JYVÄSKYLÄ STUDIES IN COMPUTING
183

Guy Wolf

Big High-Dimensional Data Analysis with Diffusion Maps

JYVÄSKYLÄN YLIOPISTO

JYVÄSKYLÄ STUDIES IN COMPUTING 183

Guy Wolf

Big High-Dimensional Data Analysis with Diffusion Maps

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella
julkisesti tarkastettavaksi yliopiston Agora-rakennuksen Alfa-salissa
joulukuun 13. päivänä 2013 kello 12.

Academic dissertation to be publicly discussed, by permission of
the Faculty of Information Technology of the University of Jyväskylä,
in building Agora, Alfa hall, on December 13, 2013 at 12 o'clock noon.

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2013

Big High-Dimensional Data Analysis with Diffusion Maps

JYVÄSKYLÄ STUDIES IN COMPUTING 183

Guy Wolf

Big High-Dimensional Data Analysis with Diffusion Maps

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2013

Editors
Timo Männikkö
Department of Mathematical Information Technology, University of Jyväskylä
Pekka Olsbo, Ville Korkiakangas
Publishing Unit, University Library of Jyväskylä

URN:ISBN:978-951-39-5534-2
ISBN 978-951-39-5534-2 (PDF)

ISBN 978-951-39-5533-5 (nid.)
ISSN 1456-5390

Jyväskylä University Printing House, Jyväskylä 2013

ABSTRACT

Wolf, Guy
Big High-Dimensional Data Analysis with Diffusion Maps
Jyväskylä: University of Jyväskylä, 2013, 30 p.(+included articles)
(Jyväskylä Studies in Computing
ISSN 1456-5390; 183)
ISBN 978-951-39-5533-5 (nid.)
ISBN 978-951-39-5534-2 (PDF)
Finnish summary
Diss.

In order to process big high-dimensional data, this thesis proposes a combination of techniques such as dimensionality reduction, coarse-graining, dictionary constructions, and out-of-sample extensions. The introduced tools and methodologies cope with both the dimensionality and the size of analyzed datasets. The thesis proposes to enhance the Diffusion Maps (DM) dimensionality reduction method from the data point level to a data cluster level. The thesis proposes two approaches for applying DM to data clusters or patches. The first approach considers the DM properties that originate from its Markovian diffusion process. This approach directly coarse-grains this process and prunes local data clusters while ensuring the important stochastic properties of the process are preserved. The second approach utilizes a manifold geometry data model and enhances the diffusion kernel to consider nonscalar affinities between local manifold patches. These affinities combine positional information on the manifold together with relations between manifold tangent spaces in the compared patches. The resulting embedding maps each patch to an embedded tensor. Then, a patch-based dictionary is introduced to retrieve a small representative set of patches that are sufficient for approximating the embedded tensor space. In both cases, the analyzed kernel size is significantly reduced since it is only affected by the various geometric areas (e.g., manifold patches) in which the data is spread instead of the total number of data points.

In addition, this thesis also proposes methods for updating the initial DM embedding as more information is streamed or becomes available. Two types of such information are considered: 1. new data points that should be added to the analysis, and 2. updates and modifications to existing data points that should be updated. For the first update type, this thesis provides a patch-based out-of-sample extension of vector fields. For the second type, this thesis introduces a method to efficiently update kernel-based embeddings without recomputing the spectral decomposition of the entire kernel. This method is especially suitable in cases when the amount of updates is small and can be considered as a perturbation of kernel values.

Keywords: Big Data, data analysis, manifold learning, diffusion maps

Author

Guy Wolf
Department of Mathematical Information Technology
University of Jyväskylä
Finland

Supervisors

Professor Amir Averbuch
School of Computer Science
Tel Aviv University
Israel

Professor Pekka Neittaanmäki
Department of Mathematical Information Technology
University of Jyväskylä
Finland

Reviewers

Doctor Dan Kushnir
CTO Technical Staff
Alcatel Lucent Bell Labs
New Jersey, United States of America

Professor Keijo Ruotsalainen
Faculty of Technology
University of Oulu
Finland

Opponent

Doctor Neta Rabin
Department of Exact Sciences
Afeka Tel-Aviv Academic College of Engineering
Israel

ACKNOWLEDGEMENTS

I would like to thank my supervisors, Prof. Amir Averbuch and Prof. Pekka Neittaanmäki, for all their guidance, their help, and for making this research possible. Amir has guided me through my first steps into academic research since I first started working on my M.Sc. thesis and I always found his advices to be priceless. I can honestly say I could not have hoped for a better research experience than working with him. I am also grateful for the oppostunity of working with Pekka, and I would like to thank him for our fruitful research and for his great hospitality.

I would like to thank all my colleagues with whom I worked on the research in this thesis. In particular, many thanks are reserved to the coauthors of the papers in this thesis, namely, Moshe Salhov, Dr. Amir Bermanis, Aviv Rotbart, Yaniv Shmueli, and Dr. Gil David. I wish to express my great appreciation to Prof. Yoel Shkolinsky and Prof. Raphi Coifman for fruitful discussions that had an important impact on the nature of my research. I am certain that without all these people this research would not have been so enjoyable, enriching and rewarding.

It should be mentioned that the research in this thesis was partly completed during the work on SCOPE (Scientific innovation product concept) and Diagnostics TEKES (Finnish Technology Agency) projects in 2011-12.

Finally, on a personal note, I would like to thank my family for accompanying me through my academic journey, and especially my wife for her endless love and support.

LIST OF FIGURES

CONTENTS

LIST OF INCLUDED ARTICLES

PI Guy Wolf, Aviv Rotbart, Gil David, and Amir Averbuch. Coarse-grained localized diffusion. *Applied and Computational Harmonic Analysis*, 33(3):388–400, 2012.

PII Moshe Salhov, Guy Wolf, Amir Averbuch. Patch-to-Tensor Embedding. *Applied and Computational Harmonic Analysis*, 33(2):182–203, 2012.

PIII Guy Wolf and Amir Averbuch. Linear-Projection Diffusion on Smooth Euclidean Submanifolds. *Applied and Computational Harmonic Analysis*, 34(1):1–14, 2013.

PIV Moshe Salhov, Amit Bermanis, Guy Wolf, Amir Averbuch. Approximate Patch-to-Tensor Embedding via Dictionary Construction. *Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013.

PV Yaniv Shmuelli, Guy Wolf, Amir Averbuch. Updating kernel methods in spectral decomposition by affinity perturbations. *Linear Algebra and its Applications*, 437(6):1356–1365, 2012.

In [PI], the author was the main contributor of the presented mathematical modeling, and led the proving process of the theorems. In [PII], the author was an integral part of developing and analyzing the proposed embedding. Specifically, Sections 4 and 5 were mostly written by the author, including the proofs of the theorems in these sections, as well as the proof of Theorem 3.1 in Section 3. The article [PIII] was mainly written by the author, including the presented analysis, theorems and proofs. In [PIV], the author strongly contributed to the writing process, provided theoretical foundations to the proposed algorithms, and participated in their mathematical analysis. In [PV], the author provided a mathematical modeling of the analyzed problem and theoretical foundations for the proposed approach, and participated in the design process of the presented algorithm.

1 INTRODUCTION

Big high-dimensional datasets have become increasingly common in many areas, due to high availability of data and continuous technological advances. These datasets are characterized by overwhelming amounts of collected data in them. These amounts are affect both the number of observations in such datasets as well as the number of features (measured, collected or computed) that are used to quantify them. The number of such features in a dataset is also called the dimensionality of the dataset, since each observation corresponds to a vector whose coordinates are determined by these features.

When a dataset contains many measured, collected, streamed and calculated features, the data points in it are expressed as vectors in a high dimensional space. Analyzing them directly poses many challenges for machine learning and data analysis methods, which are generally referred to as the "curse of dimensionality"[1]. The main common theme of "curse of dimensionality" problems is the relation between the high dimensionality of a dataset and the volume taken by its data points in the space defined by data features. As the dimensionality of the dataset increases, the data points occupy an increasingly smaller portion of the feature space (i.e., the high dimensional space defined by the features of the dataset). As a result, the high dimensional representation of the data becomes too sparse to directly obtain practical useful information from it.

Recent analysis methods, which originate from the field of machine learning, utilize a locally low-dimensional geometric structure (e.g., a manifold) to model the data. These methods assumes the analyzed phenomena originate from a small set of underlying factors that generate the observable (i.e., measured, collected, streamed, or computed) features in the dataset via nonlinear mappings. This provides the motivation for utilizing dimensionality reduction techniques for data analysis. Instead of directly analyzing the dataset in its feature space, such methods embed the data into a low-dimensional representation in which important information, patterns and structures are revealed. Then, the analysis (e.g., clustering and anomaly detection) can be done on the obtained representation. An example of such methods is the Independent Component Analysis

[1] To the author's best knowledge, this term was coined by Richard E. Bellman in [5].

(ICA) approach, which aims to find the independent components (i.e., the underlying factors) of the data by applying a known model to the nonlinear maps that generate the analyzed data [32, 33].

Kernel methods present a common approach for obtaining meaningful dimensionality reduction of high dimensional data. These methods aim to preserve local similarities between data points. In other words, data points that are similar according to their feature values in the data set are mapped to similar low dimensional points in the embedded space. Conceptually, these methods extend the well known MDS [11, 22] method. They are based on a construction of an affinity kernel that encapsulates the relations (distances, similarities, or correlations) between data points. Spectral analysis of this kernel provides a representation of the data that simplifies its analysis. The nonparametric nature of this analysis uncovers important patterns in the data and reveals their geometry. In practice, the dimensionality of the obtained representation is usually significantly lower than the dimensionality of the input dataset.

The MDS method uses the eigenvectors of a Gram matrix, which contains the inner products between the data points in the analyzed dataset, to define a mapping of these data points into an embedded space that preserves (or approximates) most of these inner products. This method is equivalent to PCA [20, 18], which projects the data onto the span of the principal directions of the variance of the data. Both of these methods capture linear structures in the data. They separate between meaningful directions, which represent the distribution of the data, and noisy uncorrelated directions. The former ones are associated with significant eigenvalues (and eigenvectors) of the Gram matrix, while the latter ones are associated with small eigenvalues.!!!

Kernel methods, such as Isomap [40], LLE [29], Laplacian Eigenmaps [4], Hessian Eigenmaps [14] and Local Tangent Apace Alignment [42, 43], extend the MDS paradigm by considering locally-linear structures in the data. A convenient interpretation of these structures is provided by the assumptions that they form a low-dimensional manifold that captures the dependencies between the observable features in the data. This is called the *manifold assumption*. Namely, the data is assumed to be sampled from this manifold. The resulting spectrally-embedded space in these methods preserves the intrinsic geometry of the manifold, which incorporates and correlates with the underlying factors of the analyzed phenomena in the data.

Kernel methods are also inspired from spectral graph theory [8]. The defined kernel can be interpreted as a weighted adjacency matrix of a graph whose vertices are the data points. The edges of this graph are defined and weighted by the local relations (or similarities) in the kernel matrix. The analysis of the eigenvalues and the corresponding eigenvectors of this matrix reveals many qualities and connections in this graph, which serves as a discretization of the continuous manifold geometry.

The diffusion maps (DM) kernel method [9] utilizes a stochastic diffusion process to analyze data. It defines diffusion affinities via symmetric conjugation of a transition probability operator. These probabilities are based on local dis-

tances between data points. The Euclidean distances in the DM embedded space correspond to a diffusion distance metric in the observable space. This distance metric quantifies the connectivity between data points by incorporating all the diffusion paths between them. When the data is sampled from a low-dimensional manifold, these diffusion paths follow the intrinsic geometry of the manifold (i.e., geodesic paths on it). Therefore, the resulting diffusion distances capture the underlying manifold geometry of the data.

The diffusion distance metric was utilized for clustering & classification [13], parametrization of linear systems [39], and shape recognition [7]. Furthermore, the DM method was used in a wide variety of data analysis and pattern recognition applications. Examples include audio quality improvement by suppressing transient interference [38], moving vehicle detection [31], scene classification [19], gene expression analysis [30] and source localization [37].

In addition, several extensions of DM methodology was recently proposed. For example, in [23, 21], the DM methodology is extended to consider several data sources and fuse the generated datasets into a single embedded representation. In [35, 36, 34], the DM affinities were extended to consider the orientation and local neighborhoods of the underlying manifold, and the resulting embedded space was used for Cryo EM applications.

This thesis extends the DM method to provide a framework for analyzing big high-dimensional data. It deals with practical challenges that are posed by handling big volumes of data that are updated on a daily basis (if not at a higher rate). The thesis proposes theoretical and practical approaches to model and analyze such data. The presented tools enable efficient, sound and practical utilizations of DM in particular and kernel methods in general for analyzing modern datasets.

2 CONTRIBUTION OF THE THESIS

The thesis consists of a collection of papers that address practical challenges of analyzing big high-dimensional data with kernel methods in general and diffusion maps in particular. The relations between these papers are illustrated in Fig. 1. They cover three main approaches for performing data analysis by utilizing DM. The first approach enhances kernel methods and DM to consider and analyze clusters (or patches) of data points instead of analyzing them individually. The second approach defines, explores and utilizes dictionary constructions to reduce the size of the diffusion kernel and enable the utilization of DM for

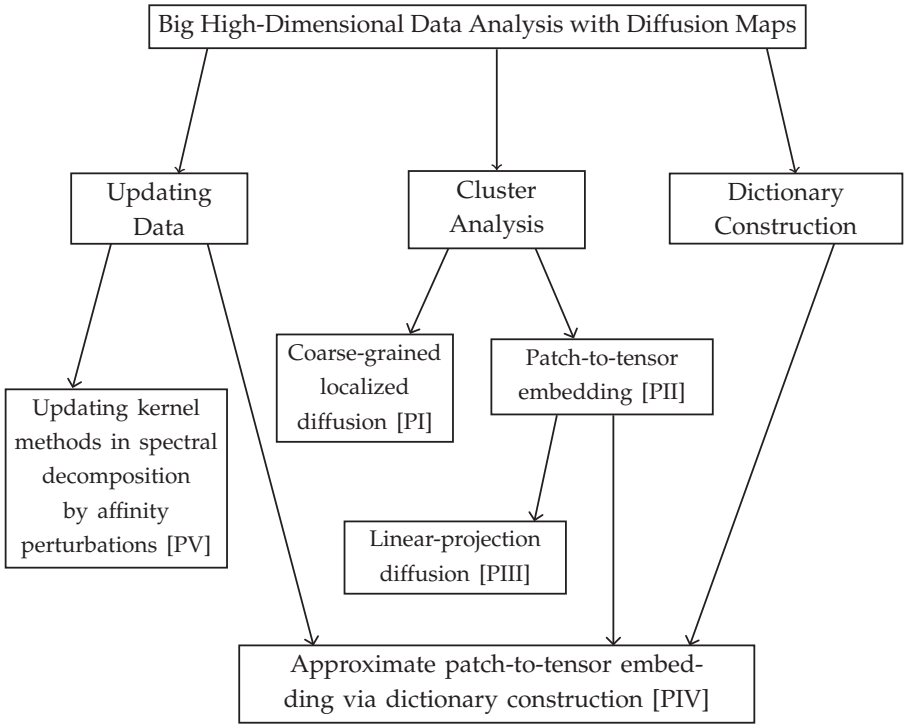


FIGURE 1 Illustration of the structure of the thesis

bigger datasets. The third approach is aimed at coping with updating data by analyzing the effects of perturbations on the DM embedded space and by using out-of-sample extension techniques to efficiently computing the embedding of new data points. Detailed descriptions of these approaches are presented in sections 2.1, 2.2 and 2.3.

2.1 Cluster and patch analysis

Data-analysis methods nowadays are expected to deal with increasingly large amounts of data. Such massive datasets often contain many redundancies. One effect from these redundancies is the high-dimensionality of datasets, which is handled by dimensionality reduction techniques such as DM. Another effect is the duplicity of very similar data-points that can be analyzed together as a cluster. To cope with this effect, thesis presents two novel approaches in [PII, PI] for analyzing data clusters (or local patches) rather than individual data points. These approaches allow the DM dimensionality reduction framework to move from the data point level to the cluster level. This way, the size of the analyzed dataset is decreased by only referring to data clusters or patches. Then, the dimensionality of the dataset can be decreased by the DM embedding.

In order to provide motivation and justification for the presented approaches, two main questions should be addressed: 1. Why is patch processing, which is also called vector processing, the right way to go when we want to manipulate high-dimensional data? 2. Do these patches exist in real-life datasets? Brief answers to both questions are provided here as well as in [PII].

Data analysis methods often assume that the processed data have been generated by some physical phenomenon, which is governed by an underlying potential [26, 27]. Therefore, the affinity kernel will reveal clustered areas that correspond to neighborhoods of the local minima of this potential. In other words, these high-dimensional data points reside on several patches located on the low dimensional underlying manifold. On the other hand, if the data is spread sparsely over the manifold in the high-dimensional ambient space, then the application of an affinity kernel to the data will not reveal any patches/clusters. In this case, the data is too sparse to represent or detect the underlying manifold structure, and the only available processing tools are variations of nearest-neighbor algorithms. Therefore, data points on a low-dimensional manifold in a high-dimensional ambient space can either reside in locally-defined patches, and then the methods in this thesis are applicable to it, or scattered sparsely all over the manifold and thus there is no detectable coherent physical phenomenon that can provide an underlying structure for it. Since the algorithms in this thesis are based on a manifold learning approach, it is inapplicable in the latter case.

In general, all the tools that extract intelligence from high-dimensional data assume that under some affinity kernel there are data points that reside on locally-related patches, otherwise no intelligence (or correlations) will be extracted from

the data and it can be classified as noise of uncorrelated data points. Therefore, the local patches, and not the individual points, are the basic building blocks for correlations and underlying structures in the dataset, and their analysis can provide a more natural representation of meaningful insights to the patterns that govern the analyzed phenomenon.

The DM methodology and the proposed methodologies in the thesis are classified as spectral methods. Spectral methods are global in the sense that they usually require the relations between all the samples in the dataset. This global consideration hinders their use in practical large-scale problems due to high memory (e.g., fitting the kernel matrix in memory) and computational costs. However, in many cases there are many duplicities, or near duplicities, in massive datasets and a the number of different clusters (or patches) of closely-related data-points is significantly less then the number of samples in the dataset. Processing data clusters and patches, instead of individual data points, reduces the many redundancies that usually occur in big datasets, thus, it enables also to localize spectral processing and reduce these overheads and impracticalities.

We consider two approaches to achieve a DM analysis of clusters or patches. The first approach in [PI] is based on the structure of the stochastic Markovian diffusion process in DM. This approach prunes data clusters and coarse grains the DM transition probabilities between them while preserving the main DM stochastic properties. The second approach in [PII] is based on the local neighborhood structure of a manifold model for the data geometry. Under this model, the data points are assumed to be sampled from a low-dimensional manifold and local data patches are approximated by tangent spaces of the manifold. The scalar DM affinities are extended in this approach to matrix affinities that encompass multidimensional similarities between local neighborhoods of data points on the manifold. Overviews of these approaches are presented in Sections 2.1.1 and 2.1.2 with further discussion on each of them.

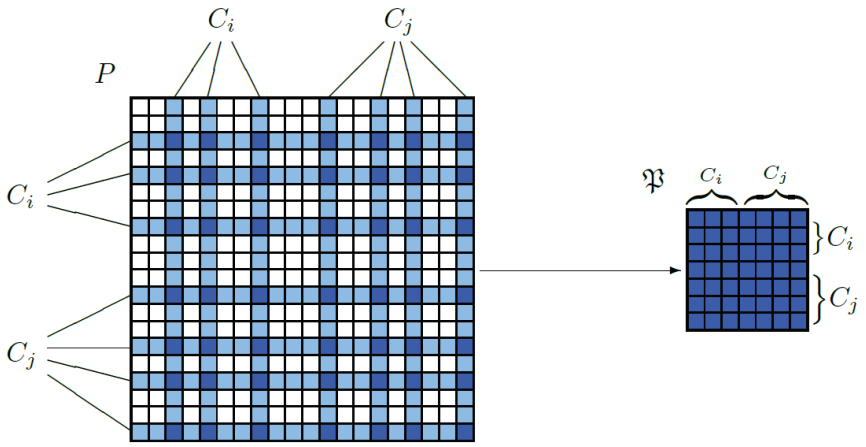
2.1.1 Coarse grained diffusion maps

In [PI], a coarse-graining approach is proposed for dealing with both the high-dimensionality of the data and the big size of the modern dataset. The presented method prunes data clusters to be considered as the analyzed elements of the DM dimensionality reduction framework. This way, the DM kernel size is determined by the number of pruned clusters, which is usually significantly smaller than the number of data points in the analyzed dataset. Then, the coarse-grained DM can be applied in order to compute a low-dimensional embedding of these data clusters. We show that the essential properties (e.g., ergodicity) of the underlying diffusion process of DM are preserved by the coarse-graining. The affinity that is generated by the coarse-grained process, which we call *Localized Diffusion Process* (LDP), is strongly related to the recently introduced *Localized Diffusion Folders* (LDF) [13] hierarchical clustering algorithm. We show that the LDP coarse-graining is in fact equivalent to the affinity-pruning that is achieved at each folder-level in the LDF hierarchy.

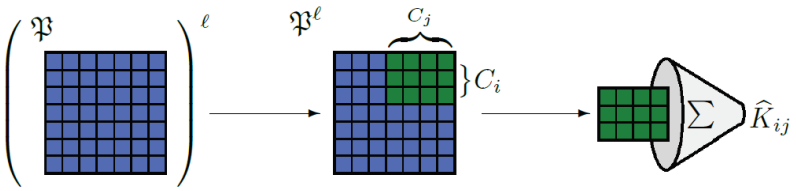
The LDF method performs an iterative process that obtains a folder hierarchy that represents the points in the dataset. Each level in the hierarchy is constructed by pruning clusters of folders (or data points) from the previous level. The iterative process has two main phases in each iteration:

1. **Clustering phase:** the "shake & bake" method is used to cluster the folders (or data-points) of the current level in the hierarchy by using a diffusion affinity matrix.
2. **Pruning phase:** the clusters of the current level are pruned and given as folders of the next level in the hierarchy. The diffusion affinity is also pruned to represent affinities between pruned clusters (i.e., folders of the next level in the hierarchy) instead of folders in the current hierarchial level.

In [PI], we focus on exploring the pruning that is performed in the second phase of this process, while considering the clustering of the data, which may be per-



(a) Let P be the DM transition probability matrix between all the data points in a dataset M. For two disjoint clusters $C_i, C_j \subseteq M$, $C_i \cap C_j = \emptyset$, consider only the transition probabilities between data points in $C_i \cup C_j$ and store them in the matrix $\mathfrak{P}$.



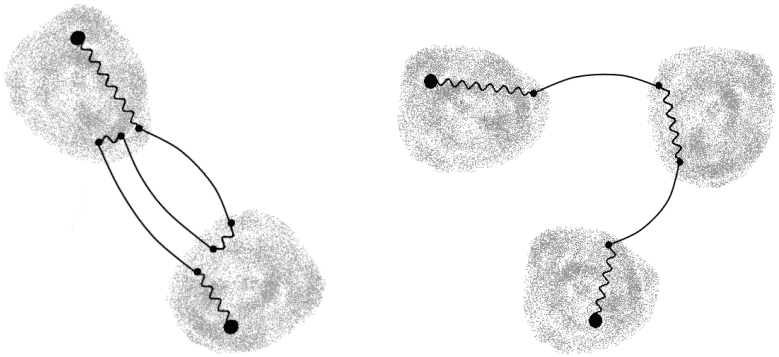
(b) Consider the top right submatrix of $\mathfrak{P}$, which only considers rows that correspond to data points in C_i and columns that correspond to data points in C_j. Let $\hat{K}_{ij}$ be a wighted sum of all the probabilities in this submatrix. The transition probabilities between pruned clusters are obtained by normalizing the resulting kernel $\hat{K}$, which contains the cells $\hat{K}_{ij}$ between all data clusters, to be row stochastic.

FIGURE 2 Illustration of the diffusion transition probability pruning in [PI], which is equivalent to the LDF affinity pruning phase in [13]

formed by "shake & bake" process [13] or by another clustering algorithm, as prior knowledge. Figure 2 illustrates the introduced transition probability pruning between clusters in [PI], which is similar to the affinity pruning in LDF. In fact, the resulting pruned probabilities and affinities in these two methods are shown in [PI] to be equivalent.

Essentially, the LDF algorithm provides an hierarchical data clustering with additional affinity information for each level in the hierarchy. While there are many empirical justifications for the merits of LDF and its utilization in various fields (e.g., unsupervised learning and image processing), it lacked theoretical justifications. The introduced coarse-grained localized diffusion process in [PI] preserves essential properties of the original DM process that enable its utilization for dimensionality reduction tasks. This process and the diffusion affinity generated by it are related in the paper to the one achieved by the LDF pruning phase. This relation adds the needed complimentary foundations for the LDF framework by providing theoretical justifications for its already-obtained empirical support. Additionally, the presented relation shows that the applications presented in [13] in fact demonstrate the utilization of the LDP for data-analysis tasks and the results presented there provide empirical support of its benefits.

A similar coarse-graining approach was also presented in [24]. The approach there is based on a graph representation of the diffusion random-walk process. The clustering of data-points was performed by graph partitioning. Then, transition probabilities between partitions were achieved by averaging transition probabilities between their vertices. The resulting random-walk process maintains most of the spectral properties of the original diffusion process and its eigendecomposition can be approximated by the original spectral decomposition. However, the approximation error strongly depends on the exact partitioning used. In addition, since all the random-walk paths are considered in the



(a) (b)

FIGURE 3 Illustration of the difference between (a) a localized path, which only traverses through data points in its source and destination clusters, and (b) a nonlocalized path, which traverses through one or more intermediary clusters

averaging process, there is a limited number of viable time-scales (in the diffusion process) that can be used by this process before it converges to the averaging of the stationary distribution.

The presented coarse-graining of the diffusion process in [PI] copes with the rapid convergence toward the stationary distribution by only preserving localized paths between clusters while ignoring paths that are "global" from the cluster point-of-view. The difference between localized and nonlocalized paths is illustrated in Fig. 3 While it is desirable that the clusters will be sufficiently coherent to consist of a continuous partitioning of the dataset and its underlying manifold, the properties of the presented coarse-graining process are neither depend on such assumptions nor on the exact clustering method used.

2.1.2 Patch-based diffusion maps

In [PII], we extend the original Diffusion Maps method in particular and kernel methods in general by suggesting the concept of a super-kernel. We aim to analyze patches of an underlying data manifold instead of analyzing single data points on this manifold, which represents the geometrical structure of the analyzed data. Each patch is defined as a local neighborhood of a point in a dataset sampled from the underlying manifold. The relation between two patches is described by a matrix rather than by a scalar value. This matrix represents both the affinity between the points at the centers of these patches and the similarity between their local coordinate systems. The constructed matrices between all patches are then combined in a block matrix, which we call a super-kernel. The structure of the constructed super-kernel is illustrated in Fig. 4. The presented super-kernels provide an extension of scalar-affinity kernels that are used in kernel methods.

We suggest several methods for constructing super-kernels. In particular,

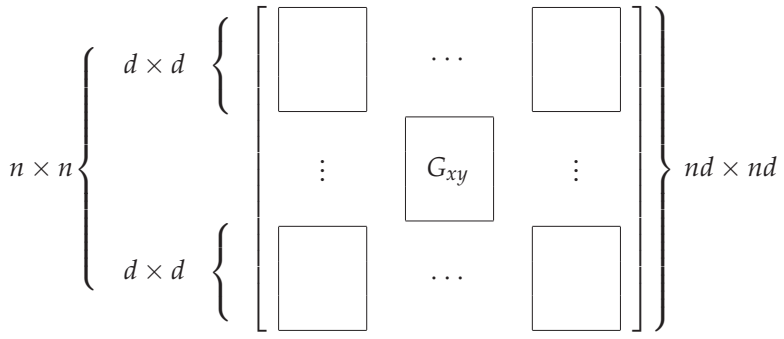
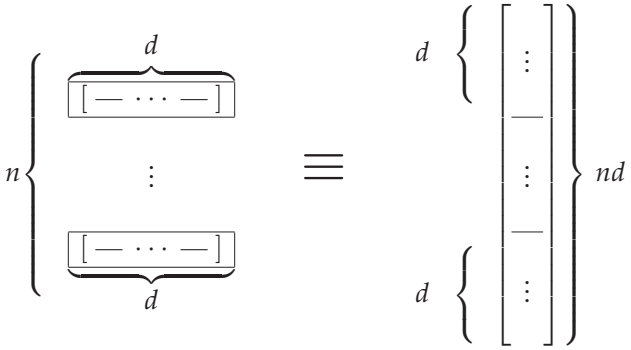


FIGURE 4 The structure of a super-kernel G over a dataset M with n data points that are sampled from a d-dimensional manifold immersed in a high dimensional space. The super-kernel G is a block matrix, where each block G_{xy}, $x, y \in M$, is a $d \times d$ matrix that represents a non-scalar affinity between the manifold patches around x and around y.

linear-projection operators between tangent spaces of data points are suggested for expressing the similarities between the local coordinate systems of their patches. Other constructions such as ones based on orthogonal transformations can also be used. Such constructions will be explored in future works. We also suggest using the original diffusion kernel for expressing the affinities between points on the manifold. We examine and determine the bounds for the spectra (i.e., the eigenvalues) of the suggested constructions. Then, the eigenvalues and the eigenvectors of the constructed super-kernels are used to embed the patches of the manifold into a tensor space. This embedding is illustrated in Fig. 5. We relate the Frobenius distance metric between the coordinate matrices of the embedded tensors to a new distance metric between the patches in the original space. We show that this metric can be regarded as an extension of the diffusion distance metric, which is related to the original Diffusion Maps method [9].



(a) Each eigenvector ϕ of the $nd \times nd$ super-kernel G is a vector of length nd. Equivalently, it contains n subvectors of length d that define a map $\vec{\phi}(x)$, $x \in M$, of patches around data points in M.

$$\mathcal{T}_x = \left(\begin{array}{c} \boxed{\lambda_1^t} \cdot \boxed{[\, — \vec{\phi}_1(x) — \,]} \\ \vdots \\ \boxed{\lambda_\delta^t} \cdot \boxed{[\, — \vec{\phi}_\delta(x) — \,]} \end{array} \right) \Bigg\} \delta \times d$$

(b) By using δ eigenvectors of the super-kernel, each patch around a data point x is mapped to δ subvectors $\vec{\phi}_1(x), \ldots, \vec{phi}_\delta(x)$ of length d. These subvectors are then organized in a coordinate matrix of a tensor $\mathcal{T}_x \in \mathbb{R}^\delta \otimes \mathbb{R}^d$ that provides the embedding of the patch around $x \in M$.

FIGURE 5 Illustration of the achieved patch-to-tensor embedding based on the eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_\delta$ and the eigenvectors $\phi_1, \phi_2, \ldots, \phi_\delta$ of the super-kernel G from Fig. 4. The exact value of δ is determined by the decay of the spectrum of G.

The linear-projection diffusion (LPD) super-kernel that is introduced in [PII] is further explored in [PIII]. The main focus in [PIII] is on the theoretical properties of the LPD super-kernel that was presented in [PII]. This super-kernel is a specific type of linear-projection super-kernels, whose spectra (i.e., eigenvalues)

were shown to be non-negative. In the case of the LPD super-kernel, all the eigenvalues are between zero and one. The LPD-based embedding in [PII] results in a tensor space in which Frobenius distances (between coordinate matrices of embedded tensors) extend of the original diffusion distance from [9]. This extension incorporates information about the proximities of tangential points in the diffusion process together with the projections between the corresponding tangent spaces that represent the patches [PII].

The results obtained in [PII] for finite constructions of LPD super-kernels are extended in [PIII] by exploring their properties when they becomes continuous. This paper enhances the properties of LPD super-kernels two-folds: 1. It shows that the infinitesimal generator of the LPD super-kernel converges to a natural extension of the original diffusion operator from scalar functions to vector fields. This operator was shown to be locally equivalent to a composition of linear projections between tangent spaces and the vector-Laplacians on them. 2. It introduces the stochastic process defined by the LPD super-kernels and demonstrated it on a synthetic manifold.

The presented PTE method in [PII, PIII] generalizes the DM framework and its diffusion distance metric by incorporating matrix similarity relations into a single super-kernel. However, the use of these multidimensional similarities results in a bigger kernel matrix, which significantly increases the computational complexity of PTE due to its reliance on a spectral decomposition of the kernel. In [PIV], we present an efficient approximation for this spectral decomposition that enables us to utilize an enhanced diffusion distance for the analysis of large datasets. This enhancement is discussed in Section [PIV]

Among other benefits, the patch-processing approach introduced here, together with suitable dictionary constructions such as the one in [PIV], enable the reduction of wide redundancies in many large-scale datasets. It provides a meaningful representation of the essential intelligence from the analyzed data without any superfluous information that does not benefit the sought-after patterns and can thus be regarded as noise from the analysis point of view. The presented LPD super-kernel and related vector propagating diffusion process can also be utilized for out-of-sample extensions of vector fields. This utilization is explored in [PIV] and is further discussed in Section 2.3.

2.2 Patch-based dictionary construction

The DM method uses a Markovian diffusion process to model and analyze data. A spectral analysis of the DM kernel yields a map of the data into a low dimensional space, where Euclidean distances between the mapped data points represent the diffusion distances between the corresponding high dimensional data points. Many machine learning methods, which are based on the Euclidean metric, can be applied to the mapped data points in order to take advantage of the diffusion relations between them. However, a significant drawback of the DM in

particular, and kernel methods in general, is the need to apply spectral decomposition to a kernel matrix, which becomes infeasible for large datasets.

For a sufficiently small dataset, kernel methods can be implemented and executed on relatively standard computing devices. However, even for moderate size datasets, the necessary computational requirements to process them are unreasonable and, in many cases, impractical. For example, a segmentation of a medium size image with 512×512 pixels requires a $2^{18} \times 2^{18}$ kernel matrix. The size of such a matrix necessitated about 270 gigabytes of memory assuming double precision. Furthermore, the spectral decomposition procedure applied to such a matrix will be a formidable slow task. Hence, there is a growing need to have more computationally efficient methods that are practical for processing large datasets. This need becomes even more crucial when using non-scalar affinities between data patches (or clusters) as suggested in [PII] and discussed in Section 2.1.

Sparsification by a sparse eigensolver such as Lanczos, which computes the relevant eigenvectors [12] of the kernel matrix, is widely used to reduce the computational load involved in processing a kernel matrix. Another sparsification approach is to transform the dense kernel matrix into a sparse matrix by selectively truncating elements outside a given neighborhood radius of each dataset member. Other approaches to achieve matrix sparsification are described in [41]. Given a dataset with n data points, common approaches for processing kernel methods, including the ones described in this thesis, require at least $O(n^2)$ operations to determine which entries to either calculate or to threshold. While there are methods to alleviate these computational complexities [1], kernel sparsification might result in a significant loss of intrinsic geometric information such as distances and similarities.

The main computational load associated with kernel methods is generated by the application of a spectral decomposition to a kernel matrix. Considerable efforts have been invested (e.g., in [15, 2] and others) in approximating the spectral decomposition operator to become a feasible computation. For example, the dictionary approach in [15] constructs a dictionary of representatives that are sufficient for approximating the full kernel. This method outputs the smaller dictionary-based kernel and the corresponding extension coefficients to the full dataset (or kernel). The number of dictionary members depends on the given data, kernel configuration and the desired quality of the full kernel approximation.

Another prominent approach to reduce the discussed computational load is based on the Nyström extension method [16], which estimates the eigenvectors needed for an embedding. This approach is based on three phases:

1. The dataset is subsampled uniformly over the set of indices that are randomly chosen without repetition.
2. The subsamples define a smaller (than the dataset size) kernel. SVD is applied to the small kernel.
3. Spectral decomposition of a small kernel is extended by the application of the Nyström extension method to the entire dataset.

This three-phase approach reduces the computational load, but the approximated spectral decomposition output suffers from several major problems. Subsampling affects the quality of the spectral approximation. In addition, the Nyström extension method exhibits ill-conditioned behavior that also affects the spectral approximation [6]. Uniform subsampling of a sufficient number of data points captures most of the data probability distribution. However, rare events, compared to the subsampled size, might get lost. The results from this loss of information degrades the quality of the estimated embedded distances.

In [PIV], the dictionary construction approach in [15] is utilized to approximate the spectral decomposition of a non-scalar affinity kernel under the settings of [PII, PIII], which are described in Section 2.1. In this case the need for efficient dictionary constructions becomes even more important since a dataset with n data points from a d-dimensional manifold produces a $nd \times nd$ super-kernel matrix (see Fig. 4). The presented dictionary construction in [PIV] utilizes the underlying patch structure of data that originate from a low-dimensional manifold in a high-dimensional ambient space. This paper describes the necessary condition for updating a non-scalar dictionary for achieving a bound on the approximation error.

Although the proposed method is applicable to many such kernels, we focus on the linear-projection super-kernel construction described in [PII]. The extension of the dictionary construction from [15] is done by an efficient algorithm that assumes the data is sampled from an underlying manifold and utilizes the non-scalar relations and the similarities between manifold patches instead of utilizing scalar relations between individual data points. The constructed dictionary contains a small set of representative manifold patches, which are represented by the embedded tensors from [PII]. This representative set is shown to be sufficient, by construction, for representing the entire structure of the super-kernel and the non-scalar affinities in it. Therefore, the achieved patch dictionary encompasses multidimensional similarities between local areas of the data. The presented dictionary-based analysis reduces the computational costs of the spectral analysis in comparison to the straight-forward embedding method in [PII]. Hence, it enables the patch-based embedding to be applied to datasets that are impractical to process and embed without using the dictionary construction.

2.3 Dealing with updating data

Many machine learning algorithms contain a training step that is done once. The training step is usually computationally expensive since it involves processing big matrices (e.g., spectral decomposition of a big kernel matrix). If the analyzed data originates from an evolving dynamic system (or phenomena), it has to be updated as the underlying system changes over time. There are two main types of updates that can be considered in such systems: 1. Some features of the training dataset are changed without adding new samples. 2. New samples are col-

lected, measured or streamed over time and the results of the analysis should be extended to them. In this section we discuss methods for handling both cases.

The first challenge is addressed by [PV], which presents an algorithm for updating kernel-based embeddings under sufficiently small perturbations of the data affinities. The second challenge is well known as out-of-sample extension and there are several methods of coping with it using scalar-affinity kernels. In [PIV] we present such an extension scheme for vector-fields (rather than scalar functions) that arise in modern applications. Specifically, the presented method enables the out-of-sample extension of patch-based embeddings such as the PTE from [PII]. More detailed discussions of these methods are presented in Section 2.3.1 and 2.3.2.

2.3.1 Updating kernel methods by affinity perturbations

Studying a dataset while being able to extract constructive information from it is a challenging task. The computational complexity increases when processing evolving data that requires frequent updates of the profile that represents the training set we use. As time advances, the training profile, which was previously extracted from evolving dynamic data, may not represent accurately the behavior of the current data. Therefore, this scenario requires not only the extension of a known stable profile to new samples (as done in out-of-sample techniques), but rather the updating of the extracted training profile. A straightforward approach to update this profile is to repeat the entire computational process that previously generated it. However, this approach becomes computationally impractical when dealing with big data while the changes in the training profile are relatively small.

In [PV], the problem of efficiently updating the training profile of analyzed data is explored under the setting of constantly evolving data. The data is modeled by a kernel matrix and processed by spectral decomposition. In many algorithms for clustering and classification, a low dimensional representation of the affinity kernel graph of the embedded training dataset is computed. Then, it is used to classify newly arrived data points. This paper proposes methods for updating such kernel-based embeddings of the training dataset in an incremental way without the need to perform the entire computation upon changes in a small number of the training samples. An efficient computation of this algorithm is critical in many web-based applications.

The presented methods in [PV] efficiently update the training profile while performing a limited computation that only takes into consideration the modified features instead of considering all the features in the dataset. It is based on extending the Power Iteration algorithm from [25]. This algorithm has been proved to be effective when calculating the principle eigenvector of a matrix. However, this method was not suitable for find the other eigenvectors of the matrix. The presented algorithm in [PV], which we call Recursive Power Iteration (RPI), uses an iterative approach that uncovers one eigenvector at a time. In each step, the power iteration is performed on a modified kernel matrix whose principal eigenvector corresponds to the next eigenvector to uncover from the full kernel matrix.

In general, an initial guess of the eigenvectors is important to guarantee fast convergence of the algorithm. The presented RPI algorithm uses the original eigenvectors of the unperturbed kernel as the initial guess for each power iteration. An additional optimization is achieved by using first order approximation of the perturbed eigenvectors. The justification for this approach is that the first order approximation of the perturbed eigenvector is inexpensive, and each RPI step will guarantee that this approximation converges to the actual eigenvector of the perturbed kernel. The first order approximation should be close to the required solution and therefore requires fewer iterations steps to converge. The correctness of the presented algorithm is proved in [PV] and its performances are demonstrated on real data.

2.3.2 Out-of-sample extensions of vector fields

The challenge of extending achieved data analysis results (e.g., the DM embedding) to newly arrived data points is addressed by out-of-sample extension methods. Several kernel approaches have been applied for this tasks. A classical kernel-based technique is the Nyström extension method [2, 28]. This method is based on inverting a kernel matrix that is assumed to be derived from a uniform sampling of the data. More recent methods are Geometric Harmonics [10] and the Multiscale Extension (MSE) scheme in [6]. These methods use the spectral decomposition of the kernel (i.e., its eigenvalues and eigenvectors) as a basis of its range. The eigenfunctions are shown to be easily extended to new data points, thus any function in its range, which can be expressed as a linear combination of these eigenfunctions, is also easily extended. Functions that are not in the range of the kernel are extended by projecting them on the kernel's range and using the resulting function (and extension) as an approximation of the original function.

The MSE scheme [6] in particular was suggested as an alternative to the Nyström extension. This scheme, which samples scattered data and extends functions defined on sampled data points, overcomes some of the limitations of the Nyström method due to ill-conditioned matrix inversions that are involved in its computation. The MSE method is based on mutual distances between data points. It uses a coarse-to-fine hierarchy of a multiscale decomposition of a Gaussian kernel to overcome ill-conditioned phenomena and to speed the computations.

The main focus of [PIV] is the dictionary construction that enables the application of patch-based DM analysis to big data. However, the presented construction also provides a natural diffusion-based out-of-sample extension of vector fields. This type of extension is beneficial when the analyzed data consists of directional information in addition to positional information on the manifold. For example, the goal in [3] is to recover missing data in images utilizing interpolation of the appropriate vector field. Another example is the utilization of tangential vector fields interpolation on $\mathcal{S}^2$ for modeling atmospheric air flow and oceanic water velocity [17].

The dictionary construction in [PIV] is aimed to approximate the spectral

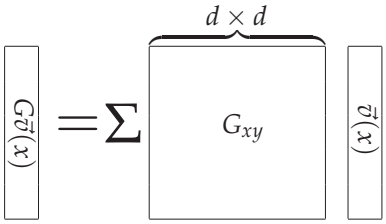


FIGURE 6 Illustration of the super-kernel as an operator on vector-fields. Let $\mathcal{M}$ be the d-dimensional underlying data manifold and let $T_x(\mathcal{M}) \equiv \mathbb{R}^d$, $x \in \mathcal{M}$, be the d-dimensional tangent spaces of the manifold at every data point. Then, the constructed super-kernel provides an operator on tangent vector fields $\vec{v} : \mathcal{M} \to \mathbb{R}^d$ such that $\vec{v}(x) \in T_x(\mathcal{M}) \overset{G}{\mapsto} \sum G_{xy}\vec{v}(y) = G\vec{v}(x) \in T_x(\mathcal{M})$

decomposition a super-kernel that consists of non-scalar affinities between manifold patches. This approximation is achieved by computing the super-kernel decomposition on a sufficiently small set of representatives (i.e., the dictionary) and extend the results of this computation to the entire dataset. The presented extension method from the dictionary set can also be utilized to extend the super-kernel spectral decomposition (and the resulting embedding) to new manifold areas (or patches), either from the sampled dataset or even from the (smaller) dictionary. According to [PIII], the LPD super-kernel provides a diffusion operator for tangent vector fields of the underlying data manifold. This interpretation of the super-kernel is illustrated in Fig. 6. Thus, its spectral decomposition consists of eigenvector fields that span the range of the LPD super-kernel. Therefore, the achieved extension method in [PIV] for super-kernel eigenvector fields is equivalent to the out-of-sample extension of tangent vector fields of the underlying data manifold.

YHTEENVETO (FINNISH SUMMARY)

Tämä opinnäyte esittelee joukon menetelmiä suurten, korkeaulotteisten datamassojen käsittelyä varten. Esiteltävät menetelmät ovat ulotteisuuden vähentäminen, karkeajakoistus, sanakirjarakennelmat ja aineiston ulkopuoliset laajennukset. Esitetyt metodit pystyvät käsittelemään kooltaan ja ulottuvuuksiltaan vaativia aineistoja. Teoksessa esitetään kaksi tapaa laajentaa diffuusiokuvauksiin perustuva ulotteisuuden vähentäminen yksittäisten datapisteiden tasolta dataklustereihin. Ensimmäinen tapa tarkastelee menetelmän käyttämän Markov-diffuusioprosessin ominaisuuksia. Se karkeajakoistaa suoraan tätä prosessia ja karsii paikallisia dataklustereita. Samalla myös prosessin stokastiset ominaisuudet säilyvät. Toinen lähestymistapa hyödyntää monistogeometrista datamallia. Se laajentaa diffuusioytimen huomioimaan paikallisten data-alueiden epäskalaarit samankaltaisuudet. Nämä samankaltaisuudet yhdistävät paikkatiedon ja vertailtujen data-alueiden moniston tangenttiavaruuksien väliset suhteet. Tuloksena saatava upotuskuvaus tuottaa jokaisesta alueesta upotetun tensorin. Seuraavaksi muodostetaan alueesta sanakirja, joka mahdollistaa upotetun tensoriavaruuden likimääräisesti esittävän, pienen aluejoukon valinnan. Molemmissa tavoissa tarkasteltavan diffuusioytimen koko pienenee merkittävästi, koska ydin on riippuvainen vain datapisteiden muodostamista geometrisista alueista datapisteiden kokonaismäärän sijaan.

Lisäksi esitetään menetelmiä alkuperäisen diffuusioupotuksen päivittämiseksi, kun lisää dataa tulee saataville. Uusien, analyysiin lisättävien datapisteiden käsittelyyn esitellään aluepohjainen aineiston ulkopuolisten vektorikenttien laajennus. Päivitettyjä datapisteitä varten esitetään tehokas menetelmä ydinpohjaisten upotusten päivittämiseen laskematta uudelleen koko ytimen spektraalihajotelmaa. Tämä menetelmä soveltuu erityisesti määrältään vähäisten päivitysten käsittelyyn, ja on tarkasteltavissa ytimen arvojen vaihteluna.

REFERENCES

[1] D. Achlioptas and F. McSherry. Fast computation of low rank matrix approximations. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, STOC '01, pages 611–618. ACM, 2001.

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

[3] C. Ballester, M. Bertalmio, G. Sapiro, and J. Verdera. Filling-in by joint interpolation of vector fields and gray levels. *IEEE Transactions on Image Processing*, 10:1200–1211, 2001.

[4] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003.

[5] R.E. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, NJ, 1957.

[6] A. Bermanis, A. Averbuch, and R.R. Coifman. Multiscale data sampling and function extension. *Applied and Computational Harmonic Analysis*, 34(1):15 – 29, 2013.

[7] M.M. Bronstein and A.M. Bronstein. Shape recognition with spectral distances. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):1065–1071, 2011.

[8] F. Chung. *Spectral Graph Theory*, volume 92. CBMS-AMS, May 1997.

[9] R.R. Coifman and S. Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006.

[10] R.R. Coifman and S. Lafon. Geometric harmonics: A novel tool for multiscale out-of-sample extension of empirical functions. *Applied and Computational Harmonic Analysis*, 21(1):31–52, 2006.

[11] T. Cox and M. Cox. *Multidimensional Scaling*. Chapman and Hall, London, UK, 1994.

[12] J.K. Cullum and R.A. Willoughby. *Lanczos Algorithms for Large Symmetric Eigenvalue Computations I: Theory*, volume 41 of *Classics in Applied Mathematics*. SIAM, 2002.

[13] G. David and A. Averbuch. Hierarchical data organization, clustering and denoising via localized diffusion folders. *Applied and Computational Harmonic Analysis*, 33(1):1–23, 2012.

[14] D.L. Donoho and C. Grimes. Hessian eigenmaps: New locally linear embedding techniques for high dimensional data. *Proceedings of the National Academy of Sciences of the United States of America*, 100:5591–5596, May 2003.

[15] Y. Engel, S. Mannor, and R. Meir. The kernel recursive least-squares algorithm. *Signal Processing, IEEE Transactions on*, 52(8):2275–2285, 2004.

[16] C. Fowlkes, S. Belongie, F. Chung, and J. Malik. Spectral grouping using the nyström method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):214–225, 2004.

[17] E. J. Fuselier and G.B. Wright. Stability and error estimates for vector field interpolation and decomposition on the sphere with rbfs. *SIAM Journal on Numerical Analysis*, 47(5):3213–3239, 2009.

[18] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24, 1933.

[19] L. Jingen, Y. Yang, and M. Shah. Learning semantic visual vocabularies using diffusion distance. In *CVPR 2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 461–468, 2009.

[20] I.T. Jolliffe. *Principal Component Analysis*. Springer, New York, NY, 1986.

[21] Y. Keller, R.R. Coifman, S. Lafon, and S.W. Zucker. Audio-visual group recognition using diffusion maps. *IEEE Transactions on Signal Processing*, 58(1):403–413, 2010.

[22] J.B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29:1–27, 1964.

[23] S. Lafon, Y. Keller, and R.R. Coifman. Data fusion and multicue data matching by diffusion maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1784–1797, 2006.

[24] S. Lafon and A.B. Lee. Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1393–1403, 2006.

[25] A.N. Langville and C.D. Meyer. Updating markov chains with an eye on google's pagerank. *SIAM Journal on Matrix Analysis and Applications*, 27(4):968–987, 2006.

[26] B. Nadler, S. Lafon, R.R. Coifman, and I.G. Kevrekidis. Diffusion maps, spectral clustering and eigenfunctions of fokker-planck operators. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 955–962. MIT Press, Cambridge, MA, 2006.

[27] B. Nadler, S. Lafon, R.R. Coifman, and I.G. Kevrekidis. Diffusion maps, spectral clustering and reaction coordinates of dynamical systems. *Applied and Computational Harmonic Analysis*, 21(1):113–127, 2006.

[28] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 2nd edition, 1992.

[29] S.T. Roweis and L.K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, December 2000.

[30] X. Rui, S. Damelin, and D.C. Wunsch. Applications of diffusion maps in gene expression data-based cancer diagnosis analysis. In *EMBS 2007: the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 4613–4616, 2007.

[31] A. Schclar, A. Averbuch, N. Rabin, V. Zheludev, and K. Hochman. A diffusion framework for detection of moving vehicles. *Digital Signal Processing*, 20(1):111–122, 2010.

[32] A. Singer. Spectral independent component analysis. *Applied and Computational Harmonic Analysis*, 21(1):135–144, 2006.

[33] A. Singer and R.R. Coifman. Non-linear independent component analysis with diffusion maps. *Applied and Computational Harmonic Analysis*, 25(2):226–239, 2008.

[34] A. Singer, Y. Shkolnisky, R. Hadani, and Z. Zhao. Viewing angle classification of cryo-electron microscopy images using eigenvectors. *SIAM Journal on Imaging Sciences, accepted for publication*, 2011.

[35] A. Singer and H.-t. Wu. Orientability and diffusion maps. *Applied and Computational Harmonic Analysis*, 31(1):44–58, 2011.

[36] A. Singer and H.-t. Wu. Vector diffusion maps and the connection laplacian. *Communications on Pure and Applied Mathematics*, 65(8):1067–1144, 2012.

[37] R. Talmon, I. Cohen, and S. Gannot. Supervised source localization using diffusion kernels. In *WASPAA 2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 245–248, 2011.

[38] R. Talmon, I. Cohen, and S. Gannot. Single-channel transient interference suppression with diffusion maps. *IEEE transactions on audio, speech, and language processing*, 21(1-2):132–144, 2013.

[39] R. Talmon, D. Kushnir, R.R. Coifman, I. Cohen, and S. Gannot. Parametrization of linear systems using diffusion kernels. *IEEE Transactions on Signal Processing*, 60(3):1159–1173, 2012.

[40] J.B. Tenenbaum, V. de Silva, and J.C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[41] U. von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[42] G. Yang, X. Xu, and J. Zhang. Manifold alignment via local tangent space alignment. *International Conference on Computer Science and Software Engineering*, December 2008.

[43] Z. Zhang and H. Zha. Principal manifolds and nonlinear dimension reduction via local tangent space alignment. Technical Report CSE-02-019, Department of Computer Science and Engineering, Pennsylvania State University, 2002.