Autoregressive ConvLSTM Framework for fMRI Time Series Forecasting in Alzheimer's Disease

Ahmed Alshembari[1] **Anima Kujur**[1] **Zahra Monfared**[1,2]
[1]Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, Heidelberg, Germany
[2]Department of Mathematics and Computer Science, Heidelberg University, Heidelberg, Germany
{ahmed.alshembari, anima.kujur, zahra.monfared}@iwr.muni-heidelberg.de

Abstract

Resting-state functional magnetic resonance imaging (fMRI) provides a noninvasive window into brain dynamics and has emerged as a powerful tool for studying neurodegenerative disorders. We develop an autoregressive deep learning framework that employs convolutional long short-term memory (ConvLSTM) units to forecast future brain states in resting-state fMRI sequences from patients with Alzheimer's disease (AD). Unlike traditional linear autoregressive models or hybrid CNN-LSTM approaches, which often ignore spatial structure or flatten brain images, our method integrates convolution directly into the LSTM gates. This design reduces the number of parameters and maintains spatial coherence, preserving the intrinsic 2D structure of brain images while capturing temporal dependencies. To enhance prediction quality, we introduce a custom loss function that jointly optimizes mean squared error and structural similarity index. Experiments on the ADNI fMRI dataset demonstrate that our model generates high-fidelity brain state predictions and achieves substantial performance gains over pure LSTM, and CNN-LSTM baselines. Cross-validation further confirms the robustness of our approach across subjects, which highlights its potential for early biomarker discovery and disease progression monitoring in AD.

1 Introduction

Alzheimer's disease (AD), a leading cause of dementia, is a progressive brain disorder characterized by memory loss, cognitive decline, and widespread neural damage[10, 16]. Pathological hallmarks such as amyloid-beta plaques and tau tangles disrupt large-scale brain networks, including the hippocampus and the default mode network (DMN) [3, 15]. Detecting these disruptions at an early stage is crucial for timely intervention and disease management [32]. Functional magnetic resonance imaging (fMRI) provides a non-invasive window into brain activity through blood-oxygen-level-dependent (BOLD) signals [9, 19]. Traditional analyses rely on static functional connectivity, which assumes stationarity of neural activity, and therefore misses temporal fluctuations in connectivity [14]. Dynamic functional connectivity (dFC) methods attempt to capture such variability but remain limited by window-size constraints and their sensitivity to noise [21]. Deep learning has emerged as a powerful alternative for modeling complex fMRI patterns. Convolutional neural networks (CNNs) effectively learn spatial features, while recurrent architectures such as long-short-term memory (LSTM) networks capture temporal dependencies[6, 24]. However, CNNs ignore temporal dynamics, and LSTMs discard spatial organization by flattening high-dimensional brain images, leading to information loss and inflated parameter counts [13, 17, 18, 27]. Hybrid CNN-LSTM models partly address this issue, but spatial coherence is still compromised when CNN features are vectorized for sequence modeling[5, 29, 31]. Moreover, the integration of CNNs and LSTMs is non-trivial, as naïve combinations often introduce architectural inefficiencies or fail to fully exploit spatiotemporal

NeurIPS 2025 Workshop: Learning from Time-Series for Health.

dependencies. To address these limitations, we propose an autoregressive (AR) framework based on ConvLSTM2D [25, 28]. This approach combines convolutional and recurrent operations *effectively* to maintain spatial structure while modeling temporal dynamics in fMRI sequences. We benchmark this model against LSTM, and CNN-LSTM baselines and assess performance through cross-validation on the Alzheimer's disease neuroimaging initiative (ADNI) dataset. The main contributions of this work are:

- **ConvLSTM-based autoregressive framework:** A ConvLSTM2D model that effectively captures both spatial and temporal dependencies in resting-state fMRI, addressing the limitations of CNN, LSTM, and CNN-LSTM baselines.
- **Custom multi-objective loss:** A loss function that balances voxel-level accuracy with structural coherence by combining mean squared error (MSE), structural similarity index (SSIM), mean absolute error (MAE), and peak signal-to-noise ratio (PSNR).
- **Comprehensive benchmarking:** Benchmarking against CNN, LSTM, and CNN-LSTM baselines shows significant and consistent improvements across all performance metrics.
- **Robust validation:** Five-fold cross-validation on the ADNI dataset confirms robustness and generalizability, with implications for biomarker discovery and modeling AD progression.

2 Related Work

Early fMRI Analysis of AD. Initial fMRI studies relied on static functional connectivity methods such as seed-based correlations and independent component analysis (ICA) [20] to identify disrupted networks such as the DMN [3]. While these methods revealed connectivity reductions in AD patients, they assumed the stationarity of brain activity. Dynamic functional connectivity (dFC) approaches, including sliding-window correlations, introduced temporal variability analysis [1, 21], but remained limited by noise sensitivity and arbitrary window sizes.

Deep Learning in AD fMRI. Deep learning has substantially advanced AD fMRI analysis. CNNs have been widely used to extract spatial features from fMRI volumes. For instance, [23] applied CNNs to resting-state fMRI and achieved a classification accuracy of 96.86%, while [22] combined fMRI and structural MRI to achieve an area under the ROC curve (AUC) of 85.12. Despite their success in classification, CNNs process each time frame independently, thereby neglecting temporal dynamics essential for disease progression modeling [2].

LSTM and Hybrid CNN-LSTM Models. Recurrent networks such as LSTMs can capture temporal dependencies in fMRI data, but they require flattening 3D brain volumes into vectors, which discards spatial organization. Hybrid CNN–LSTM architectures attempt to mitigate this by combining spatial and temporal learning. For example, [26] proposed a multimodal CNN-LSTM for AD classification with $\approx$86% accuracy, while [7] employed 3D-CNNs with bidirectional LSTMs for fMRI, reaching an accuracy of 94.82%. Although effective for classification, these hybrid models disrupt spatial-temporal coherence during feature flattening, which limits their ability to predict fMRI sequences.

ConvLSTM for fMRI Sequence Prediction. Convolutional LSTM (ConvLSTM) networks, originally proposed for video forecasting [25], embed convolution directly into recurrent gates, thereby preserving spatial structure while modeling temporal dynamics. While ConvLSTMs have been applied to fMRI classification [4], their potential for AR sequence prediction in AD has remained largely unexplored. Our work addresses this gap by developing a ConvLSTM2D AR framework tailored for resting-state fMRI forecasting and AD progression modeling.

3 Methodology

The proposed pipeline, summarized in Fig. 1, consists of four stages: data and preprocessing, model architectures, training protocol, and evaluation. The figure illustrates the architectures of the proposed ConvLSTM2D framework and the main CNN-LSTM baseline. For brevity, the pure LSTM baselines are not shown, as they are simple single-component models, but they are described in Sect. 3.2.

3.1 Preprocessing and Training Setup

We used resting-state fMRI data from ADNI, comprising time series of 3D volumes from five patients diagnosed with AD. Each volume was reformatted into a 2D grid of axial slices (704×704). Pixel

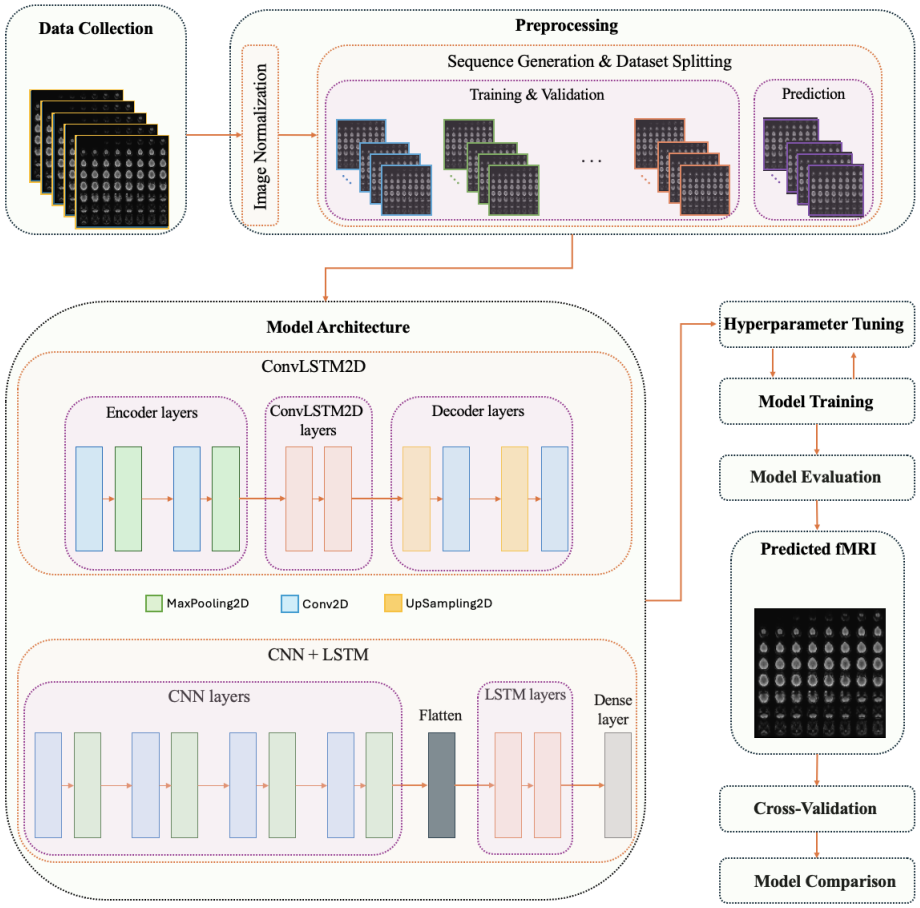


Figure 1: Overview of the proposed methodology.

intensities were normalized to the $[0, 1]$ range. Training sequences were constructed using a sliding window of length 10, where the model predicts the subsequent frame from 10 consecutive inputs (see Appendix B). Dataset statistics are reported in Appendix A. Sequences were split chronologically into training (80%) and validation (20%) sets, with one additional sequence held out for final evaluation. Models were trained for a maximum of 100 epochs with early stopping based on validation loss (see Appendix C).

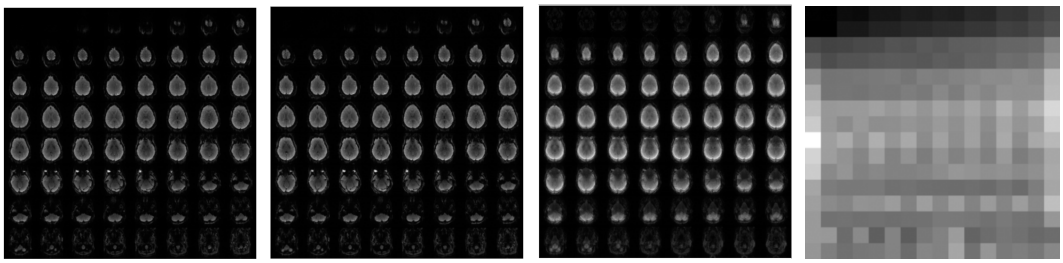
3.2 Models

We evaluate three architectures for next-frame fMRI prediction: (i) **ConvLSTM2D**, our primary model, which embeds convolutional operations within recurrent gates to preserve the 2D spatial structure while modeling temporal dynamics; (ii) **CNN-LSTM**, the main baseline, which first extracts spatial features via convolutional layers and then applies an LSTM on vectorized outputs to capture temporal dependencies; (iii) a **pure LSTM** baseline, which models temporal sequences without spatial convolutions; and Comparative results are presented in Sect. 4.1. Detailed model architectures are described in Appendix D. We measure performance using MSE [11], MAE, SSIM [30], and PSNR [8]. Formal definitions of these metrics are provided in Appendix E.

4 Experiments

4.1 Model Comparison with Optimized Hyperparameters

Hyperparameters for both ConvLSTM2D and CNN-LSTM models were tuned via grid search prior to evaluation. The optimal ConvLSTM2D configuration used two stacked layers with 128–256



(a) Actual fMRI image (b) ConvLSTM prediction (c) CNN-LSTM prediction (d) LSTM prediction

Figure 2: Comparison of actual and predicted fMRI images.

Table 1: Average performance metrics from 5-fold cross-validation. Best per column in **bold**.

Method	MSE	MAE	SSIM	PSNR (dB)
LSTM	0.0181	0.0652	0.5064	17.6526
CNN-LSTM	0.0179	0.0655	0.4210	17.9718
ConvLSTM2D	**0.0003**	**0.0086**	**0.9609**	**35.5123**

units, while the CNN-LSTM performed best with four convolutional layers, as shown in Fig. 1. The full hyperparameter search space and the corresponding optimal configurations are provided in Appendix G. Using these tuned models, we compared the performance of ConvLSTM2D, CNN-LSTM, LSTM, in predicting next-step fMRI frames. On the reserved prediction sequence, the ConvLSTM2D model achieved high predictive accuracy and structural fidelity, with an MSE of 0.00028, MAE of 0.0082, SSIM of 0.9621, and PSNR of $35.8921\,\mathrm{dB}$. In contrast, the baseline CNN-LSTM yielded an MSE of 0.0032, MAE of 0.0286, SSIM of 0.7400, and PSNR of $24.9910\,\mathrm{dB}$. The pure LSTM, yielded an MSE of 0.0294, MAE of 0.0843, SSIM of 0.4686, and PSNR of $15.3236\,\mathrm{dB}$. Qualitatively, the predictions of the pure LSTM (Fig. 2d) failed to capture the fine-grained spatial structure of the fMRI frames, appearing blurred and highly pixelated compared to the actual data. This observation is consistent with the quantitative results, which confirm the model's limited ability to represent spatial information when temporal dependencies are modeled without convolutional operations. By contrast, higher SSIM and PSNR values for ConvLSTM2D indicate superior preservation of structural details and image quality, as visually confirmed in Fig. 2b. Overall, these results demonstrate the superior ability of ConvLSTM2D to capture spatio-temporal patterns, which is critical for AR prediction in AD research.

4.2 Model performance under cross-validation

To evaluate the robustness and generalizability of the proposed AR ConvLSTM2D model, a 5-fold cross-validation was conducted. Performance was assessed using MSE, MAE, SSIM, and PSNR, with metrics averaged across all folds (Table 1). These results indicate that the ConvLSTM2D outperforms in achieving lower error values and higher structural similarity and image quality while also generalizing reliably across different data splits, underscoring its potential for robust spatio-temporal modeling.

5 Conclusion and Future Work

This work introduces a novel AR framework using a ConvLSTM2D model to predict future brain states in AD patients from resting-state fMRI data. By combining convolutional operations for spatial features with recurrent dynamics for temporal dependencies, the model learns spatio-temporal patterns that are important for AD progression. The proposed ConvLSTM2D model outperformed pure LSTM and CNN-LSTM baselines in predicting future brain states, showing it can model long-term changes of the brain activity. This reveals the potential of combining spatial and temporal learning for forecasting in neurodegenerative research.

Limitations: The main limitations include the scarcity of high-quality, well-annotated fMRI datasets for AD, particularly those with longitudinal labels suitable for progression modeling, which may limit generalization. The model was designed only for fMRI sequence prediction and did not incorporate multimodal inputs, such as MRI, positron emission tomography (PET), or clinical measures. In addition, residual noise from fMRI preprocessing may still influence dynamic connectivity signals.

Future work: Future directions include improving and stabilizing training strategies to reduce error accumulation in long AR rollouts, as well as designing loss functions that enforce temporal consistency or emphasize clinically important regions. Another avenue is to explore alternative RNN architectures, such as shallow piecewise linear RNNs (shPLRNNs) [12], instead of LSTM. When combined with generalized teacher forcing, shPLRNNs have been shown to yield strictly bounded loss gradients during training, even on complex or chaotic data. Furthermore, extending the framework to incorporate multimodal inputs and integrating explainable AI techniques may enhance the clinical applicability of predictive modeling for AD progression. Finally, addressing the inherent *sparsity* of AD datasets remains an important challenge for developing robust and reliable models.

Acknowledgements

Z.M. is grateful to the Bundesministerium für Forschung, Technologie und Raumfahrt (BMFTR, Federal Ministry for Research, Technology and Space) for funding through project OIDLITDSM, No. 01IS24061.

References

[1] E. A. Allen, E. Damaraju, S. M. Plis, E. B. Erhardt, T. Eichele, and V. D. Calhoun. Tracking whole-brain connectivity dynamics in the resting state. *Cerebral Cortex*, 24(3):663–676, 2014.

[2] Andrew P Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

[3] R. L. Buckner, J. R. Andrews-Hanna, and D. L. Schacter. The brain's default network: Anatomy, function, and relevance to disease. *Annals of the New York Academy of Sciences*, 1124(1):1–38, 2008.

[4] Yaojia Chen, Jiacheng Wang, Chuyu Wang, Mingxin Liu, and Quan Zou. Deep learning models for disease-associated circrna prediction: a review. *Briefings in bioinformatics*, 23(6):bbac364, 2022.

[5] Mohit Dua, Drishti Makhija, PYL Manasa, and Prashant Mishra. A cnn–rnn–lstm based amalgamation for alzheimer's disease detection. *Journal of Medical and Biological Engineering*, 40(5):688–706, 2020.

[6] Nicha C Dvornek, Pamela Ventola, Kevin A Pelphrey, and James S Duncan. Identifying autism from resting-state fmri using long short-term memory networks. In *International Workshop on Machine Learning in Medical Imaging*, pages 362–370. Springer, 2017.

[7] Chiyu Feng, Ahmed Elazab, Peng Yang, Tianfu Wang, Feng Zhou, Huoyou Hu, Xiaohua Xiao, and Baiying Lei. Deep learning framework for alzheimer's disease diagnosis via 3d-cnn and fsbi-lstm. *IEEE Access*, 7:63605–63618, 2019.

[8] Rafael C Gonzalez and Richard E Woods. *Digital image processing*. Pearson, 3 edition, 2008.

[9] M. D. Greicius, G. Srivastava, A. L. Reiss, and V. Menon. Default-mode network activity distinguishes alzheimer's disease from healthy aging: Evidence from functional mri. *Proceedings of the National Academy of Sciences*, 101(13):4637–4642, 2004.

[10] H Hampel, F Padberg, K Buch, J Unger, S Stübner, and HJ Möller. Diagnosis and treatment of alzheimer-type dementia. *Deutsche Medizinische Wochenschrift (1946)*, 124(5):124–129, 1999.

[11] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: Data mining, inference, and prediction*. Springer, 2 edition, 2009.

[12] Florian Hess, Zahra Monfared, Manuel Brenner, and Daniel Durstewitz. Generalized teacher forcing for learning chaotic dynamics. *arXiv preprint arXiv:2306.04406*, 2023.

[13] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

[14] R. M. Hutchison, T. Womelsdorf, E. A. Allen, P. A. Bandettini, V. D. Calhoun, M. Corbetta, et al. Dynamic functional connectivity: Promise, issues, and interpretations. *NeuroImage*, 80: 360–378, 2013.

[15] C. R. Jack, D. A. Bennett, K. Blennow, M. C. Carrillo, B. Dunn, S. B. Haeberlein, et al. Nia-aa research framework: Toward a biological definition of alzheimer's disease. *Alzheimer's & Dementia*, 14(4):535–562, 2018.

[16] David S Knopman, Helene Amieva, Ronald C Petersen, Gäel Chételat, David M Holtzman, Bradley T Hyman, Ralph A Nixon, and David T Jones. Alzheimer disease. *Nature reviews Disease primers*, 7(1):33, 2021.

[17] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[18] Jiyun Li, Binbin Song, and Chen Qian. Diagnosis of alzheimer's disease by feature weighted-lstm: a preliminary study of temporal features in brain resting-state fmri. *Journal of Integrative Neuroscience*, 21(2):56, 2022.

[19] Nikos K Logothetis. The neural basis of the blood–oxygen–level–dependent functional magnetic resonance imaging signal. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 357(1424):1003–1037, 2002.

[20] Ganesh R Naik and Dinesh K Kumar. An overview of independent component analysis and its applications. *Informatica*, 35(1), 2011.

[21] M. G. Preti, T. A. Bolton, and D. Van De Ville. The dynamic functional connectome: State-of-the-art and perspectives. *NeuroImage*, 160:41–54, 2017.

[22] Shangran Qiu, Prajakta S Joshi, Matthew I Miller, Chonghua Xue, Xiao Zhou, Cody Karjadi, Gary H Chang, Anant S Joshi, Brigid Dwyer, Shuhan Zhu, et al. Development and validation of an interpretable deep learning framework for alzheimer's disease classification. *Brain*, 143(6): 1920–1933, 2020.

[23] Saman Sarraf and Ghassem Tofighi. Deep learning-based pipeline to recognize alzheimer's disease using fmri data. In *2016 future technologies conference (FTC)*, pages 816–820. IEEE, 2016.

[24] D. Shen, G. Wu, and H.-I. Suk. Deep learning in medical image analysis. *Annual Review of Biomedical Engineering*, 19:221–248, 2017.

[25] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 802–810, 2015.

[26] S. Spasov, L. Passamonti, A. Duggento, P. Liò, and N. Toschi. A parameter-efficient deep learning approach to predict conversion from mild cognitive impairment to alzheimer's disease. *NeuroImage*, 189:276–287, 2019.

[27] Haijing Sun, Anna Wang, and Shanshan He. Temporal and spatial analysis of alzheimer's disease based on an improved convolutional neural network and a resting-state fmri brain functional network. *International Journal of Environmental Research and Public Health*, 19(8): 4508, 2022.

[28] Oskar Triebe, Nikolay Laptev, and Ram Rajagopal. Ar-net: A simple auto-regressive neural network for time-series. *arXiv preprint arXiv:1911.12436*, 2019.

[29] Ruofan Wang, Qiguang He, Chunxiao Han, Haodong Wang, Lianshuan Shi, and Yanqiu Che. A deep learning framework for identifying alzheimer's disease using fmri-based brain network. *Frontiers in Neuroscience*, 17:1177424, 2023.

[30] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[31] Simon Wein, Alina Schüller, Ana Maria Tomé, Wilhelm M Malloni, Mark W Greenlee, and Elmar W Lang. Forecasting brain activity based on models of spatiotemporal brain dynamics: A comparison of graph neural network architectures. *Network Neuroscience*, 6(3):665–701, 2022.

[32] Yara Yakoub, Nicholas J Ashton, Cherie Strikwerda-Brown, Laia Montoliu-Gaya, Thomas K Karikari, Przemysław R Kac, Fernando Gonzalez-Ortiz, Jonathan Gallego-Rudolf, Pierre-François Meyer, Frédéric St-Onge, et al. Longitudinal blood biomarker trajectories in preclinical alzheimer's disease. *Alzheimer's & Dementia*, 19(12):5620–5631, 2023.

A Dataset Description

We used resting-state fMRI data from the ADNI database (`http://adni.loni.usc.edu`, accessed 31 October 2024). The dataset includes a total of 2,026 3D fMRI volumes from five AD patients (denoted P1–P5). Table 2 summarizes the number of volumes per patient.

Table 2: Summary of ADNI fMRI dataset used in this study.

Patient ID	**Number of Volumes**
P1	482
P2	482
P3	482
P4	369
P5	211
Total	2026

Each 3D volume was reformatted into a 2D image by arranging axial slices into a 704×704 grid. This produced sequences of the form

$$X \in \mathbb{R}^{T\times 704\times 704}, \quad T = 2026,$$

where T is the total number of time points across patients.

B Preprocessing and Sequence Generation

Pixel intensities were normalized to the $[0, 1]$ range by dividing each image by its maximum intensity. Training sequences were constructed using a sliding window of length 10. Specifically, for time index $j \in \{1, \ldots, T - 10\}$, the input was

$$[I_j, I_{j+1}, \ldots, I_{j+9}],$$

and the target was I_{j+10}. This yielded $M = T - 10$ training samples.

C Training Protocol

From the generated sequences, one ten-frame sequence was reserved as a held-out test set. The remaining sequences were divided into training and validation sets using an 80/20 chronological split (no shuffling to preserve temporal dependencies).

All models were trained for up to 100 epochs with early stopping based on validation loss, restoring the best-performing weights. Full training hyperparameters are summarized in Appendix G.

D Models

D.1 ConvLSTM2D Architecture

The ConvLSTM2D model employs an encoder-decoder framework. The encoder applies TimeDistributed Conv2D layers to extract spatial features from each input sequence frame, followed by MaxPooling2D layers to downsample and reduce computational complexity. The core includes one or more ConvLSTM2D layers, extending traditional LSTMs with convolutional operations, as defined by:

$$\begin{aligned}
i_t &= \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + b_i),\\
f_t &= \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + b_f),\\
c_t &= f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c),\\
o_t &= \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + b_o),\\
H_t &= o_t \odot \tanh(c_t).
\end{aligned}$$

where $*$ denotes convolution, X_t is the input at time t, H_t is the hidden state, and c_t is the cell state. The decoder uses UpSampling2D layers to restore spatial dimensions and a Conv2D layer to predict frames with the same dimension as the input. The model iteratively predicts frames by feeding outputs as inputs.

Training uses a custom loss function combining MSE and SSIM, defined as

$$\mathcal{L} = \alpha \cdot \mathrm{MSE} + (1 - \alpha) \cdot \big(1 - \mathrm{SSIM}\big).$$

where $\alpha \in [0, 1]$ balances the contributions of MSE (pixel-wise accuracy) and SSIM (structural fidelity), tuned empirically to optimize performance for AD fMRI sequences, as detailed in Sect. **??**.

D.2 CNN-LSTM Architecture

The CNN-LSTM model separates spatial and temporal processing into two distinct stages. In the first stage, a stack of four 2D convolutional layers (each followed by ReLU activations and MaxPooling2D operations) is applied in a TimeDistributed manner to extract spatial features from individual frames of the fMRI sequence. This convolutional block encodes each frame into a compact representation that preserves salient local patterns while reducing dimensionality.

In the second stage, these extracted frame-level embeddings are passed to stacked LSTM layers that capture temporal dependencies across the sequence. Unlike ConvLSTM2D, where convolutional operations are integrated directly within the recurrent cell, CNN-LSTM treats the temporal and spatial learning separately: CNN layers encode spatial features, and LSTM layers model their evolution over time. Formally, given an input sequence $\{X_t\}_{t=1}^{T}$, the feature representation for frame t is

$$F_t = \mathrm{CNN}(X_t),$$

which is then processed sequentially by the LSTM as

$$H_t, c_t = \mathrm{LSTM}(F_t, H_{t-1}, c_{t-1}),$$

where H_t and c_t denote the hidden and cell states, respectively.

The final LSTM output is passed through a dense layer with a sigmoid activation to produce predictions. The model is trained with mean squared error (MSE) loss, focusing on frame-wise reconstruction accuracy without explicit structural similarity regularization.

D.3 Pure LSTM Architecture (no convolutions)

As a convolution-free baseline, we evaluate a purely recurrent model that separates per-frame embedding from temporal modeling and uses a non-convolutional decoder. Input series are resized to 704×704, normalized to $[0, 1]$ by the per-frame maximum, and AR training pairs are formed from 10 input frames to predict the $(t{+}1)$ frame (subject-wise 5-fold CV, one subject held out per fold).

Encoder (per frame). Each frame is downsampled to 44×44, flattened, layer-normalized, and projected to a 128-D embedding via a Dense layer with ReLU:

$$F_t = \mathrm{Dense}_{128}\big(\mathrm{LayerNorm}(\mathrm{Flatten}(\mathrm{Resize}_{44\times44}(X_t)))\big).$$

Temporal core. The sequence $\{F_t\}_{t=1}^{10}$ is processed by a single `LSTM(128)` (no return of intermediate states), yielding a latent vector h which is further mapped by a Dense layer:

$$h = \mathrm{LSTM}_{128}(F_{1:10}), \quad z = \mathrm{Dense}_{128}(h).$$

Decoder. The latent z is expanded to a $16\times16\times64$ seed grid with a Dense+Reshape block. The grid is upsampled four times with nearest-neighbor `UpSampling2D`; after each upsample, a per-pixel Dense (functionally a 1×1 conv) mixes channels and halves width until a floor of 16 channels. Finally, the feature map is resized back to 704×704 and a Dense(1) with sigmoid produces the next-frame prediction.

E Evaluation Metric Definitions

For completeness, we define the performance metrics used in our experiments. Let $\mathbf{y}$ denote a ground–truth image and $\hat{\mathbf{y}}$ its prediction, each with N pixels. The mean squared error (MSE) measures pixel–level accuracy:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}\big(y_i - \hat{y}_i\big)^2.$$

The mean absolute error (MAE) quantifies the average absolute difference between corresponding pixels:

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|.$$

The structural similarity index (SSIM) assesses perceptual similarity between two images x and y and is defined as

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)},$$

where μ_x, μ_y are the mean intensities, σ_x, σ_y are the standard deviations, σ_{xy} is the covariance, and C_1, C_2 are small constants to stabilize the division. SSIM values range from -1 (dissimilar) to 1 (identical).

The peak signal–to–noise ratio (PSNR) evaluates overall image quality in decibels:

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}}\right),$$

where MAX denotes the maximum possible pixel value (1 for normalized images).

F Hardware

The computational efficiency of the AR ConvLSTM2D model, applied to fMRI sequence prediction for Alzheimer's disease (AD) research, was assessed by measuring both training and inference times. All experiments were conducted on a high-performance computing system equipped with **2 × 64-Core AMD EPYC 9534 CPUs**, **2048GB RAM**, and **8 × NVIDIA H200 GPUs**.

The ConvLSTM2D model was trained with for 100 epochs, using the optimal hyperparameters identified through grid search (as detailed in Appendix G). Each epoch averaged **7 minutes**, resulting in a total training time of approximately **11.67 hours**. Validation, performed after each epoch, was significantly faster, averaging **30 seconds per epoch**. For inference, the model processed a single sequence in approximately **15 milliseconds**, demonstrating its potential for real-time applications in fMRI sequence prediction. The AR nature of the model, which predicts future frames iteratively, introduces additional computational overhead during inference compared to non-autoregressive models. However, this is offset by the efficient architecture and high-performance hardware, enabling near real-time predictions. Disabling shuffling during training preserved the temporal dependencies of the fMRI data, further optimizing computational efficiency. Overall, the model achieved a balance between predictive accuracy and computational performance, making it suitable for large-scale fMRI analysis in AD research.

The baseline models exhibited expected computational profiles: the pure LSTM was considerably slower and memory-intensive because of the need to flatten high-dimensional fMRI volumes; and the hybrid CNN-LSTM achieved intermediate efficiency but still required flattening of CNN features, which limited both scalability and accuracy. In contrast, the ConvLSTM2D offered a favorable balance by jointly modeling spatial and temporal patterns with manageable computational overhead.

These computational characteristics informed the subsequent cross-validation evaluation, as discussed in Sect. 4.2.

Table 3: Optimal hyperparameter configurations for ConvLSTM2D and CNN-LSTM models.

Parameter	**ConvLSTM2D**	**CNN-LSTM**
Convolutional depth	2	4
Number of filters	64	64
LSTM units	512	256
Number of layers	2	2
Learning rate	1×10^{-4}	1×10^{-3}
Filter size	(3,3)	(3,3)
Pool size	(2,2)	(2,2)
Epochs	100	50
Optimizer	Adam	Adam
Validation loss	0.1316 (MSE+SSIM)	3.9e-3 (MSE)

Table 4: Architecture-specific hyperparameters for the pure LSTM baseline.

Per-frame downsample	44×44 (resize)
Per-frame embedding	Dense 128 (after flatten)
Temporal core	`LSTM(128)` (return_sequences=False)
Latent projection	Dense 128
Decoder seed grid	$16 \times 16 \times 64$
Upsampling stages	4 (nearest neighbor)
Channel mixing	Per-pixel Dense ($\approx 1\times1$ conv) after each upsample
Output layer	Sigmoid to 1 channel

G Optimal Hyperparameter Configurations

G.1 ConvLSTM2D and CNN-LSTM Models

Hyperparameters for ConvLSTM2D and CNN-LSTM were tuned via grid search. The search space included: convolutional depth $\{1, 2, 3, 4\}$, number of filters $\{16, 32, 64\}$, ConvLSTM2D units $\{64, 128, 256, 512\}$, and layers $\{1, 2\}$. All models used ReLU activations in intermediate layers, a sigmoid output layer, (3×3) filters, and (2×2) max pooling. ConvLSTM2D was trained with a composite MSE+SSIM loss, while CNN-LSTM was optimized with MSE loss.

The grid search indicated that depths beyond 2 gave diminishing returns for ConvLSTM2D, whereas CNN-LSTM benefited from 4 convolutional layers. The final selected configurations and validation losses are summarized in Table 3.

G.2 Pure LSTM Model

Table 4 summarizes the configuration used for the pure LSTM baseline. Unlike CNN-LSTM and ConvLSTM2D, we did not perform a hyperparameter search; model-specific hyperparameters were fixed at default values.