3-D functional brain network classification using Convolutional Neural Networks

Conference Paper · April 2017

DOI: 10.1109/ISBI.2017.7950736

6 authors, including:

Yu Zhao
Siemens Healthineers
37 PUBLICATIONS **369** CITATIONS

Hanbo Chen
University of Georgia
90 PUBLICATIONS **1,316** CITATIONS

Qinglin Dong
Harvard Medical School
25 PUBLICATIONS **177** CITATIONS

Jinglei Lv
Queensland Institute of Medical Research
121 PUBLICATIONS **1,026** CITATIONS

3-D Functional Brain Network Classification using Convolutional Neural Networks

Dehua Ren[1], Yu Zhao[2]*, Hanbo Chen[2], Qinglin Dong[2], Jinglei Lv[3], Tianming Liu[2]*
**Co-first authors*

[1]College of Computer Science and Information Engineering, Tianjin University of Science and Technology, Tianjin, 300222, P.R. China; [2]Cortical Architecture Imaging and Discovery Lab, Department of Computer Science and Bioimaging Research Center, the University of Georgia, Athens, GA, 30605, USA; [3]Queensland Institute of Medical Research (QIMR) Berghofer, Herston, QLD, Australia.

ABSTRACT

Several recent studies have shown that dictionary learning and sparse representation can effectively reconstruct hundreds of interacting functional brain networks simultaneously from whole-brain fMRI data. However, accurate classification and recognition of those hundreds of functional networks from an individual or a population of many subjects is still a challenging and open problem due to the intrinsic variability of functional networks and other noise sources. To tackle this problem, this paper presents an effective deep learning framework to train convolutional neural networks from a large dataset of hundreds of thousands of available brain network volume maps, which was then applied on testing samples for network classification and recognition. We effectively applied computer-labeled data as training set so the whole process can be automated. Experimental results showed that the proposed method is quite robust in handling noisy patterns in the dataset, which suggests that our work offers a new computational framework for modeling functional connectomes from fMRI big data in the future.

Index Terms— deep learning, 3D convolutional neural networks, functional brain networks, classification

1. INTRODUCTION

Several recent studies have shown that dictionary learning and sparse representation can effectively reconstruct hundreds of concurrent interacting functional brain networks from whole-brain functional magnetic resonance imaging (fMRI) data [1], [2]. However, accurate classification and recognition of those hundreds of functional networks from an individual or a population of many subjects is still a challenging and open problem due to the intrinsic variability of functional networks and other noise sources, which motivates the investigation of classification and recognition of those functional network volume maps.

Besides widely used visual inspection and manual check approaches [3], many previous studies on automated classification of the functional brain networks have been carried out in the literature. A classification method is usually composed by two key parts: 1) a set of features or measurements to quantify the similarity between brain networks; 2) a classification method to learn the boundary between different groups and classify each network based on similarity measurements. The classic way to quantify similarity between functional networks is to calculate the voxel-wise overlap rate between the volumetric activation maps of functional networks [4], [5]. The accuracy of this measurement, however, is sensitive to the performance of brain registration and the artifact noises in spatial maps. Besides, due to the high dimension of volumetric data, the computation is very time consuming, which makes group-wise consistent network identifications from fMRI big data very hard to achieve.

Advanced techniques employing machine learning algorithms were also applied to tackle this problem in the literature [6]. In [7], classification accuracy for the intrinsic connectivity networks (ICNs) decomposed by independent component analysis (ICA) method can be quite high by using SVM. However, when applied to resting state fMRI, with lots of reconstructed components reflecting noises and artifacts, classification of spatial components based on strong expectations on the profile of the component may generate unreliable result, due to the major problem of complicated parameter tuning and overfitting issues.

Deep learning has gained great interest recently in many research fields. It is capable of modeling complex, hierarchical features in data and solves feature extraction and classification problems simultaneously. Convolutional Neural Network (CNN) [8] is a state-of-the-art deep learning method and has been widely applied in different image processing applications, especially in the aspect of image classification and recognition [9]. However, most CNN-related researches are 2D-centric [10], which is not optimal for 3D volumetric representation and will overlook 3D structure information. In this work, we adopt and improve an effective 3D CNN framework called VoxNet [11] to train convolutional neural networks to classify ICNs reconstructed by dictionary learning and sparse coding. Specifically, 217,600 ICNs were applied for classifier training and testing. Notably, due to the big size of the data, instead of asking experts to manually label the images based on visual inspection, the training data are automatically labeled based on overlap measurement between ICNs and the templates. Though overlap measurement will introduce possible errors and outliers in training data, our result shows that VoxNet is quite robust in handling outliers and can accurately classify most ICNs that cannot be correctly classified based on overlap method. This promising result suggests that our work offers a new computational framework for modeling functional connectomes mapped from fMRI big data in the future.

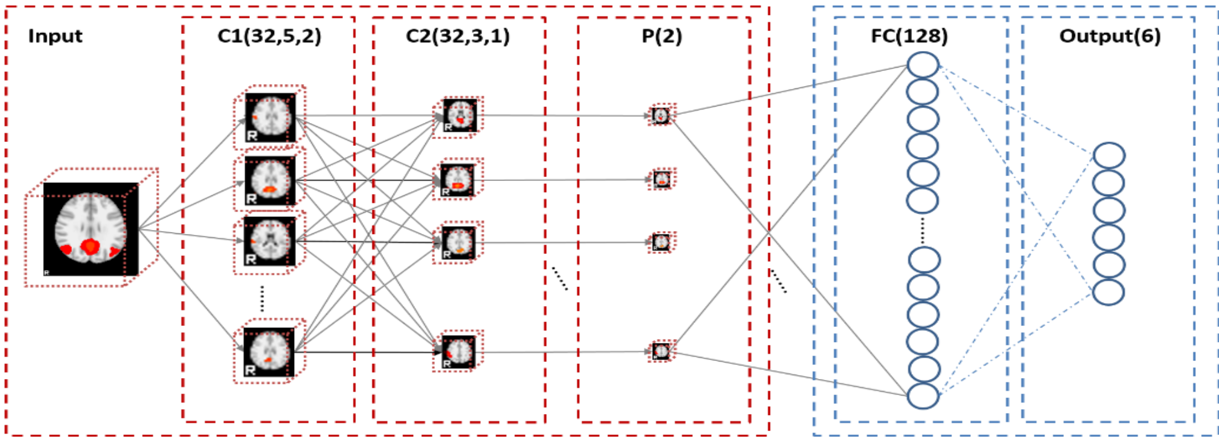

Fig. 1. 3D CNN structure used in experiment. Using VoxNet to exploit the 3D spatial structure of functional brain components, including input layer, two convolutional layers C1, C2, max pooling layer P, Fully Connected Layer FC, and output layer O.

2. METHOD

2.1. Experimental Dataset and Pre-processing

Our experiment was performed on the publicly released Human Connectome Project (HCP) fMRI data (Q1 release) [12]. The fMRI data of 68 individuals scanned during 7 tasks and resting state were used in this study. The acquisition parameters were as follows: 90×104 matrix, 220mm FOV, 72 slices, TR=0.72s, TE=33.1ms, flip angle = 52°, BW =2290 Hz/Px, in-plane FOV = 208 × 180 mm, 2.0 mm isotropic voxels. More details of task design and acquisition procedure can be found in [12]. Data preprocessing was similar to that in [13], which includes motion correction, spatial smoothing, temporal pre-whitening, slice time correction, global drift removal, and linear registration to the MNI space.

Based on dictionary learning and sparse representation [13], the whole-brain fMRI signals of each individual were decomposed into 400 ICNs. In total, 68 × 400 × 8 = 217600 ICNs are obtained for training and testing. The size of each ICN volume was initially 91×109×91 and was down sampled to 45×54×45 for the relief of computation burden without losing much accuracy.

2.2. 3D CNN Structure

Deep learning models are a set of algorithms that learn a hierarchy of features. By stacking layers in the model, the features are built from low-level to high-level in the increasingly complex hierarchical architecture. Such learning algorithm can be trained with either supervised or unsupervised approaches. Currently, the deep learning models have been shown to achieve competitive result in 1D or 2D inputs for speech recognition, image recognition and natural language processing. In the state of the art 2D CNNs, the convolution operators are applied on the image input layer, yielding 2D feature maps. One of the advantages of this type of models is that the features and the classifiers are jointly learned from the data.

In this paper, our goal is to classify the brain's functional network components, which are represented by 3D volumetric images. For the purpose of classification, the 3D spatial structures within the components need to be well presented and revealed. Thus we propose to use a 3D CNN for modeling brain networks. Similar to 2D CNNs, the 3D CNN applies a 3D convolution operator to the cube input, yielding 3D feature maps. VoxNet [11] can well incorporate 3D structure information as intrinsic feature for performing classification. Besides, the deep-layered nature of VoxNet can extract more abstract feature expression of the input with deeper layers. These promising characteristics of VoxNet make it suitable for the application of our 3D functional brain networks classification.

The network structure of VoxNet used in this paper consists of six layers (Fig. 1), including one input layer, two convolutional layers, one pooling layer (P), one fully connected layer (FC) and one output layer. Detailed layer information is listed below:

- *Input Layer – I:* This layer was modified to take 3D functional brain volumes as input. The node size is each volume size, $45 \times 54 \times 45 = 109350$ in this paper.
- *Convolutional Layer - C(f,d,s):* The convolutional layer is represented by C(f,d,s), where f is the number of filters, also the number of feature maps after filtered; d is the size of the 3D filter; s is stride. The output nonlinear action function uses a leaky rectified nonlinearity unit (ReLU) [14] with parameter 0.1.
- *Pooling Layer - P(m):* m is the size of matrices, to which the max-pooling operation is applied.
- *Fully Connected Layer - FC(n):* There are n nodes in this layer, all fully connected to previous layer nodes. The output nonlinear action function is the same as convolution layer.
- *Output Layer - O(k):* Softmax action function is adopted. The node number k equals to the number of component types, which is 6 here. And all the nodes in this layer are fully connected to previous layers.

In order to prevent the CNNs from overfitting, dropouts were attached after C1, P and FC with parameter set to 0.2, 0.3, and 0.4, respectively. In summary, The VoxNet Structure in this experiment is: C1(32, 5,2)-C2(32, 3,1)-P(2)-FC(128)-O(6), as shown in Fig. 1. More specifically, the target Function is set as: multinomial negative log-likelihood plus 0.001 times the L2 weight norm for regularization. The GPU (Nvidia Tesla K20Xm) environments were used as the hardware support for the training phase. Two training strategies are used in the experiment. Particularly, the Stochastic Gradient Descent (SGD) with momentum method was used as the training strategy where momentum is set to 0.9 and batch size is 32.

Training and Testing Data

In this work, we attempt to classify ICNs by five most consistent functional brain networks reported in previous literatures [15]. The averaged functional activations of these 5 networks were downloaded and applied as templates. Base on the overlap with template, each ICN will be labeled by one of these template networks (from 1 to 5 accordingly) or none of these networks (0). To quantify the overlap between ICN V and template W, the overlap rate similarity (ORS) was applied:

$$\mathrm{ORS} = \sum_{k=1}^{|V|} \frac{\min(V_k, W_k)}{(V_k + W_k)/2} \qquad (1)$$

where V_k and W_k are the activation score of voxel k in image V and W. For each ICN, the template with the highest ORS is defined as its best fit. And when the ORS between ICN and its best fit is larger than 0.2, the ICN will be labeled following the template (network 1-5 accordingly). Otherwise, it will be labeled by 0 (unknown ICN).

The total number of ICNs labeled by template 1-5 are 1075, 1075, 766, 1008 and 666, respectively (4590 in total), which is denoted by L as labeled set. And the rest of unknown ICN set is denoted by U (213010 in total).

To train a good classifier based on 3D CNN VoxNet, the training set should: (1) be as large as possible; and (2) has balanced number of training samples in each category. Since the size of U is incomparably larger than other categories in L (230 times), and the size of L is not big enough for a good CNN training, in order to have a balanced training set with a decent size, U and L are partitioned and combined by the following procedures to generate training, testing, and validation sets:

- L is randomly partitioned into 3 parts: L=L1+L2+L3. The size of L1 and L2 are 10% of L while L3 is the remaining 80% ICNs in L.
- U is randomly partitioned into 31 parts: $\mathbf{U} = \sum_{i=1}^{31} \mathbf{U_i}$. The size of U1 and U2 are one fifth of L, and U3 …U31 evenly partitioned the rest of U (each contains about 7000 ICNs).
- Training set = L3 ⊗ 10 + Ui (i=1…31), where L3 ⊗ 10 is to duplicate all ICNs in L3 by 10 times to balance the size of different categories. 29 training sets are generated and the size of each training set is about 43,000. Parameters of 3D CNN VoxNet classifier will be trained through the training sets
- Validation set = L1 ⊗ 10 + U1. L1 was also duplicated 10 times to balance the size. The validation set is for parameter tuning during training process.
- Testing set = L2 ⊗ 10 + U2. The final classification performance will be tested and discussed based on testing set (size: 5508).

3. RESULTS

29 CNN classifiers were trained based on 29 training sets separately and each classifier was trained for 40 epochs. The average training time for each classifier is 9 hours. Once the classifier is trained, the average time required to predict an ICN's label is 0.015 second, which is quite fast. Each ICN in the testing set will be assigned a label by each classifier.89.4 % testing ICNs are classified to the same category by all classifiers. For the rest ICNs, they will be classified to the category receiving the highest vote among all classifiers. We further analyzed ICNs that are disagreed between classifiers. Most of them are either classified as category 0 (unknown) or a certain functional network. Despite that, the rest of disagreements are mostly on whether an ICN should be classified as network 1 or network 2. Since, network 1 and 2 are both on visual cortex and close to each other, sometimes it is also difficult for a human expert to tell whether an ICN should be labeled by 1 or 2. This kind of disagreement between classifiers is natural and reasonable.

We then compared the labels assigned by classifiers with the initial labels assigned by ORS measurement to examine the consensus between them. The Jaccard index between two labels are calculated and listed in Table 1. It can be seen that the consensus between two approaches is quite high. By visual inspection, when the same label is achieved by both methods, the classification is quite reasonable.

As to the cases disagreed by ORS measurement and trained CNN classifiers, in almost all cases (only 1 exception), ORS labeled the ICN as unknown and CNN classifiers will label it as one of the target functional networks. After visual inspection, we found that the category suggested by CNN classifier is usually more reasonable. Some examples are selected and shown in Fig. 2. It is obvious that all these ICNs are quite noisy. The noisy patterns reduced the ORS value regardless of the patterns similar to target networks, while CNN classifiers are quite robust and powerful in handling these noisy patterns.

Nevertheless, we still identified 22 cases (1.65% of all cases) where CNN classifiers made the wrong decision. 4 examples are shown in Fig. 3. One possible reason causing these errors is due to the training set. Since our training sets are automatically generated without visual inspection, errors and vague cases are inevitable. It should be noted that, among those cases, only 3 are voted by all 29 classifiers to be classified as a target functional network.

Given the fact that the whole procedure is automatically performed without human intervention, in spite of the few wrong cases, CNN classifier performs reasonably well in correctly classifying ICNs and comprehensively outperforms traditional ORS-based functional brain network identification/classification strategy. Biases like noises or partly overlapped patterns can be exluded and intrinsic 3D structure information of the functional brain network components can be well examined by CNN to make reliable predictions or identifications.

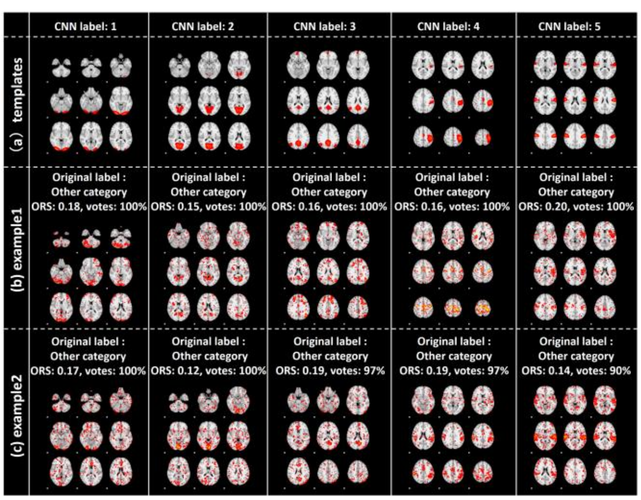

Fig. 2. Examples of improved classification by using CNN. The first row shows the template of 5 networks we used for classification. The second and third rows are two example ICNs that were initially labeled as ‘unknown’ but classified as the functional network shown on its top by CNN. 9 most informative axial slices of activation map are selected for visualization. The ORS and CNN votes are listed on the top of each sub-figure.

Table 1. Jaccard index between labels assigned by ORS and CNN classification result.

Classifier	1	2	3	4	5	6	7	8	9	10
Jaccard	95%	92.7%	95.4%	97.2%	95.7%	96.9%	94.1%	95.5%	95.3%	95.7%
Classifier	11	12	13	14	15	16	17	18	19	20
Jaccard	96%	94.8%	96.9%	96.8%	94.8%	93.6%	94.8%	94.8%	93.9%	93.6%
Classifier	21	22	23	24	25	26	27	28	29	voted
Jaccard	95.7%	96.9%	94.8%	94.4%	94.7%	95.1%	96.6%	94.8%	94.9%	95.7%

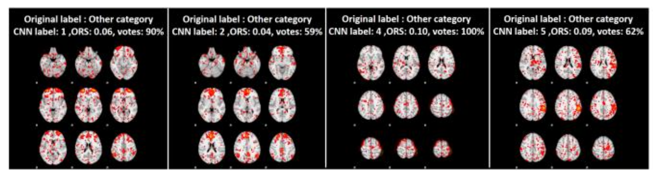

Fig. 3. Examples of failed classification by using CNN. The ICNs were initially labeled as ‘unknown’ but classified as functional network 1 - 5 listed on its top by CNN. The CNN votes as well as the ORS between ICN and the predicted network template are listed on the top.

4. CONCLUSION

A novel 3D CNN deep network classifier based on VoxNet was developed and applied to fMRI-based functional brain network components classification/identification. With simple ORS labelled training sets, CNN classifiers can be trained so that future big unlabeled dataset can be predicted using those classifiers precisely and quickly. In addition to the accurate and reliable classification results obtained, the proposed method has two methodological merits. First, it is based on deep learning method which is adaptive to different datasets. Second, different from traditional machine learning methods that require experts to manually annotate dataset for training, the training set is automatically generated in our method and the powerful 3D CNN classifier will handle the possible outliers introduced during computation. Finally, high accuracy of the predictions and high fault-tolerance characteristics demonstrated the promising potential of the application of deep learning networks such as CNN in functional network components classifications in the context of fMRI big data in the future.

5. REFERENCES

[1] M. Zibulevsky and B. A. Pearlmutter, “Blind Source Separation by Sparse Decomposition,” 1999.

[2] Y.-B. Lee, J. Lee, S. Tak, K. Lee, D. L. Na, S. W. Seo, Y. Jeong, and J. C. Ye, “Sparse SPM: Group Sparse-dictionary learning in SPM framework for resting-state functional connectivity MRI analysis,” *Neuroimage*, vol. 125, pp. 1032–1045, 2016.

[3] R. E. Kelly, G. S. Alexopoulos, Z. Wang, F. M. Gunning, C. F. Murphy, S. S. Morimoto, D. Kanellopoulos, Z. Jia, K. O. Lim, and M. J. Hoptman, “Visual inspection of independent components: defining a procedure for artifact removal from fMRI data.,” *J. Neurosci. Methods*, vol. 189, no. 2, pp. 233–45, Jun. 2010.

[4] Z. Wang and B. S. Peterson, “Partner-matching for the automated identification of reproducible ICA components from fMRI datasets: algorithm and validation.,” *Hum. Brain Mapp.*, vol. 29, no. 8, pp. 875–93, Aug. 2008.

[5] B. Chai, D. B. Walther, D. M. Beck, and L. Fei-Fei, “Exploring Functional Connectivity of the Human Brain using Multivariate Information Analysis.”

[6] F. De Martino, F. Gentile, F. Esposito, M. Balsi, F. Di Salle, R. Goebel, and E. Formisano, “Classification of fMRI independent components using IC-fingerprints and support vector machine classifiers,” *Neuroimage*, vol. 34, no. 1, pp. 177–194, 2007.

[7] Y. Wang and T.-Q. Li, “Dimensionality of ICA in resting-state fMRI investigated by feature optimized classification of independent components with SVM.,” *Front. Hum. Neurosci.*, vol. 9, p. 259, 2015.

[8] Y. Lecun, L. Eon Bottou, Y. Bengio, and P. Haaner, “Gradient-Based Learning Applied to Document Recognition RS-SVM Reduced-set support vector method. SDNN Space displacement neural network. SVM Support vector method. TDNN Time delay neural network. V-SVM Virtual support vector method,” *PROC. IEEE*, 1998.

[9] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, “ImageNet Large Scale Visual Recognition Challenge,” *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[10] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, “Learning Rich Features from RGB-D Images for Object Detection and Segmentation.”

[11] D. Maturana and S. Scherer, “VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition.”

[12] D. M. Barch, G. C. Burgess, M. P. Harms, S. E. Petersen, B. L. Schlaggar, M. Corbetta, M. F. Glasser, S. Curtiss, S. Dixit, C. Feldt, D. Nolan, E. Bryant, T. Hartley, O. Footer, J. M. Bjork, R. Poldrack, S. Smith, H. Johansen-Berg, A. Z. Snyder, D. C. Van Essen, and WU-Minn HCP Consortium, “Function in the human connectome: task-fMRI and individual differences in behavior.,” *Neuroimage*, vol. 80, pp. 169–89, Oct. 2013.

[13] J. Lv, X. Jiang, X. Li, D. Zhu, S. Zhang, S. Zhao, H. Chen, T. Zhang, X. Hu, J. Han, J. Ye, L. Guo, and T. Liu, “Holistic atlases of functional networks and interactions reveal reciprocal organizational architecture of cortical function.,” *IEEE Trans. Biomed. Eng.*, vol. 62, no. 4, pp. 1120–31, Apr. 2015.

[14] A. L. Maas, A. Y. Hannun, and A. Y. Ng, “Rectifier Nonlinearities Improve Neural Network Acoustic Models.”

[15] S. M. Smith, P. T. Fox, K. L. Miller, D. C. Glahn, P. M. Fox, C. E. Mackay, N. Filippini, K. E. Watkins, R. Toro, A. R. Laird, and C. F. Beckmann, “Correspondence of the brain’s functional architecture during activation and rest.,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 106, no. 31, pp. 13040–5, Aug. 2009.