THE NYSTRÖM EXTENSION FOR SIGNALS DEFINED ON A GRAPH

Ayelet Heimowitz

The Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ, USA
aheimowitz@math.princeton.edu

Yonina C. Eldar

Department of Electrical Engineering, Technion, Haifa, Israel
yonina@ee.technion.ac.il

ABSTRACT

In this paper we introduce a computationally efficient solution to the problem of graph signal interpolation. Our solution is derived using the Nyström extension and is due to the properties of the Markov matrix which we use as our graph shift operator, inspired by diffusion maps. We focus on graph signals that are smooth over the graph. This assumption cements the relationship between the graph and the graph signal. We experimentally verify our suggested framework on the MNIST data set of handwritten digits.

***Index Terms*—** Signal processing on graphs, Diffusion maps, Nyström extension.

I. INTRODUCTION

The advancement of technology over the past few decades has made vast amounts of data available from various sources such as social networks, government agencies, commercial and academic bodies and more. For this reason in many applications we find ourselves limited not by the amount of data available, but by the time necessary for processing big data.

A popular approach to data processing is to model the underlying geometry of the data as weighted graphs. The different data elements (data points) are modeled as nodes, while the pairwise relationships between these elements are modeled as edges. For example, an image may be modeled as a graph where the nodes represent the basic elements (pixels, super-pixels) while the edges reflect similarity between these elements. This similarity can be in color, texture or some other feature.

When modeling an image as a graph, there is no obvious definition for the direction of an edge. For this reason images are often modeled as undirected graphs. Undirected graphs are graphs possessing bidirectional edges, as opposed to directed graphs in which the pairwise relationships between nodes are a-symmetric. One may use a directed graph, for example, to model citations of academic papers as a paper may only cite previously published papers.

In recent years scientists have began to use weighted graphs in order to process signals residing over irregular domains [1], [2], [3], [4], [5]. There are two main categories for this signal processing. The first category concerns undirected graphs with non-negative weights, and uses results from spectral graph theory [1]. The second category contains general graphs. In this case signal processing may rely upon algebraic signal processing theory [2]. In either approach, the graph signals are defined as a mapping from each node of the graph to some value. In this paper we define a graph signal as a mapping that preserves the geometry of the graph in the sense that closely connected nodes are mapped to similar values.

In addition to this definition, we restrict the discussion to weighted undirected graphs with non-negative weights. This restriction allows us to represent the graph as a Markov matrix, which is a matrix containing transition probabilities between the nodes in the graph. This representation is useful for two reasons. First, the eigenvalues and eigenvectors of the Markov matrix define an embedding of the graph in the Euclidean space. This embedding, introduced by Coifman and Lafon [6], is called a diffusion map and has been shown to preserve the local geometry of the graph. Thus the graph is expressed as a cloud of points in a Euclidean space where nearby points correspond to similar nodes. Since the graph signal maps each node of the graph to some value, it also maps each diffusion embedding vector to this value.

Another advantage of using the Markov matrix representation is that it is similar to a positive semi-definite (PSD) matrix called the normalized graph Laplacian. This similarity allows to efficiently approximate the eigendecomposition of the Markov matrix. Our suggested method of approximation is a variation on the Nyström extension [7], [8], [9], [10] which allows to efficiently approximate the eigendecomposition of PSD matrices.

In this paper we propose an efficient estimation of the eigendecomposition of the Markov matrix, leading to an efficient solution to the problem of graph signal interpolation. In this problem, the graph signal maps some subset of the nodes of the graph to known values, and the goal is to recover the value to which the remaining nodes are mapped.

We test our resulting efficient method of graph signal interpolation on the application of digit recognition. Specifically, we formulate the problem of identifying a hand written digit from an image as a graph signal interpolation problem and show that our suggested framework acheives high accuracy with low runtime.

This paper is organized as follows. Section II contains a summary of the diffusion maps framework. In Section III we introduce the field of signal processing on graphs. Section IV presents a variation on Nyström's extension allowing to apply it to the Markov matrix. In addition, we derive an efficient graph signal interpolation framework. In Section V we present experimental results of our graph signal interpolation framework. We apply this framework to the MNIST data set of handwritten digits [11], [12].

II. DIFFUSION MAPS

The diffusion maps framework uses a graph to represent some data set $X = \{x_1, \ldots, x_N\}$. For weighted graphs we denote the affinity matrix containing edge weights as $\mathbf{W}$. The ijth element of the affinity matrix specifies the weight of an edge between node v_i and node v_j. If no edge exists between these nodes, then $W_{i,j}$ is set to zero.

The Markov transition matrix of a weighted undirected graph with non-negative weights is a normalization of the affinity matrix $\mathbf{W}$ such

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 646804-ERC- COG-BNYQ

This research was carried out while the first author was a postdoctoral researcher at the Department of Electrical Engineering, Technion, Haifa, Israel.

that each row sums to 1. Mathematically, we express the Markov matrix as

$$\mathbf{P} = \mathbf{D}^{-1}\mathbf{W}, \tag{1}$$

where $\mathbf{D}$ is the diagonal matrix of node degrees

$$d(v_i) = \mathbf{D}_{i,i} = \sum_{j=1}^{N} \mathbf{W}_{i,j}. \tag{2}$$

This normalization allows to consider the entries of the Markov matrix to be transition probabilities between nodes of the graph.

Powers of the Markov matrix can be interpreted as transition probabilities between nodes of the graph when allowing the transition to occur over several steps. For example,

$$p_2(x_i, x_j) = \mathbf{P}^2_{i,j} = \sum_{m=1}^{N} \mathbf{P}_{i,m}\mathbf{P}_{m,j} \tag{3}$$

which is the probability of transition from node v_i to node v_j in two steps. Another way to view powers of the Markov matrix is as an expression of pairwise relationships in the graph at different scales. The lowest scale considers only adjacent nodes, while the t-th scale takes into consideration nodes connected by length t paths.

Coifman and Lafon [6] focus on a scale-dependent distance between nodes of a graph known as the diffusion distance. Formally, the diffusion distance at scale t is defined as [6], [13], [14]

$$D^2_t(i,j) = \|p_t(x_i,\cdot) - p_t(x_j,\cdot)\|^2_{\phi_0} \tag{4}$$

where ϕ_0 is the leading left eigenvector of the Markov matrix and $p_t(\cdot,\cdot)$ are elements of $\mathbf{P}^t$. It can be shown that ϕ_0 is the stationary distribution of the Markov chain [13],

$$\phi_0(i) = \frac{d(v_i)}{\sum_{l=1}^{N} d(v_l)}. \tag{5}$$

For simplicity, we denote

$$d(\mathcal{G}) = \sum_{l=1}^{N} d(v_l).$$

The diffusion distance can then be expressed as

$$D^2_t(x_i, x_j) = \sum_{m=1}^{N} \left(\left(\mathbf{P}^t\right)_{i,m} - \left(\mathbf{P}^t\right)_{j,m}\right)^2 \cdot \frac{d(\mathcal{G})}{d(v_m)}. \tag{6}$$

The diffusion distance expresses the connectivity of the graph. If two nodes, v_i and v_j, have many short paths connecting them then the probability of transition between them ($P^t_{i,j}$) will be high. This means that the probability of each of these nodes transitioning to some general node v_m is similar, causing a low diffusion distance $D^2(i,j)$. On the other hand, if v_i and v_j are relatively disconnected, then the probability of each node transitioning to some general node v_m is different, causing the diffusion distance to be high. In this way the diffusion distance contains the connectivity information of the graph.

The diffusion distances at scale t are locally preserved when each node v_i of the graph is embedded in a Euclidean space by the diffusion embedding vector

$$\Psi_t(i) = \begin{bmatrix} \lambda_2^t \psi_2(i) \\ \lambda_3^t \psi_3(i) \\ \vdots \\ \lambda_N^t \psi_N(i) \end{bmatrix}, \tag{7}$$

where ψ_j is the jth eigenvector of $\mathbf{P}$ and λ_j is the jth eigenvalue. This embedding was introduced by Coifman and Lafon [6].

Keller and Gur [13] use the diffusion maps embedding (7) to solve the sensor localization problem. They define the location of a sensor as a function over the diffusion embedding vectors of that sensor. This function is approximated from a small set of known sensor locations and then applied to all diffusion embedding vectors. They named this framework spectral regression. When a graph is represented by the Markov matrix, we show that spectral regression can be applied to graph signal interpolation. We discuss this problem in Section V.

III. SIGNAL PROCESSING ON GRAPHS

As in diffusion maps, in the field of signal processing on graphs a data set is represented by a weighted graph. The pairwise (edge) information of the graph is contained in the graph shift operator $\mathbf{A}$ [2]. This is a matrix where the ijth entry corresponds to the relationship between v_i and v_j. If these nodes are not adjacent, then we set $\mathbf{A}_{i,j} = 0$.

A graph signal is defined in the literature as a mapping from each node v_i to some value s_i [2], [3], [4], [15]. In this paper, we define a graph signal as having an inherent connection to the graph it is defined on. This connection should cause a manifestation of the graph connectivity in the graph signal, leading to a slow change of the graph signal over the edges of the graph.

An example of a graph signal can be taken from computer vision. We construct a graph in which each node represents a basic image element (pixel, superpixel). The edges of the graph contain information of color similarity between adjacent image elements. The graph signal maps each image element v_i to one of two classes foreground ($\mathbf{s}_i = 1$) or background ($\mathbf{s}_i = 0$). For obvious reasons, adjacent image elements with similar color information should be mapped to the same class. This means the graph signal must change slowly over the edges of the graph.

The field of signal processing on graphs extends traditional signal processing to signals defined over irregular domains represented by graphs. For example, the time shift operation is extended to a graph shift operation [2]. This new operation is defined as

$$\tilde{\mathbf{s}} = \mathbf{A}\mathbf{s}. \tag{8}$$

Another operation generalized for signals defined over graphs is the Fourier transform. The Graph Fourier transform (GFT) [2] is defined as

$$\hat{\mathbf{s}} = \mathbf{V}^{-1}\mathbf{s} \tag{9}$$

where $\mathbf{V}$ is the matrix of eigenvectors (or generalized eigenvectors where necessary) of the graph shift operator. The inverse graph Fourier transform (IGFT) is

$$\mathbf{s} = \mathbf{V}\hat{\mathbf{s}}. \tag{10}$$

The graph shift operator is very important to the field of signal processing on graphs. However, its definition as a weighted adjacency matrix is not unique. In [16] we suggested defining the graph shift operator $\mathbf{A}$ as the Markov matrix $\mathbf{P}$. In the remainder of this paper we refer only to the case that $\mathbf{A} = \mathbf{P}$, and provide a computationally efficient method of computing $\mathbf{V}$ for our suggested definition of the graph shift operator.

IV. NYSTRÖM EXTENSION

IV-A. Nyström Extension for Positive Semi-definite Matrices

The Nyström Extension is a method for approximating the eigen-decomposition of PSD matrices. While the Markov matrix is not PSD, in this section we show that the Nyström extension can be applied to it.

First, we briefly present the extension for some PSD matrix $\mathbf{K}$. This matrix can be considered as a combination of four block matrices,

$$\mathbf{K} = \begin{bmatrix} \mathbf{E} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{C} \end{bmatrix}, \tag{11}$$

where $\mathbf{E} \in \mathbb{R}^{r\times r}$, $\mathbf{B} \in \mathbb{R}^{N-r\times r}$, and $\mathbf{C} \in \mathbb{R}^{N-r\times N-r}$ for some $0 < r < N$. The Nyström extension [7], [8], [9], [10] is a method for extending the eigenvectors of $\mathbf{E}$ to create an estimate of r eigenvectors of $\mathbf{K}$.

Since $\mathbf{K}$ is symmetric, the matrix $\mathbf{E}$ must be symmetric as well. Thus it is diagonalizable as

$$\mathbf{E} = \mathbf{Z}\mathbf{Q}\mathbf{Z}^T, \tag{12}$$

where $\mathbf{Z}$ is the unitary matrix that contains in its columns the eigenvectors of $\mathbf{E}$ and $\mathbf{Q}$ is a diagonal matrix that contains the eigenvalues of $\mathbf{E}$ in its main diagonal. The Nyström extension of the matrix of eigenvectors of $\mathbf{K}$ is given by [7]

$$\tilde{\mathbf{Z}} = \begin{bmatrix} \mathbf{Z} \\ \mathbf{B}\mathbf{Z}\mathbf{Q}^{-1} \end{bmatrix}. \tag{13}$$

The diagonal of the matrix $\mathbf{Q}$ is the approximation of the eigenvalues of $\mathbf{K}$.

IV-B. Tailoring Nyström Extension to the Markov Matrix

It is important to note that the Markov matrix $\mathbf{P}$ is not PSD. The eigenvalues of $\mathbf{P}$ range from -1 to 1, and $\mathbf{P}$ is not even symmetric. However, the Markov matrix is similar to the normalized graph Laplacian

$$\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}\left(\mathbf{D} - \mathbf{W}\right)\mathbf{D}^{-\frac{1}{2}} = \mathbf{I}_N - \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}} \tag{14}$$

which is PSD. It is easy to show that the relationship between the eigenvectors of the Markov matrix ψ_i and the eigenvectors of the graph Laplacian $\hat{\psi}_i$ are given by

$$\psi_i = \mathbf{D}^{-\frac{1}{2}}\hat{\psi}_i. \tag{15}$$

The eigenvalues of the Markov matrix λ_i are related to the eigenvalues of the graph Laplacian $\hat{\lambda}_i$ as

$$\lambda_i = 1 - \hat{\lambda}_i. \tag{16}$$

Since the Laplacian is a PSD matrix the Nyström extension can be preformed on the graph Laplacian $\mathbf{L}$. The simple connection between the eigendecomposition of the Laplacian and the Markov matrix allows us to use the Nyström extension on the Markov matrix. This is done in two steps. The first step is to compute the eigendecomposition of the Laplacian ($\tilde{\mathbf{Z}}$, $\mathbf{Q}$). The second step is to convert these matrices to the eigendecomposition of the Markov matrix via (15) and (16).

In other words, when the graph shift operator is the Markov matrix, the Nyström extension can be used for estimation of the GFT. In turn, as we will now show, a small variation on the Nyström extension will allow us to solve the problem of graph signal interpolation efficiently.

In the graph signal interpolation problem, we assume the graph signal at a set of indices $\mathcal{R} \in \{1,\ldots,N\}^r$ is known. The goal is to estimate the remaining $N-r$ entries. We examine the GFT of $\mathbf{s}$ using the approximated eigenvectors of the graph shift

$$\mathbf{s} = \mathbf{V}\mathbf{x}. \tag{17}$$

The vector $\mathbf{x}$ contains the frequencies of the graph signal $\mathbf{s}$.

For convenience, we assign indices to nodes so as to ensure that $\mathcal{R} = \{1,\ldots,r\}$. This is possible since the indices of nodes are assigned arbitrarily. A change of indices is no more than a permutation of rows and columns in the graph shift operator. Under this definition, the Nyström extension becomes

$$\mathbf{s} = \mathbf{D}^{-\frac{1}{2}}\begin{bmatrix} \mathbf{Z} \\ \mathbf{B}\mathbf{Z}\mathbf{Q}^{-1} \end{bmatrix}\mathbf{x}. \tag{18}$$

One issue with this procedure is that the term $\mathbf{Q}^{-1}$ in (18) is a division by eigenvalues. The eigenvalues of the normalized graph Laplacian are known to be bounded between 0 and 2. Thus, due to the devision by eigenvalues, there may be numerical instabilities. This may cause some of the eigenvectors to have very large values in the $(r+1),\ldots,N$ coordinates. We thus suggest to change the extension as follows:

$$\tilde{\mathbf{Z}} = \begin{bmatrix} \mathbf{Z} \\ \mathbf{B}\mathbf{Z} \end{bmatrix}. \tag{19}$$

We note that we could not omit the division by eigenvalues in (19) if we were attempting to find a rank r approximation of the Markov matrix. However, in our case this omission will cause the graph signal to be approximated by

$$\mathbf{s} = \mathbf{D}^{-\frac{1}{2}}\begin{bmatrix} \mathbf{Z} \\ \mathbf{B}\mathbf{Z} \end{bmatrix}\mathbf{x}. \tag{20}$$

The matrix $\mathbf{D}^{-\frac{1}{2}}$ can be defined as the following block matrix,

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_e & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_b \end{bmatrix}, \tag{21}$$

where $\mathbf{D}_e$ is of size $r\times r$ and $\mathbf{D}_b$ is of size $N-r\times N-r$. It follows that

$$\mathbf{s} = \mathbf{D}^{-\frac{1}{2}}\begin{bmatrix} \mathbf{Z} \\ \mathbf{B}\mathbf{Z} \end{bmatrix}\mathbf{x} = \begin{bmatrix} \mathbf{D}_e^{-\frac{1}{2}}\mathbf{Z}\mathbf{x} \\ \mathbf{D}_b^{-\frac{1}{2}}\mathbf{B}\mathbf{D}_e^{\frac{1}{2}}\mathbf{D}_e^{-\frac{1}{2}}\mathbf{Z}\mathbf{x} \end{bmatrix} = \begin{bmatrix} \mathbf{s}_{\mathcal{R}} \\ \mathbf{D}_b^{-\frac{1}{2}}\mathbf{B}\mathbf{D}_e^{\frac{1}{2}}\mathbf{s}_{\mathcal{R}} \end{bmatrix}, \tag{22}$$

which enforces the smoothness of the graph signal. Strongly connected nodes will be mapped to similar values, while the known entries of the graph signal are adhered to as well.

In addition to solving the problem of numerical instability while enforcing the smoothness of the graph signal, the variation (19) on the Nyström extension also provides a simple closed form expression for the interpolation of smooth graph signals.

V. EXPERIMENTAL RESULTS

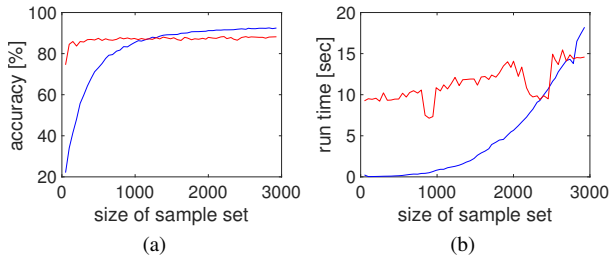

Fig. 1. Graph signal interpolation on the MNIST data set. Results for the Nyström based method are presented in blue. Results of Spectral Regression are presented in red. (a) size of training set (r) vs. rate of accurate graph signal reconstruction. (b) size of training set (r) vs. total time for graph signal reconstruction.

The MNIST data set consists of a training set of 60000 images of handwritten digits, and a test set of 10000 such images. The goal is to determine for each image in the test set the digit it depicts.

This problem can be formulated as a graph signal interpolation problem. We build a weighted graph wherein each image is represented by a single node. As suggested in [4], the edge weights are calculated according to the Euclidean distances between vectorizations of each two images $\mathbf{x}_i$ and $\mathbf{x}_j$. We build a matrix of distances $\mathbf{F}$, where $F_{i,j}$ contains the distance between image $\mathbf{x}_i$ and image $\mathbf{x}_j$. For each row n in $\mathbf{F}$ we keep only the 200 smallest entries, and ignore the rest. The set of indices of entries kept from row n is denoted as $\mathcal{M}_n$. We follow

the practice suggested in [4], and set the similarity between each two images to be

$$W_{i,j} = \begin{cases} \frac{F_{i,j} \cdot 70000^2}{\sum_{n=1}^{N} \sum_{m \in \mathcal{M}_n} F_{n,m}} & \text{for } j \in \mathcal{M}_i \\ 0 & \text{for } j \notin \mathcal{M}_i \end{cases}. \quad (23)$$

We then symmetrize this matrix

$$W_{i,j} = \max\left(W_{i,j}, W_{j,i}\right). \quad (24)$$

Now that the graph is defined, we turn to define the graph signal. Each image in the data set depicts one of ten digits. We define ten graph signals $\mathbf{s^0}, \ldots, \mathbf{s^9}$. Each graph signal contains

$$s_i^k = \begin{cases} 1 & \text{if } v_i \text{ represents an image of the digit } k, \\ 0 & \text{otherwise.} \end{cases} \quad (25)$$

The graph signals $\mathbf{s^0}, \ldots, \mathbf{s^9}$ are known over a set of r randomly chosen (sampled) nodes from the training set. We vary r from 50 to 3000 and use (22) to recover the vectors $\mathbf{s}^0, \ldots, \mathbf{s}^9$ for the test set. If $|s_i^k| > |s_i^m|$ for all $k \neq m$, then we say that node v_i represents an image of digit k.

Figure 1 presents the accuracy and runtime of the graph signal interpolation framework presented in Section IV.

We compare the results of our variation on the Nyström extension to results from the Spectral regression framework [13], [16]. In this framework, the graph signal interpolation is done through an optimization problem. As most of the graph geometry is contained in the leading eigenvectors of the Markov matrix [16], spectral regression searches for a band limited $\mathbf{x}$ that complies with the known entries of the graph signal, *i.e.*

$$\mathbf{x} = \arg\min_{\mathbf{x}^*} \|\mathbf{x}^*\|_0 \quad \text{such that} \quad \mathbf{s}_{\mathcal{R}} = \mathbf{V}_{\mathcal{R}} \mathbf{x}^*, \quad (26)$$

where for vector or matrix $\mathbf{l}$, $\mathbf{l}_{\mathcal{R}}$ denotes the rows of $\mathbf{l}$ whose indices are in the set $\mathcal{R}$. The graph signal is then interpolated via (17).

As the eigenvectors are now calculated exactly, it is prohibitive in both memory consumption and runtime to calculate the eigenvectors of the 70000×70000 Markov matrix. Instead, we sample the Markov matrix, creating a sub-matrix containing the set of r randomly chosen (sampled) nodes from the training set and the 10000 nodes of the test set. In addition, we only compute 20 eigenvectors of this sub-matrix for each value of r. This is done in order to make runtimes comparable. The optimization problem (26) is solved using the SPGL1 package[1] [17], [18].

A comparison of accuracy and runtime of the interpolations is presented in Figure 1. We note that in the Nyström-based interpolation, as the size of the training set (r) grows, the complexity of calculating (12) increases. In the Spectral Regression interpolation framework, as the size of the training set (r) grows so does the size of the graph shift. However, the number of calculated eigenvectors remains unchanged. Thus, when $r = 2600$ runtime for both frameworks is approximately equal. In addition, at this value of r, the results of the Nyström-based interpolation are more accurate due to the limitation imposed on the number of eigenvalues computed in the Spectral Regression framework. In order to achieve comparable accuracy, the number of eigenvectors computed would need to be increased, causing an increase in runtime as well. Our variation on the Nyström extension performs faster while still achieving good accuracy.

[1] https://github.com/mpf/spgl1.

VI. CONCLUSION

The field of signal processing on graphs is a relatively new area of study. Signal processing on graphs strives to generalize definitions and operations from signal processing to data represented by a graph. An important definition in this field is the graph shift operator. In this paper we show that when defining the graph shift operator to be the Markov matrix, the framework of signal processing on graphs is strongly related to the diffusion map framework. We then present a computationally efficient solution to graph signal interpolation problem based on our suggested definition for the graph shift operator.

VII. REFERENCES

[1] 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 Processing Magazine*, vol. 30, no. 3, pp. 83–98, May 2013.

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

[3] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3042–3054, June 2014.

[4] S. Chen, R. Varma, A. Sandryhaila, and J. Kovacevic, "Discrete signal processing on graphs: Sampling theory," *IEEE Transactions on Signal Processing*, vol. 63, no. 24, pp. 6510–6523, December 2015.

[5] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst, "Learning laplacian matrix in smooth graph signal representations," *IEEE Transactions on Signal Processing*, vol. 64, no. 23, pp. 6160–6173, December 2016.

[6] R. R. Coifman and S. Lafon, "Diffusion maps," *Applied and Computational Harmonic Analysiss: Special issue on Diffusion Maps and Wavelets*, vol. 21, pp. 5–30, July 2006.

[7] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral grouping using the Nyström method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 214-225, pp. 1–52, January 2004.

[8] E. J. Nyström, "Über die praktische auflösung von linearen integralgleichungen mit anwendungen auf randwewtaufgaben der potentialtheorie," *Commentationes Physico-Mathematicae*, vol. 4, no. 15, pp. 1–52, 1928.

[9] C. T. H. Baker, *The Numerical Treatment of Integral Equations*, Oxford: Clarendon Press, 1977.

[10] W. H. Press, S.A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipies in C*, Cambridge University Press, second edition, 1992.

[11] L. Deng, "The mnist database of handwritten digit images for machine learning research [best of the web]," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141–142, November 2012.

[12] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, November 1998.

[13] Y. Keller and Y. Gur, "A diffusion approach to network localization," *IEEE Transactions on Signal Processing*, vol. 59, no. 6, pp. 2642 –2654, June 2011.

[14] R. Talmon, I. Cohen, S. Gannot, and R. R. Coifman, "Diffusion maps for signal processing: A deeper look at manifold-learning techniques based on kernels and graphs," *IEEE Signal Processing Magazine*, vol. 30, no. 4, pp. 75–86, July 2013.

[15] A. Sandryhaila and J. M. F. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 80–90, September 2014.

[16] A. Heimowitz and Y.C. Eldar, "A unified view of diffusion maps

and signal processing on graphs," in *International Conference on Sampling Theory and Applications (SampTA)*, 2017.

[17] Ewout van den Berg and Michael P. Friedlander, "Probing the pareto frontier for basis pursuit solutions," *SIAM J. on Scientific Computing*, vol. 31, no. 2, pp. 890 – 912, November 2008.

[18] Ewout van den Berg and Michael P. Friedlander, "Sparse optimization with least-squares constraints," Tech. Rep., Univ of British Columbia, January 2010.