GRAPH INFERENCE ENHANCEMENT WITH CLUSTERING: APPLICATION TO GENE REGULATORY NETWORK RECONSTRUCTION

Aurélie Pirayre, Camille Couprie, Laurent Duval

IFP Energies nouvelles
1 et 4 avenue de Bois-Préau
92852 Rueil-Malmaison, France

Jean-Christophe Pesquet

Université Paris-Est
LIGM, UMR CNRS 8049,
Champs-sur-Marne, France

ABSTRACT

The obtention of representative graphs is a key problem in an increasing number of fields, such as computer graphics, social sciences, and biology to name a few. Due to the large number of possible solutions from the available amount of data, building meaningful graphs is often challenging. Nonetheless, enforcing a priori on the graph structure, such as a modularity, may reduce the underdetermination in the underlying problem. In this work, we introduce such a methodology in the context of Gene Regulatory Network inference. These networks are useful to visualize gene interactions occurring in living organisms: some genes regulate the expression of others, structuring the network into modules where they play a central role. Our approach consists in jointly inferring the graph and performing a clustering using the graph-Laplacian-based random walker algorithm. We validate our approach on the DREAM4 dataset, showing significant improvement over state-of-the-art GRN inference methods.

Index Terms— genomic data analysis, graph construction, combinatorial Dirichlet problem, random walker

1. INTRODUCTION

In many applications [1–3], the design of a graph structure capturing the essence of a set of observed data constitutes a central problem. The latter may be tackled with two main types of approaches. The first one relates to the definition of an appropriate statistical model, e.g. a Gaussian graphical model. It often resorts to the estimation of a large covariance matrix [4, 5], also called dispersion matrix. Its inverse, the concentration or precision matrix, may directly be interpreted as the adjacency matrix defining the underlying network organization. The edge presence or absence thus encodes information in the associated graph. The second type of approaches, less popular in the signal processing community, but more likely to be encountered in computer vision, formulates problems from a combinatorial optimization standpoint. We adopt the second strategy throughout this work and express the desired graph as the solution of a variational formulation.

Additionally, assumptions on the network topology [6] alleviate part of the optimization burden and promote more pragmatic solutions. In several contexts, for instance social networks, road and traffic networks or biological networks, the graphs to infer exhibit a community structure or modularity, driven by a limited number of specific nodes. From the knowledge of these particular nodes, we constrain such a modular structure into the network using the "random walker" clustering algorithm [7].

In the biological context of our application, for each gene of a studied organism placed in different living conditions, a signal consisting of a sequence of expression levels of the gene is collected. From these data, regulatory processes between genes may be recovered and are represented in the shape of a graph where nodes are associated to genes and edges to regulations relationships. Such networks are called Gene Regulatory Networks (GRNs). In addition to the GRNs, the biology community has at its disposal another kind of tools, based on clustering [8, 9] to extract useful information from gene expression data. To the best of our knowledge, only very few methods, often relying upon graphical models [10–12], perform joint clustering and graph inference in order to improve the network.

GRN inference is a well-studied topic, see for instance [13–15] for recent overviews. However, satisfactory results remain difficult to obtain, due to the very limited length of expression signals, in comparison to the number of genes. A first line of methods approaches the problem by computing a similarity score between expression profiles of pairs of genes, such as the mutual information (e.g. ARACNE [16] and CLR [17]). Among the top performing GRN inference methods, GENIE3 [18] expresses the whole inference as a set of regression problems solved by using random forests. Alternatively, a vast literature relies on graphical models methods, e.g. SIMoNe [12]. Similarly however, model estimation becomes inaccurate if the number of samples (the signal length) is small compared to the number of variables, which often happens in GRN inference. Adopting a strategy similar to ours, [10] suggests the adjunction of clustering results to improve GRN inference. However, the clustering step is decou-

pled from the graph construction at the detriment of the spatial smoothness promoted in our approach. In the "Module networks" approach of [19], the authors design a module inference method based on probabilistic graphical models, specific to gene expression signals. Similarly to [19], we assume that a list of putative transcription factors (TFs), proteins coded by genes regulating the expression of other genes, is available.

Our formulation requires the construction of a fully connected graph, where nodes correspond to genes, and edges connecting every pair of genes are weighted by similarity measures between the gene expression signals.

The variational approach we propose both generates the topology and the clustering of the GRN graph. The graph structure, defined by variables on the edges on the graph, and the clustering solution, defined by variables on the nodes, are computed through an optimization procedure. Our objective function is designed to constrain the construction of a modular network. Its clustering step involves a seeded graph-based technique where the modular structure is driven by marked nodes. In our application, these nodes correspond to known TFs.

The paper is organized as follows: in Section 2, we formulate our joint clustering and graph inference approach as an optimization problem and detail the optimization strategy to solve the problem. The evaluation of the proposed method, in the GRN inference context, is performed and results are given in Section 3, before our conclusions in Section 4.

2. JOINT INFERENCE/CLUSTERING MODEL

2.1. Classical inference graph construction

As described in [20], graph inference may be expressed as an edge selection problem. Our goal is to compute a GRN inference graph $\mathcal{G}^*$ where nodes correspond to genes and edges correspond to true regulation relationships between genes. We assume in this work that a list of T module central nodes is available. In our situation, this means that genes in this list are known to code for TFs (transcription factors). We denote this list by $\mathcal{T} = \{t_1, \ldots, t_T\}$. As we assume that regulations are oriented from TFs to non-TF genes, we do not infer the edge directions.

We thus define an initial undirected non-reflexive graph $\mathcal{G}$ with a set of g nodes denoted by $\mathcal{V}$, and a set of n weighted edges $\mathcal{E}$ between every couple of nodes. We have $\mathcal{V} = \{1, \ldots, g\}$, and the initial number of edges equals $n = g(g-1)/2$. We denote by $(i_\tau)_{1 \leq \tau \leq T}$ the indices of the nodes in $\mathcal{T}$. Let us define an edge label $x_{i,j}$ that indicates the presence of an edge between genes i and j, i.e. $x_{i,j} = 1$ if the edge $e_{i,j}$ is present in the inferred graph, and $x_{i,j} = 0$ otherwise.

We first compute similarity measures $w_{i,j}$ between the expression signals of every couple $(i,j) \in \mathbb{E}$, where $\mathbb{E}$ is the set of edge indices, using for instance mutual information e.g. [17], or one of the graph inference methods available in the literature e.g. GENIE3 [18]. These weights are normalized to belong to the interval $[0,1]$. The results of classical inference methods may often be recovered by solving

$$\underset{x \in \{0,1\}^n}{\text{maximize}} \sum_{(i,j) \in \mathbb{E}} w_{i,j} x_{i,j} + \lambda(1 - x_{i,j}), \tag{1}$$

where $\lambda \in [0,1]$ is a scalar corresponding to a threshold on the weights w. In this case, the optimal x^* is given by the explicit solution:

$$\forall (i,j) \in \mathbb{E}, \quad x_{i,j} = \begin{cases} 1 & \text{if } w_{i,j} > \lambda, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

2.2. Using clustering in the graph construction

Let $y \in \mathbb{N}^g$ denote the labeling of a clustering of the nodes. In order to perform a joint clustering and graph construction, we propose to modify the variational Problem (1) as follows

$$\underset{x \in \{0,1\}^n, y \in \mathbb{N}^g}{\text{maximize}} \sum_{(i,j) \in \mathbb{E}} w_{i,j} x_{i,j} f(y_i, y_j) + \lambda(1 - x_{i,j}) \tag{3}$$

where $f \colon \mathbb{R}^2 \to \mathbb{R}$ is a function that maps (y_i, y_j) to

- a high value if i and j are in the same cluster,
- a low value if i and j are in different clusters.

The previous graph inference is hence influenced by the clustering, preventing edges to appear across different clusters. A simple choice for f is

$$f(y_i, y_j) = \frac{\beta - \mathbb{1}(y_i \neq y_j)}{\beta} \in [0,1]. \tag{4}$$

The parameter β, a real number greater than one, controls the influence of the clustering prior: a large value reduces its impact. Our method requires the introduction of some markers, also called seeds, to avoid a trivial solution. Assuming a modular structure of the network organized around the nodes in $\mathcal{T}$, we add to Problem (3) the following affine constraint:

$$y \in C = \{(z_i)_{1 \leq i \leq g} \in \mathbb{R}^g \mid \forall \tau \in \{1, \ldots, T\}, z_{i_\tau} = t_\tau\}. \tag{5}$$

Two results are derived if we now look at Problem (3) subject to Constraint (5).

- At fixed y, and variable x, an explicit solution exists, given by

$$x_{ij}* = \mathbb{1}(w_{ij} - \frac{w_{ij}}{\beta}\mathbb{1}(y_i \neq y_j) > \lambda). \tag{6}$$

- At fixed x, and variable y: the optimization problem reduces to

$$\underset{y \in \mathbb{N}^g \cap C}{\text{minimize}} \sum_{(i,j) \in \mathbb{E}} \alpha_{i,j} \mathbb{1}(y_i \neq y_j) \tag{7}$$

where

$$\alpha_{i,j} = \begin{cases} 0 & \text{if } w_{i,j} \leq \lambda \\ w_{i,j} - \lambda & \text{if } \lambda < w_{i,j} \leq \frac{\lambda\beta}{\beta-1} \\ \frac{w_{i,j}}{\beta} & \text{if } w_{i,j} > \frac{\lambda\beta}{\beta-1}. \end{cases} \quad (8)$$

This provides an optimization formulation for solving the joint clustering and graph construction problem. However, it turns out that Problem (7) is NP-hard [21]. In order to circumvent this difficulty, a continuous relaxation of this combinatorial problem can be introduced. To do so, assume that L is the number of clusters and introduce L vector variables $y^{(1)}, \ldots, y^{(L)}$ of size g, whose components are:

$$\forall i \in \mathcal{V}, \forall l \in \{1, \ldots, L\}, \quad y_i^{(l)} = \begin{cases} 1 & \text{if } y_i = l \\ 0 & \text{otherwise.} \end{cases} \quad (9)$$

Problem (7) is then equivalent to:

$$\underset{\substack{y^{(1)} \in C^{(1)}, \ldots, y^{(L)} \in C^{(L)} \\ (y^{(1)}, \ldots, y^{(L)}) \in D}}{\text{minimize}} \sum_{l=1}^{L} \Big(\sum_{(i,j) \in \mathbb{E}} \alpha_{i,j} (y_i^{(l)} - y_j^{(l)})^2 \Big) \quad (10)$$

where, for every $l \in \{1, \ldots, L\}$,

$$C^{(l)} = \{(z_i^{(l)})_{1 \leq i \leq g} \in \mathbb{R}^g \mid \forall \tau \in \{1, \ldots, T\}, z_{i_\tau}^{(l)} = t_\tau^{(l)}\}, \quad (11)$$

$t^{(l)}$ being defined from $t \in \mathcal{T}$ by a relation similar to (9), and

$$D = \{(y^{(1)}, \ldots, y^{(L)}) \in (\{0,1\}^g)^L \mid \sum_{l=1}^{L} y^{(l)} = 1_g\} \quad (12)$$

with $1_g = (1, \ldots, 1)^\top \in \mathbb{R}^g$. A convex relaxation of Problem (10) is then obtained by replacing D by its convex hull

$$\widehat{D} = \left\{ (y^{(1)}, \ldots, y^{(L)}) \in ([0,1]^g)^L \mid \sum_{l=1}^{L} y^{(l)} = 1_g \right\}.$$

A further simplification arises by dropping the latter constraint, in which case the optimization problem decouples into L quadratic convex problems. Provided that there is at least one marker in each connected component of the graph, each of these problems, known as the combinatorial Dirichlet problem, has a unique solution which can be obtained by solving a linear system of equations [7]. In addition, it can be shown that this solution actually belongs to the set $\widehat{D}$, so that $L-1$ linear systems only need to be solved. We note here that a continuous-valued Markov Random Field interpretation of the combinatorial Dirichlet problem is provided in [22].
Then, the final clustering label variable $y^* = (y_i^*)_{1 \leq i \leq g}$ is given by

$$\forall i \in \mathcal{V}, \quad y_i^* = \text{argmax}_{l \in \{1, \ldots, L\}} y_i^{(l)}. \quad (13)$$

Finally, the optimal clustering y^* is inserted in (6) to obtain the final edge labeling of the GRN.

3. RESULTS

We called our approach BRANE Clust in reference to "Biologically Related A priori for Network Enhancement via Clustering". We now demonstrate its performance on two GRN inference datasets, the DREAM4 multifactorial challenge [13] and DREAM5 [14], by comparing our results to three GRN inference methods, namely ARACNE [16], CLR [17], and GENIE3 [18]. The DREAM4 multifactorial dataset contains five ground truth networks extracted from the GRNs of *E. coli* and *S. cerevisiae*, provided with simulated expression signals of length 100. They all contain 100 genes, from which about half of them are known as TFs. The DREAM5 network used in our evaluation is a simulated network of 1643 genes including 195 TFs and signals of length 487.

To evaluate the obtained networks, we compare them with the true networks using two measures derived from the standard confusion matrix. The Precision is defined as $\frac{\mathrm{TP}}{\mathrm{TP+FP}}$ and the Recall is computed by $\frac{\mathrm{TP}}{\mathrm{TP+FN}}$, where TP is the number of true positive, FP is the number of false positive and FN is the number of false negative. The Precision value indicates the proportion of correctly inferred edges (TP) compared to the total number of inferred edges (TP + FP). The Recall value reveals the proportion of correctly inferred edges (TP) compared to the total number of expected edges given by the gold standard (TP + FN).

Each evaluated method produces a graph depending upon a threshold parameter defined on their weights. In our case, the threshold is given by λ. The variation of this parameter allows us to compute a Precision-Recall curve, and to deduce another performance measure, the Area Under Precision-Recall (AUPR) curve. A larger value of AUPR reflects a better accuracy.

We choose for BRANE Clust the weights given by the best available method, namely GENIE3. The AUPR obtained using the different methods including BRANE Clust are reported in Table 1. In all our tests, better results are obtained for a β parameter value close to 2.

On DREAM4, BRANE Clust results in about a 10 % improvement over CLR and GENIE3, in terms of AUPR. We note that the improvement mostly takes place at the beginning of the Precision-Recall curve, meaning that BRANE

Network index	1	2	3	4	5
GENIE3 [18]	0.239	0.260	0.323	0.301	0.295
CLR [17]	**0.245**	0.255	0.299	0.298	0.299
BRANE Clust	0.243	**0.277**	**0.369**	**0.328**	**0.332**

Table 1. Area Under Precision-Recall for CLR, GENIE3 and BRANE Clust methods on the DREAM4 dataset. For GENIE3, only TF genes are used as input genes. BRANE Clust was initialized using GENIE3 weights.

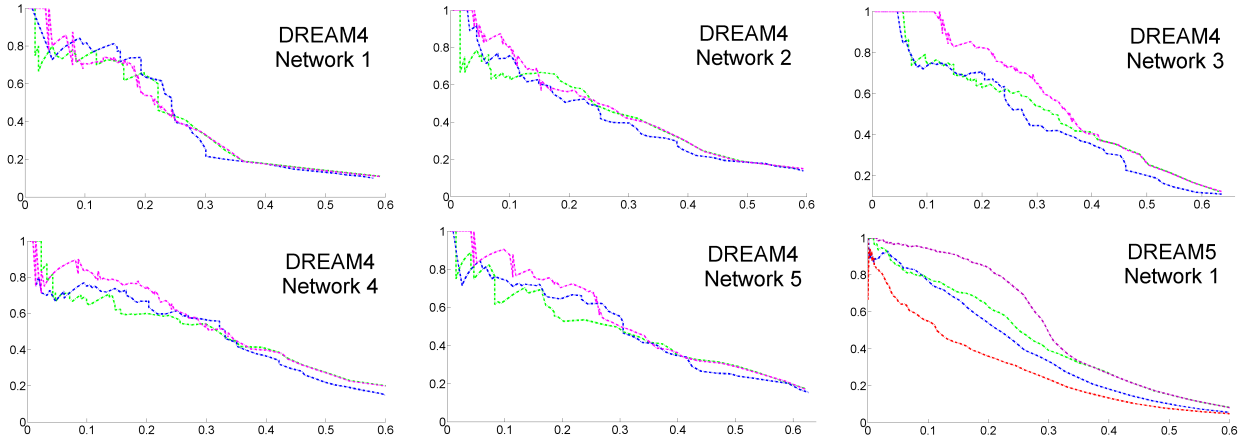


Fig. 1. CLR (blue), GENIE3 (green) and BRANE Clust (pink) Precision-Recall curves on the DREAM4 dataset and on the first DREAM5 network. On the DREAM5 network also appears the ARACNE Precision-Recall curve in red. For a better readibility, all curves are truncated at the same point on the x axis.

Clust improves the accuracy of inferred networks of relatively small size but high precision. The evaluation of [12] on DREAM4 leads to poor results: the average of the maximal Precision over the five network does not exceed 35 %. On the larger network of DREAM5, the BRANE Clust gain is even more important, with a relative improvement of 19 % over GENIE3 and 38% over CLR. The corresponding Precision-Recall curves appear in Figure 1. Regarding computation times, our BRANE Clust approach is extremely fast, as it only takes 3 seconds on the DREAM5 network. In comparison to the weight computation times, that represent approximately 10 minutes for CLR and one hour for GENIE3, this cost is negligible.

A visualization of the inferred graph at a threshold resulting in a Precision of 0.75 is given in Figure 2. This means that $3/4$ of the graph is correctly predicted, clearly showing the modular structure of the network. The modularity is still visible at a lower Precision, but reaches its limits in improving GRN inference results.

4. CONCLUSIONS

In contrast to the large majority of clustering and graph inference method which heavily relies on Gaussian graphical models, our model incorporates a priori by fixing a small set of pre-labeled nodes and constraining the graph construction using the random walker segmentation algorithm. The significant improvement obtained on GRN inference datasets shows the potential efficiency that spatial coherency enforcement can bring to graph construction in a more general context. Our work could thus be inspiring for graph construction problems encountered in other applications.

REFERENCES

[1] L. J. Grady and J. Polimeni, *Discrete calculus: Applied analysis on graphs for computational science*, Springer, 2010.

[2] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.

[3] T. Maugey, A. Ortega, and P. Frossard, "Graph-based representation for multiview image geometry," *IEEE Trans. Image Process.*, 2015, In press.

[4] A. Wiesel, Y. C. Eldar, and A. O. Hero III, "Covariance estimation in decomposable Gaussian graphical models," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1482–1492, Mar. 2010.

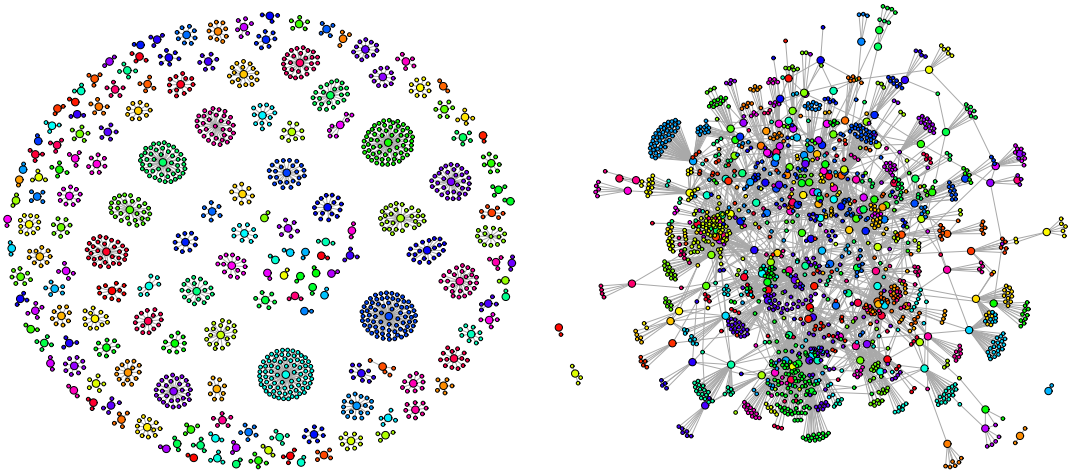
[5] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst, "Learning graphs from signal observations under smoothness prior," *PREPRINT*, 2014.

[6] M. E. J. Newman, "Communities, modules and large-scale structure in networks," *Nat. Phys.*, vol. 8, no. 1, pp. 25–31, 2012.

[7] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, Nov. 2006.

[8] M. E. Newman, "Modularity and community structure in networks," *Proc. Nat. Acad. Sci. U.S.A.*, vol. 103, no. 23, pp. 8577–8582, Jun. 2006.

[9] B. Abu-Jamous, F. Rui, D. J. Roberts, and A. K. Nandi, "Bi-CoPaM ensemble clustering application to five *Escherichia coli* bacterial datasets," in *Proc. Eur. Sig. Im-*



Clustering obtained at a Precision of 0.75 Clustering obtained at a Precision of 0.34

Fig. 2. Graph x and clustering results y on the Dream 5 network at different Precisions.

age Proc. Conf., Lisbon, Portugal, Sep. 1-5, 2014, pp. 2485–2489.

[10] H. Toh and K. Horimoto, "Inference of a genetic network by a combined approach of cluster analysis and graphical Gaussian modeling," *Bioinformatics*, vol. 18, no. 2, pp. 287–297, Feb. 2002.

[11] N. Friedman, "Inferring cellular networks using probabilistic graphical models," *Science*, vol. 303, no. 5659, pp. 799–805, Feb. 2004.

[12] J. Chiquet, A. Smith, G. Grasseau, C. Matias, and C. Ambroise, "SIMoNe: Statistical Inference for MOdular NEtworks," *Bioinformatics*, vol. 25, no. 3, pp. 417–418, 2009.

[13] D. Marbach, R. J. Prill, T. Schaffter, C. Mattiussi, D. Floreano, and G. Stolovitzky, "Revealing strengths and weaknesses of methods for gene network inference," *Proc. Nat. Acad. Sci. U.S.A.*, vol. 107, no. 14, pp. 6286–6291, Apr. 2010.

[14] D. Marbach, J. C. Costello, R. Küffner, N. M. Vega, R. J. Prill, D. M. Camacho, K. R. Allison, The DREAM5 Consortium, M. Kellis, J. J. Collins, and G. Stolovitzky, "Wisdom of crowds for robust gene network inference," *Nat. Meth.*, vol. 9, no. 8, pp. 796–804, 2012.

[15] Z. Kurt, N. Aydin, and G. Altay, "A comprehensive comparison of association estimators for gene network inference algorithms," *Bioinformatics*, vol. 30, no. 15, pp. 2142–2149, Aug. 2014.

[16] A. A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. Dalla Favera, and A. Califano, "ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context," *BMC Bioinformatics*, vol. 7 (Suppl. 1), no. 5, pp. S7, 2006.

[17] J. J. Faith, B. Hayete, J. T. Thaden, I. Mogno, J. Wierzbowski, G. Cottarel, S. Kasif, J. J. Collins, and T. S. Gardner, "Large-scale mapping and validation of *Escherichia coli* transcriptional regulation from a compendium of expression profiles," *PLoS Biol.*, vol. 5, no. 1, pp. 54–66, 2007.

[18] V. A. Huynh-Thu, A. Irrthum, L. Wehenkel, and P. Geurts, "Inferring regulatory networks from expression data using tree-based methods," *PLoS One*, vol. 5, no. 9, pp. 1–10, Sep. 2010.

[19] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, and N. Friedman, "Module networks: Discovering regulatory modules and their condition specific regulators from gene expression data," *Nat. Genet.*, vol. 34, no. 166–176, pp. 2003, 2003.

[20] A. Pirayre, C. Couprie, L. Duval, and J.-C. Pesquet, "Fast convex optimization for connectivity enforcement in gene regulatory network inference," in *Proc. Int. Conf. Acoust. Speech Signal Process.*, Melbourne, Australia, Apr. 19-24, 2015.

[21] J. Darbon, "Global optimization for first order Markov Random Fields with submodular priors," *Discrete Appl. Math.*, vol. 157, no. 16, pp. 3412–3423, 2009, Combinatorial Approach to Image Analysis.

[22] D. Singaraju, L. Grady, A. K. Sinop, and R. Vidal, "Continuous valued MRFs for image segmentation," in *Markov Random Fields for Vision and Image Processing*, A. Blake, P. Kohli, and C. Rother, Eds., pp. 127–142. MIT Press, 2011.