Signal Denoising on Graphs via Graph Filtering

Siheng Chen
Department of ECE
Center for Bioimage Informatics
Carnegie Mellon University
Pittsburgh, PA, USA
sihengc@andrew.cmu.edu

Aliaksei Sandryhaila
HP Vertica
Pittsburgh, PA, USA
aliaksei.sandryhaila@hp.com

José M. F. Moura
Department of ECE
Carnegie Mellon University
Pittsburgh, PA, USA
moura@ece.cmu.edu

Jelena Kovačević
Department of ECE & BME
Center for Bioimage Informatics
Carnegie Mellon University
Pittsburgh, PA, USA
jelenak@cmu.edu

***Abstract*—Signal recovery from noisy measurements is an important task that arises in many areas of signal processing. In this paper, we consider this problem for signals represented with graphs using a recently developed framework of discrete signal processing on graphs. We formulate graph signal denoising as an optimization problem and derive an exact closed-form solution expressed by an inverse graph filter, as well as an approximate iterative solution expressed by a standard graph filter. We evaluate the obtained algorithms by applying them to measurement denoising for temperature sensors and opinion combination for multiple experts.**

I. Introduction

Signals collected in physical, engineering, social and other domains are often characterized by nonlinear, complex structure that is captured by graphs and networks [1], [2]. They require novel representation and analysis techniques. Recently, a new framework of *discrete signal processing on graphs* was proposed for analyzing signals represented with graphs [3], [4]. This is a theoretical framework that generalizes fundamental concepts of the classical signal processing from regular domains, such as lines and rectangular lattices, to general graphs. Signal processing on graphs has found multiple applications, including approximation [5], sampling [6], [7], classification [8], [9], inpainting [10] and clustering [11] of signals on graphs.

In this paper, we consider a problem of signal denoising, that is, recovery of a true signal from a noisy measurement. There is extensive literature on this topic, focusing mostly on signals with regular structures, such as time series and digital images. Existing denoising techniques are often based on signal smoothing via lowpass filtering [12], [13], regularization by total variation [14], [15] and multiresolution analysis [16]–[18].

This work generalizes the problem of signal denoising from signals with regular structure to those with complex, irregular structure that is commonly represented by graphs. We propose a novel approach to signal denoising on graphs based on the total variation regularization; we derive an exact closed-form solution and an approximate iterative solution based, respectively, on an inverse and a standard graph filters. To evaluate the derived algorithms, we apply them to denoising of measurements from temperature sensors and to combining opinions from multiple experts.

II. Discrete signal processing on graphs

Here, we briefly review relevant concepts of the discrete signal processing on graphs (DSP$_{\rm G}$); a thorough introduction can be found in [3], [4].

Graph signals. DSP$_{\rm G}$ represent the structure of a signal with a graph $G = (\mathcal{V}, {\rm A})$, where $\mathcal{V} = \{v_n\}_{n=0}^{N-1}$ is the set of *vertices* and ${\rm A} \in \mathbb{C}^{N \times N}$ is a *graph shift*, or called weighted adjacency matrix, representing connections between vertices. Note that The nth signal coefficient corresponds to node v_n, and the weight ${\rm A}_{n,m}$ of an edge from the mth to the nth node characterizes the relation, such as similarity or dependency, between the corresponding signal values. Here a graph shift can be either symmetric or asymmetric to represent arbitrary relational dependencies. Using the representation graph G, a *graph signal* is defined as a map that assigns a signal coefficient $x_n \in \mathbb{C}$ to the graph node v_n. Graph signals can be written as a vector

$$\mathbf{x} \ = \ \begin{bmatrix} x_0 & x_1 & \ldots & x_{N-1} \end{bmatrix}^T \in \mathbb{C}^N, \qquad (1)$$

where the nth vector element x_n is indexed by the node v_n.

Graph filters. A graph filter is a system that takes a graph signal as an input and produces another graph signal as an output. The most elementary nontrivial graph filter, graph shift, replaces the signal value at a node with a weighted linear combination of values at its neighbors. This operation is written as

$$\mathbf{y} \ = \ {\rm A}\,\mathbf{x}.$$

Every linear, shift-invariant graph filter is a polynomial in the graph shift [3]

$$h({\rm A}) \ = \ \sum_{\ell=0}^{L-1} h_\ell \, {\rm A}^\ell = h_0 \, {\rm I} + h_1 \, {\rm A} + \ldots + h_{L-1} \, {\rm A}^{L-1}, \quad (2)$$

and its output is given by the matrix-vector product

$$\mathbf{y} \ = \ h({\rm A})\,\mathbf{x}.$$

Graph Fourier transform. Assume that the spectral decomposition, or eigendecomposition of the graph shift matrix A is

$${\rm A} = {\rm V}\,\Lambda\,{\rm V}^{-1}, \qquad (3)$$

where $\Lambda = \mathrm{diag}(\lambda_0, \ldots, \lambda_{N-1})$ is the diagonal matrix of N distinct eigenvalues and V is the matrix of corresponding eigenvectors.[1] The eigenvalues of A represent the graph *frequencies* and the eigenvectors form a basis of *spectral components*. The *graph Fourier transform* corresponds to the expansion of a graph signal (1) into the basis of spectral components and can be written as

$$\widehat{\mathbf{x}} = \begin{bmatrix}\widehat{x}_0 & \widehat{x}_1 & \ldots & \widehat{x}_{N-1}\end{bmatrix}^T = \mathrm{F}\,\mathbf{x}, \tag{4}$$

where $\mathrm{F} = \mathrm{V}^{-1}$ is the graph Fourier matrix. Respectively, the *inverse graph Fourier transform* reconstructs the signal from its frequency representation as

$$\mathbf{x} = \mathrm{F}^{-1}\,\widehat{\mathbf{x}}. \tag{5}$$

Graph total variation. "Smoothness" of a graph signal, that is, how much the signal varies with respect to the underlying graph, can be quantified by the *graph total variation* function [4],

$$\mathrm{TV}_{\mathrm{A}}(\mathbf{x}) = \left|\left|\mathbf{x} - \frac{1}{|\lambda_{max}(\mathrm{A})|}\,\mathrm{A}\,\mathbf{x}\right|\right|_1, \tag{6}$$

where $\lambda_{max}(\mathrm{A})$ denotes the eigenvalue of A with the largest magnitude. For simplicity of discussion, we assume $\lambda_{max}(\mathrm{A}) = 1$, which can always be achieved by normalizing A. A related measure of a graph signal's smoothness is the quadratic form of total variation,

$$\mathrm{S}_2(\mathbf{x}) = \frac{1}{2}\,||\mathbf{x} - \mathrm{A}\,\mathbf{x}||_2^2\,. \tag{7}$$

For both (6) and (7), the smaller the value of the function, the "smoother" the graph signal $\mathbf{x}$. From now on, we use the quadratic form of total variation to measure the smoothness of graph signals for the computational simplicity.

III. Graph signal denoising via regularization

In this section, we formulate an optimization problem for graph signal denoising and derive its exact and approximate solutions.

Consider a noisy graph signal measurement

$$\mathbf{t} = \mathbf{x} + \mathbf{w},$$

where $\mathbf{x}$ is the true signal and $\mathbf{w}$ is noise. The goal of graph signal denoising is to recover $\mathbf{x}$ from $\mathbf{t}$ by removing the noise $\mathbf{w}$. We assume that the true signal $\mathbf{x}$ is "smooth" with respect to the underlying graph and the noise $\mathbf{w}$ is randomly distributed. In this case, inspired by the classical signal processing, where time series and digital images are denoised via regularization [14], [15], we formulate graph signal denoising as the optimization problem

$$\widetilde{\mathbf{x}} = \mathbf{argmin}\ \frac{1}{2}||\mathbf{x} - \mathbf{t}||_2^2 + \alpha\,\mathrm{S}_2(\mathbf{x}). \tag{8}$$

Here, the first term keeps the denoised signal similar to the measurement and the second term enforces the "smoothness" of the solution using the quadratic form (7). The tuning parameter α controls the trade-off between two terms of the objective function

The objective function in (8) is a linear combination of two quadratic functions of $\mathbf{x}$. Its derivative is

$$\begin{aligned} & \frac{\partial}{\partial\,\mathbf{x}}\left(\frac{1}{2}||\mathbf{x} - \mathbf{t}||_2^2 + \alpha\,\mathrm{S}_2(\mathbf{x})\right) \\ = {} & \frac{1}{2}\frac{\partial}{\partial\,\mathbf{x}}\left((\mathbf{x} - \mathbf{t})^*(\mathbf{x} - \mathbf{t}) + \alpha\,\mathbf{x}^*(\mathrm{I} - \mathrm{A})^*(\mathrm{I} - \mathrm{A})\,\mathbf{x}\right) \\ = {} & (\mathbf{x} - \mathbf{t}) + \alpha(\mathrm{I} - \mathrm{A})^*(\mathrm{I} - \mathrm{A})\,\mathbf{x}, \end{aligned} \tag{9}$$

where $*$ denotes the Hermitian transpose. Setting the right-hand side of (9) to zero, we obtain the solution to the optimization problem (8)

$$\widetilde{\mathbf{x}} = (\mathrm{I} + \alpha(\mathrm{I} - \mathrm{A})^*(\mathrm{I} - \mathrm{A}))^{-1}\,\mathbf{t}. \tag{10}$$

The advantage of the solution (10) is that it is exact and in a closed form. However, the inversion of an $N \times N$ matrix in (10) requires $O(N^3)$ arithmetic operations, and for large values of N this operation is prohibitively expensive and numerically unstable. Furthermore, in many real-world applications the graph shift matrix A is sparse and has only a few nonzero entries, its inverse, as well as the inverse matrix in (10), are dense matrices that require $O(N^2)$ arithmetic operations, again an expensive operation for large values of N. To avoid these issues, in the following section we derive an alternative solution that implements or approximates (10) with a graph filter (2).

IV. Graph filter solution

To avoid the inversion and achieve a faster implementation, instead of using conjugate gradient descent algorithm or other iterative methods [19] to perform matrix inversion in (10), we propose to find a graph filter $h(\mathrm{A})$ that solves (8) as

$$\widetilde{\mathbf{x}} \;=\; h(\mathrm{A})\mathbf{t}. \tag{11}$$

This solution avoids expensive matrix inversion and multiplication with a dense matrix, and instead performs iterative multiplications with the matrix A, which in many practical cases is highly sparse. In general, if A is K-sparse, that is, on average every row of A contains only K nonzero elements, the solution (11) requires only $O(LKN)$ arithmetic operations, where L is the number of filter taps in (2).

Comparing (11) with (10), we conclude that the graph filter must satisfy $h(\mathrm{A}) = (\mathrm{I} + \alpha(\mathrm{I} - \mathrm{A})^*(\mathrm{I} - \mathrm{A}))^{-1}$. The following results establish when this equality holds.

Lemma 1 *If the eigendecomposition of* A *is as in* (3) *and if matrix* $\mathrm{V}^{-1}\,\mathrm{A}^*\,\mathrm{V}$ *is diagonal, then* A^* *is a graph filter, that is, there exists a polynomial* (2) *that satisfies*

$$h(\mathrm{A}) = \mathrm{A}^*\,. \tag{12}$$

[1] The assumptions on the existence of the eigendecomposition and uniqueness of eigenvalues are made solely for the simplicity of discussion in this paper. In general, $\mathrm{DSP}_{\mathrm{G}}$ applies to arbitrary graph, including those with non-diagonalizable adjacency matrices and with repeating eigenvalues [3].

Proof: Assume that $\mathrm{V}^{-1}\,\mathrm{A}^*\,\mathrm{V} = \mathrm{diag}(d_0, \dots, d_{N-1}) = \mathrm{D}$ is a diagonal matrix. Then, (12) can be written as

$$\begin{aligned} \mathrm{D} &= \mathrm{V}^{-1}\,h(\mathrm{A})\,\mathrm{V} \\ &= \mathrm{V}^{-1}\,h(\mathrm{V}\,\Lambda\,\mathrm{V}^{-1})\,\mathrm{V} \\ &= h(\Lambda) \\ &= \mathrm{diag}\left(h(\lambda_0), \dots, h(\lambda_{N-1})\right). \end{aligned}$$

Hence, the graph filter must satisfy the system of equations $h(\lambda_n) = d_n$ for $0 \le n < N$.

This system always has a solution if $L = N$, that is, if the graph filter (2) is a polynomial of degree N. Thus, if $\mathrm{V}^{-1}\,\mathrm{A}^*\,\mathrm{V}$ is a diagonal matrix, then A^* can be implemented by a graph filter of degree N. ■

The following theorem shows that if A^* is a graph filter, the inverse matrix in (10) is also a graph filter.

Theorem 1 *Assume that* A^* *is a graph filter, we then have*

$$(\mathrm{I} + \alpha(\mathrm{I} - \mathrm{A})^*(\mathrm{I} - \mathrm{A}))^{-1} = g(\mathrm{A}), \qquad (13)$$

where the graph filter $g(\mathrm{A})$ *satisfies, for* $0 \le n < N$,

$$\begin{aligned} g(\lambda_n) &= 1/((1 + \alpha - \alpha h_0) - \alpha\lambda_n \qquad (14) \\ &\quad + \alpha \sum_{k=2}^{N-1} (h_{k-1} - h_k)\lambda_n^k + \alpha h_{N-1}\lambda_n^N). \end{aligned}$$

Proof: From Lemma 1,

$$\begin{aligned} (\mathrm{I} - \mathrm{A})^*(\mathrm{I} - \mathrm{A}) &= \mathrm{I} - \mathrm{A} - \mathrm{A}^* + \mathrm{A}^*\,\mathrm{A} \\ &= \mathrm{I} - \mathrm{A} - h(\mathrm{A}) + h(\mathrm{A})\,\mathrm{A} \\ &= \mathrm{V}\left[\mathrm{I} - \Lambda - h(\Lambda) + h(\Lambda)\,\Lambda\right]\mathrm{V}^{-1}, \end{aligned}$$

from which

$$\begin{aligned} &(\mathrm{I} + \alpha(\mathrm{I} - \mathrm{A})^*(\mathrm{I} - \mathrm{A}))^{-1} \\ &= \mathrm{V}\left[\mathrm{I} + \alpha(\mathrm{I} - \Lambda - h(\Lambda) + h(\Lambda)\,\Lambda)\right]^{-1}\mathrm{V}^{-1} \\ &= \mathrm{V}\,g(\Lambda)\,\mathrm{V}^{-1} \\ &= g(\mathrm{A}), \end{aligned}$$

where polynomial $g(\mathrm{A})$ satisfies

$$\begin{aligned} g(\lambda_n) &= \frac{1}{1 + \alpha(1 - \lambda_n - h(\lambda_n) + h(\lambda_n)\lambda_n)} \\ &= 1/((1 + \alpha - \alpha h_0) - \alpha\lambda_n \\ &\quad + \alpha \sum_{k=2}^{N-1} (g_{k-1} - g_k)\lambda_n^k + \alpha h_{N-1}\lambda_n^N). \end{aligned}$$

■

Corollary 1 *If the graph shift is Hermitian, the graph filter* $g(\mathrm{A})$ *in* (13) *satisfies*

$$g(\lambda_n) = \frac{1}{1 + \alpha(1 - \lambda_n)^2}. \qquad (15)$$

Proof: If A is Hermitian, it satisfies $\mathrm{A} = \mathrm{A}^*$, which simplifies (12) to $h(\mathrm{A}) = \mathrm{A}$. Hence, $h_0 = 1$ and $h_\ell = 0$ for $\ell \ge 1$, and (14) becomes (15). ■

Approximate solution. The result of Theorem 1 holds, and hence, an exact graph filter solution (11) to the denoising problem (8) exists, only if Lemma 1 holds. In other cases, we can find an approximate solution for (12) as $h(\mathrm{A}) = \mathrm{V}\,\widehat{\mathrm{D}}\,\mathrm{V}^{-1}$, where $\widehat{\mathrm{D}}$ is a solution to the optimization problem

$$\widehat{\mathrm{D}} = \mathbf{argmin}_{\mathrm{D} \in \mathcal{D}}\ ||\mathrm{V}\,\mathrm{D}\,\mathrm{V}^{-1} - \mathrm{A}^*||_F^2 \qquad (16)$$

over the set $\mathcal{D}$ of all diagonal matrices. Here, $||\cdot||_F$ denotes the Frobenius norm.

V. Experimental Results

In this section, we evaluate the presented solutions by applying them to denoising of sensor measurements and combining of experts' opinions. Since a graph shift can be either symmetric or asymmetric, in both experiments, we build a directed graph, which cannot be handled by graph-Laplacian based methods [20].

A. Sensor measurement denoising

Setup. We consider a dataset of daily temperature measurements collected by 150 weather stations across the United States [3]. Measurements from all stations on a single day represent a graph signal (1) of length $N = 150$. The dataset contains one full year of measurements, that is, 365 graph signals. The measured temperatures are in the range from -50 to 120 degree Fahrenheit.

The graph representing this dataset is obtained from the geodesic distances between stations. We use an 8-nearest neighbor graph with nodes representing stations. Each node is connected to eight of its geographically closest neighbors. The corresponding edge weights are $\mathrm{A}_{n,m} = \mathrm{P}_{n,m}/\sum_n \mathrm{P}_{n,m}$, where

$$\mathrm{P}_{n,m} = \exp\left(-\frac{d_{n,m}}{\sum_{n,m} d_{n,m}/N^2}\right), \quad 0 \le n, m < N,$$

and where $d_{n,m}$ is the geodesic distance between the nth and the mth weather stations.

Experiments. We simulate noisy measurements by adding random noise to the original data. The noise is generated using four different distributions: a Gaussian distribution with zero mean and standard deviations 5 and 10, denoted, respectively, as $\mathcal{N}(0, 5^2)$ and $\mathcal{N}(0, 10^2)$, and a uniform distribution over intervals $[-5, 5]$ and $[-10, 10]$, denoted, respectively, as $\mathcal{U}([-5, 5])$ and $\mathcal{U}([-10, 10])$.

Given the graph shift and the noisy graph signals, we apply the exact graph total variation regularization (GTVR) solution (10) and the graph filter (GF) solution (11). In the latter case, the conditions of Lemma 1 are not satisfied; hence, we use the approximate solution obtained from (16). Note that, if Lemma 1 held, the results of GTV and GF approaches would have been exactly the same.

Results. For each of 365 graph signals, we calculate the root mean square error (RMSE) between the ground truth (the noiseless signal) and the denoised signal obtained using GTVR and GF approaches. Table I shows the average RMSE over all 365 graph signals. for the four considered noise

Noise	Method	Tuning parameter α					
		0	10^{-2}	10^{-1}	1	10	100
$\mathcal{N}(0, 5^2)$	**GTVR**	4.98	4.93	4.52	**3.35**	4.66	6.78
	GF	4.98	4.93	4.52	**3.35**	4.73	7.23
$\mathcal{N}(0, 10^2)$	**GTVR**	9.96	9.85	9.01	5.67	**5.07**	6.91
	GF	9.96	9.85	9.02	5.68	**5.13**	7.34
$\mathcal{U}([-5, 5])$	**GTVR**	2.89	2.86	2.64	**2.57**	4.56	6.76
	GF	2.89	2.86	2.64	**2.57**	4.64	7.21
$\mathcal{U}([-10, 10])$	**GTVR**	5.78	5.72	5.24	**3.70**	4.71	6.80
	GF	5.78	5.72	5.24	**3.70**	4.78	7.25

TABLE I: Average RMSE of denoised temperature measurements over the period of one year.

distributions and different values of the trade-off parameter α in (8). As the results demonstrate, both the exact and approximate solutions to graph signal denoising perform well, leading to average errors as small as 2.57 degrees Fahrenheit. Furthermore, although GF is an approximation of the exact solution GTVR, it produces very similar denoising errors, which highlights its practical usefulness in data denoising.

B. Combining multiple opinions

In many real-world classification problems, the way to obtain the ground truth is through experts' opinions, but it is sometimes hard to have access to it. For instance, when a dataset is too large, obtaining experts' opinion is too expensive, or experts' opinions differ from each other, which happens, for example, in biomedical image classification [21]. In this case, a popular solution is to use multiple users, experts, or classifiers to label dataset elements and then combine their opinions into the final estimate of the ground truth [22]. As we demonstrate here, opinion combining can be formulated and solved as graph signal denoising problem.

Setup. We consider the task of recognizing hand-written digits. In particular, we study two binary classification problems: distinguishing between images of digits 1 and 8, and between images of digits 4 and 9. The former is a less difficult task, since digits 1 and 8 look significantly different, and the latter is a more difficult task, since digits 4 and 9 have similar shapes. We randomly pick 500 images of each digit from the MNIST dataset [23]. For each classification problem, the corresponding $N = 1000$ images are represented by a 8-nearest neighbor graph, where each node represents an image and is connected to eight most similar images, where similarity is measured by the ℓ_2 norm of difference between two images. Note that the digits in the MNIST dataset are aligned well and we do not consider the translation problem. Corresponding edge weights are defined as $\mathrm{A}_{n,m} = \mathrm{P}_{n,m} / \sum_n \mathrm{P}_{n,m}$, where

$$\mathrm{P}_{n,m} = \exp\left(\frac{-||\mathbf{s}_n - \mathbf{s}_m||_2}{\sum_{n,m} ||\mathbf{s}_n - \mathbf{s}_m||_2 / N^2} \right), \quad 0 \le n, m < N,$$

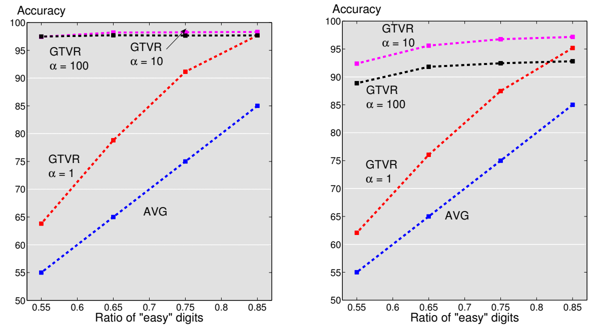
where $\mathbf{s}_n$ is the vectorized representation of the nth image.

Experiments. For each binary classification problem, we simulate $K = 100$ experts that label all N images. Each expert labels one image of a digit as $+1$ and the other as -1 on to produce a vector $\mathbf{t}_k \in \{+1, -1\}^N$. Note that each expert makes mistakes when labeling digits. We think of $\mathbf{t}_k$ as a graph signal with noise that represents the kth expert's errors.

Since some images are harder to classify than others (for instance, some people write 4 and 9 almost identically, while others write these digits differently), we split the dataset of N images into "easy" and "hard" images and assume that there is a 90% chance that an expert classifies an "easy" image correctly and only a 30% chance that an expert classifies a "hard" image correctly. We consider four cases of "easy" images making up 55%, 65%, 75%, and 85% of the entire dataset.

A baseline solution is to average (AVG) all experts opinions into vector $\mathbf{t}_{\rm avg} = (\sum_k \mathbf{t}_k)/K$ and then use the signs $\mathrm{sign}(\mathbf{t}_{\rm avg})$ vector as the labels to images. We compare the baseline solution with the GTVR solution (10), where we first denoise every signal $\mathbf{t}_k$ and then compute the average of denoised signals $\widetilde{\mathbf{t}}_{\rm avg} = (\sum_k \widetilde{\mathbf{t}}_k)/K$ and use $\mathrm{sign}(\widetilde{\mathbf{t}}_{\rm avg})$ as labels to images. For the trade-off parameter α in (8), we consider values $\alpha \in \{1, 10, 100\}$.

Results. Figure 1 shows the classification accuracy of our approach and the baseline averaging approach. Viewing experts's opinions as noisy graph signals and denoising them before averaging significantly improves the classification accuracy by up to 40% accuracy. Although distinguishing between digits 4 and 9 is more difficult that between digits 1 and 8, we still achieve accuracy above 90% in almost all cases. Also, observe that the trade-off parameter α has a significant influence on the classification accuracy.



(a) Digits 1 and 8.

(b) Digits 4 and 9.

Fig. 1: Classification accuracy of handwritten digits by combining multiple experts' opinions.

VI. Conclusions

We presented a new algorithm for denoising of signals residing on arbitrary graphs. We approached this task through the regularization of the graph total variation of noisy signals and formulated the corresponding optimization problem. We derived an exact, closed-form solution expressed with an inverse graph filter and an iterative solution expressed with a standard graph filter. We also identified conditions when the graph filter solution is exact or approximate. To evaluate the proposed approach, we applied it to denoising of sensor measurements and to classification via combination of experts' opinions and showed that our approach performs well in these applications.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge support from the NSF through awards 1130616 and 1017278, as well as CMU Carnegie Institute of Technology Infrastructure Award.

REFERENCES

[1] M. Jackson, *Social and Economic Networks*. Princeton University Press, 2008.

[2] M. Newman, *Networks: An Introduction*. Oxford University Press, 2010.

[3] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 61, no. 7, pp. 1644–1656, 2013.

[4] ——, "Discrete signal processing on graphs: Frequency analysis," *IEEE Trans. Signal Process.*, vol. 62, pp. 3042 – 3054, 2014.

[5] D. Thanou, D. I. Shuman, and P. Frossard, "Parametric dictionary learning for graph signals," in *Proc. IEEE Glob. Conf. Signal Information Process.*, Austin, TX, Dec. 2013, pp. 487–490.

[6] S. K. Narang, A. Gadde, and A. Ortega, "Signal processing techniques for interpolation in graph structured data," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2013, pp. 5445–5449.

[7] P. Liu, X. Wang, and Y. Gu, "Coarsening graph signal with spectral invariance," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2014, pp. 1075–1079.

[8] S. Chen, F. Cerda, P. Rizzo, J. Bielak, J. H. Garrett, and J. Kovačević, "Semi-supervised multiresolution classification using adaptive graph filtering with application to indirect bridge structural health monitoring," *IEEE Trans. Signal Process.*, vol. 62, no. 11, Jun. 2014.

[9] A. Sandryhaila and J. M. F. Moura, "Classification via regularization on graphs," in *Proc. IEEE Glob. Conf. Signal Information Process.*, Austin, TX, Dec. 2013, pp. 495–498.

[10] S. Chen, A. Sandryhaila, G. Lederman, Z. Wang, J. M. F. Moura, P. Rizzo, J. Bielak, J. H. Garrett, and J. Kovačević, "Signal inpainting on graphs via total variation minimization," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Florence, May 2014.

[11] P.-Y. Chen and A. Hero, "Local fiedler vector centrality for detection of deep and overlapping communities in networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2014.

[12] M. Vetterli and J. Kovačević, *Wavelets and Subband Coding*, ser. Signal Processing. Englewood Cliffs, NJ: Prentice Hall, 1995, http://waveletsandsubbandcoding.org/.

[13] S. Mallat, *A Wavelet Tour of Signal Processing*, 3rd ed. New York, NY: Academic Press, 2009.

[14] L. I. Rudin, Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, no. 1–4, pp. 259–268, 1992.

[15] S. B. A. Elmoataz, O. Lezoray, "Nonlocal discrete regularization on weighted graphs: a framework for image and manifold processing," *ieee-trans-ip*, no. 7, pp. 1047–1060, 2008.

[16] J. Portilla, M. Wainwright, V. Strela, and E. Simoncelli, "Image denoising using a scale mixture of Gaussians in the wavelet domain," *IEEE Trans. Image Process.*, vol. 12, no. 11, pp. 1338–1351, Nov. 2003.

[17] P. L. Dragotti, V. Velisavljević, M. Vetterli, and B. Beferull-Lozano, "Discrete directional wavelet bases and frames for image compression and denoising," in *Proc. SPIE Conf. Wavelet Appl. Signal Image Process.*, Aug. 2003.

[18] A. K. Fletcher, V. K. Goyal, and K. Ramchandran, "Iterative projective wavelet methods for denoising," in *Proc. SPIE Conf. Wavelet Appl. Signal Image Process.*, vol. 5207, San Diego, CA, Aug. 2003, pp. 9–15.

[19] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, Mar. 2004.

[20] A. J. Smola and R. Kondor, "Kernels and regularization on graphs," in *Learning theory and kernel machines*. Springer, 2003, pp. 144–158.

[21] A. Kuruvilla, N. Shaikh, A. Hoberman, and J. Kovačević, "Automated diagnosis of otitis media: A vocabulary and grammar," *Int. J. Biomed. Imag., sp. iss. Computer Vis. Image Process. for Computer-Aided Diagnosis*, Aug. 2013.

[22] S. R. Cholleti, S. A. Goldman, A. Blum, D. G. Politte, S. Don, K. Smith, and F. Prior, "Veritas: Combining expert opinions without labeled data." *International Journal on Artificial Intelligence Tools*, vol. 18, no. 5, pp. 633–651, 2009.

[23] Y. Lecun, B. Lon, B. Yoshua, and H. Patrick, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, vol. 86, no. 11, Nov. 1998, pp. 2278–2324.