# DEEP NEURAL NETWORKS ON GRAPH SIGNALS FOR BRAIN IMAGING ANALYSIS

*Yiluan Guo, Hossein Nejati, Ngai-Man Cheung*

Singapore University of Technology and Design (SUTD), Singapore 487372

## ABSTRACT

Brain imaging data such as EEG or MEG are high-dimensional spatiotemporal data often degraded by complex, non-Gaussian noise. For reliable analysis of brain imaging data, it is important to extract discriminative, low-dimensional intrinsic representation of the recorded data. This work proposes a new method to learn the low-dimensional representations from the noise-degraded measurements. In particular, our work proposes a new deep neural network design that integrates graph information such as brain connectivity with fully-connected layers. Our work leverages efficient graph filter design using Chebyshev polynomial and recent work on convolutional nets on graph-structured data. Our approach exploits graph structure as the prior side information, localized graph filter for feature extraction and neural networks for high capacity learning. Experiments on real MEG datasets show that our approach can extract more discriminative representations, leading to improved accuracy in a supervised classification task.

***Index Terms***— Brain imaging, autoencoder, convolutional nets, graph signal processing, dimensionality reduction

## 1. INTRODUCTION

Conventional imaging sensors detect signals lying on regular grids. On the other hand, recent advances and proliferation in sensing have led to new imaging signals lying on irregular domains. An example is brain imaging data such as Electroencephalography (EEG) and Magnetoencephalography (MEG). Some example of MEG data used in our experiments is shown in Figure 1(a). The color in Figure 1(a) is indicative of the intensity and influx / outflux of magnetic fields. The data are different from conventional 2D image data in that they lie irregularly on the brain structure. The data are captured by a recumbent Elekta MEG scanner with 306 sensors distributed across the scalp to record the cortical activations for 1100 milliseconds (Figure 1(b)). Therefore, MEG are high-dimensional spatiotemporal data often degraded by complex, non-Gaussian noise. For reliable analysis of MEG data, it is important to learn discriminative, low-dimensional intrinsic representation of the recorded data [1, 2].

Several methods have been applied to perform dimensionality analysis of brain imaging data, e.g., principal component analysis (PCA) and its numerous variants (see [1] for a recent review). In addition, it has been recognized that there are patterns of anatomical links, statistical dependencies or causal interactions between distinct units within a nervous system [3, 4, 5]. By modeling brain imaging data as signals residing on *brain connectivity graphs*, some methods have been proposed to apply the recent graph signal processing [6] to analyze brain imaging data [7, 8, 9, 10].

Deep learning, on the other hand, has achieved breakthroughs in image and video analysis, thanks to its hierarchical neural network structures with layer-wise non-linear activation and high capacity[11]. As an important deep learning model, autoen-
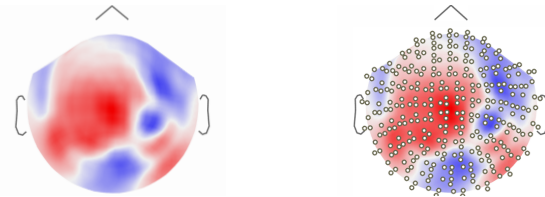


(a) Top view of MEG brain imaging.  (b) Top view with the sensors.

**Fig. 1**. Example of MEG brain imaging. The color indicates the intensity and directon of the magnetic fields. The nodes in (b) represent the sensors.

coders(AE) / stacked autoencoders(SAE) has achieved state-of-the-art performance in extraction of meaningful low-dimensional representations for input data in an unsupervised way[12]. However, conventional SAEs fail to take advantage the graph information when the inputs are modeled as graph signals.

In this work, we propose new AE-like neural networks that tightly integrate graph information for analysis of high-dimensional graph signals such as brain imaging data. In particular, we propose new AE networks that directly integrate graph models to extract meaningful representations. Our work leverages efficient graph filter design using Chebyshev polynomial[13] and recent work on deep learning on graph-structured data [14, 15, 16, 17]. Among these models, Convolutional Nets(ConvNets) are of great interest since they achieve state-of-the-art performance for images[18, 19] by extracting local features to build hierarchical representations. Image signals residing on regular grids are suitable for ConvNets. However, the problem to generalize ConvNets to signals on irregular domains, i.e. graphs, is a challenging one [15, 16, 20]. [20] proposed to convert the vertices on a graph into a sequence and extract locally connected regions from graphs, where the convolution is performed in spatial domain. On the contrary, the convolution in [15] is performed in spectral domain using recent graph signal processing theory [6]. [16] presented a formulation of ConvNets on graph in spectral domain and proposed fast localized convolutional filters. The filters are polynomial Chebyshev expansions where the polynomial coefficients are the parameters to be learned. [17] applied the first order approximation of [16] and achieved good results on the semi-supervised classification task on social networks.

This work is inspired by [16, 17] but focuses on new AE-like networks to extract meaningful representation in an unsupervised manner. The proposed method is depicted in Figure 2. First, brain imaging data is modelled as signals residing on connectivity graphs estimated with causality analysis. Then, the graph signals are processed by the ConvNets on graph, which output high-dimensional, rich feature maps of the graph signals. Subsequently, fully connected layers are used to extract low dimensional representations. During
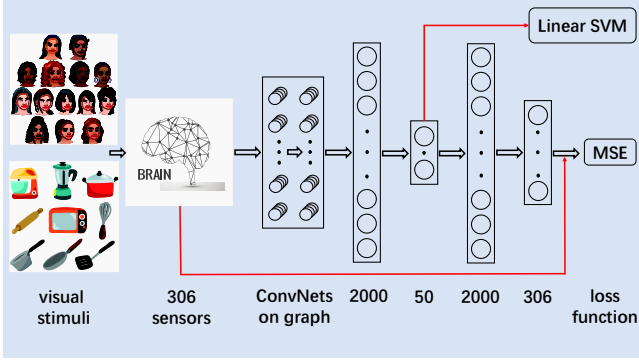
**Fig. 2**. The structure of the proposed method. 306 MEG sensors are used to record the cortical activations evoked by two categories of visual stimuli: face and object. Recorded high-dimensional MEG measurements and the prior estimated graph are the inputs to the proposed ConvNets on graph. This is followed by an autoencoder with fully connected layers of various size. The entire network is trained end-to-end with mean square error. During testing, we extract the activation of the innermost hidden layer and this is subject to a linear SVM to predict whether the subject views face or object.

testing, this low-dimensional representations are subject to a linear SVM classifier to evaluate their inclusion of discriminative information. Similar to [17], we also use the first order approximation in Chebyshev expansions [13, 16]. However, our network structure is different in that we propose an integration of ConvNets on graph with SAE. The entire network is trained end-to-end in an unsupervised way to learn the low-dimensional representations for the input brain imaging data. In other words, our work is a method of dimensionality reduction. Authors in [21] propose to use graph Laplacian to regularize the learning of autoencoder. Their work uses a *sample graph* to model the underlying data manifold. Their approach is significantly different from our work that integrates graph structure into the network. Moreover, it is non-trivial to apply their method to our problem which encodes sensor correlation with a *feature graph*.

Our contributions are threefold. First, we model the brain imaging data as graph signals with suitable brain connectivity graphs. Second, we propose new AE-like network structure that integrates ConvNets on graph with the SAE; the system is trained end-to-end in an unsupervised way. Third, we perform extensive experiments to demonstrate that our model can extract more robust and discriminative representations for brain imaging data. The proposed method can be useful for other high-dimensional graph signals.

## 2. PROPOSED METHOD

We first discuss main results from graph signal processing and ConvNets on graph. Then we discuss our proposed method.

### 2.1. GSP and convolution on graph

In conventional ConvNets, local filters are convoluted with signals on regular grids and the filter parameters are learned by back-propagation. To extend convolution from image / audio signals on regular grids to graph-structured data on irregular domain, recent graph signal processing[6] provides theoretical results. In particular, we consider an undirected, connected, weighted graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$, which has a number of vertices $|\mathcal{V}| = N$ and

an edge set $\mathcal{E}$. $W$ is the symmetric weighted adjacency matrix encoding the edge weights. Graph Laplacian, or combinatorial Laplacian is defined as $L = D - W$, where $D$ is the diagonal degree matrix with diagonal element $D_{ii} = \sum_{j=1}^{N} W_{ij}$. Since $L$ is an symmetric matrix, it can be eigen-decomposed as $L = U\Lambda U^T$ and has a complete set of orthonormal eigenvectors, denoted as $u_l$, for $l = 0, 1, ..., N-1$, and sorted real associated eigenvalues $\lambda_l$, known as the frequencies. In other words, we have $Lu_l = \lambda_l u_l$ for $l = 0, 1, ..., N-1$ and $0 \le \lambda_0 < \lambda_1 < ... < \lambda_{N-1}$. Normalized graph Laplacian, defined as $\tilde{L} = I - D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$, is also widely used due to the property that all the eigenvalues of it lie in the interval $[0, 2]$. $\{u_l\}$ acts like the Fourier basis in analogy to the eigen-functions of Laplace operator in classical signal processing. The graph Fourier transform(GFT) for a signal $\mathbf{x} \in \mathbb{R}^N$ on vertices of the graph $\mathcal{G}$ is defined as $\tilde{x}(\lambda_l) = \langle u_l, \mathbf{x} \rangle = u_l^T \mathbf{x}$.

GFT plays a fundamental role to define filtering and convolution operations for graph signals. Convolution theorem [22] states that convolution in spatial domain equals element-wise multiplication in spectral domain. Given the signal $\mathbf{x}$ and a filter $\mathbf{h} \in \mathbb{R}^N$ on graph $\mathcal{G}$, the convolution $*_\mathcal{G}$ between $\mathbf{x}$ and $\mathbf{h}$ is

$$\mathbf{x} *_\mathcal{G} \mathbf{h} = U((U^T\mathbf{h}) \odot (U^T\mathbf{x})), \qquad (1)$$

where $\odot$ indicates element-wise multiplication.

In [15], the authors proposed spectral neural networks to learn the filters $\mathbf{h}$ in spectral domain. There are two limitations in this approach. First, it is computationally-intensive to perform GFT and inverse GFT in each feed forward pass. Second, the learned filters using this approach are not explicitly localized, which differ from the filters in conventional ConvNets on images. To overcome these limitations, authors of [16] proposed to use polynomial filters and Chebyshev expansions [13]:

$$\mathbf{x} *_\mathcal{G} \mathbf{h} \approx \sum_{k=0}^{K-1} \theta'_k T_k(\hat{L})\mathbf{x}, \qquad (2)$$

where $\theta'_k$ are the polynomial filter coefficients to be learned, $\hat{L} = \frac{2}{\lambda_{max}}\tilde{L} - I_N$, and $T_k(\cdot)$ is the Chebyshev polynomial generated recursively. $K$ is the order of the polynomial, which means that the filter is $K$-hop localized. See [13, 16] for further details.

### 2.2. Model structure

Our proposed networks use ConvNets on graph to compute rich features for the input graph signals. In particular, ConvNets on graph leverage the underlying graph structure of the data to extract local features. Then, we use fully-connected layers and AE-like structure to extract intrinsic representations from the features.

#### 2.2.1. ConvNets on graph

The structure of the ConvNets on graph is shown in Figure 3, which integrates the graph information into the neural network. We use the first order approximation of Equation (2) [17]. Since we use normalized Laplacian and all the eigenvalues of it are in the interval $[0, 2]$, we let $\lambda_{max} \approx 2$. Further, we restrict $\theta = \theta'_0 = -\theta'_1$ to reduce overfitting and computation cost. We also use a renormalization technique proposed in [17], which converts $I_N + D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$ ($A$ is the adjacency matrix) into $\widehat{D}^{-\frac{1}{2}}\widehat{A}\widehat{D}^{-\frac{1}{2}}$, where $\widehat{A} = A + I_N$ and $\widehat{D}$ is the corresponding degree matrix of $\widehat{A}$. The reason for renormalization is that the eigenvalues of $I_N + D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$ are in the interval

[0, 2], which makes training of this neural network unstable due to gradient explosion[17]. After the renormalization, we have[17]

$$\mathbf{x} *_{\mathcal{G}} \mathbf{h}_{\boldsymbol{\theta}} \approx \theta \widetilde{A} \mathbf{x}, \tag{3}$$

where $\widetilde{A} = \widehat{D}^{-\frac{1}{2}} \widehat{A} \widehat{D}^{-\frac{1}{2}}$ is the new normalized adjacency matrix for the graph, which takes self-connections into consideration. $\mathbf{h}_{\boldsymbol{\theta}}$ indicates the filter $\mathbf{h}$ is parameterized by $\theta$, which transforms the graph signal from one channel to another channel.
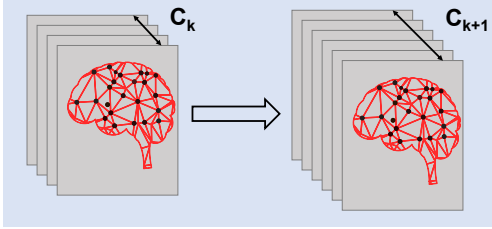


**Fig. 3**. Network structure of the ConvNets on graph. $C_k$ and $C_{k+1}$ are the number of channels at the $k$-th and $(k + 1)$-th layers resp.

Recent work [17] uses ConvNets on graph for semi-supervised classification tasks, e.g., semi-supervised document classification in citation networks. The entire dataset (e.g. full dataset of documents) is modeled as a *sample graph* with each vertex representing a sample (e.g., a labeled or unlabeled document). Therefore, the number of vertices equals to the number of samples. In their work, they apply two-layer ConvNets on graph to compute a feature vector for each vertex, which is then used to classify a unlabeled vertex. In particular, their network processes the whole graph (e.g. entire dataset of documents) as a full-batch. It is unclear how to scale the design for large dataset. On the contrary, our network processes individual graph signals in separate passes. The graph signals are modeled by a *feature graph* that encodes the correlation between features. The feature graph has $N$ vertices, with $N$ being the dimensionality of a graph signal (for MEG brain imaging data, $N = 306$, the number of sensors). Individual low-dimensional representations of the graph signals are subject to classification independently.

In our design, the $k$-th network layer takes as input a graph signal $\mathbf{x_k} \in \mathbb{R}^{N \times C_k}$, which means that this signal lies on a graph with $N$ vertices and has $C_k$ channels on each vertex. The output is a graph signal $\mathbf{x_{k+1}} \in \mathbb{R}^{N \times C_{k+1}}$. The transformation equation for the $k$-th network layer is

$$\mathbf{x_{k+1}} = \sigma \left( \widetilde{A} \mathbf{x_k} \Theta \right). \tag{4}$$

Here $\sigma(\cdot)$ is the element-wise non-linear activation function; $\Theta \in \mathbb{R}^{C_k \times C_{k+1}}$ is the parameter matrix to be learned. Note that $\Theta$ generalizes the $\theta$ in (3) for multiple channels. $\Theta$ has dimension $C_k \times C_{k+1}$: the input signal with $C_k$ channels is transformed into one with $C_{k+1}$ channels. With the normalized adjacency matrix $\widetilde{A}$ in (4), the network layer considers correlation between individual vertices and their 1-hop neighbors. To take $m$-hop neighbours into account, $m$ layers need to be stacked. In our experiment, we only stack two ConvNets on graph layers and this shows competitive performance. Note that $\widetilde{A}$ plays the role of specifying the receptive field for one feature: one feature is convoluted with its neighbours on the graph with different weights, which are determined by the nonzero value of $\widetilde{A}$. This is different from conventional ConvNets for images, where the weights is learned by back-propagation. In our work, the neural

networks instead learn the weights for transforming the channels of the input graph signal. Note that with the non-linear activation function, the transformation in each network layer is not simply multiplication.

In comparison, conventional neural networks can also expand or compress number of the channels with $1 \times 1$ convolution. Specifically, this is the ConvNets on graph when $\widetilde{A} = I$, where $I$ is the identity matrix. This is a limited model due to small kernel size. In fact, when $\widetilde{A} = I$, the ConvNets on graph reduce to fully connected layers in a conventional AE. Similarly, removing the non-linearity activation function limits the model capacity. Even with larger receptive field for one feature, the output becomes linear combination of the neighbours on graph of this feature. We observe in our experiment (Section 3) that without $\widetilde{A}$ and non-linearity activation function, our design has similar performance as conventional AEs.

### 2.2.2. *Fully connected layers and loss function*

After $k$ layers of ConvNets on graph, we obtain a graph signal $\mathbf{x_k} \in \mathbb{R}^{N \times C_k}$ of features. Each row vector is the multichannel feature of one vertex. We concatenate the row vectors and obtain $\mathbf{x_k} \in \mathbb{R}^{N \cdot C_k}$ as the output of ConvNets on graph. Since our goal is to extract low dimensional and semantically discriminative representations for each signal in an unsupervised way, we introduce stacked autoencoder(SAE) [12] here. SAE has been shown by recent research that it consistently produces high-quality semantic representations on several real-world datasets[23]. The difference between our work and SAE is that SAE takes the original signal as input while our work takes as input the high dimensional, rich feature map of the graph signal, which is the output of ConvNets on graph. The dimension of the SAE output $\mathbf{y}$ is the same as the original signal. The training of the entire network is end-to-end by minimizing mean square error between input $\mathbf{x}$ and $\mathbf{y}$, i.e. $\|\mathbf{x} - \mathbf{y}\|_2^2$.

## 3. EXPERIMENT

### 3.1. Datasets

We test our model on real MEG signal datasets. The MEG signals record the brain responses to two categories of visual stimulus: human face and object. The subjects were shown 322 human-face and 197 object images randomly while MEG signals were collected by 306 sensors on the brain. The signals were recorded 100ms before the stimulus and until 1000ms after the stimulus onset. Each image was shown to the subjects for 300ms. We focus on MEG data from 96ms to 110ms after the visual stimulus onset, as it has been recognized that the cortical activities in this duration contain rich information [24]. We model the MEG signals as graph signals by regarding the 306 sensor measurements as signals on a graph of 306 vertices. The underlying graph, which represents the complex brain network[25], is estimated by Granger Causality connectivity(GCC) analysis using the Matlab open-source toolbox BrainStorm[26]. Note that we have to renormalize the connectivity matrix following our discussion in Section 2.2.

### 3.2. Implementation

We use TensorFlow[27] to implement our networks. The numbers of channels for the two-layer ConvNets on graph are set to be 16 and 5. The subsequent fully-connected layers have dimension $d - 2000 - 50 - 2000 - 306$, where $d$ is the dimension after concatenation of the row vectors of the output of ConvNets. Adam[28] is

adopted to minimize the MSE with learning rate 0.001. Dropout[29] is used to avoid overfitting. We also include the $L2$ regularization in the loss function for the fully connected layers. For comparison, we train two different SAEs with the same schemes. After training all the networks for 300 epochs, we use linear SVM to predict whether the subject viewed face or object based on the 50-dimensional representation of the original MEG imaging data. We use 10-fold cross validation and report the average accuracy. All the experiments are performed on each subject separately.

## 3.3. Results

We compare our results with several unsupervised dimensionality reduction methods: PCA, GBF, Robust PCA and SAE. PCA is a commonly used dimensionality reduction technique by projecting data to the axis with first $n$ largest variance. GBF [30, 9] projects the MEG signals to a linear subspace spanned by the first $n$ eigenvectors of the normalized graph Laplacian. Robust PCA(RPCA) [31] decomposes the data into two parts: low rank representation and sparse perturbation. For non-linear transformation, we test two SAEs, one is with symmetric structure $306-2000-50$ and the other $306-5000-1500-2000-50$.

**Table 1**. Average classification accuracy with different methods on MEG brain imaging data.

| Method | Accuracy | | |
|---|---|---|---|
| | subject A | subject B | subject C |
| original data | 0.6482 | 0.6015 | 0.6338 |
| PCA | 0.6529 | 0.5957 | 0.6100 |
| RPCA | 0.6656 | 0.5925 | 0.6186 |
| GBF | 0.6638 | 0.6026 | 0.5970 |
| 2-layer AE | 0.6610 | 0.5983 | 0.6302 |
| 4-layer AE | 0.6693 | 0.5939 | 0.6323 |
| proposed model | **0.6833** | **0.6414** | **0.6435** |

The results are shown in Table 1. It can be observed that accuracy for the original 306-dimensional data is inferior or similar to other methods. Thus, it is advantageous to perform dimensionality reduction and feature extraction. Improvement using PCA is limited as it is not robust to the existing non-Gaussian noise. For subject A and B, RPCA achieves similar result as GBF, which leverages Granger Causality connectivity(GCC) of subjects' brain as side information. PCA, RPCA and GBF are linear transformations failing to capture the non-linearity property of the brain imaging data, which limits the performance. SAEs with 2 layers and 4 layers also outperform PCA by introducing non-linear transformation. [19] has shown that increasing the depth of networks helps improve performance by a large margin. Nevertheless, the results are similar for the two SAEs. We conjecture that the optimization stops at saddle points or local minima[32]. Our proposed model achieves the highest accuracy comparing to other methods. The reasons are that our approach 1) considers connectivity as the prior side information and 2) uses neural networks with high capacity to learn the discriminative representation.

## 3.4. Discussion

### 3.4.1. Contribution of the graph

We may ask whether the graph information is truly helpful and necessary for this task. To answer this question and better understand the importance and necessity of incorporating the graph information in the neural networks, we replace the graph adjacency matrix estimated by GCC with an identity matrix and a random symmetric matrix and train the model. Table 2 shows that GCC indeed helps the networks to extract expressive features. Replacing GCC with identity matrix ignores the prior feature correlation, resulting in accuracy similar to SAEs. Random symmetric matrix confuses the neural networks and thus the accuracy drops drastically.

**Table 2**. Classification accuracy with different adjacency matrix.

| Graph | Accuracy | | |
|---|---|---|---|
| | subject A | subject B | subject C |
| GCC | 0.6833 | 0.6414 | 0.6435 |
| Identity Matrix | 0.6616 | 0.6052 | 0.6213 |
| Random Matrix | 0.5941 | 0.5589 | 0.5332 |

### 3.4.2. Contribution of nonlinear transformation

Since we expand our single channel MEG data to multiple channels, there is concern that the transformation is a trivial multiplication with a scaler in graph ConvNets. Therefore, in this experiment, we remove the non-linearity activation function in ConvNets on graph. By doing this, the outputs of the graph ConvNets become the average of the input weighted by the graph adjacency matrix, which is equivalent to linear combination of the inputs. Thus, the accuracy should be similar to SAEs. This can be observed in Table 3. With non-linear activation function, ConvNets on graph can fully exploit the graph information.

**Table 3**. Classification accuracy with different activation function.

| Activation Function | Accuracy | | |
|---|---|---|---|
| | subject A | subject B | subject C |
| Non-linear | 0.6833 | 0.6414 | 0.6435 |
| Linear | 0.6656 | 0.6016 | 0.6132 |

## 4. CONCLUSION

In this work, we propose AE-like deep neural network that integrates ConvNets on graph with fully-connected layers. The proposed network is used to learn the low-dimensional, discriminative representations for brain imaging data. Experiments on real MEG datasets suggest that our design extracts more discriminative information than other advanced methods such as RPCA and autoencoders. The improvement is due to the exploitation of graph structure as side information. For future work, we apply recent graph learning techniques [33, 34] to improve the estimation of the underlying connectivity graph. Moreover, we address the problem of deploying the networks for real-time analysis in brain computer interface applications. Furthermore, we explore applications of our ConvNets on graph integrated AE for other image / video applications [35, 36].

## 5. REFERENCES

[1] Mwangi B, Tian TS, and Soares JC, "A review of feature reduction techniques in neuroimaging," *Neuroinformatics*, vol. 12, no. 2, pp. 229–244, 2014.

[2] Kleovoulos Tsourides, Shahriar Shariat, Hossein Nejati, Tapan K Gandhi, Annie Cardinaux, Christopher T Simons, Ngai-Man Cheung, Vladimir Pavlovic, and Pawan Sinha, "Neural correlates of the food/non-food visual distinction," *Biological Psychology*, 2016.

[3] Ed Bullmore and Olaf Sporns, "Complex brain networks: graph theoretical analysis of structural and functional systems," *Nature Reviews Neuroscience*, vol. 10, no. 3, pp. 186–198, 2009.

[4] James S Hyde and Andrzej Jesmanowicz, "Cross-correlation: an fmri signal-processing strategy," *NeuroImage*, vol. 62, no. 2, pp. 848–851, 2012.

[5] Andrea Brovelli, Mingzhou Ding, Anders Ledberg, Yonghong Chen, Richard Nakamura, and Steven L Bressler, "Beta oscillations in a large-scale sensorimotor cortical network: directional influences revealed by granger causality," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 26, pp. 9849–9854, 2004.

[6] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[7] Hamid Behjat, Nora Leonardi, Leif Sörnmo, and Dimitri Van De Ville, "Anatomically-adapted graph wavelets for improved group-level fmri activation mapping," *NeuroImage*, vol. 123, pp. 185–199, 2015.

[8] Weiyu Huang, Leah Goldsberry, Nicholas F Wymbs, Scott T Grafton, Danielle S Bassett, and Alejandro Ribeiro, "Graph frequency analysis of brain signals," *arXiv preprint arXiv:1512.00037v2*, 2016.

[9] Liu Rui, Hossein Nejati, and Ngai-Man Cheung, "Dimensionality reduction of brain imaging data using graph signal processing," in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1329–1333.

[10] Rui Liu, Hossein Nejati, and Ngai-Man Cheung, "Simultaneous low-rank component and graph estimation for high-dimensional graph signals: Application to brain imaging," in *Proc. ICASSP*, 2017.

[11] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[12] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[13] David K Hammond, Pierre Vandergheynst, and Rémi Gribonval, "Wavelets on graphs via spectral graph theory," *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, pp. 129–150, 2011.

[14] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2009.

[15] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun, "Spectral networks and locally connected networks on graphs," *arXiv preprint arXiv:1312.6203*, 2013.

[16] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Advances in Neural Information Processing Systems*, 2016, pp. 3837–3845.

[17] Thomas N Kipf and Max Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[20] Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov, "Learning convolutional neural networks for graphs," in *Proceedings of the 33rd annual international conference on machine learning. ACM*, 2016.

[21] Kui Jia, Lin Sun, Shenghua Gao, Zhan Song, and Bertram E. Shi, "Laplacian auto-encoders: An explicit learning of nonlinear data manifold," *Neurocomputing*, vol. 160, pp. 250 – 260, 2015.

[22] Stéphane Mallat, *A wavelet tour of signal processing*, Academic press, 1999.

[23] Quoc V Le, "Building high-level features using large scale unsupervised learning," in *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2013, pp. 8595–8598.

[24] S. Thorpe, D. Fize, and C. Marlot, "Speed of processing in the human visual system," *Nature*, 1996.

[25] Maxime Guye, Gaelle Bettus, Fabrice Bartolomei, and Patrick J Cozzone, "Graph theoretical analysis of structural and functional connectivity mri in normal and pathological brain networks," *Magnetic Resonance Materials in Physics, Biology and Medicine*, vol. 23, no. 5-6, pp. 409–421, 2010.

[26] François Tadel, Sylvain Baillet, John C Mosher, Dimitrios Pantazis, and Richard M Leahy, "Brainstorm: a user-friendly application for meg/eeg analysis," *Computational intelligence and neuroscience*, vol. 2011, pp. 8, 2011.

[27] Martın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al., "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[28] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[29] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting.," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[30] Hilmi E Egilmez and Antonio Ortega, "Spectral anomaly detection using graph-based filtering for wireless sensor networks," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 1085–1089.

[31] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright, "Robust principal component analysis?," *Journal of the ACM (JACM)*, vol. 58, no. 3, pp. 11, 2011.

[32] Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio, "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization," in *Advances in neural information processing systems*, 2014, pp. 2933–2941.

[33] Jiun-Yu Kao, Dong Tian, Hassan Mansour, Antonio Ortega, and Anthony Vetro, "Disc-glasso: Discriminative graph learning with sparsity regularization," in *Proc. ICASSP*, 2017.

[34] Hermina Petric Maretic, Dorina Thanou, and Pascal Frossard, "Graph learning under sparsity priors," in *Proc. ICASSP*, 2017.

[35] Ngai-Man Cheung and Antonio Ortega, "Distributed source coding application to low-delay free viewpoint switching in multiview video compression," in *Proc. Picture Coding Symposium*, 2007.

[36] Lu Fang, Ngai-Man Cheung, Dong Tian, Anthony Vetro, Huifang Sun, and O Au, "An analytical model for synthesis distortion estimation in 3d video," *IEEE Transactions on Image Processing*, 2014.