AUTOMATIC ARTIFACT REMOVAL OF RESTING-STATE FMRI WITH DEEP NEURAL NETWORKS

Christos Theodoropoulos[⋆] *Christos Chatzichristos*[†] *Sabine Van Huffel*[†]

[⋆] Department of Computer Science, LIIR Lab, KU Leuven, Belgium
[†] Department of Electrical Engineering, STADIUS, KU Leuven, Belgium

ABSTRACT

Functional Magnetic Resonance Imaging (fMRI) is a non-invasive technique for studying brain activity. During an fMRI session, the subject executes a set of tasks (task-related fMRI study) or no tasks (resting-state fMRI), and a sequence of 3-D brain images is obtained for further analysis. In the course of fMRI, some sources of activation are caused by noise and artifacts. The removal of these sources is essential before the analysis of the brain activations. Deep Neural Network (DNN) architectures can be used for denoising and artifact removal. The main advantage of DNN models is the automatic learning of abstract and meaningful features, given the raw data. This work presents advanced DNN architectures for noise and artifact classification, using both spatial and temporal information in resting-state fMRI sessions. The highest performance is achieved by a voting schema using information from all the domains, with an average accuracy of over 98% and a very good balance between the metrics of sensitivity and specificity (98.5% and 97.5% respectively).

Index Terms— Resting-state fMRI, Independent Component Analysis, Denoising, Deep Neural Networks

1. INTRODUCTION

Currently, one of the most widely used techniques for studying and analyzing brain connectivity and activity is fMRI. During an fMRI experiment, random noise and artifacts are introduced (e.g. heartbeat, head motion, thermal noise, etc). Moreover, the noise can be related to the specific hardware and the nature of the experiment. A successful and substantial analysis of the fMRI session requires high quality, noise-free data. Hence, the robust denoising and artifact removal is a crucial step of the fMRI processing [1]. This task is challenging because some types of noise are difficult to be detected due to the fact that they are very rare or quite similar to regular components [2].

Blind Source Separation (BSS) [3] is a very important step for interpreting and analyzing the fMRI data. The localization of the activated brain areas is a challenging BSS task, in which the sources consist of a combination of spatial maps (areas activated) and time-courses (timings of activation) [4]. The sources should be classified, for clean-up purposes, as artifacts or neuronal signals. Both temporal and spatial information is used to categorize the source as noise/artifact or neuronal signal, the sources classified as artifacts are removed during the reconstruction of the signal. Independent Component Analysis (ICA) [5] is a statistical method which tries to find a linear transformation of the observable space into a new space such that the individual new variables are mutually independent. ICA is a powerful technique for separating the various source of fluctuations and, ICA assumes that statistically independent spatial maps are mixed with the use of corresponding time-courses in an associated (mixing) matrix.

The most widely used Machine Learning based approach for artifact removal is FIX ("FMRIB's ICA-based X-noiseifier") [6], [7]. It is an ICA-based framework using FastICA algorithm (as implemented in Melodic toolbox [8]). Principal Component Analysis (PCA) [9] is applied as a preprocessing step, for dimensionality reduction and reduction of unstructured noise. The features (over 180) are manually engineered in order to capture aspects of spatial maps (e.g. size of the clusters and voxels overlaying bright/dark raw data voxels) timeseries, and frequency spectrum (e.g. autoregressive and distributional properties, jump amplitudes). The hand-crafted features are sensitive to the acquisition and preprocessing parameters. Hence, the re-training of the model is essential when the data differ a lot from the initial data, which were used for the training of the models. Finally, multiple different classifiers are stacked in order to extract the final decision.

In the view of the DNN success in various biomedical problems [10], [11], a Deep Learning [12] framework is proposed for automatic noise and artifact detection in resting state fMRI data [13], which exhibits good performance. The dataset of the study is taken from Baby Connectome Project (BCP [14]) and contains resting state sessions from 32 subjects/infants. ICA is applied on the data and 150 components per subject are extracted. Trained raters decided whether a component is related to noise or a nuisance signal. Normalization (standardization) is applied on each extracted 3D spatial map. The proposed framework contains a 3D Convolution Neural Network (CNN) [15] model which receives the spatial maps as input and extracts meaningful spatial features. The temporal information is analyzed by a 1D CNN model, which learns high-level temporal representations. For each convolu-

tional layer, ReLU is used as activation function. Finally, a stack of fully connected layers is added in each model, in order to perform the classification of the component (signal or noise/artifact). A majority voting schema is also applied for the final classification.

In this study, advanced Deep Learning architectures are used for denoising and artifact removal. Having as starting point the proposed models of [13], we want to explore the effectiveness of more complex architectures and the addition of frequency information as input. The labeled extracted components are used for training and evaluation of the Deep Learning models. After necessary data-preprocessing and manipulation, the training and testing of the different DNN models are executed. The performance is tested, using the spatial, temporal, and frequency information independently and jointly. The main outcome of the study is twofold; spectral information boosts the overall performance and a weighted voting schema achieves the best results.

2. DEEP LEARNING METHODS

The proposed DNN models can be separated based on the given input (spatial, temporal, and frequency). The main layer of the models using 3D spatial maps as input is the convolutional layer, which is capable of extracting high-level feature representations taking into account the local connectivity between the elements of the input. We employ models using both temporal and frequency information in order to test whether the assumption used in [13], that a neural network using temporal information can infer all the meaningful frequency features, is valid, and whether we can improve the total performance.

2.1. Models using spatial information

The first model (CNN_{sm_1}), shown in Fig. 1, is similar to the one proposed in [13] and is considered as the baseline model. The main difference is that the stride of every convolution operation is set to 1, while in [13] stride values of 2 and 3 are used. The second model (CNN_{sm_2}, Fig. 2) has a slight difference with the first one. Batch Normalization (BN) [16] layers are used after each convolutional layer. BN layer [17], [18] helps the network to get trained smoother and faster, decreases the sensitivity to the weight initialization and can be used as a type of regularization. Hence, the second model tests whether the addition of BN layer is advantageous in our task. The third model (CNN_{sm_3}, Fig. 3) includes the idea of residual blocks. This type of block is initially proposed in ResNet architecture [19] and contains skip connections, which help the network to learn additional residual features. Learning residual features boosts the performance in many computer vision tasks [19], [20]. Hence, we want to investigate whether residual blocks are efficient in our study. ReLU is used as the activation function in all of the layers (3D convolutional layers and fully connected layers) of the proposed models.

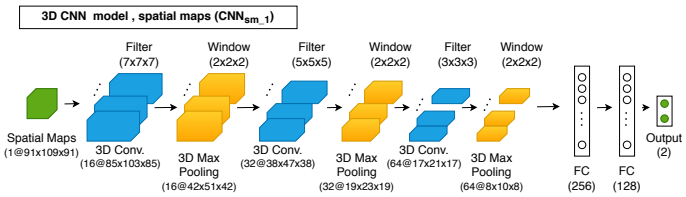

Fig. 1: CNN_{sm_1} model

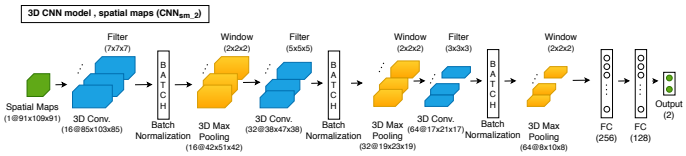

Fig. 2: CNN_{sm_2} model

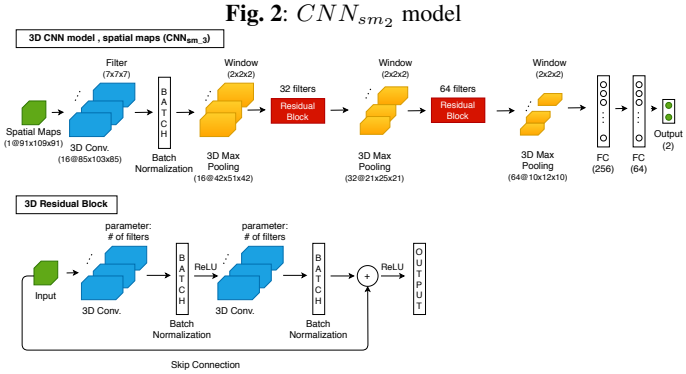

Fig. 3: CNN_{sm_3} model

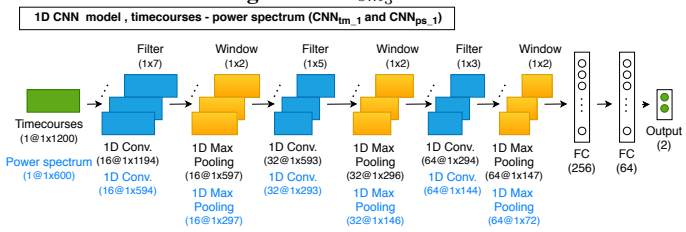

Fig. 4: CNN_{tm_1} and CNN_{ps_1} models

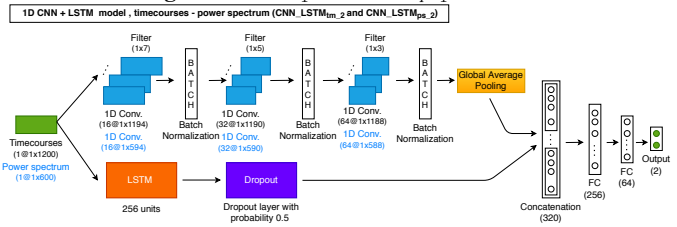

Fig. 5: $CNN-LSTM_{tm_2}$ and $CNN-LSTM_{ps_2}$ models

Only the last output layer uses the Sigmoid activation function in order to extract the final probability (1: perfect noise, 0: pure signal).

2.2. Models using temporal and frequency information

The proposed architectures using temporal and frequency information are identical. The first model (CNN_{tm_1}, CNN_{ps_1}, Fig. 4), which is used as baseline model, employs a sequence of 1D convolutional and max pooling layers. It is similar to the model proposed in [13]. The second model ($CNN-LSTM_{tm_2}$, $CNN-LSTM_{ps_2}$, Fig. 5) introduces a parallel architecture, which also includes an LSTM

block [21], followed by a dropout layer. The usage of LSTM block [22], [23] provides the capability of learning long-term time-dependent patterns. The dropout layer is used for regularization in order to avoid overfitting.

3. RESULTS

The dataset consists of resting-state fMRI data of young healthy adults from Human Connectome Project [24], [25]. ICA is applied on the data for unmixing the different sources and each extracted component is labeled as noise/artifact or signal. The first step of the experimental process is the separation of the three different subsets of the dataset: training, validation, and test set. Taking into account the computational cost of the training of the models, 80 subjects are included in the training set and 20 subjects in the validation set. In the training set, a random sampling is performed for each different split (5-fold cross-validation technique) in order to balance the classes and handle the imbalance problem, as the class which contains the noisy components is dominant. The remaining 294 subjects are used as test set. Hence, as the number of subjects in the test set is large, the evaluation process indicates robustly the generalization capabilities of the models.

As 5-fold cross-validation technique is used, the models are trained five times. For all the models, Adam [26] is used as optimizer with learning rate equal to 0.001. For the models using spatial information (CNN_{sm_1}, CNN_{sm_2}, and CNN_{sm_3}) the batch size is set to 16 and early stopping is applied after 3 epochs, when no performance improvement is achieved in the validation set. For the models using temporal and frequency information (CNN_{tm_1}, $CNN-LSTM_{tm_2}$, CNN_{ps_1}, and $CNN-LSTM_{ps_2}$) the batch size is set to 128 and early stopping is applied after 4 epochs.

Other than training the different models separately, we also train four combinations of them with the addition of a concatenation layer and two fully connected layers with 128 and 32 neurons, in order to check for a possible increment in the performance. The tested combinations are the following:

- $Comb_1$: CNN_{sm_1}, CNN_{tm_1} and CNN_{ps_1}
- $Comb_2$: CNN_{tm_1} and CNN_{ps_1}
- $Comb_3$: CNN_{sm_1} and CNN_{tm_1}
- $Comb_4$: $CNN-LSTM_{tm_2}$ and $CNN-LSTM_{ps_2}$.

The final step of the experimental procedure is the evaluation phase. All the trained models are evaluated in the same test set. Accuracy, precision, sensitivity, and specificity are calculated. The final predictions are extracted separately from each trained model, however different voting schemes using the extracted probabilities are also applied. The models are tested using 294 subjects (test set). For each split (5-fold cross-validation) the four performance metrics (accuracy: ACC, precision: PREC, sensitivity: SEN, and specificity: SPEC) are calculated. Moreover, a voting schema for the final decision is applied in order to evaluate whether combinations of the distinct models result in better performance.

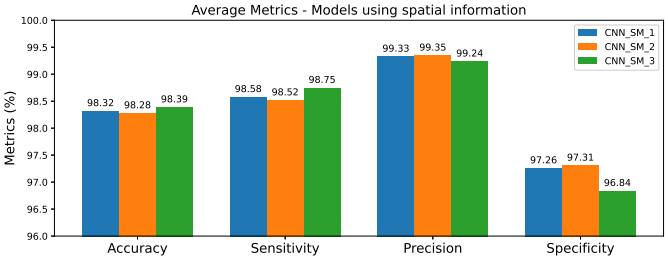

Fig. 6: Evaluation of CNN_{sm_1}, CNN_{sm_2}, and CNN_{sm_3} models - Average metrics

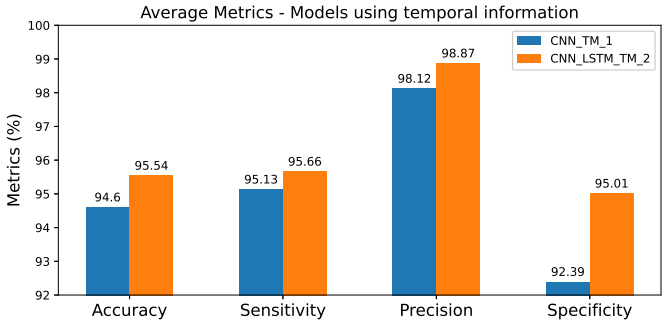

Fig. 7: Evaluation of CNN_{tm_1} and $CNN-LSTM_{tm_2}$ models - Average metrics

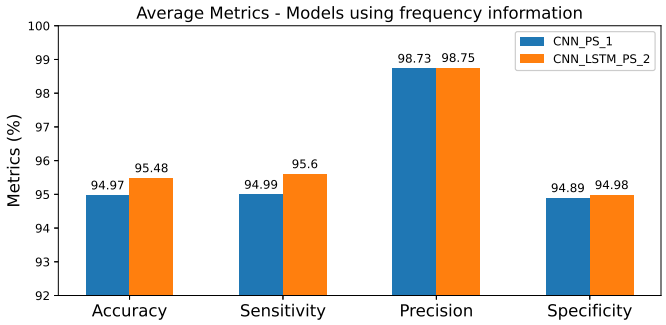

Fig. 8: Evaluation of CNN_{ps_1} and $CNN-LSTM_{ps_2}$ models - Average metrics

Model	ACC	SEN	PREC	SPEC
$Comb_1$	95.66	96	98.59	94.29
$Comb_2$	95.62	95.69	**98.85**	**95.37**
$Comb_3$	95.77	96.48	98.26	92.83
$Comb_4$	**96.27**	**96.9**	98.46	94.67

Table 1: Evaluation of the combined models - Average metrics (%)

A general description of the weighted voting schemes with n different models is the following:

$$Prob_{out} = w_1 Prob_1 + ... + w_n Prob_n, \sum_{i=1}^{n} w_i = 1, \quad (1)$$

where w_i and $Prob_i$ are the voting weight and the extracted probability of the ith model, respectively. If $Prob_{out} > 0.5$ (threshold) then the component is considered classified as an artifact, else it is classified as a neuronal signal. Both the time and frequency information are derived from the same data (time courses of the mixing matrix), hence, we selected

the weights in order to balance the contribution of the spatial maps and time courses in the decision function. The evaluated voting schemes (inside the parentheses are the corresponding voting weights) are the following:

- $Schema_1$: CNN_{sm_1} (0.5), CNN_{tm_1} (0.25), and CNN_{ps_1} (0.25)
- $Schema_2$: CNN_{sm_1} (0.5), $CNN-LSTM_{tm_2}$ (0.25), and $CNN-LSTM_{ps_2}$ (0.25)
- $Schema_3$: CNN_{sm_2} (0.5), $CNN-LSTM_{tm_2}$ (0.25), and $CNN-LSTM_{ps_2}$ (0.25)
- $Schema_4$: $CNN-LSTM_{tm_2}$ (0.5), and $CNN-LSTM_{ps_2}$ (0.5)

Figure 6 indicates that the performance of the three different models using spatial information is very similar. The accuracy is over 98%, so the possible improvement is limited. The addition of the residual blocks (CNN_{sm_3} model) increases the complexity of the model, but the performance does not improve significantly. Moreover, BN layers which are included in CNN_{sm_2} model do not affect the performance.

The models using temporal information (CNN_{tm_1} and $CNN-LSTM_{tm_2}$) perform worse than those using spatial information as the accuracy decreases approximately by 3%. The high resolution of the spatial maps is an important aspect of the models' efficiency. Figure 7 shows that the addition of the LSTM block in $CNN-LSTM_{tm_2}$ model results in better performance as the model is capable of learning better the sequential patterns. The models using frequency information (CNN_{ps_1} and $CNN-LSTM_{ps_2}$) perform similarly to the models using temporal information. The $CNN-LSTM_{ps_2}$ model with the LSTM block achieves better performance (Figure 8).

The evaluation of the combined models $Comb_1$ and $Comb_3$ (Table 1) demonstrates that the end-to-end training using multiple sources of information (spatial, temporal, and frequency) is not advantageous. The $Comb_2$ and $Comb_4$ models perform better than those using one source of information (temporal or frequency). Figure 9 presents the results of the different voting schemes. The performance of the voting schemes 1, 2, and 3 ($Schema_1$, $Schema_2$, and $Schema_3$) is almost identical. However, $Schema_3$ is slightly more robust and stable as it seems to generalize significantly well using the different splits (5-fold cross validation).

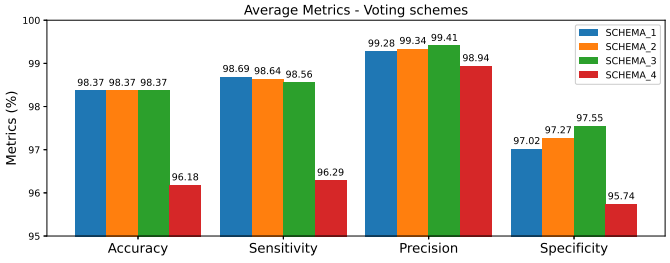

Fig. 9: Evaluation of the voting schemes - Average metrics

4. DISCUSSION AND CONCLUSION

The results of this study indicate that the denoising and artifact removal of resting-state fMRI can be very effectively implemented using a DNN framework. The models of spatial maps (CNN_{sm_1}, CNN_{sm_2}, and CNN_{sm_3}) perform almost identically and the accuracy is over 98%. This finding demonstrates that the usage of high-resolution spatial information, without the addition of temporal information, can present exceptional performance. The temporal models (CNN_{tm_1}, and $CNN-LSTM_{tm_2}$) are less efficient than spatial models. It is worth mentioning that the addition of the LSTM block in $CNN-LSTM_{tm_2}$ model boosts the performance significantly with an accuracy increment of almost 1% (around 95.5%). Similarly, the frequency models (CNN_{ps_1}, and $CNN-LSTM_{ps_2}$) perform worse than spatial models and the enhanced model $CNN-LSTM_{ps_2}$ with the LSTM block achieves higher evaluation metrics compared to CNN_{ps_1}. Notably, the evaluation of combined models ($Comb_2$, and $Comb_4$) and the voting schema ($Schema_4$) points out that the combination of timecourses and power spectrum as inputs is valuable and increases the performance (accuracy over 96%). Hence, the hypothesis that the DNN models learn the features related to frequency automatically, given the temporal information (timecourses), does not entirely hold [13] and adding the frequency information can result in an improved performance of the employed scheme.

The evaluation of the combined models ($Comb_1$, and $Comb_3$) demonstrates that the joint training using the three channels of information (spatial maps, timecourses, and power spectrum) is not advantageous. Finally, the best results are obtained by the voting $Schema_3$ with average accuracy of 98.37% and a very good balance between the metrics of sensitivity and specificity. Moreover, this schema shows very stable performance using the different splits in 5-fold cross validation. More precisely, the accuracy is varying from 98.31% (1st split) to 98.42% (4th split).

The main drawback of the proposed schemes (compared to FIX) is the fact that only healthy adult brains have been used for training the models. Hence, in order to use the proposed scheme in studies with brains of different size or anatomy (e.g. pediatric subjects), we would either need to retrain the selected scheme or use transfer learning. As future work, we intend to explore such cases with transfer learning approaches in order to evaluate the performance of our models in task-related fMRI studies and also in pediatric subjects. Furthermore, inception modules [27] can be tested in the DNN models, as they have shown state of the art results in many Deep Learning tasks. In addition, attention mechanisms can be included in the temporal and frequency models.

5. ACKNOWLEDGMENTS

This research received funding from EIT 19263 – "SeizeIT2: Discreet Personalized Epileptic Seizure Detection Device" and from the Flemish Government (AI Research Program).

6. REFERENCES

[1] M. A. Lindquist, "The statistical analysis of fMRI data," *Statistical Science*, vol. 23, no. 4, pp. 439–464, Jun. 2008.

[2] M. G. Bright and K. Murphy, "Is fMRI "noise" really noise? Resting state nuisance regressors remove variance with network structure," *NeuroImage*, vol. 114, pp. 158–169, Jul. 2015.

[3] S. Theodoridis, *Machine Learning: A Bayesian and Optimization Perspective*. Academic Press, Apr. 2015.

[4] C. Chatzichristos *et al.*, "Blind fMRI source unmixing via higher-order tensor decompositions," *J. Neuroscience Methods*, vol. 315, pp. 17–47, Mar. 2019.

[5] V. D. Calhoun *et al.*, "A review of group ica for fmri data and ica for joint inference of imaging, genetic, and erp data," *NeuroImage*, vol. 45, no. 1, pp. S163–S172, Mar. 2009.

[6] G. Salimi-Khorshidi *et al.*, "Automatic denoising of functional mri data: combining independent component analysis and hierarchical fusion of classifiers," *NeuroImage*, vol. 90, pp. 449–468, Apr. 2014.

[7] L. Griffanti *et al.*, "Ica-based artefact removal and accelerated fmri acquisition for improved resting state network imaging," *NeuroImage*, vol. 95, pp. 232–247, Jul. 2014.

[8] C. F. Beckmann and S. M. Smith, "Probabilistic independent component analysis for functional magnetic resonance imaging," *IEEE transactions on medical imaging*, vol. 23, no. 2, pp. 137–152, Feb. 2004.

[9] S. Wold *et al.*, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, Aug. 1987.

[10] Z. Mao *et al.*, "Spatio-temporal deep learning method for adhd fmri classification," *Information Sciences*, vol. 499, pp. 1–11, Oct. 2019.

[11] N. T. Duc *et al.*, "3d-deep learning based automatic diagnosis of alzheimer's disease with joint mmse prediction using resting-state fmri," *Neuroinformatics*, vol. 18, no. 1, pp. 71–86, Jan. 2020.

[12] I. Goodfellow *et al.*, *Deep learning*. MIT press Cambridge, Nov. 2016, vol. 1.

[13] T.-E. Kam *et al.*, "A deep learning framework for noise component detection from resting-state functional mri," in *International Conference on MICCAI*. Shenzhen, China: Springer, Oct. 2019, pp. 754–762.

[14] B. R. Howell *et al.*, "The unc/umn baby connectome project (bcp): an overview of the study design and protocol development," *NeuroImage*, vol. 185, pp. 891–905, Jan. 2019.

[15] Y. LeCun *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, Jun. 1995.

[16] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, Feb. 2015.

[17] Z. Liao and G. Carneiro, "On the importance of normalisation layers in deep learning with piecewise linear activation units," in *2016 IEEE WACV*. Lake Placid, NY, USA: IEEE, Mar. 2016, pp. 1–8.

[18] C. Tian *et al.*, "Enhanced cnn for image denoising," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 1, pp. 17–23, Mar. 2019.

[19] K. He *et al.*, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on CVPR*, Las Vegas, NV, USA, Dec. 2016, pp. 770–778.

[20] C. Szegedy *et al.*, "Inception-v4, inception-resnet and the impact of residual connections on learning," *arXiv preprint arXiv:1602.07261*, Feb. 2016.

[21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, Dec. 1997.

[22] N. C. Dvornek *et al.*, "Combining phenotypic and resting-state fmri data for autism classification with recurrent neural networks," in *2018 IEEE 15th ISBI*. Washington, D.C.: IEEE, Apr. 2018, pp. 725–728.

[23] W. Yan *et al.*, "Discriminating schizophrenia using recurrent neural network applied on time courses of multi-site fmri data," *EBioMedicine*, vol. 47, pp. 543–552, Sept. 2019.

[24] S. M. Smith *et al.*, "Resting-state fmri in the human connectome project," *NeuroImage*, vol. 80, pp. 144–168, Oct. 2013.

[25] D. C. Van Essen *et al.*, "The wu-minn human connectome project: an overview," *NeuroImage*, vol. 80, pp. 62–79, Oct. 2013.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, Dec. 2014.

[27] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on CVPR*, Boston, MA, USA, Jun. 2015, pp. 1–9.