Manual segmentation is time-consuming and hard to scale.

Recent studies like Liu et al. (2022) shifted to full-brain 3D CNNs, showing improved performance and scalability without requiring handcrafted features.

OUR CONTRIBUTIONS & INNOVATIONS

Our goal was to reproduce and improve upon the model by Liu et al., with key innovations enabled by high-performance computing (HPC) resources.

- Used the **MeluXina** Supercomputer to train efficiently.
- Utilized a **deeper** Multi Layer Perceptron to improve the classification.
- Introduced advanced **data augmentation** and hyperparameter tuning.
- Trained on a **larger batch size** with improved GPU parallelism.

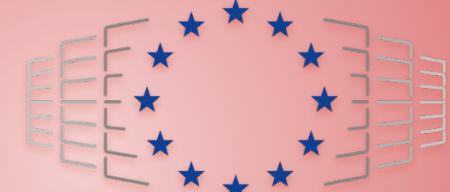
These enhancements significantly elevated both model performance and scalability.

FROM GALILEO100 TO MELUXINA

We began our project on CINECA's CPU-only **Galileo100** cluster, but its lack of GPU support limited our ability to train deep learning models effectively.

- To overcome these constraints, we transitioned to **MeluXina**, a GPU-powered supercomputer optimized for AI workloads.
- **Access was granted via the EuroHPC Joint Undertaking**, following support received at the EuroHPC Summit.

This shift enabled full-scale training, real-time augmentation, and extensive experimentation, transforming our work from replication to innovation.



EuroHPC
Joint Undertaking

DATASET OVERVIEW - ADNI

We used the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset, which provides high-quality, T1-weighted structural MRI scans labeled by cognitive status: cognitively normal (CN), mild cognitive impairment (MCI), and Alzheimer's Disease (AD).

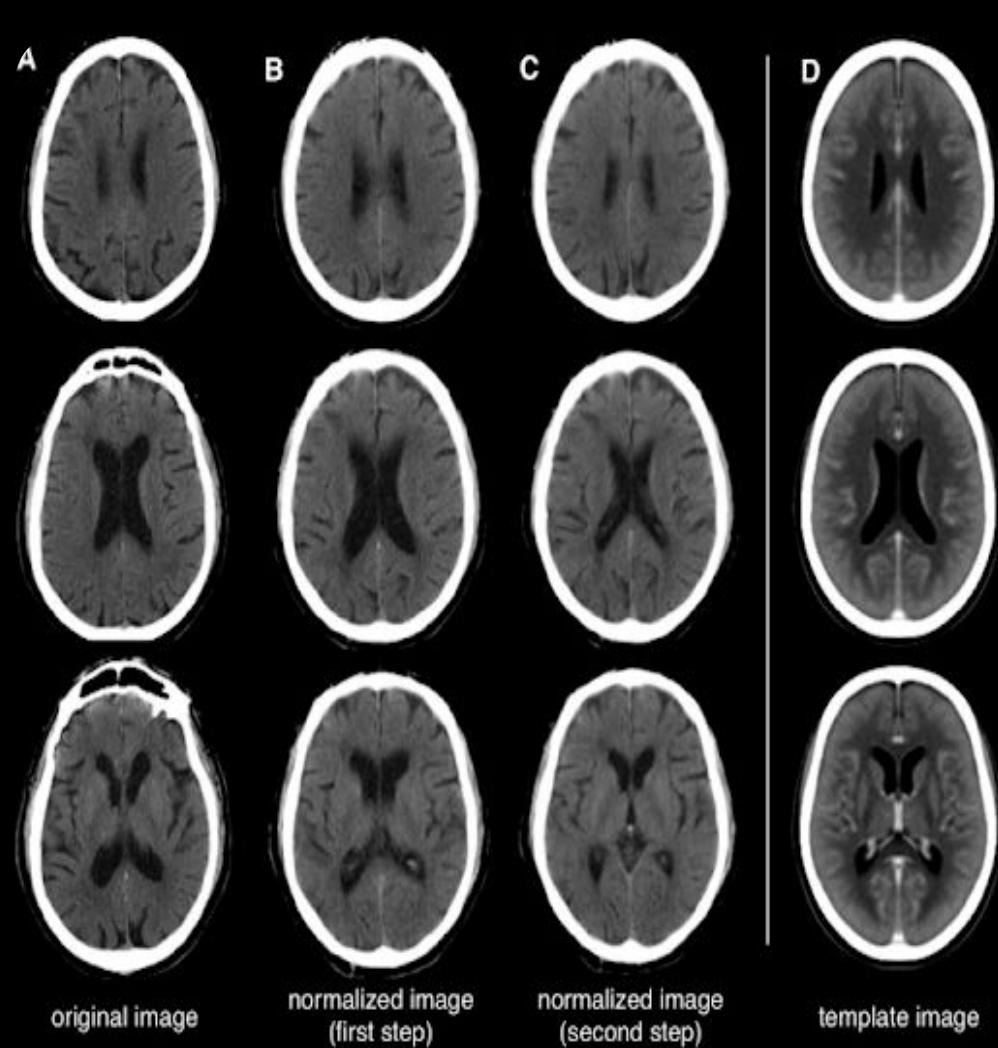
Each scan includes detailed metadata, such as age, sex, acquisition date, and preprocessing steps like GradWarp, B1 correction, and N3 bias field correction, ensuring data consistency and clinical relevance.

All data was converted to the BIDS (Brain Imaging Data Structure) format, a standardized framework that facilitates automated preprocessing, reproducibility, and seamless integration into deep learning pipelines.



PREPROCESSING PIPELINE

ADNI → BIDS | Clinica | Data Augmentation



CLINICA

We used Clinica, an automated neuroimaging pipeline, to preprocess ADNI MRI data efficiently and reproducibly.

Key steps included:

- **clinica convert adni-to-bids:** conversion for standardized data structure
- **clinica run t1-volume:** spatial normalization, bias correction and intensity standardization to reduce scanner artifacts.

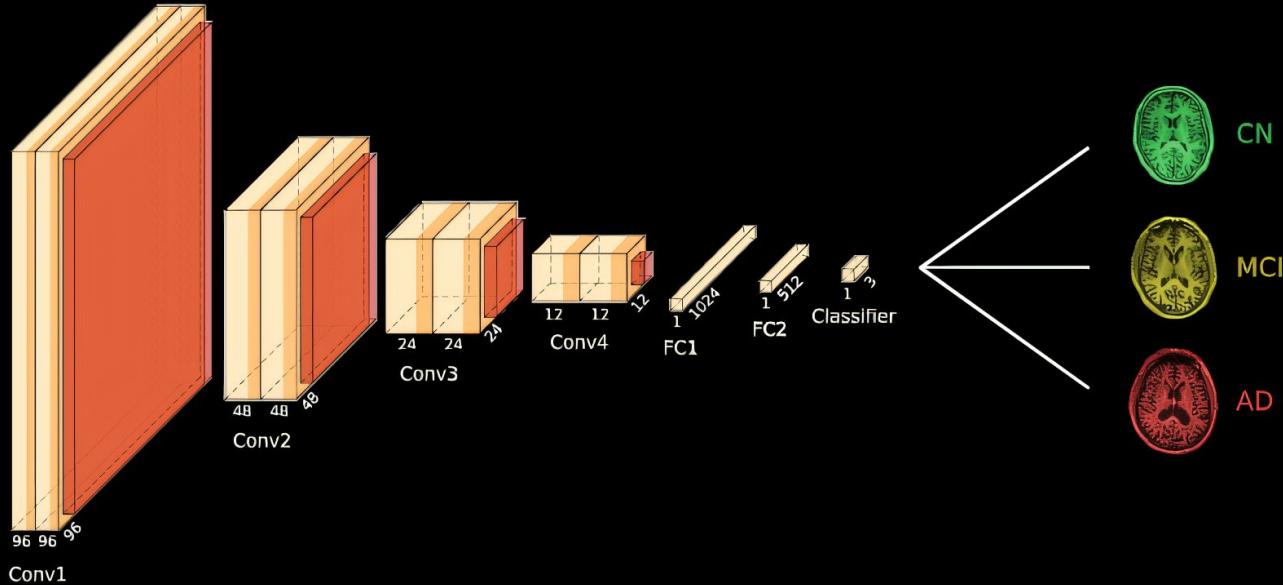
Clinica's integration with SPM enabled scalable, high-quality preprocessing across thousands of MRI scans.

*Detailed description on how to use them correctly inside INSTALL.MD

MODEL ARCHITECTURE

Convolution | Pooling | Fully-Connected

ARCHITECTURAL COMPOSITION



Feature Extraction Network

- 4 Convolutional layers
- Instance Norm
- ReLU activation
- MaxPooling

Multi Layer Perceptron

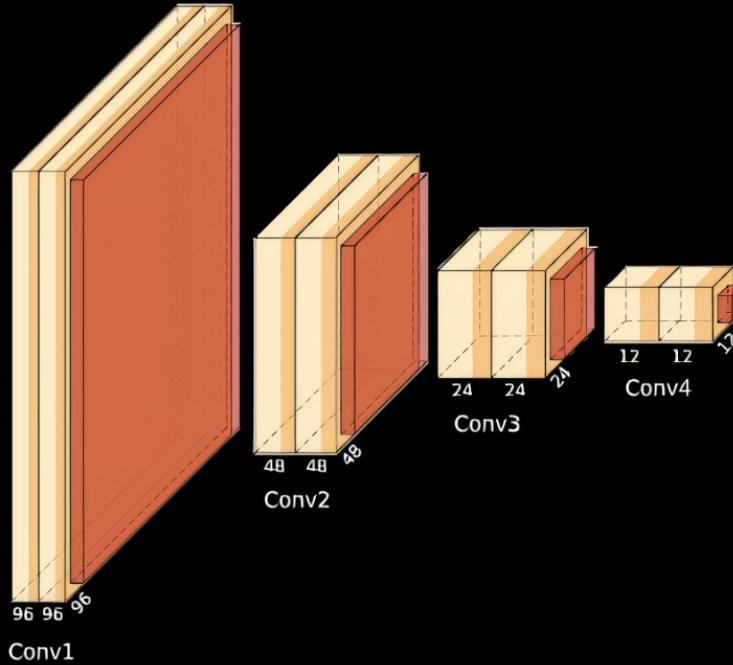
- 3 FeedForward layers

Total Weights

- 24.736.067 weights

$$f_{\theta} = \varphi_{\text{MLP}} \circ (\psi_4 \circ \psi_3 \circ \psi_2 \circ \psi_1) (\mathbf{z}) = \varphi_{\text{MLP}} \circ \psi(\mathbf{z})$$

CONVOLUTIONAL BACKBONE



↑ Channels, ↓ Voxels :

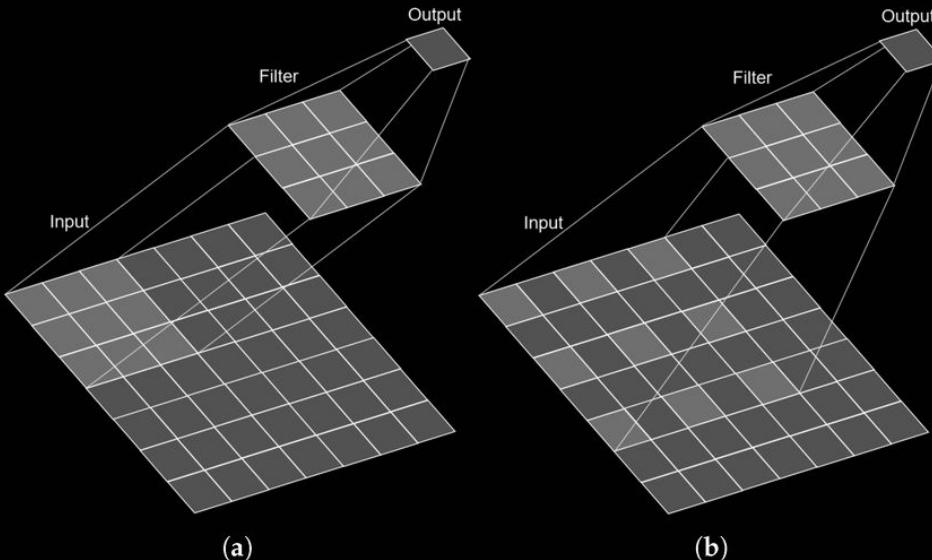
- **Block 1:** $1 \rightarrow 32$ channels
 $k=1$, MaxPool(3,2)
- **Block 2:** $32 \rightarrow 256$ channels
 $k=3$, dil=2, MaxPool(3,2)
- **Block 3:** $256 \rightarrow 512$ channels
 $k=5$, dil=2, pad=2, MaxPool(3,2)
- **Block 4:** $512 \rightarrow 512$ channels
 $k=3$, dil=2, pad=1, MaxPool(5,2)

For each layer:

- Instance Normalization
- ReLU activation function

$$\psi_k(\mathbf{z}) = \text{MP}_k\left(\sigma\left(\text{Norm}_k(\text{Conv}_k(\mathbf{z}))\right)\right)$$

CONVOLUTIONAL BACKBONE



(a) Standard convolution.
(b) Dilated convolution with factor 2.

↑ Channels, ↓ Voxels :

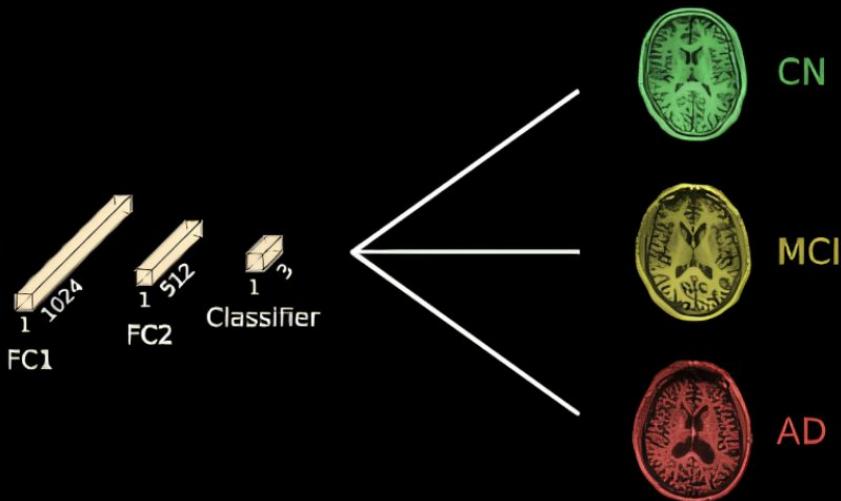
- **Block 1:** $1 \rightarrow 32$ channels
 $k=1$, MaxPool(3,2)
- **Block 2:** $32 \rightarrow 256$ channels
 $k=3$, dil=2, MaxPool(3,2)
- **Block 3:** $256 \rightarrow 512$ channels
 $k=5$, dil=2, pad=2, MaxPool(3,2)
- **Block 4:** $512 \rightarrow 512$ channels
 $k=3$, dil=2, pad=1, MaxPool(5,2)

For each layer:

- Instance Normalization
- ReLU activation function

$$\psi_k(\mathbf{z}) = \text{MP}_k\left(\sigma\left(\text{Norm}_k(\text{Conv}_k(\mathbf{z}))\right)\right)$$

MULTI-LAYER PERCEPTRON



F = flatten dim of convolutional features

- $W_0 : \mathbb{R}^F \rightarrow \mathbb{R}^{1024}$ (Linear + ReLU)
- $W_1 : \mathbb{R}^{1024} \rightarrow \mathbb{R}^{512}$ (Linear + ReLU)
- $W_2 : \mathbb{R}^{512} \rightarrow \mathbb{R}^3$ (Linear + SoftMax)

SoftMax \rightarrow Class Probabilities

Compared to the original architecture proposed by Liu et al. (2022), we extended the MLP head by **adding two fully connected layers** with ReLU activations before the final classification layer.

$$\hat{\mathbf{y}} = \text{softmax}\left(W_2 \sigma\left(W_1 \sigma\left(W_0 \mathbf{v}_0 + b_0\right) + b_1\right) + b_2\right)$$

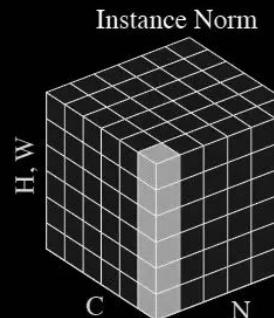
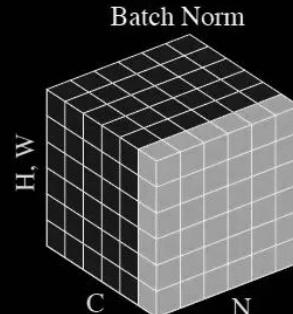
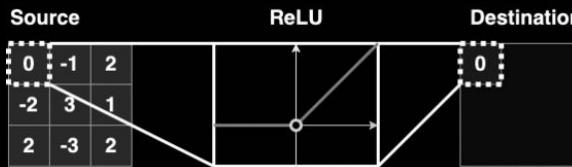
GRAD FLOW, ACTIVATION AND OPTIMISATION

ReLU advantages

- Piece-wise linear, non-saturating → gradients stay > 0 for $x > 0$
- Covered by Universal Approximation (Cybenko '89; Mhaskar-Poggio-Yarotsky '16)

Gradient Flow:

- Instance / Batch Norm keep activations in effective ReLU range
- Dilated convs give wide context with fewer stacked layers
- Cross-entropy loss supplies well-scaled output gradients
→ Vanishing-gradient issues largely mitigated without residuals

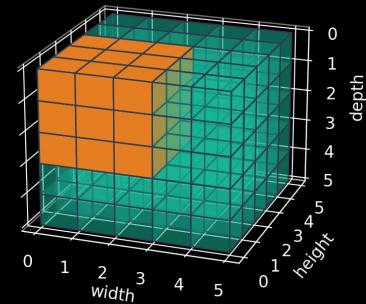


3D Convolution

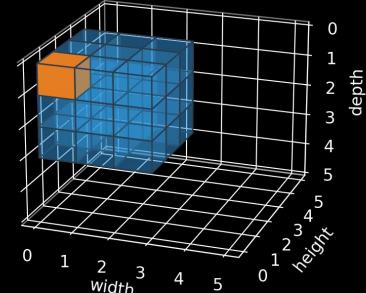
stride: (1, 1, 1)

padding: (0, 0, 0)

Input Volume (5x5x5)



Output Volume (3x3x3)



ADAM vs SGD+MOMENTUM

Baseline (Liut et al.) Our Approach

- **Optimizer:** SGD + Momentum (0.9)
- **Learning rate:** 0.01
- **Optimizer:** Adam
- **Learning rate:** 0.0018
- **Other hyperparams:** PyTorch defaults ($\beta_1=0.9$, $\beta_2=0.999$, $\varepsilon=10^{-8}$)

Why we kept Adam:

Adam update (per-parameter pre-conditioning):

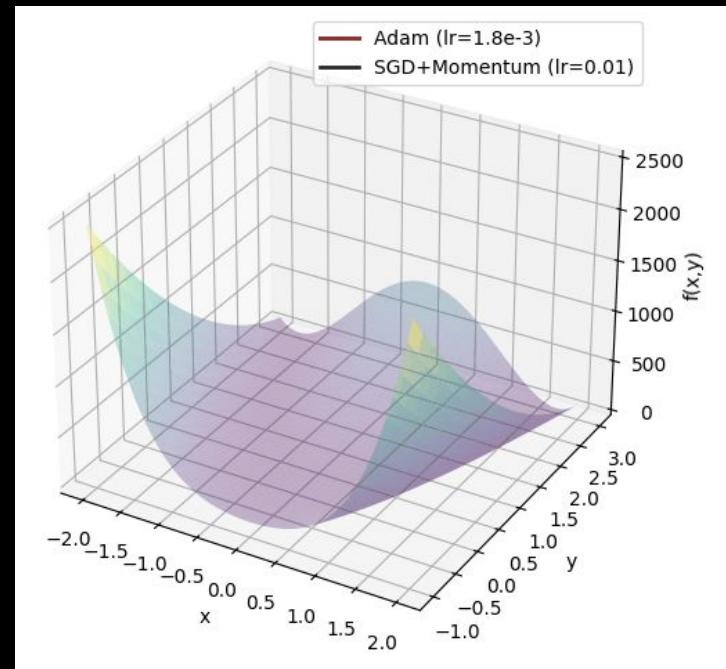
$$\theta_{t+1} = \theta_t - \eta \cdot \frac{\hat{m}^t}{\sqrt{\hat{v}} + \varepsilon}$$

$$\eta = 1.8 \cdot 10^{-3}, \beta_1 = 0.9, \beta_2 = 0.999$$

Acts like a diagonal Newton step at linear cost, therefore yielding:

- No fresh SGD+M runs \rightarrow reference baseline from Liu et al. (2022)
- Adam reached higher stability and faster convergence per GPU-hour vs their SGD+M results

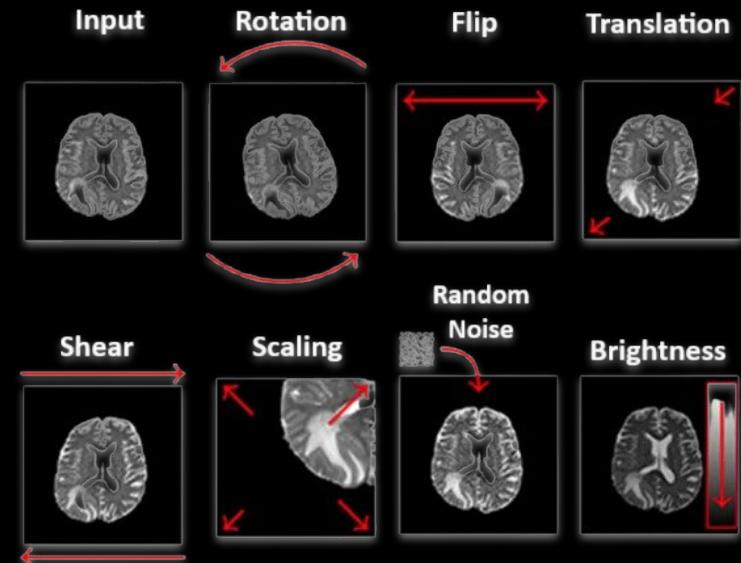
Illustrative Rosenbrock Example



$$f(x, y) = (1 - x)^2 + 100(y - x^2)^2$$

DATA AUGMENTATION

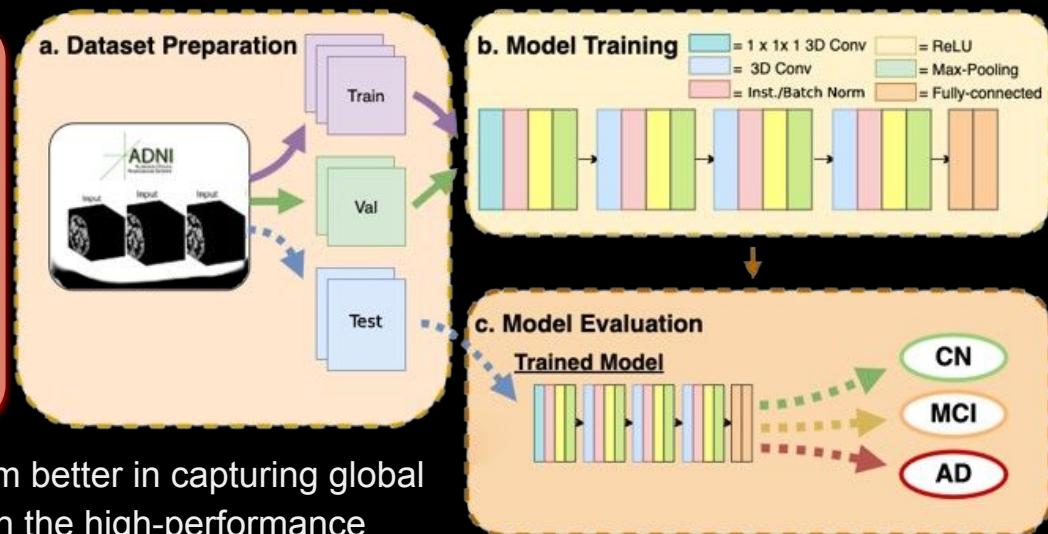
- Gaussian Blurring
- Random Cropping
- Center Cropping for Val/Test



FINAL HYPERPARAMETERS

- **Total Epochs:** 100
- **Instance Normalization**
- **Batch size:** 36
- **Optimizer:** Adam (LR = 0.0018)
- **Loss:** Cross Entropy

This selection of hyperparameters helped us perform better in capturing global brain patterns efficiently when performing training on the high-performance environment of MeluXina.



VISUALIZATION

Grad-CAM | Thresholding | Saliency

GRAD-CAM

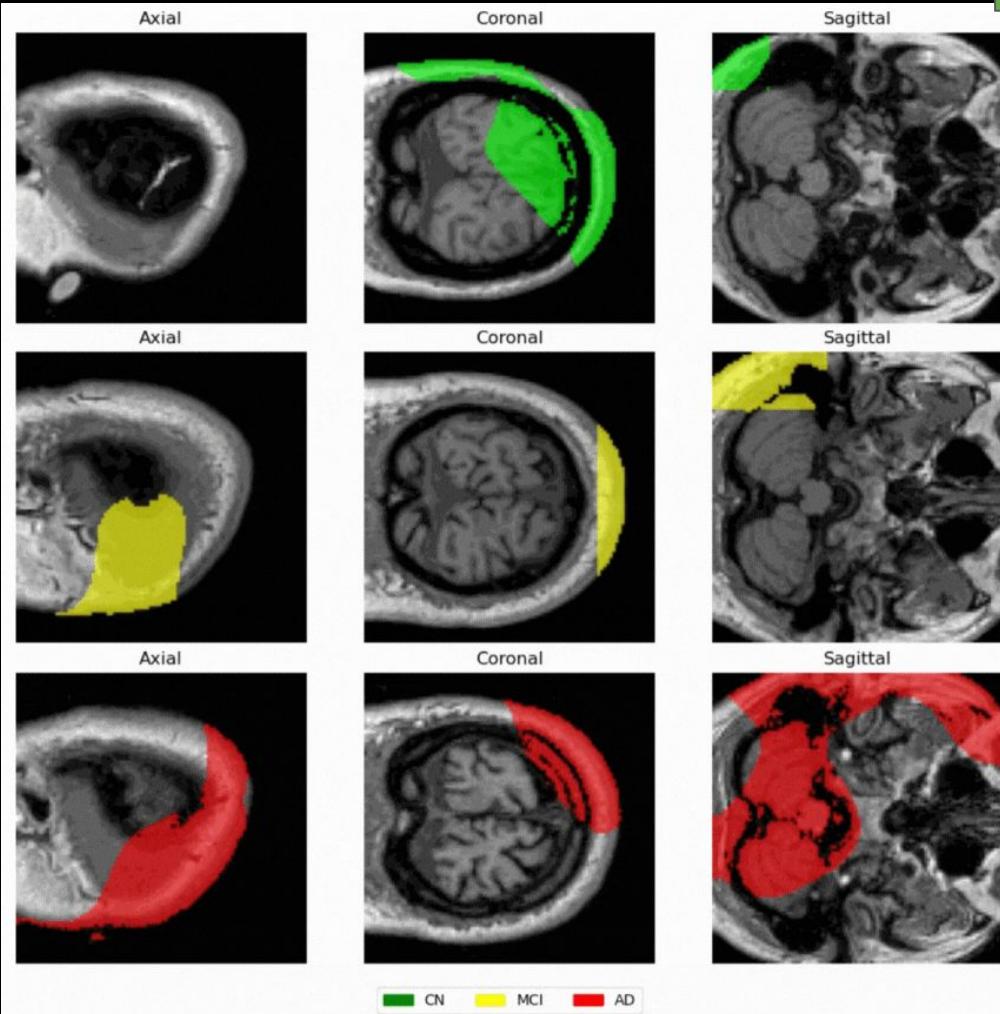
To enhance interpretability, we applied **Grad-CAM** to visualize the brain regions influencing the model's decisions.

This approach is crucial for bridging deep learning predictions with clinical understanding in Alzheimer's diagnosis.

To **fine-tune the sensitivity** of the activation maps, we implemented **three thresholding strategies**:

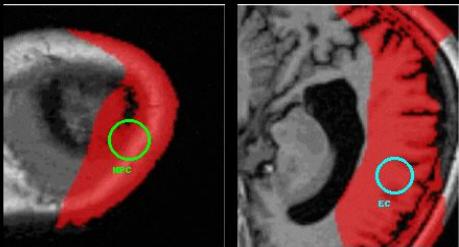
- **Percentile-based** (top $p\%$ values)
- **Otsu's method** (adaptive histogram thresholding)
- **Mean + std** (voxels above $\mu+k\sigma$)

We generated animated heatmaps across axial, coronal, and sagittal views, color-coded by predicted class.

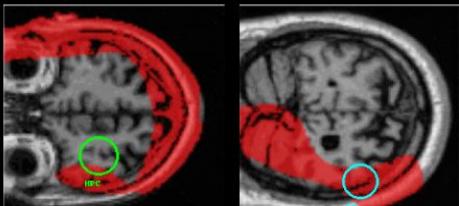


IS GRAD-CAM PREDICTION ACCURATE?

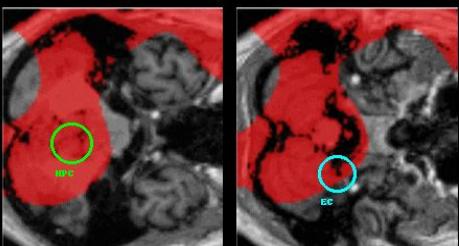
Axial (HPC & EC)



Coronal (HPC & EC)



Sagittal (HPC & EC)



As discussed earlier, Alzheimer's disease is strongly associated with early structural changes in specific brain regions, in particular the **hippocampus** and the **entorhinal cortex**.

While Grad-CAM often tends to produce broad and diffuse activation maps, it still provides valuable insight into the model's decision process.

The hereby examples show that, despite the spatial spread of the heatmaps, there is a clear overlap between the highlighted regions and known anatomical targets of AD progression, offering a visual confirmation that the classifier has learned to associate meaningful structural patterns with disease severity.

RESULTS & EVALUATION

Python | C++ | Model Accuracy

MODEL COMPARISON

Python

PyTorch

vs

C++

LibTorch

OUR ACCURACY

Python Model

95.18%

AUC-ROC (CN)

88.17%

AUC-ROC (MCI)

87.43%

AUC-ROC (AD)

C++ Model

96.05%

AUC-ROC (CN)

91.74%

AUC-ROC (MCI)

88.13%

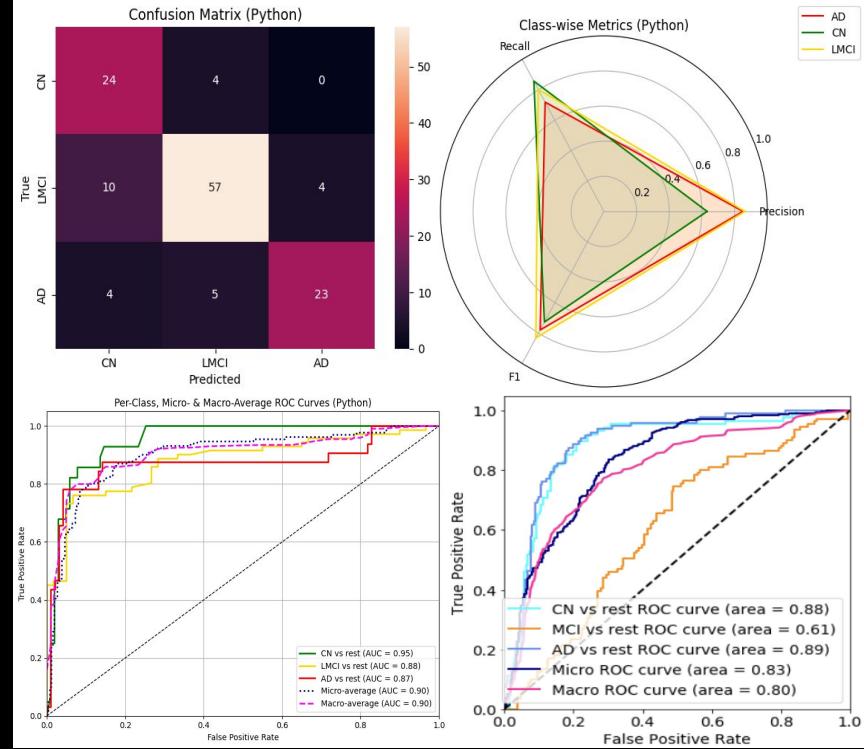
AUC-ROC (AD)

Comparison with Liu et al. (2022): Liu et al. reported AUC-ROC of 87.59% (CN), 62.59% (MCI) and 89.21% (AD). Our models show a substantial improvement, especially for CN class, suggesting better sensitivity to intermediate stages of cognitive decline, a key aspect for early detection strategies.

PYTHON RESULTS

Config (BS _t –BS _{vt} , lr)	Acc. (%)	BA (%)	Macro-AUC
24–48, 0.0014	58.02	54.80	0.754
16–32, 0.0010	56.49	44.39	0.799
32–64, 0.0010	57.25	49.61	0.774
Config (BS _t –BS _{vt} , lr)	Acc. (%)	BA (%)	Macro-AUC
8–16, 0.0010	78.63	75.94	0.891
16–32, 0.0010	79.39	78.12	0.904
24–48, 0.0014	79.39	77.28	0.903
28–32, 0.0012	80.92	79.21	0.900
36–36, 0.0018	82.44	81.44	0.903
32–64, 0.0010	77.10	77.44	0.894
20–40, 0.0012	76.34	75.10	0.882
16–32, 0.0016	73.28	73.22	0.862
12–24, 0.0007	68.70	64.41	0.835
4–2, 0.0010	71.76	68.00	0.876

Hyperparameters Fine Tuning:
Batch vs Instance Norm.



Python Model vs Liu's Model

C++ RESULTS

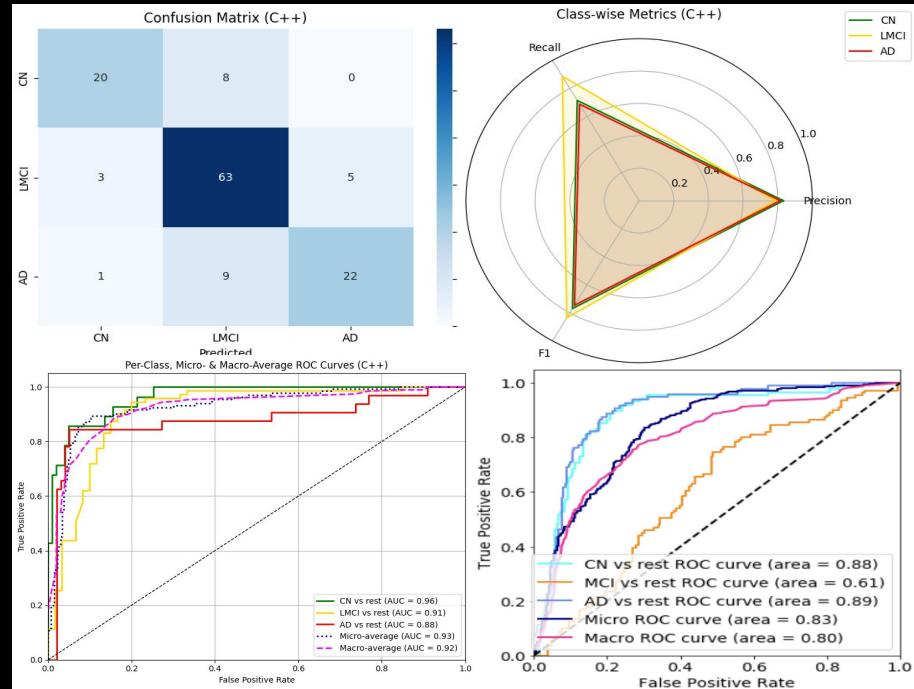
Python Model

Threads	Elapsed (min)	TotalCPU (min)	MaxRSS (MB)
8	88.05	161.55	5538.94
16	93.08	161.25	5440.12
32	98.08	160.87	5440.23
64	49.70	118.23	5528.05

C++

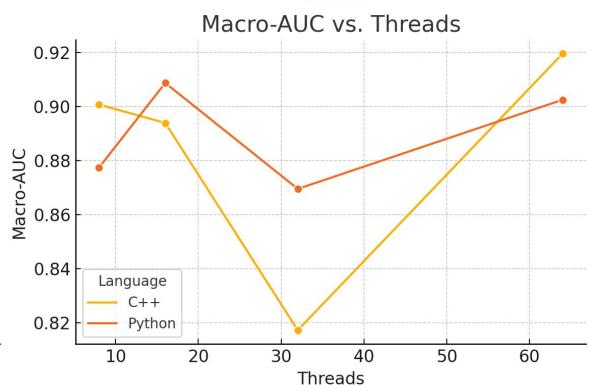
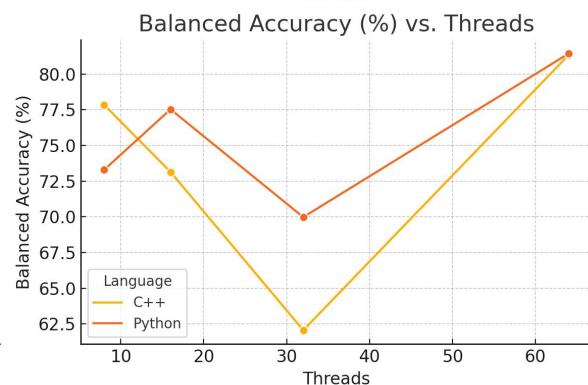
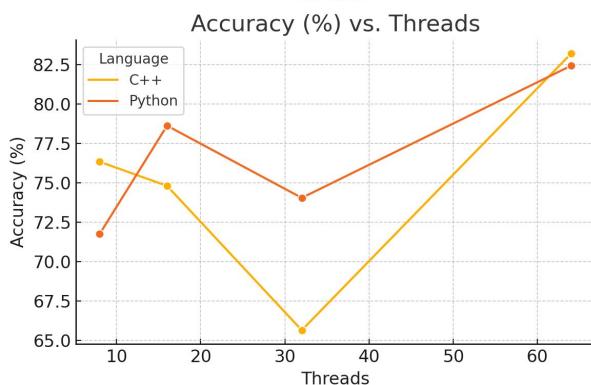
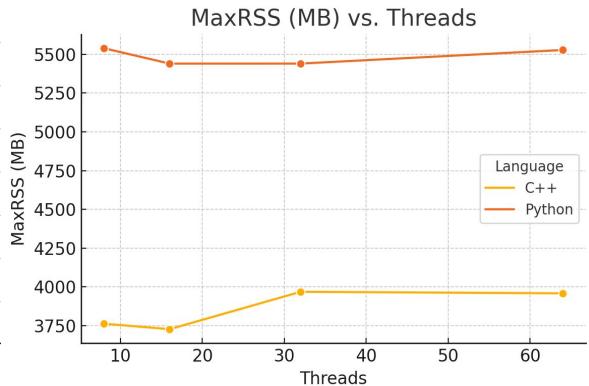
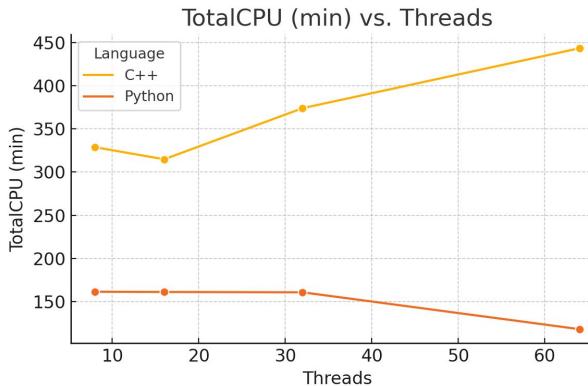
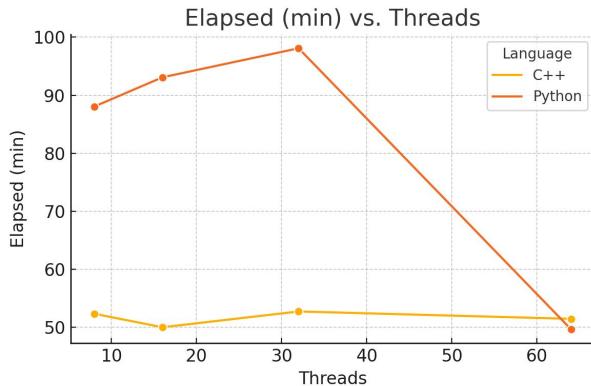
Threads	Elapsed (min)	TotalCPU (min)	MaxRSS (MB)
8	52.35	328.75	3761.94
16	50.03	314.75	3726.84
32	52.75	373.85	3968.07
64	51.48	443.15	3958.01

Scalability: Python vs C++



C++ Model vs Liu's Model

PERFORMANCE: PYTHON vs C++



HOW DID WE OUTPERFORM THE ORIGINAL MODEL?

Our best models achieved a **Macro-AUC of 0.9197** (**C++** model with **LibTorch**, 64 threads) and a **Macro-AUC of 0.903** (**Python** model with **PyTorch**, 64 threads), when classifying between cognitively normal (CN), mild cognitive impairment (MCI), and Alzheimer's disease (AD) using full-brain structural MRI volumes.

This performance **surpasses the original model by Liu et al.** (AUC = 0.8512 for CN vs MCI/AD) and improves upon it in **all per-class AUCs**, particularly for CN classification (AUC = 0.961 vs 0.875).

Key contributors to this boost include:

- A **deeper MLP**, enables learning more complex non-linear feature mappings.
- Use of the **Adam optimizer** (vs. Liu's SGD + momentum), yielding better convergence.
- A **larger batch size (36)** thanks to GPU acceleration on the MeluXina supercomputer.
- **Extended training (100 epochs)** with regular validation.
- **Careful hyperparameter tuning**, benefiting from greater computational flexibility.

The resulting ROC curves confirmed **high separability** across all classes, even on unseen validation data, supporting the robustness and generalization of our approach.



CONCLUSIONS

This project highlights how combining **structural MRI**, **deep learning**, and **high-performance computing** can significantly advance early diagnosis of Alzheimer's disease.

- By using **full-brain 3D CNNs**, we avoided the limitations of traditional ROI-based approaches, allowing the model to learn from the entire brain volume rather than predefined regions. This led to superior classification performance and broader anatomical insight.
- Critical to our success were strong **preprocessing pipelines** (using Clinica and SPM), robust **data augmentation**, and access to **GPU-accelerated infrastructure** via the MeluXina supercomputer. These enabled larger batch sizes, deeper training, and scalable experimentation, key factors in surpassing existing benchmarks.
- Finally, our integration of **Grad-CAM-based visualizations** provided transparent, anatomically meaningful insights into the model's predictions. This step bridges the gap between black-box AI models and clinical interpretability, making the approach more viable in real-world medical settings.

Together, these contributions show how **AI** and **HPC** can be effectively combined to develop accurate, explainable, and scalable tools for neurodegenerative disease detection.



ACKNOWLEDGMENTS

We extend our heartfelt gratitude to Professors Edie Miglio, Paola Antonietti and Luca Formaggia for their unwavering support, guidance, and dedication throughout the entirety of this project. Their expertise and encouragement have been instrumental in every phase of our work.

A special thank you is also due to Politecnico di Milano for offering us the invaluable opportunity to take part in this project. Through its support, we were able to immerse ourselves in the world of academic research and gain a profound appreciation for its challenges and rewards.

This journey allowed us to explore the fascinating intersection of neuroscience and machine learning, two worlds that, when combined, open up remarkable possibilities for understanding the human brain and advancing healthcare. Engaging with these fields side by side has not only deepened our technical and scientific knowledge but has also inspired us to continue pursuing research at the frontier of innovation.



**Vittorio
Cozzoli**



**Alberto
Taddei**



**Tommaso
Crippa**