



BrainParcel: A Brain Parcellation Algorithm for Cognitive State Classification

Hazal Mogultay^(✉) and Fatos Tunay Yarman Vural

Department of Computer Engineering, Middle East Technical University,
Ankara, Turkey
{hazal,vural}@ceng.metu.edu.tr

Abstract. In this study, we propose a novel brain parcellation algorithm, called BrainParcel. BrainParcel defines a set of supervoxels by partitioning a voxel level brain graph into a number of subgraphs, which are assumed to represent “homogeneous” brain regions with respect to a predefined criteria. Aforementioned brain graph is constructed by a set of local meshes, called mesh networks. Then, the supervoxels are obtained using a graph partitioning algorithm. The supervoxels form partitions of brain as an alternative to anatomical regions (AAL). Compared to AAL, supervoxels gather functionally and spatially close voxels. This study shows that BrainParcel can achieve higher accuracies in cognitive state classification compared to AAL. It has a better representation power compared to similar brain segmentation methods, reported the literature.

Keywords: fMRI · Brain partitioning · Mesh model

1 Introduction

Functional Magnetic Resonance Imaging (fMRI) is one of the most common imaging techniques for detecting the activation levels of human brain, during a cognitive process. fMRI measures the change of oxygen level in the brain with respect to neural activities. In principle, oxygen dependencies of neuron groups fluctuate in accordance with the activation and MRI machines can detect those changes through the scan. An intensity value is recorded at each 1–2s for a neuron group called voxel. Each voxel is a cubic volume element around 1–2 mm³ size. Classification of the cognitive stimulus from the voxel intensity values are called brain decoding and the pioneering studies in this area are called Multi Voxel Pattern Analysis (MVPA) [9, 11]. MVPA involves recognizing the cognitive states represented by voxel intensity values of fMRI data, using machine learning techniques. A set of features is extracted from voxel intensity values recorded during each cognitive task. However, due to the large feature space formed by

voxels (about 100,000–200,000 voxels per brain volume), dimension reduction techniques are required, such as, clustering the voxels groups into homogeneous regions.

Anatomical regions, defined by experimental neuroscience can be used as brain parcels. In most common approach, called AAL, there are 116 major regions and each region is assumed to contain voxel groups which work together. In order to reduce the dimension of the feature space, representative signals can be selected for each region or average time series can be computed within each region [1, 16]. However, anatomical regions lose the subject-specific and task dependent information of brain activities. Besides, sizes of the regions vary extremely and activation levels of voxels may not be homogeneous within an anatomic region.

In order to partition the voxels into a set of homogeneous regions, well-defined clustering methods such as k-means [6, 7, 10], hierarchical clustering [1, 4], and spectral clustering [17, 20] can be used. The pros and cons of these clustering algorithms are widely studied in fMRI literature on a variety of datasets [8, 18]. Some studies bring spatially close voxels together considering only the location information in analogy with the AAL [6]. Although this method improves the strict norms of AALs, it lacks the functional similarity of voxel time series, which belongs to the same regions. Recent literature reveals that functionally close voxels tend to contribute to the same cognitive task, thus, form homogeneous regions. Therefore, one needs to bring both functionally similar and spatially contiguous voxels together to define homogeneous brain regions [21]. Similarly, Wang et al. suggest to combine n-cut segmentation algorithm with simple linear iterative clustering (SLIC) [21]. Blumensath et al. use region growing for brain segmentation with functional metrics and spatial constraints between samples [3]. Bellec et al. also use region growing with functional metrics within the 26 spatial neighborhood in 3-Dimensional space [2]. Background on neuronal activity, also, supports this idea, such that physically close neurons are in chemical interaction with each other and this interaction can be interpreted as functional similarity. With these objectives in mind, many different clustering algorithms are applied to create data dependent homogeneous brain parcels. Depending on the predefined distance measure, the clustering algorithms can group spatially or functionally similar voxels under the same cluster. Craddock et al. adopt this idea and propose a brain parcellation method, in which they represent the voxels in a graph structure and used n-cut on a spatially constrained brain graph with functional edges [5]. In order to achieve spatial contiguity they connected each voxel to its 26 closest neighbors in 3D space. On the other hand, to accomplish functional homogeneity, they set edge weights of the graph to the correlation between the time series of two voxels as follows;

$$e_{i,j} = \begin{cases} \text{corr}(\mathbf{v}_i, \mathbf{v}_j) & , \text{dist}(\mathbf{v}_i, \mathbf{v}_j) \leq d_t \\ 0 & , \text{otherwise,} \end{cases} \quad (1)$$

where d_t is selected to be $\sqrt{3}$ and $\text{corr}(\mathbf{v}_i, \mathbf{v}_j)$ is the Pearson Correlation between the intensity values of voxels \mathbf{v}_i and \mathbf{v}_j . They, also, remove the edges with cor-

relation values less than 0.5 to reduce the weak connections. Then, they define a brain graph $G = (V, E)$, where the set of voxels $V = [\mathbf{v}_1, \mathbf{v}_2 \dots \mathbf{v}_N]$ are the nodes of the graph, and $E = [e_{1,1}, e_{1,2}, \dots e_{N,N}]$ are the edge weights computed according to Eq. 1. They partition the graph G into subgraphs by removing the edges iteratively using N-cut segmentation method using the following formula,

$$N_cut = \frac{\sum_{\mathbf{v}_i \in A, \mathbf{v}_j \in B} e_{i,j}}{\sum_{\mathbf{v}_i \in A, \mathbf{v}_n \in V} e_{i,n}} + \frac{\sum_{\mathbf{v}_i \in A, \mathbf{v}_j \in B} e_{i,j}}{\sum_{\mathbf{v}_j \in B, \mathbf{v}_n \in V} e_{j,n}}. \quad (2)$$

As it is mentioned above, conventional MVPA methods create features sets from the selected voxel intensity values or use some averaging techniques to represent each brain region. This approach is quite restrictive to represent cognitive states. Recent studies suggest to model the relationships among voxels rather than using voxel intensity values. Ozay et al., demonstrate this idea by suggesting the Mesh Model which is a graph structure that identifies the connectivity among voxels [15].

Mesh Model (MM) represents intensity values of voxels as a weighted linear combination of its neighboring voxels, defined on a neighborhood system. The estimated weights represent the arc weights between the voxels and the voxels represents a node in the overall brain graph. A star mesh is formed around each voxel and its p neighbors, independently. In each mesh, the voxel at the center is called seed-voxel and the surrounding voxels are called neighbors. p nearest neighbors of voxel \mathbf{v}_i for cognitive stimulus k are shown as $\eta_{\mathbf{v}_i(k)}^p$ and they can be selected spatially (Spatial Mesh Model - SMM) [12, 14] or functionally (Functional Mesh Model - FMM) [12, 13] such that, spatial neighbors has the smallest Euclidean distance with the seed-voxel whereas functional neighbors has maximum functional similarity. Meshes are formed using the full length time series for voxels, recorded during an fMRI experiment session. Assuming s measurements are taken for each cognitive stimulus, time series of a voxel \mathbf{v}_i for stimulus k is an s dimensional vector shown as $\mathbf{v}_i(k) = [\mathbf{v}_i(k)^1, \mathbf{v}_i(k)^2, \dots \mathbf{v}_i(k)^s]$. Spatial Mesh Model (SMM) selects the neighbors according to the physical distances among voxels in 3-dimensional space by Euclidean distance [12, 14]. On the other hand, Functional Mesh Model (FMM), proposed by Onal et al., selects functional neighbors with the highest p -correlation values obtained by Pearson Correlation [12, 13]. Afterwards, time series of the seed voxel is represented as a weighted combination of its neighbors by the following equation for each cognitive stimuli:

$$\mathbf{v}_i(k) = \sum_{\mathbf{v}_j(k) \in \eta_{\mathbf{v}_i(k)}^p} a_{i,j,k} \mathbf{v}_j(k) + \epsilon_{i,j}, \quad (3)$$

where $\eta_{\mathbf{v}_i(k)}^p$ is the p nearest neighbors of voxel v_i for sample k and $a_{i,j,k}$ are the arc weights of the mesh network between the voxels and they are called Mesh Arc Descriptors (MADs). MADs are estimated using regularized Ridge regression method by the minimization of error term $\epsilon_{i,j}$. Concatenating each MAD for each voxel and cognitive task creates a new feature space and classification is performed on this feature space.

In this study we combine classical brain parcellation approach proposed by Craddock et al. and Mesh Model and propose a novel brain parcellation algorithm, called BrainParcel. Unlike current methods, we partition the graph formed by star meshes and partition this graph into brain regions. We show that brain partitions obtained by BrainParcel have better representation power than the partitions obtained by state of the art clustering methods and AAL in cognitive state classification problem.



Fig. 1. Overall architecture of the BrainParcel algorithm.

2 BrainParcel

Brain parcel is a brain partitioning algorithm that uses graph theoretic approaches. First, we form a brain graph by ensembling the meshes estimated around each voxel. Then, we partition this graph using n-cut segmentation algorithm. Each region is represented by the average time series of all voxels in that region. Then, these representative time series are fed to a machine learning algorithm to classify the underlying cognitive states. Figure 1 indicates the stages of suggested BrainParcel method for brain decoding problem. Each stage is explained in the following subsections.

2.1 Neighborhood System

In order to estimate a star mesh around each voxel independently, we need to define a neighborhood system. The concept of neighborhood takes an important place in this study. We inspire from the biological structure of human brain, where spatially close neurons act together by means of some electro-chemical interactions. Additionally, experimental evidence indicates that physically far apart neurons may contribute to the same cognitive process through the brain connectome. We try to utilize these observations in our brain parcellation model by defining a neighborhood system around each voxel and employ multiple connections between the neighboring voxels.

Neighborhood of the i^{th} voxel \mathbf{v}_i , is defined as the set of voxels that are closest to \mathbf{v}_i according to a predefined rule. Assuming p_c many neighbors around a voxel, neighborhood of \mathbf{v}_i is represented by $\eta_{v_i}^{p_c}$.

Letting N be the number of voxels, we define an N -by- N adjacency matrix, ND , to represent the neighborhood of voxels. Each entry of ND is calculated as follows;

$$ND(i, j) = \begin{cases} 1 & , v_j \in \eta_{v_i}^{p_c} \\ 0 & , otherwise. \end{cases} \quad (4)$$

In this study, we define two types of neighborhood, given below:

Spatial neighborhood $\eta_{v_i}^{p_c}$ is defined as the set of voxels, which has the p_c smallest Euclidean distance in 3-Dimensional space to voxel \mathbf{v}_i . This neighborhood system ensures resulting brain parcels to be spatially contiguous.

Functional neighborhood $\eta_{v_i}^{p_c}$ is defined as the set of voxels, which has the highest p_c -Pearson Correlation to voxel \mathbf{v}_i . This neighborhood system connects functionally similar voxels, even if they are physically apart from each other.

Note that, selection of the number of neighbors, p_c , and the type of the neighborhood system highly effects the rest of the steps of BrainParcel. Specifically, functional neighborhood relaxes the spatial similarity, selecting the neighboring voxels which are physically far apart. Therefore, the resulting brain parcels are not guaranteed to be spatially contiguous. It is very crucial to define a sort of balance in these two types of neighborhood, so that the resulting brain parcels consist of functionally similar and spatially contiguous voxels.

2.2 Extracting Mesh Arc Descriptors (MADs) Among Voxels

Each voxel is connected to its neighboring voxels according to one of the above neighborhood systems to form a star mesh around that voxel. The structure of star mesh depends on the type of the neighborhood system defined above. The arc weights of each local mesh are estimated by adopting the mesh model of Onal et al. [12–14]. As opposed to the current studies, we form the meshes, based on the complete time series recorded at each voxel rather than forming a different mesh for each cognitive task. This approach enables us to form a shared brain partition across all of the cognitive tasks

fMRI technique collects a time series for each voxel, when the subject is exposed to a cognitive stimulus. In the case of a block experiment design, which we have used, subjects are exposed to a stimulus for a specific time interval and the voxel time series over the entire brain volume are collected. Then, after a rest period, another stimulus is given to the subject. The time series recorded during a stimulus at i^{th} voxel is represented by the vector \mathbf{v}_i . Based on the idea of mesh model, we represent each \mathbf{v}_i as weighted sum of other voxels in the $\eta_{v_i}^{p_c}$ neighborhood of \mathbf{v}_i according to Eq. 3. Notice that Mesh Arc Descriptors (MADs) for classification are calculated per cognitive stimulus. However, we compute MADs from the entire time series of the voxels. Therefore, k index, which indicates a specific cognitive task, is removed from Eq. 3, since we compute MADs for the entire time duration of fMRI recordings. This representation is carried with a linear equation by the following formula;

$$\mathbf{v}_i = \sum_{v_j \in \eta_{v_i}^{p_c}} a_{i,j} \mathbf{v}_j + \epsilon_{i,j}. \quad (5)$$

Weights of the representation, called Mesh Arc Descriptors (MADs) are shown as $a_{i,j}$ and are estimated by Regularized Ridge Regression which minimizes the mean squared error $\epsilon_{i,j}^2$ [12–14].

2.3 Constructing a Voxel-Level Brain Graph

In order to construct a brain graph from the estimated MADs, we ensemble all the local meshes under the same graph, $G_m = (V, E_m)$. The set of nodes of this graph correspond the set of voxels $V = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N]$. The set of edges corresponds to set of all MADs, $a_{i,j} \in E_m$. Note that, since $a_{i,j} \neq a_{j,i}$, the graph G_m is directed. On the other hand, the graph partitioning methods, such as n-cut requires undirected graphs, in which each edge weight, $e_{i,j}$ is represented by a scalar number. In order to obtain an undirected graph from the directed graph G_m , a set of heuristic rules are used. Suppose that the mesh is formed for the voxel \mathbf{v}_i , and \mathbf{v}_j is in the neighborhood of \mathbf{v}_i with mesh arc-weight $a_{i,j}$. Edge value $e_{i,j}$ is determined, based on the following rules:

- **Case 1:** IF $\mathbf{v}_i \notin \eta_{v_j}^{p_c}$ AND $\mathbf{v}_j \in \eta_{v_i}^{p_c}$ THEN $e_{i,j} = a_{i,j}$
- **Case 2:** IF $\mathbf{v}_i \in \eta_{v_j}^{p_c}$ AND $\mathbf{v}_j \in \eta_{v_i}^{p_c}$ THEN this case requires further analysis. Assuming highly correlated voxels should have a stronger edge between them, we employ the following thresholding method;
 - IF $\text{corr}(v_i, v_j) \geq 0$, THEN $e_{i,j} = \max(a_{i,j}, a_{j,i})$
 - IF $\text{corr}(v_i, v_j) < 0$, THEN $e_{i,j} = \min(a_{i,j}, a_{j,i})$.

Above rules prune the directed graph G_m to an undirected graph G to be partitioned for obtaining homogeneous brain regions, called supervoxels.

2.4 Graph Partitioning for Obtaining Supervoxels

After constituting the brain graph G , n-cut segmentation method is used for clustering this graph. N-cut is a graph partitioning algorithm which carries a graph cut method on a given undirected graph. Given G , n-cut cuts the edges one by one in an iterative manner. With each cut, the graph is split into two smaller connected components. Letting N be the number of voxels, n-cut method requires the representative graph G , which is actually an $N - by - N$ adjacency matrix explained in the previous sections. The number of intended brain parcels is set to \mathbb{C} . With graph cut operations, graph is split into C connected components where $C \leq \mathbb{C}$. Each sample is a member of one of this clusters and assigned with a cluster index. In other words, n-cut method returns an $1 - by - N$ dimensional vector $\mathbb{L}_{\mathbb{C}} = [l_1^c, l_2^c, \dots, l_N^c]$ where each l_i^c is a number between 1 and C . The n-cut method, as applied to undirected graph G is called BrainParcel. The output of this algorithm yields a set of supervoxels, which are homogeneous with respect to the subgraphs of mesh network.

Recall that, anatomical regions (AAL) produce an experimentally neuroscientific parcellation of the brain. In order to compare the brain decoding performances, we conducted our experiment, where we form mesh network for both among anatomical regions and the network formed among supervoxels obtained at the output of BrainParcel. There are 116 basic brain regions in AAL and each voxel resides in one and only one region. Let us represent the anatomically defined region indices of voxels with $\mathbb{L}_{\mathbb{A}} = [l_1^a, l_2^a, \dots, l_N^a]$, in order to avoid

confusions. Notice that, with $\mathbb{L}_{\mathbb{A}}$ we skip all of the brain parcellation steps. Also, let us call $\mathbb{L} = [l_1, l_2, \dots, l_N]$ to all kinds of brain segmentations; in our case it means $\mathbb{L} \supset (\mathbb{L}_{\mathbb{C}} \cap \mathbb{L}_{\mathbb{A}})$.

2.5 Representation of Supervoxels

We need to calculate a representative signal for each supervoxel. For this purpose, we take an average among the time series of voxels within each supervoxel. With C supervoxels, we calculate set of vectors $U = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_C]$, where each u_i is the representative vector of supervoxel i and they are calculated as follows;

$$\mathbf{u}_i = \frac{\sum_{l_j==i} \mathbf{v}_j}{\sum_{l_j==i} 1}. \quad (6)$$

In the dataset on which we have performed our experiments, six measurements were taken for each cognitive stimulus. Assuming K stimuli were shown to each subject, time series of each voxel has a length $\mathbb{K} = 6K$. Therefore, at the output of the clustering algorithm we construct a data matrix U of size $C - by - \mathbb{K}$, where each row represents a feature, and each column corresponds to a cognitive stimulus.

2.6 Constructing Supervoxel-Level Brain Graph

The original area of utilization of the mesh model was to model the relationships among voxels and use this relationship for decoding the cognitive processes. Both spatial and functional neighborhoods were considered, and their representation powers were demonstrated by relatively high recognition performances compared to the available state of the art network models. Specifically, Functional Mesh Model (FMM) outperform most of the MVPA and Spatial Mesh Model (SMM) results. Therefore, we use FMM for classifying the cognitive states.

Data matrix U , defined in the previous section, is feed into the FMM algorithm. Each supervoxel \mathbf{u}_i is represented by linear combination of its functional neighbors, the arc weights are estimated at each mesh using Ridge Regression for each cognitive stimulus. Recall that fMRI collects multiple measurements during the time course of each cognitive stimulus. In our dataset 6 measurements are collected for each stimulus, and \mathbf{u}_i is a vector of length $\mathbb{K} = 6K$ for K stimuli. Let us represent the vector of the stimulus k by $\mathbf{u}_i(\mathbf{k})$. First, functionally closest p_m neighbors of $\mathbf{u}_i(\mathbf{k})$; $\eta_{u_i(k)}^{p_m}$, are selected from the supervoxels \mathbf{u}_j which has the highest correlation with supervoxel \mathbf{u}_i according to Pearson Correlation. Then, the mean square error $E(\epsilon_{i,j}^2)$ is minimized to estimate $a_{i,j,k}$ of the Eq. 3. Estimated MADs are concatenated so that they represent a more powerful feature space compared to the raw fMRI signal intensity values that is used in MVPA studies. We concatenate all the MADs and represent the stimulus in a feature space formed by MADs.

2.7 Classification

MADs estimated at supervoxel-level, are concatenated under a feature vector for classifying the cognitive states. 6 fold cross validation schema is applied on the dataset, where at each fold, one run is reserved from the data as a test set. Logistic regression is used for classification.

3 Experiments

3.1 Dataset

In this study, we use a dataset called “Object Experiment”. This dataset is recorded by the team of ImageLab of METU members at Bilkent University UMRAM Center. It consists of 4 subjects in the age of twenties. Each subject is shown various bird and flower pictures. In between those stimuli, simple mathematics questions are shown as transition. There are 6 runs in the experiment and in each run, 36 pictures are shown to each subject. Thus, there are total of 216 samples. Number of samples are balanced for the two classes (bird and flower). Preprocessing of the dataset is carried with the SPM toolbox and the number of voxels is decreased to 20,000 for each subject. Also, there are 116 labeled anatomical regions, defined under MNI coordinate system [19]. We provide experimental results, where each given accuracy is the output of a six fold cross validation. Recall that, each subject is given 6 runs of stimuli. At each fold, we split a run for testing and use the other 5 runs for training. The reported accuracies are the average of these 6 folds for each subject.

3.2 Comparative Results

In this section, we provide a comparison between BrainParcel and the parcellation algorithm suggested by Craddock et al. Table 1 shows the classification performances for various number of parcels. The results are reported after optimizing the mesh sizes empirically. Recall that, functionally constrained systems that construct the graph with *Functional_ND* neighborhood system does not provide any spatial integrity within the brain parcels, since the brain graph is not formed on these grounds. On the other hand, spatially constrained systems provide both spatial continuity and functional homogeneity since the brain graph is formed by spatial restrictions and edges are weighted in terms of functional connectivity.

In Table 1, the first and third column give the best results for the brain parcellation method suggested by Craddock et al. (called classical, in the Table), and the other two gives the results obtained by BrainParcel that we have proposed. Each row of this table gives the results for a different number of supervoxels (SV). Notice that spatially constrained BrainParcel gives the best classification performances in the overall schema.

These results point to the idea that, in order to achieve better representational power for cognitive state classification, one needs spatially contiguous

and functionally homogeneous brain parcels, which is accomplished by spatial BrainParcel. Moreover, recall that we have offered BrainParcel as an alternative to AAL, which has 116 basic anatomic regions and gives 53% performance on average. A compatible parcellation scheme consists of 100 super voxels, where, Spatial BrainParcel results in higher classification accuracies compared to the other methods.

Table 1. Overall 2-class classification accuracy acquired from the MADs constructed among super voxels and method suggested by Craddock et al. (called, classical). These results suggest that Spatial BrainParcel gives higher performances, since it provides spatial continuity and functional homogeneity within each brain parcel.

# of SV	Spatial constraints		Functional constraints	
	Classical	BrainParcel	Classical	BrainParcel
100	67.79	74.00	70.63	73.29
250	72.96	78.71	76.79	77.54
500	75.46	79.42	77.33	77.54
750	77.46	78.04	79.38	77.54
1000	78.08	78.83	79.96	80.08

4 Conclusion

In this study, we offer a brain parcellation methodology, which combines the spatial and functional connectivity of brain on a novel graph representation. This approach offers a better alternative to the current brain parcellation methods in the literature [8, 18], for brain decoding problems. BrainParcel uses spectral clustering methods, which represents the voxel space as a graph formed by mesh model. Common studies compute the edge weights of the brain graph as the pairwise correlation between voxels, whereas we computed the edge weights by estimating them using the mesh model among a group of voxels. Then, brain graph is partitioned with n-cut segmentation method to generate supervoxels.

As suggested, using task dependent brain parcellation methods enable better brain decoding performances compared to anatomical regions. Moreover, it is demonstrated that functional connectivity, united with the spatial contiguity is the best approach to represent homogeneous brain regions.

Also, results show that using the MADs of the mesh model for classification, improves the brain decoding performances in all of the experiment setups.

Our study reveals that mesh model not only improves the classification performance, but also creates a brain graph, where the nodes represent homogeneous super voxels with a better representation power for brain decoding. Although the performance increase looks relatively small, when the large size of the data set is considered, the performance boost becomes quite meaningful.

In the future, experimental set up can be refined for parameter selection.

Acknowledgement. This project is supported by TUBITAK under grant number 116E091. We thank UMRAM Center of Bilkent University for opening their facilities to collect fMRI dataset. We also thank to Dr. Itir Onal Ertugrul and Dr. Orhan Firat for their contribution and effort of data collection.

References

1. Alkan, S., Yarman-Vural, F.T.: Ensembling brain regions for brain decoding. In: 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 2948–2951. IEEE (2015)
2. Bellec, P., et al.: Identification of large-scale networks in the brain using fMRI. *Neuroimage* **29**(4), 1231–1243 (2006)
3. Blumensath, T., et al.: Spatially constrained hierarchical parcellation of the brain with resting-state fMRI. *Neuroimage* **76**, 313–324 (2013)
4. Cordes, D., Haughton, V., Carew, J.D., Arfanakis, K., Maravilla, K.: Hierarchical clustering to measure connectivity in fMRI resting-state data. *Magnetic Reson. Imaging* **20**(4), 305–317 (2002)
5. Craddock, R.C., James, G.A., Holtzheimer, P.E., Hu, X.P., Mayberg, H.S.: A whole brain fMRI atlas generated via spatially constrained spectral clustering. *Hum. Brain Mapp.* **33**(8), 1914–1928 (2012)
6. Flandin, G., Kherif, F., Pennec, X., Malandain, G., Ayache, N., Poline, J.-B.: Improved detection sensitivity in functional MRI data using a brain parcelling technique. In: Dohi, T., Kikinis, R. (eds.) MICCAI 2002. LNCS, vol. 2488, pp. 467–474. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45786-0_58
7. Flandin, G., Kherif, F., Pennec, X., Riviere, D., Ayache, N., Poline, J.B.: Parcelation of brain images with anatomical and functional constraints for fmri data analysis, pp. 907–910 (2002)
8. Liao, T.W.: Clustering of time series data—a survey. *Pattern Recogn.* **38**(11), 1857–1874 (2005)
9. Mitchell, T., et al.: Predicting human brain activity associated with the meanings of nouns. *Science* **320**(5880), 1191–1195 (2008)
10. Moğultay, H., Alkan, S., Yarman-Vural, F.T.: Classification of fMRI data by using clustering. In: 23th Signal Processing and Communications Applications Conference, SIU, pp. 2381–2383. IEEE (2015)
11. Norman, K.A., Polyn, S.M., Detre, G.J., Haxby, J.V.: Beyond mind-reading: multi-voxel pattern analysis of fMRI data. *Trends Cogn. Sci.* **10**(9), 424–430 (2006)
12. Onal, I., Ozay, M., Mizrak, E., Öztekin, I., Yarman-Vural, F.T.: A new representation of fMRI signal by a set of local meshes for brain decoding. *IEEE Trans. Sig. Inf. Process. Over Netw.* (2017). <https://doi.org/10.1109/TSIPN.2017.2679491>
13. Onal, I., Ozay, M., Yarman-Vural, F.T.: Functional mesh model with temporal measurements for brain decoding. In: 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 2624–2628. IEEE (2015)
14. Onal, I., Ozay, M., Yarman-Vural, F.T.: Modeling voxel connectivity for brain decoding. In: International Workshop on Pattern Recognition in NeuroImaging (PRNI), pp. 5–8. IEEE (2015)
15. Ozay, M., Öztekin, I., Öztekin, U., Yarman-Vural, F.T.: Mesh learning for classifying cognitive processes (2012). arXiv preprint [arXiv:1205.2382](https://arxiv.org/abs/1205.2382)
16. Richiardi, J., Eryilmaz, H., Schwartz, S., Vuilleumier, P., Van De Ville, D.: Decoding brain states from fMRI connectivity graphs. *Neuroimage* **56**(2), 616–626 (2011)

17. Shen, X., Papademetris, X., Constable, R.T.: Graph-theory based parcellation of functional subunits in the brain from resting-state fMRI data. *Neuroimage* **50**(3), 1027–1035 (2010)
18. Thirion, B., Varoquaux, G., Dohmatob, E., Poline, J.B.: Which fMRI clustering gives good brain parcellations? *Front. Neurosci.* **8** (2014)
19. Tzourio-Mazoyer, N., et al.: Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. *Neuroimage* **15**(1), 273–289 (2002)
20. Van Den Heuvel, M., Mandl, R., Pol, H.H.: Normalized cut group clustering of resting-state fMRI data. *PLoS One* **3**(4), e2001 (2008)
21. Wang, J., Wang, H.: A supervoxel-based method for groupwise whole brain parcellation with resting-state fMRI data. *Front. Hum. Neurosci.* **10** (2016)