Appl. Comput. Harmon. Anal. 33 (2012) 182–203

Contents lists available at SciVerse ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

Patch-to-tensor embedding

Moshe Salhov, Guy Wolf, Amir Averbuch *

School of Computer Science, Tel Aviv University, Tel Aviv 69978, Israel

ARTICLE INFO

Article history:
Received 2 January 2011
Revised 12 September 2011
Accepted 13 November 2011
Available online 16 November 2011
Communicated by Mauro Maggioni

Keywords:
Dimensionality reduction
Manifold learning
Kernel PCA
Diffusion Maps
Patch processing
Vector processing

ABSTRACT

A popular approach to deal with the "curse of dimensionality" in relation with high-dimensional data analysis is to assume that points in these datasets lie on a low-dimensional manifold immersed in a high-dimensional ambient space. Kernel methods operate on this assumption and introduce the notion of local affinities between data points via the construction of a suitable kernel. Spectral analysis of this kernel provides a global, preferably low-dimensional, coordinate system that preserves the qualities of the manifold. In this paper, we extend the *scalar* relations used in this framework to *matrix* relations, which can encompass multidimensional similarities between local neighborhoods of points on the manifold. We utilize the diffusion maps methodology together with linear-projection operators between tangent spaces of the manifold to construct a super-kernel that represents these relations. The properties of the presented super-kernels are explored and their spectral decompositions are utilized to embed the patches of the manifold into a tensor space in which the relations between them are revealed. We present two applications that utilize the patch-to-tensor embedding framework: data classification and data clustering.

1. Introduction

High-dimensional datasets have become increasingly common in many areas due to high availability of data and continuous technological advances. Classical methods for statistical analysis fail on such datasets because of a problem known as "curse of dimensionality". More recent methods, originated from the field of machine learning, assume that the observable parameters in such datasets are related to a small number of underlying factors via a set of non-linear mappings. Mathematically, this assumption is characterized by a manifold structure on which data points are assumed to lie. This underlying manifold is immersed (or submersed) in an ambient space that is defined by observable parameters. Usually, the intrinsic dimension of the underlying manifold is significantly smaller than the dimension of the ambient space.

Several methods have been suggested to provide a global coordinate system that represents the structure of the underlying manifold of a high-dimensional dataset. Kernel methods such as k-PCA [1,2] and Diffusion Maps [3] and its geometric harmonics [4,5] have shown good results. These methods are based on the construction of a kernel that introduces the notion of similarity, proximity, or affinity between data points. Spectral analysis of this kernel is used to obtain an embedding of the data points into a Euclidean space in a manner that preserves the qualities represented by the used kernel.

Kernel methods extend two classical methods that uncover linear structures in datasets. These methods are Principal Component Analysis (PCA) [6,7] and Multi-Dimensional Scaling (MDS) [8,9]. The PCA method uses a covariance matrix between the parameters of the analyzed datasets, and projects the data points on a space spanned by the most significant eigenvectors of this matrix. The MDS method uses the eigenvectors of a Gram matrix, which contains the inner-products

* Corresponding author. Fax: +972 3 6422020.
E-mail address: amir@math.tau.ac.il (A. Averbuch).

1063-5203/$ – see front matter
doi:10.1016/j.acha.2011.11.003

between the points in the analyzed dataset, to define a mapping of data points into an embedded space that preserves most of these inner-products. Both methods are equivalent. They represent data points that use directions in which most of the variance in the dataset is located.

Kernel methods aim at extending the essence of the MDS method by replacing the Gram matrix with a kernel matrix while preserving the qualities represented by it instead of the inner-products that are preserved by the MDS method. Some examples of these methods are LLE [10], Isomaps [11], Laplacian eigenmaps [12], Hessian eigenmaps [13], local tangent space alignment [14,15] and Diffusion maps [3]. These methods are also inspired from spectral graph theory [16]. The defined kernel can be thought of as an adjacency matrix of a graph whose vertices are the points in the dataset. The analysis of the eigenvalues and the corresponding eigenvectors of this matrix can reveal many qualities and connections in the graph.

A recent work [17] suggests to enrich the information represented by a simplified version of the kernel used in the Diffusion Maps method. The original kernel expresses the notion of proximity or the neighborhood structure of the manifold. The enriched kernel also maintains the information about the orientation of the coordinate systems in each neighborhood. This information allows the resulting eigenmap (i.e., the map constructed by the eigenvalues and eigenvectors of the kernel) to be used for determining the orientability of the underlying manifold. In cases when the manifold is orientable, this method finds a suitable global orientation together with the global coordinate system of the embedded space. If the manifold is not orientable, a modification of the used kernel can be utilized to find a double-cover of this manifold.

In this paper, we extend the original Diffusion Maps method in particular and kernel methods in general by suggesting the concept of a super-kernel. We aim at analyzing patches of the manifold instead of analyzing single points on the manifold. Each patch is defined as a local neighborhood of a point in a dataset sampled from an 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.

We suggest a few methods for constructing super-kernels. In particular, linear-projection operators between tangent spaces of data points are suggested for expressing the similarities between the local coordinate systems of their patches. 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. 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 [3].

An alternative method for constructing super-kernels was presented in [18], where parallel transport operators on the underlying manifold were utilized to define the similarities between the patches of the manifold. The resulting super-kernel was utilized there to construct a Vector Diffusion Map (VDM) via spectral analysis. The continuous parallel transport operators were approximated there, in the finite case, by orthogonal transformations that achieve minimal Frobenius distances from the linear-projection operators that are used in this paper. Algorithmically, this orthogonalization step seems like a small difference between projection-based super-kernels, which are presented here, and the ones presented in [18]. However, the theoretical implications of this additional step are significant. While the linear-projections incorporate the effects of the curvature of the manifold on the relations between patches in the super-kernel, these effects are canceled in the orthogonalization process, and only intrinsic quantities to the compared patches (i.e., not the general manifold) are preserved. The resulting VDM embedding shares many of the qualities of the original diffusion maps embedding [3], when the scalar (i.e., 0-form) operators translated to 1-forms setting. Specifically, the infinitesimal generator of the VDM super-kernel converges to the connection-Laplacian, which is related to the heat kernel on 1-forms.

One important quality of the VDM construction [18] is that it does not require an ambient space in which the underlying manifold lies. Therefore, it can be utilized to analyze general graphs. The linear-projection approach, on the other hand, relies on the existence of an ambient space. In practice, most analyzed datasets inherently define an ambient space by the measured features of the data, and thus its existence is well established. However, there are image-processing applications in which the VDM approach would be preferable, since the orthogonal transformations used there can be interpreted as isometries that achieve the best fitting between pairs of images. One such example that utilizes the VDM for the analysis of images in cryo-electron microscopy is presented in [18,19]. In this example [19], noisy two-dimensional EM snapshots of molecules were gathered from many unknown viewing angles, and the embedding performed by VDM was utilized to order the analyzed snapshots according to these angles. Once the viewing angles are known, a three-dimensional illustration of the analyzed molecule can be constructed from these 2D snapshots, but this step is not relevant to the presented methods in here and in [18].

Another approach for applying spectral analysis of non-scalar affinities to data-analysis tasks is to consider non-pairwise relations between data points. One example of this approach is shown in [20], where a hyper-graph was used to model the relations between data points. Each hyper-edge in this hyper-graph represents a relation between an unordered set of data points, and is assigned a weight that quantifies this relation. By expanding the hyper-edges to cliques of related data points, this hyper-graph can be reduced to a standard graph on which well-known partitioning algorithms can be performed to achieve clustering of the original data.

A different example of the utilization of non-pairwise affinities for clustering is presented in [21]. Instead of constructing a hyper-graph to represent the non-pairwise affinities, and then reducing it to a standard graph, an affinity tensor (i.e., a N-way array) is constructed and analyzed directly. This super-symmetric tensor replaces the standard affinity matrix that is usually used in kernel methods. The data clusters are achieved by probabilistic clustering that is performed on the constructed affinity tensor.

The approach used in [20,21] to extend kernel methods to use non-scalar affinities is significantly different from the one presented in this paper and in [17,18]. First, this approach does not utilize the locally-linear structure of the underlying manifold when defining the relations between data points. Secondly, the analyzed items in this approach are still individual data points, even though the considered relations between them are more complex than in classical kernel methods. The patch-processing approach (used in here and in [17,18]), on the other hand, considered *pairwise* affinities between local patches on the manifold. While the complexity (i.e., non-scalarity) of the affinities in [20,21] comes from the nature of the relations between individual data points, the complexity in our case comes from the analyzed items themselves, which are patches instead of data points. This property is best seen by considering the structure of the extended affinity kernel (or super-kernel), which is a block *matrix* in our case and a N-way array (i.e., not a matrix) in [21].

The paper has the following structure: The benefits of patch processing are discussed in Section 1.1. Section 2 contains an overview that includes the problem setup (Section 2.1), a description of Diffusion Maps (Section 2.2) and a description of the general patch-to-tensor embedding scheme based on the construction of a super-kernel (Section 2.3). Linear-projection super-kernels are discussed in Section 3. Description of the diffusion super-kernel is given in Section 4. Description of the linear-projection diffusion super-kernel is given in Section 5. Numerical examples, which demonstrate some aspects of the above constructions, are presented in Section 6. The application of the proposed patch-to-tensor embedding for data-analysis tasks is demonstrated in Section 7. Technical proofs are given in Appendix A.

1.1. Benefits of patch processing

In this section, we provide additional motivation and justification for the approach of analyzing patches rather than individual points. The two main questions that should be addressed for such a justification are: 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? We will provide brief answers to both questions here.

We assume that the processed data have been generated by some physical phenomenon, which is governed by an underlying potential [22,23]. 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 method in this paper is 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 algorithm in this paper is 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 proposed methodology in this paper is classified as a spectral method. 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 massive datasets, there are many duplicities, or near duplicities, and the number of different patches of closely-related data points is significantly less than the number of samples in the dataset. Processing patches, instead of individual data points, reduces these redundancies, thus, it enables also to localize spectral processing, reduce these overheads and alleviate the impracticality barriers.

2. Overview

2.1. Problem setup

Let $M \subseteq \mathbb{R}^m$ be a set of n points sampled from a manifold $\mathcal{M}$ that lies in the ambient space $\mathbb{R}^m$. Let $d \ll m$ be the intrinsic dimension of $\mathcal{M}$, thus, it has a d-dimensional tangent space $T_x(\mathcal{M})$, which is a subspace of $\mathbb{R}^m$, at every point $x \in M$. If the manifold is densely sampled, the tangent space $T_x(\mathcal{M})$ can be approximated by a small enough patch (i.e., neighborhood) $N(x) \subseteq M$ around $x \in M$.

Let $o_x^1, \ldots, o_x^d \in \mathbb{R}^m$, where $o_x^i = (o_x^{i1}, \ldots, o_x^{im})^T$, $i = 1, \ldots, d$, form an orthonormal basis of $T_x(\mathcal{M})$ and let $O_x \in \mathbb{R}^{m \times d}$ be a matrix whose columns are these vectors:

$$O_x \triangleq \begin{pmatrix} | & & | & & | \\ o_x^1 & \cdots & o_x^i & \cdots & o_x^d \\ | & & | & & | \end{pmatrix}, \quad x \in M. \tag{2.1}$$

We will assume from now on that vectors in $T_x(\mathcal{M})$ are expressed by their d coordinates according to the presented basis $o_x^1, \ldots, o_x^d$. For each vector $u \in T_x(\mathcal{M})$, the vector $\tilde{u} = O_x u \in \mathbb{R}^m$ is the same vector as u represented by m coordinates, according to the basis of the ambient space. For each vector $v \in \mathbb{R}^m$ in the ambient space, the vector $v' = O_x^T v \in T_x(\mathcal{M})$ is the linear projection of v on the tangent space $T_x(\mathcal{M})$.

Section 2.2 explains the application of the original diffusion maps method for the analysis of the dataset M. Then, Section 2.3 describes the new construction we propose for embedding patches of the manifold $\mathcal{M}$ based on the points in the dataset M.

2.2. Diffusion maps

The original diffusion maps method [3,24] can be used to analyze the dataset M by exploring the geometry of the manifold $\mathcal{M}$ from which it is sampled. This method is based on defining an isotropic kernel $K \in \mathbb{R}^{n \times n}$, whose elements are defined as $k(x, y) \triangleq e^{-\frac{\|x-y\|}{\varepsilon}}$, $x, y \in M$, where ε is a meta-parameter of the algorithm. This kernel represents the affinities between points on the manifold. The kernel can be viewed as a construction of a weighted graph over the dataset M. The points in M are used as vertices and the weights of the edges are defined by the kernel K. The degree of each point (i.e., vertex) $x \in M$ in this graph is $q(x) \triangleq \sum_{y \in M} k(x, y)$. Kernel normalization with this degree produces a $n \times n$ row stochastic transition matrix P whose elements are $p(x, y) = k(x, y)/q(x)$ for $x, y \in M$, which defines a Markov process (i.e., a diffusion process) over the points in M.

The diffusion maps method computes an embedding of data points on the manifold into a Euclidean space whose dimensionality is usually significantly lower than the original data dimensionality. This embedding is a result of spectral analysis of the diffusion kernel. Thus, it is preferable to work with a symmetric conjugate to P, which is denoted by A and its elements are

$$a(x, y) = \frac{k(x, y)}{\sqrt{q(x)q(y)}} = \sqrt{q(x)}\, p(x, y) \frac{1}{\sqrt{q(y)}}, \quad x, y \in M. \tag{2.2}$$

We will refer to A as the diffusion affinity kernel or as the symmetric diffusion kernel. The eigenvalues $1 = \sigma_0 \geqslant \sigma_1 \geqslant \cdots$ of A and their corresponding eigenvectors $\psi_0, \psi_1, \ldots$ are used to construct the desired map, which embeds each data point $x \in M$ onto the point $\Psi(x) = (\sigma_i \psi_i(x))_{i=0}^{\delta}$ for a sufficiently small δ, which is the dimension of the embedded space and depends on the decay of the spectrum of A. This construction is also known as the Laplacian of the graph constructed by the diffusion kernel [16].

The diffusion maps method uses scalar values to describe the affinities between points on the manifold. We extend this method by considering affinities, or relations, between patches (i.e., neighborhoods of points) on the manifold. These relations cannot be expressed by mere scalar values, since the similarity between patches must contain information about their relative positions in the manifold, their orientations and the correlations between their coordinates. We suggest to use the tangent spaces of the manifold $\mathcal{M}$ (i.e., similarities between them) together with scalar affinities between their tangential data points, to construct a block matrix, where each block represents the affinity between two patches. The rest of this section describes the construction of such block matrices that we call *super-kernels*.

2.3. Super-kernel

Let $\Omega \in \mathbb{R}^{n \times n}$ be an affinity kernel defined on $M \subseteq \mathbb{R}^m$, i.e., each row or each column in Ω corresponds to a data point in M, and each element in it, $[\Omega]_{xy} = \omega(x, y)$, $x, y \in M$, represents an affinity between x and y. We will require, by definition, that Ω will be symmetric and positive semi-definite. Furthermore, we will require that its elements satisfy $\omega(x, y) \geqslant 0$, $x, y \in M$. The exact definition of Ω can vary. We will present few ways to define it in the following sections.

For $x, y \in M$, let $O_{xy} \in \mathbb{R}^{d \times d}$ be a $d \times d$ matrix that represents the similarity between the matrices O_x and O_y, which were defined in Eq. (2.1). The matrices O_x and O_y represent bases of the tangent spaces $T_x(\mathcal{M})$ and $T_y(\mathcal{M})$, respectively. Thus, the matrix O_{xy} represents, in some sense, the similarity between these tangent spaces. We will refer to it as a tangent similarity matrix. We will require that the tangent similarity matrices satisfy the following condition:

$$O_{xy} = O_{yx}^T, \quad x, y \in M. \tag{2.3}$$

In following sections we will present a way to define such tangent similarity matrices.

We use the affinity kernel Ω and the tangent similarity matrices O_{xy} in the following definition to introduce the concept of a *super-kernel*:

Definition 2.1 *(Super-kernel).* A super-kernel is a matrix $G \in \mathbb{R}^{nd\times nd}$ where in terms of blocks, it is a block matrix of size $n \times n$ and each block in it is a $d \times d$ matrix. Each row and each column of blocks in G corresponds to a point in M, and a single block G_{xy} (where $x, y \in M$) represents an affinity or similarity between the patches $N(x)$ and $N(y)$. Each block $G_{xy} \in \mathbb{R}^{d\times d}$ is defined as $G_{xy} \triangleq \omega(x, y) O_{xy}$, $x, y \in M$.

It is convenient to consider each single cell in G as an element in a block, i.e., $[G_{xy}]_{ij}$ where $x, y \in M$ and $i, j \in \{1, \ldots, d\}$. We can also use the vectors o_x^i and o_y^j to apply this indexing scheme and use the following notation:

$$g\big(o_x^i, o_y^j\big) \triangleq [G_{xy}]_{ij}, \quad x, y \in M,\ i, j \in \{1, \ldots, d\}. \tag{2.4}$$

In this notation, it is easy to see that G is symmetric since

$$[G_{xy}]_{ij} = [G_{yx}^T]_{ij} = [G_{yx}]_{ji}, \quad x, y \in M,\ i, j \in \{1, \ldots, d\},$$

where the first equality is due to Eq. (2.3), to the symmetry of Ω and the definition of G_{xy}. It is important to note that $g(o_x^i, o_y^j)$ is *only a notation* for convenience reasons and a single element of a block in G does not necessarily have any special meaning. The block itself, as a whole, holds meaningful similarity information.

We will use spectral decomposition for analyzing a super-kernel G, and utilize it to embed the patches $N(x)$ of the manifold (for $x \in M$) into a tensor space. Let $|\lambda_1| \geqslant |\lambda_2| \geqslant \cdots \geqslant |\lambda_\ell|$ be the ℓ most significant eigenvalues of G and let $\phi_1, \phi_2, \ldots, \phi_\ell$ be their corresponding eigenvectors. According to the spectral theorem, if ℓ is greater than the numerical rank of G, then

$$G \approx \sum_{i=1}^{\ell} \lambda_i \phi_i \phi_i^T, \tag{2.5}$$

where the eigenvectors are treated as column vectors. For convenience reasons, we will treat this approximation as an equality, since, from a theoretical point of view, ℓ can always be chosen to be large enough for actual equality to hold. In practice, the exact value of ℓ depends on the numerical rank of G, the decay of its spectrum, and the exact application of the construction. Usually, however, the affinity kernel and the tangent similarity matrices can be chosen in such a way that a small ℓ will obtain sufficient accuracy for the desired task.

Each eigenvector ϕ_i, $i = 1, \ldots, \ell$, is a vector of length nd. We use a similar notation to Eq. (2.4) to denote each of its elements as $\phi_i(o_x^j)$ where $x \in M$ and $j = 1, \ldots, d$. An eigenvector ϕ_i can also be regarded as a vector of n sections, each of which is a vector of length d that corresponds to a point $x \in M$ on the manifold. To express this notion we use the notation

$$\varphi_i^j(x) = \phi_i\big(o_x^j\big), \quad x \in M,\ i = 1, \ldots, \ell,\ j = 1, \ldots, d. \tag{2.6}$$

Thus, the section in ϕ_i, which corresponds to $x \in M$, is the vector $(\varphi_i^1(x), \ldots, \varphi_i^d(x))^T$.

We use the eigenvalues and eigenvectors of G to construct a spectral map whose definition is similar to the standard (i.e., classic) diffusion map:

$$\Phi\big(o_x^j\big) = \begin{pmatrix} \lambda_1^\mu \phi_1(o_x^j) \\ \vdots \\ \lambda_\ell^\mu \phi_\ell(o_x^j) \end{pmatrix}, \tag{2.7}$$

where μ is a meta-parameter of the embedding. It depends on the specific affinity kernel and on tangent similarity matrices that are used. In Section 3, we will use the value $\mu = \frac{1}{2}$ (for a positive semi-definite G), and in Section 4, we will use the value $\mu = 1$. By using this construction, we get nd vectors of length ℓ. Each $x \in M$ corresponds to d of these vectors, i.e., $\Phi(o_x^j)$, $j = 1, \ldots, d$.

We use these vectors to construct the tensor $\mathcal{T}_x \in \mathbb{R}^\ell \otimes \mathbb{R}^d$ for each $x \in M$, which is represented by the following $\ell \times d$ matrix:

$$\mathcal{T}_x \triangleq \begin{pmatrix} | & & | \\ \Phi(o_x^1) & \cdots & \Phi(o_x^d) \\ | & & | \end{pmatrix}, \quad x \in M. \tag{2.8}$$

In other words, the coordinates of $\mathcal{T}_x$ (i.e., the elements in this matrix) are

$$[\mathcal{T}_x]_{ij} = \lambda_i^\mu \varphi_i^j(x), \quad x \in M,\ i = 1, \ldots, \ell,\ j = 1, \ldots, d, \tag{2.9}$$

where μ is the meta-parameter that is used in Eq. (2.7). Each tensor $\mathcal{T}_x$ represents an embedding of the patch $N(x)$, $x \in M$, into the tensor space $\mathbb{R}^\ell \otimes \mathbb{R}^d$.

In the following sections, we will present several constructions for a super-kernel G and the properties of the embedded tensors, which result from its spectral analysis, are examined. Specifically, we will relate the Frobenius distance between the embedded tensors, regarded as their coordinate matrices, to the relations between their corresponding patches in the original manifold.

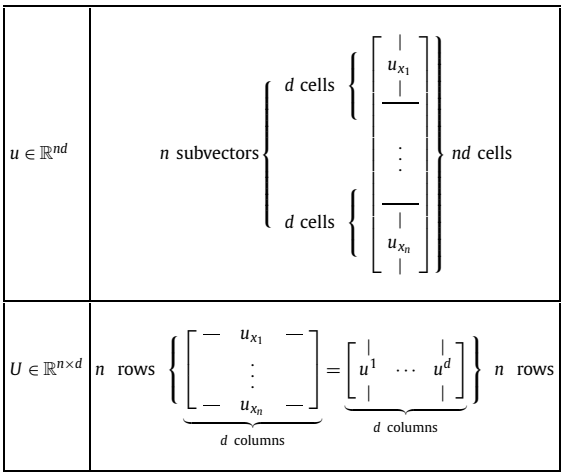


Fig. 1. An illustration of viewing an arbitrary vector $u \in \mathbb{R}^{nd}$ as a matrix $U \in \mathbb{R}^{n \times d}$. Note that $x_1, \ldots, x_n$ are used here to denote all the points in M.

3. Linear-projection super-kernel

The proposed construction of a super-kernel (see Definition 2.1) encompasses both the affinities between points on the manifold $\mathcal{M}$ and the similarities between their tangent spaces. The latter are expressed by the tangent similarity matrices, which can be defined in several ways. In this paper, we will use linear-projection operators to define these similarity matrices. Specifically, for $x, y \in M$, assume that $T_x(\mathcal{M})$ and $T_y(\mathcal{M})$ are two tangent spaces of the manifold. The operator $O_x^T O_y$, which defines a linear projection from $T_y(\mathcal{M})$ to $T_x(\mathcal{M})$ via the ambient space $\mathbb{R}^m$, is used to describe the similarity between them. The obvious extreme cases are an identity matrix, which indicates on complete similarity and a zero matrix, which indicates on orthogonality (i.e. complete dissimilarity). The following definition formalizes the use of these linear projections as tangent similarities in the construction of a super-kernel.

Definition 3.1 *(LP super-kernel).* A Linear-Projection (LP) super-kernel is a super-kernel G, as defined in Definition 2.1, where the tangent similarity matrices are defined by the linear-projection operators

$$O_{xy} = O_x^T O_y, \quad x, y \in M,$$

i.e., for every $x, y \in M$, the blocks of G are defined as $G_{xy} = \omega(x, y) O_x^T O_y$.

The linear-projection operators, which define the tangent similarity matrices by an LP super-kernel, express some important properties of the manifold structure, e.g., curvatures between patches and differences in orientation. While there might be other ways to construct a super-kernel that expresses these properties, LP super-kernels do have an important property, which is given by the following theorem:

Theorem 3.1. *An LP super-kernel G is positive semi-definite and its spectral norm satisfies $\|G\| \leqslant \|\Omega\|$, where $\|\Omega\|$ is the spectral norm of the affinity kernel.*

To prove this theorem, we first need to introduce some notations. Let $u \in \mathbb{R}^{nd}$ be an arbitrary vector of length nd. We can view u as having n subvectors of length d, where each subvector u_x corresponds to a point $x \in M$ on the manifold. Let $U \in \mathbb{R}^{n \times d}$ be a $n \times d$ matrix such that for every $x \in M$ its rows are the subvectors u_x and let $u^1, \ldots, u^d$ be the columns of this matrix. Fig. 1 illustrates these notations. An element in U, which is in a row u_x, $x \in M$, and a column u^j, $j = 1, \ldots, d$, is denoted by u_x^j.

Each subvector u_x, $x \in M$, has d elements, therefore, it can be seen as a vector on the tangent space $T_x(\mathcal{M})$. We define the same vector, presented by m coordinates of the ambient space $\mathbb{R}^m$, as $\tilde{u}_x = O_x u_x$, $x \in M$. Since both u_x and $\tilde{u}_x$ represent the same vector (in two different orthonormal coordinate systems), their norms have the same value. Indeed,

$$\|\tilde{u}_x\|^2 = \tilde{u}_x^T \tilde{u}_x = u_x^T O_x^T O_x u_x = u_x^T u_x = \|u_x\|^2, \quad x \in M. \tag{3.1}$$

We denote by $\tilde{U} \in \mathbb{R}^{n \times m}$ the $n \times m$ matrix whose rows are $\tilde{u}_x$ for every $x \in M$ and we denote its columns by $\tilde{u}^1, \ldots, \tilde{u}^m$. Each element in $\tilde{U}$, which is in a row $\tilde{u}_x$, $x \in M$, and a column $\tilde{u}^i$, $i = 1, \ldots, m$, is denoted as $\tilde{u}_x^i$.

Lemma 3.2. *Let G be an LP super-kernel and let $u \in \mathbb{R}^{nd}$ be an arbitrary vector of length nd. Then, $u^T G u = \sum_{i=1}^m (\tilde{u}^i)^T \Omega \tilde{u}^i$, where Ω is the affinity kernel, always holds.*

The proof of Lemma 3.2 is technical and it is given in Appendix A. We can now prove Theorem 3.1.

Proof of Theorem 3.1. Let $G \in \mathbb{R}^{nd\times nd}$ be an LP super-kernel and let $u \in \mathbb{R}^{nd}$ be an arbitrary vector of length nd. First, we recall that we require the affinity kernel Ω to be positive semi-definite, thus, from Lemma 3.2 we get

$$v^T \Omega v \geqslant 0, \quad v \in \mathbb{R}^n, \tag{3.2}$$

therefore,

$$u^T G u = \sum_{i=1}^{m} \left(\tilde{u}^i\right)^T \Omega \tilde{u}^i \geqslant 0. \tag{3.3}$$

Since u is an arbitrary vector of length nd, Eq. (3.3) shows that G is positive semi-definite. This proves the first part of the theorem.

Next, we denote the spectral norm of Ω by $\sigma = \|\Omega\|$, thus,

$$v^T \Omega v \leqslant \sigma \|v\|^2, \quad v \in \mathbb{R}^n \tag{3.4}$$

therefore, from Lemma 3.2 we get

$$u^T G u = \sum_{i=1}^{m} \left(\tilde{u}^i\right)^T \Omega \tilde{u}^i \leqslant \sum_{i=1}^{m} \sigma \left\|\tilde{u}^i\right\|^2 = \sigma \sum_{i=1}^{m} \sum_{z\in M} \left|\tilde{u}_z^i\right|^2 = \sigma \sum_{z\in M} \|\tilde{u}_z\|^2. \tag{3.5}$$

Then, by using Eq. (3.1) we get

$$\sum_{z\in M} \|\tilde{u}_z\|^2 = \sum_{z\in M} \|u_z\|^2 = \sum_{z\in M} \sum_{j=1}^{d} \left|u_z^j\right|^2 = \|u\|^2. \tag{3.6}$$

By combining Eqs. (3.6) and (3.5), we get

$$u^T G u \leqslant \sigma \|u\|^2. \tag{3.7}$$

Since u is an arbitrary vector of length nd, Eq. (3.7) shows that the Raleigh quotient of G is at most σ. We have already shown that G is positive semi-definite, hence, its spectral norm is its largest eigenvalue, which is also the maximal value of its Raleigh quotient. Therefore, the spectral norm of G is at most σ, and the second part of the theorem is also proved. □

In Sections 3.1 and 3.2, we present two constructions of LP super-kernels. The first construction preserves global tangent similarities by ignoring the affinity between the points in M. The second construction uses binary affinities (i.e., 0 or 1) that preserves local tangent similarities. In Section 5, we will present our final construction, which uses the diffusion affinity kernel to define an LP super-kernel that is used to define the patch-to-tensor embedding.

3.1. Global linear-projection (GLP) super-kernel

A simple way to construct an LP super-kernel is to ignore the affinity kernel completely. In other words, we can use an all-ones matrix as the affinity kernel, thus, the resulting super-kernel will contain only the information about the tangent similarities between patches. While this approach may not be useful in practice, it will provide an insight into the effect the linear projection operators have on the embedding achieved by using an LP super-kernel. The following definition formalizes the described construction of a global LP super-kernel.

Definition 3.2 *(GLP super-kernel).* A Global Linear-Projection (GLP) super-kernel is an LP super-kernel G, as was defined in Definition 3.1, where the affinity kernel is defined as a constant

$$\omega(x, y) \triangleq 1 \quad x, y \in M,$$

i.e., the affinity kernel Ω, in this case, is an all-ones matrix, and the blocks of G are defined as $G_{xy} = O_x^T O_y$, $x, y \in M$.

By definition, a GLP super-kernel G is an LP super-kernel, thus, Theorem 3.1 applies to it and G is positive semi-definite. Therefore, all the eigenvalues of G are non-negative, and a spectral map Φ (Eq. (2.7)) can be defined using $\mu = \frac{1}{2}$. The defined spectral map can then be used to embed each patch $N(x)$, $x \in M$, to a tensor $\mathcal{T}_x$ (Eq. (2.8)). In fact, such an embedding can be defined for every LP super-kernel. The following lemma shows an important relation between the blocks of an *LP* super-kernel G and the embedded tensors resulting from this construction.

Lemma 3.3. *Let $x, y \in M$ be two points on the manifold and let $\mathcal{T}_x$ and $\mathcal{T}_y$ be their embedded tensors (Eq. (2.8)). If the embedding is done by using the spectral map Φ (Eq. (2.7)) of an LP super-kernel G with the meta-parameter $\mu = \frac{1}{2}$, then*

$$G_{xy} = \mathcal{T}_x^T \mathcal{T}_y, \quad x, y \in M,$$

where the tensors are treated as matrices (i.e., their coordinate matrices).

Lemma 3.3 is a result from the construction of the embedded tensors, the definition of LP super-kernels and the application of the spectral theorem to them. A detailed proof of this lemma is given in Appendix A.

A GLP super-kernel preserves global tangent similarities, which are defined as linear-projection operators, between patches. The resulting embedded tensors can be regarded as $\ell \times d$ matrices and their distances can be defined by a matrix norm. Let $\mathcal{D}$ be a matrix norm. The distance between two tensors $\mathcal{T}_x$ and $\mathcal{T}_y$, $x, y \in \mathcal{M}$, is defined as $\mathcal{D}(\mathcal{T}_x - \mathcal{T}_y)$. Theorem 3.4 shows that for matrix norms of a certain form, this distance is equivalent to the distance between the basis matrices O_x and O_y under the same norm.

Theorem 3.4. *Let $\mathcal{D}$ be a matrix norm, defines as $\mathcal{D}(S) = f(S^T S)$ for every matrix S of arbitrary size, where f is a suitable function from the set of all matrices (of all sizes) to $\mathbb{R}$. Let $x, y \in M$ be two points on the manifold and let $\mathcal{T}_x$ and $\mathcal{T}_y$ be their embedded tensors (Eq. (2.8)). If the embedding is done by using the spectral map Φ (Eq. (2.7)) of a* GLP *super-kernel G with the meta-parameter $\mu = \frac{1}{2}$, then*

$$\mathcal{D}(\mathcal{T}_x - \mathcal{T}_y) = \mathcal{D}(O_x - O_y), \quad x, y \in M,$$

where the tensors are treated as matrices (i.e., their coordinate matrices).

Proof. For $x, y \in M$, let O_x and O_y be the matrices defined in Eq. (2.1) and let $\mathcal{D}$ be the matrix norm described in the theorem. Then, by definition,

$$\mathcal{D}(O_x - O_y) = f\big((O_x - O_y)^T (O_x - O_y)\big), \quad x, y \in M. \tag{3.8}$$

We recall the definitions of the blocks in a GLP super-kernel G, thus, the matrix product in the right-hand side Eq. (3.8) is

$$(O_x - O_y)^T (O_x - O_y) = G_{xx} - G_{xy} - G_{yx} + G_{yy}, \quad x, y \in M,$$

therefore, according to Lemma 3.3,

$$\begin{aligned}(O_x - O_y)^T (O_x - O_y) &= \mathcal{T}_x^T \mathcal{T}_x - \mathcal{T}_x^T \mathcal{T}_y - \mathcal{T}_y^T \mathcal{T}_x + \mathcal{T}_y^T \mathcal{T}_y \\ &= (\mathcal{T}_x - \mathcal{T}_y)^T (\mathcal{T}_x - \mathcal{T}_y), \quad x, y \in M.\end{aligned} \tag{3.9}$$

By combining Eqs. (3.9) and (3.8) we get

$$\mathcal{D}(O_x - O_y) = f\big((\mathcal{T}_x - \mathcal{T}_y)^T (\mathcal{T}_x - \mathcal{T}_y)\big) = \mathcal{D}(\mathcal{T}_x - \mathcal{T}_y), \quad x, y \in M,$$

as stated in the theorem. □

Theorem 3.4 shows a relation between matrix distances in the original space and the same type of distances in the embedded space. The distance metrics covered by this theorem are defined by the matrix norms of the form $\mathcal{D}(S) = f(S^T S)$. In fact, two popular matrix norms (i.e., the Frobenius norm and the spectral norm) satisfy this property, and are thus covered by this theorem. The following corollary states that this fact in a formal way.

Corollary 3.5. *Let $x, y \in M$ be two points on the manifold and let $\mathcal{T}_x$ and $\mathcal{T}_y$ be their embedded tensors (Eq. (2.8)). If the embedding is done by using the spectral map Φ (Eq. (2.7)) of a* GLP *super-kernel G, with the meta-parameter $\mu = \frac{1}{2}$, then*:

1. *The Frobenius distances, defined by the Frobenius (also called Hilbert–Schmidt) norm, in the embedded tensor space satisfy*

$$\|\mathcal{T}_x = \mathcal{T}_y\|_F = \|O_x - O_y\|_F.$$

2. *The spectral distances, defined by the spectral (also called operator) norm, in the embedded tensor space satisfy*

$$\|\mathcal{T}_x - \mathcal{T}_y\| = \|O_x - O_y\|.$$

Proof. The Frobenius norm is defined by $\|S\|_F = \operatorname{tr}(S^T S)$ and the spectral norm is defined by $\|S\| = \lambda_{\max}(S^T S)$ (where $\lambda_{\max}$ is a the largest eigenvector of a square matrix). Both definitions fit the form of the matrix norm in Theorem 3.4, thus its result applies for the distances defined by these norms. □

3.2. Local linear-projection super-kernel

We presented an important property (Theorem 3.4 and Corollary 3.5) of the GLP super-kernel construction, but it also has a critical flaw. Manifolds are based on local structures and the similarities between tangent spaces of far-away points are meaningless. The next construction introduces the notion of locality in an LP super-kernel.

We use the notion of neighboring points to define a simple local affinity kernel. We use the notation $x \sim y$ to denote the fact that two points $x, y \in M$ on the manifold are considered neighbors of one another. It means that $x \sim y \Leftrightarrow [N(x) \cap N(y) \neq \emptyset]$, i.e., x and y are neighbors if their patches have mutual points. A more restrictive definition requires neighbors to be in the patches of one another, i.e., $x \sim y \Leftrightarrow [x, y \in N(x) \cap N(y)]$. The exact definition of neighboring points is not crucial for the presented construction. The following definition uses the concept of neighboring points to construct a local LP super-kernel by using a binary affinity kernel, which indicates whether two points are neighbors (i.e., their affinity is 1) or not (i.e., their affinity is 0).

Definition 3.3 *(LLP super-kernel).* A Local Linear-Projection (LLP) super-kernel is a linear-projection super-kernel G, as was defined in Definition 3.1, where the affinity kernel is defined as

$$\omega(x, y) \triangleq \begin{cases} 1 & x \sim y, \\ 0 & \text{otherwise} \end{cases} \qquad x, y \in M,$$

i.e., the blocks of G are defined as $G_{xy} = O_x^T O_y$ for $x \sim y \in M$ and as the zero matrix for non-neighboring points in M.

Since, by definition, an LLP super-kernel is an LP super-kernel, both Theorem 3.1 and Lemma 3.3 are applicable for it. Thus, we can use it to embed patches on the manifolds to tensors by using a spectral map Φ (Eq. (2.7)), with $\mu = \frac{1}{2}$, to construct the tensors in Eq. (2.8). Theorem 3.4 showed that for a wide range of matrix distance metrics, when the embedding is done with a GLP super-kernel, the distance between embedded tensors is equal to the distance between the basis matrices (Eq. (2.1)) of the original patches. While the result in this theorem is not globally true when the embedding is done with an LLP super-kernel, Theorem 3.6 shows that a similar result does apply to neighboring points in this embedding.

Theorem 3.6. *Let $\mathcal{D}$ be a matrix norm of the same form as in Theorem* 3.4*, let $x \sim y \in M$ be two neighboring points on the manifold and let $\mathcal{T}_x$ and $\mathcal{T}_y$ be their embedded tensors* (*Eq.* (2.8)). *If the embedding is done by using the spectral map Φ* (*Eq.* (2.7)) *of an* LLP *super-kernel G, with the meta-parameter $\mu = \frac{1}{2}$, then*

$$\mathcal{D}(\mathcal{T}_x - \mathcal{T}_y) = \mathcal{D}(O_x - O_y), \quad x, y \in M,$$

where the tensors are treated as matrices (*i.e., their coordinate matrices*).

Proof. Let G be the LLP super-kernel that is used to embed the data points in the theorem. According to Definition 3.3, $G_{xy} = O_x^T O_y$ and $G_{yx} = O_y^T O_x$. Also, according to the same definition, since any point is a neighbor of itself then we get $G_{xx} = O_x^T O_x$ and $G_{yy} = O_y^T O_y$. Therefore,

$$(O_x - O_y)^T (O_x - O_y) = G_{xx} - G_{xy} - G_{yx} + G_{yy},$$

and by combining this result with Eq. (3.8) (from the proof of Theorem 3.4), which still applies here (since matrix norms of the same form are considered in both theorems), we get

$$\mathcal{D}(O_x - O_y) = f(G_{xx} - G_{xy} - G_{yx} + G_{yy}).$$

Since Lemma 3.3 applies for LLP super-kernels, a calculation similar to the one in Eq. (3.9) gives

$$\mathcal{D}(O_x - O_y) = f\big((\mathcal{T}_x - \mathcal{T}_y)^T (\mathcal{T}_x - \mathcal{T}_y)\big) = \mathcal{D}(\mathcal{T}_x - \mathcal{T}_y),$$

as stated in the theorem. □

Theorem 3.6 extends Theorem 3.4 to the case of LLP super-kernels and it shows that the embedding achieved by it is locally similar to the one achieved by a GLP super-kernel. Locally similar means that the distances between the embedded tensors are equivalent in both cases of neighboring points. Corollary 3.5 stated that the result of Theorem 3.4 applies, in particular, to the Frobenius distance and to the spectral distance. A similar corollary can be stated for the result of Theorem 3.6 and its proof is the same as in Corollary 3.5.

Corollary 3.7. *Let $x \sim y \in M$ be two neighboring points on the manifold and let $\mathcal{T}_x$ and $\mathcal{T}_y$ be their embedded tensors* (*Eq.* (2.8)). *If the embedding is done by using the spectral map Φ* (*Eq.* (2.7)) *of an* LLP *super-kernel G with the meta-parameter $\mu = \frac{1}{2}$. Then, the Frobenius distances, defined by the Frobenius norm in the embedded tensor space and the spectral distances, defined by the spectral norm, in the embedded tensor space satisfy $\|\mathcal{T}_x - \mathcal{T}_y\|_F = \|O_x - O_y\|_F$ and $\|\mathcal{T}_x - \mathcal{T}_y\| = \|O_x - O_y\|$, respectively.*

The presented construction of an LLP super-kernel takes us one step closer to our final construction of an LP super-kernel that will be used to define the desired patch-to-tensor embedding, since it considers the local nature of the manifold. Section 4 will further examine this aspect by utilizing the diffusion affinity kernel to introduce the notion of locality in the construction of a super-kernel.

4. Diffusion super-kernel

The definition of a super-kernel (Definition 2.1) is based on an affinity kernel, which describes the relations between points on the manifold, and a set of tangent similarity matrices, which describe the relations between tangent spaces of the manifold. Section 3 explored mainly the latter part of this construction (i.e., the matrices O_{xy} for $x, y \in M$), and proposed two simple definitions of an affinity kernel to use in conjunction with the proposed LP super-kernel (see Definitions 3.2 and 3.3). In this section, we set aside the exact definition of the tangent similarity matrices and focus on the affinity kernel that is used. Specifically, Definition 4.1 suggests to use the classic diffusion affinity kernel A (defined in Eq. (2.2)) to describe the affinities in the construction of a super-kernel.

Definition 4.1 *(Diffusion super-kernel).* A diffusion super-kernel is a super-kernel G, as was defined in Definition 2.1, where the affinity kernel is defined as $\omega(x, y) = a(x, y)$, $x, y \in M$, i.e., the affinity kernel is the symmetric diffusion kernel.

The Euclidean distance between data points in the embedded space, which results from the application of the usual diffusion maps, is equal to a diffusion distance in the original ambient space. This diffusion distance measures the distance between two diffusion "bumps" $a(x, \cdot)$ and $a(y, \cdot)$, each of which is a row in the symmetric diffusion kernel that defines the diffusion map. From a technical point of view, this relation means that the Euclidean distance between two arbitrary points in the range of a diffusion map is equal to the Euclidean distances between the corresponding rows of its symmetric diffusion kernel. Lemma 4.1 establishes the same technical relation between the spectral map Φ (Eq. (2.7)) of a diffusion super-kernel G and the rows of the super-kernel itself.

Lemma 4.1. *Let G be a diffusion super-kernel and let Φ be a spectral map (Eq. (2.7)) of this kernel with the meta-parameter $\mu = 1$. For every $x, y \in M$ and $j = 1, \ldots, d$,*

$$\left\| \Phi\left(o_x^j\right) - \Phi\left(o_y^j\right) \right\| = \left\| g\left(o_x^j, \cdot\right) - g\left(o_y^j, \cdot\right) \right\|,$$

where $g(o_x^j, \cdot)$ (or $g(o_y^j, \cdot)$) is a vector whose elements are $g(o_x^j, o_z^\xi)$ (or $g(o_y^j, o_z^\xi)$), which are defined in Eq. (2.4) for every $z \in M$ and $\xi = 1, \ldots, d$.

The proof of Lemma 4.1, which appears in Appendix A, is based on the spectral theorem. It is similar to the corresponding result regarding the standard diffusion maps method. The relation provided by Lemma 4.1 is useful from a technical point of view, but it does not provide meaningful information about the relation between the embedded tensors and the original patches. Theorem 4.2 shows a relation between tensor distances (in the embedded space), defined using the Frobenius norm, to an extended diffusion distance. The extended diffusion distance encompasses the information about similarities between tangent spaces, as well as the affinities between points on the manifold in a fashion similar to the definition of the original diffusion distance.

Theorem 4.2. *Let $x, y \in M$ be two points on the manifold and let $\mathcal{T}_x$ and $\mathcal{T}_y$ be their embedded tensors (Eq. (2.8)). If the embedding is done by using the spectral map Φ (Eq. (2.7)) of a diffusion super-kernel G with the meta-parameter $\mu = 1$, then*

$$\|\mathcal{T}_x - \mathcal{T}_y\|_F^2 = \sum_{z \in M} \left\| a(x, z) O_{xz} - a(y, z) O_{yz} \right\|_F^2,$$

where the tensors are treated as matrices (i.e., their coordinate matrices) when computing the Frobenius distance between them.

Proof. First, we use the definition of the Frobenius norm and the construction of the embedded tensor space to get

$$\begin{aligned} \|\mathcal{T}_x - \mathcal{T}_y\|_F^2 &= \sum_{i=1}^{l} \sum_{j=1}^{d} \left| \lambda_i \varphi_i^j(x) - \lambda_i \varphi_i^j(y) \right|^2 = \sum_{j=1}^{d} \sum_{i=1}^{l} \left| \lambda_i \phi_i\left(o_x^j\right) - \lambda_i \phi_i\left(o_y^j\right) \right|^2 \\ &= \sum_{j=1}^{d} \left\| \Phi\left(o_x^j\right) - \Phi\left(o_y^j\right) \right\|^2. \end{aligned} \tag{4.1}$$

Next, we combine this result with Lemma 4.1 to get

$$\begin{aligned}\|\mathcal{T}_x-\mathcal{T}_y\|_F^2&=\sum_{j=1}^{d}\big\|g\big(o_x^j,\cdot\big)-g\big(o_y^j,\cdot\big)\big\|^2=\sum_{j=1}^{d}\sum_{z\in M}\sum_{\xi=1}^{d}\big|g\big(o_x^j,o_z^\xi\big)-g\big(o_y^j,o_z^\xi\big)\big|^2\\&=\sum_{z\in M}\sum_{j=1}^{d}\sum_{\xi=1}^{d}\big|a(x,z)[O_{xz}]_{j\xi}-a(y,z)[O_{yz}]_{j\xi}\big|^2\\&=\sum_{z\in M}\big\|a(x,z)O_{xz}-a(y,z)O_{yz}\big\|_F^2,\end{aligned}$$

as states in the theorem. □

Corollary 4.3 reinforces our argument that the presented metric is indeed an extension of the original diffusion distance by presenting a case in which both metrics converge up to multiplication by a constant (i.e., $\sqrt{d}$).

Corollary 4.3. *In the context of Theorem* 4.2*, if all the tangent similarity matrices are orthogonal, and for every* $x, y, z \in M$*, the product* $O_{xz}^T O_{yz}$ *is symmetric and positive semi-definite, then*

$$\|\mathcal{T}_x-\mathcal{T}_y\|_F^2=d\big\|a(x,\cdot)-a(y,\cdot)\big\|^2,$$

where $a(u,\cdot)$ *denotes a vector of length n with the entries* $a(u,z)$ *for every* $z\in M$*. In other words, the extended diffusion distance in this case is the original diffusion distance multiplied by* $\sqrt{d}$*.*

Proof. According to Theorem 4.2,

$$\|\mathcal{T}_x-\mathcal{T}_y\|_F^2=\sum_{z\in M}\big\|a(x,z)O_{xz}-a(y,z)O_{yz}\big\|_F^2,\quad x,y\in M,$$

and since the Frobenius norm comes from the Frobenius inner product (denoted by ':'),

$$\begin{aligned}&\big\|a(x,z)O_{xz}-a(y,z)O_{yz}\big\|_F^2\\&\quad=\big\|a(x,z)O_{xz}\big\|_F^2-2\big(a(x,z)O_{xz}:a(y,z)O_{yz}\big)+\big\|a(y,z)O_{yz}\big\|_F^2\\&\quad=a(x,z)^2\operatorname{tr}\big(O_{xz}^TO_{xz}\big)-2a(x,z)a(y,z)\operatorname{tr}\big(O_{xz}^TO_{yz}\big)+a(y,z)^2\operatorname{tr}\big(O_{yz}^TO_{yz}\big),\quad x,y,z\in M.\end{aligned}$$

If O_{xz} and O_{yz} are both $d\times d$ orthogonal matrices, as the corollary assumes, then so are $O_{xz}^TO_{xz}$, $O_{yz}^TO_{yz}$, and $O_{xz}^TO_{yz}$. In fact, the first two are the $d\times d$ identity matrix, whose trace is d. The product $O_{xz}^TO_{yz}$ is also symmetric and positive semi-definite, by the assumption in the corollary, thus, its d eigenvalues are all ones and its trace is d. The traces of these matrices are all d, thus,

$$\big\|a(x,z)O_{xz}-a(y,z)O_{yz}\big\|_F^2=da(x,z)^2-2da(x,z)a(y,z)+da(y,z)^2,$$

therefore, if we combine this result with Theorem 4.2, we get

$$\begin{aligned}\|\mathcal{T}_x-\mathcal{T}_y\|_F^2&=d\sum_{z\in M}\big(a(x,z)^2-2a(x,z)a(y,z)+a(y,z)^2\big)\\&=d\big(\big\langle a(x,\cdot),a(x,\cdot)\big\rangle-2\big\langle a(x,\cdot),a(y,\cdot)\big\rangle+\big\langle a(y,\cdot),a(y,\cdot)\big\rangle\big)\\&=d\big\|a(x,\cdot)-a(y,\cdot)\big\|^2,\quad x,y\in M,\end{aligned}$$

as stated in the corollary. □

5. Linear-projection diffusion super-kernel

In Section 4, we utilized the diffusion affinity kernel to construct a super-kernel without defining the tangent similarity matrices. In Section 3, we presented a general construction of a super-kernel that is based on linear-projection tangent similarity matrices (see Definition 3.1) without defining the affinity kernel. Definition 5.1 combines these constructions and introduces our construction of a linear-projection diffusion super-kernel. We will construct a patch-to-tensor embedding, which maps patches of the manifold into a meaningful tensor space by using the spectral map of this super-kernel.

Definition 5.1 *(LPD super-kernel).* A Linear-Projection Diffusion (LPD) super-kernel G is both a diffusion super-kernel as was defined in Definition 4.1 and an LP super-kernel as was defined in Definition 3.1, i.e., its blocks are defined as $G_{xy}=a(x,y)O_x^TO_y$, $x,y\in M$.

Since an LPD super-kernel is an LP super-kernel, Theorem 3.1 applies to it. We recall that the spectral norm of the symmetric diffusion kernel is $\|A\| = 1$, therefore, we get Corollary 5.1 for the case of LPD super-kernels, whose proof is an immediate result of this discussion.

Corollary 5.1. *An LPD super-kernel G is positive semi-definite and its operator norm satisfies $\|G\| \leqslant 1$.*

Another immediate result of Theorem 3.1 in this case, or rather of Corollary 5.1, has to do with the eigenvalues of an LPD super-kernel:

Corollary 5.2. *All the eigenvalues of an LPD super-kernel are between 0 and 1, i.e., its eigenvalues are $1 \geqslant \lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant 0$.*

Proof. According to Corollary 5.1, an LPD super-kernel G is positive semi-definite, thus, its eigenvalues are non-negative and the largest one is equal to the spectral norm of G, which satisfies $\|G\| \leqslant 1$, according to the same corollary. Therefore, every eigenvalue of G is at least 0 and at most 1. □

We recall that the original diffusion distance between two points $x, y \in M$ is

$$\left\|a(x,\cdot) - a(y,\cdot)\right\|^2 = \sum_{z\in M}\left(a(x,z) - a(y,z)\right)^2.$$

According to Theorem 4.2 and Definition 5.1, when the spectral map Φ (Eq. (2.7)) of an LPD super-kernel is used to embed the patches of these points, the Frobenius distance between the resulting embedded tensors (regarded as their coordinate matrices) satisfies

$$\|\mathcal{T}_x - \mathcal{T}_y\|_F^2 = \sum_{z\in M}\left\|a(x,z)O_x^T O_z - a(y,z)O_y^T O_z\right\|_F^2 = \sum_{z\in M}\sum_{j=1}^{d}\left\|\left(a(x,z)O_x^T - a(y,z)O_y^T\right)o_z^j\right\|^2. \tag{5.1}$$

The vectors o_z^j in this equation are unit vectors that form an orthonormal basis of the tangent space $T_x(\mathcal{M})$ at the point $z \in M$. For each point $z \in M$, the matrix $[a(x,z)O_x^T - a(y,z)O_y^T]$ is applied to each of these unit vectors and the squared lengths of the resulting vectors are summed. These terms can be seen as extensions of the terms $(a(x,z) - a(y,z))$ of the original diffusion distance, which only consider the differences between scalar affinities.

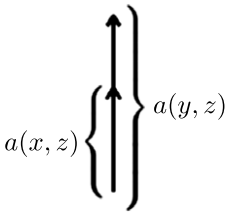
Let u be a unit vector. We examine the result of applying the matrix $[a(x,z)O_x^T - a(y,z)O_y^T]$, $x, y, z \in M$, to such a vector (see Fig. 2). First, since u is a unit vector, $a(x,z)u$ and $a(y,z)u$ are vectors of lengths $a(x,z)$ and $a(y,z)$, respectively, in the same direction as u (see Fig. 2(a)). The vector $a(x,z)O_x^T u$ is a linear projection of the vector $a(x,z)u$ on the tangent space $T_x(\mathcal{M})$, where the resulting vector is represented by the d coordinates of this tangent space (see Fig. 2(b)). Similarly, $a(y,z)O_y^T u$ is the projection of $a(y,z)u$ on $T_y(\mathcal{M})$, represented by the d local coordinates of this tangent space. The resulting vector $a(x,z)O_x^T u - a(y,z)O_y^T u$ contains the difference between the two resulting vectors of length d (see Fig. 2(c)). If the lengths of vectors in the direction of u are not changed by these projections (e.g., u is on both $T_x(\mathcal{M})$ and $T_y(\mathcal{M})$), and if the coordinate systems of these tangent spaces are equivalent, in the sense that the direction of these projections is the same in both of them, then the length of the resulting vector will simply be the scalar difference $a(x,z) - a(y,z)$. This is an extreme scenario. In most cases, these differences (i.e., the scalar difference and the length of the difference vector) will not coincide due to the curvature of the manifold and the difference in coordinate systems on the manifold.

We have shown that the embedding achieved by spectral analysis of an LPD super-kernel is similar, in some sense, to the one achieved by the original diffusion maps method. We use the name Patch-to-Tensor Embedding (PTE) for the presented embedding,

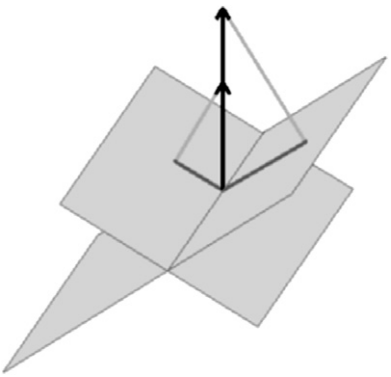
$$N(x) \overset{PTE}{\longmapsto} \mathcal{T}_x, \quad x \in M, \tag{5.2}$$

which maps each patch of the manifold to the corresponding tensor, defined by Eq. (2.8), by using the spectral map Φ (Eq. (2.7)) with $\mu = 1$, of an LPD super-kernel that is constructed over the input set M of points on the manifold. In Section 6, a simple (yet not optimal) algorithm for constructing a Patch-to-Tensor Embedding is presented. The results of its applications on synthetic datasets are presented in Section 6 and its utilizations for data clustering & classification and for image segmentation are presented in Section 7.

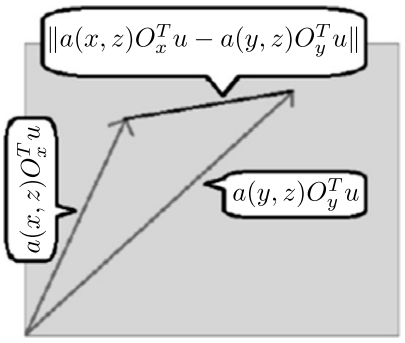
The finite LPD super-kernel that we presented here is further explored in [25], where its properties when it becomes continuous are examined and analyzed. Specifically, the infinitesimal generator of this super-kernel and the stochastic process defined by it are explored. It is shown there that the resulting infinitesimal generator of this super-kernel converges to a natural extension of the original diffusion operator from scalar functions to vector fields. This operator is shown to be locally equivalent to a composition of linear projections between tangent spaces and the vector-Laplacians on them. An LPD process can then be defined by using the LPD super-kernel as a transition operator while extending the process to be continuous.



(a) Two vectors in the direction of u.



(b) Projection of the two vectors on two tangent spaces.



(c) The difference between the local coordinates of the projected vectors.

Fig. 2. An illustration of the application of the matrix $[a(x,z)O_x^T - a(y,z)O_y^T]$ to the unit vector u.

The LPD process propagates tangent vectors over the manifold [25]. Since it is a stochastic process, it has an inherent time parameter, which we will refer to as the diffusion time in the rest of this paper. In the finite discrete case, this parameter is interpreted as the number of steps performed by the diffusion. The original DM algorithm also has such diffusion time parameter, which is expressed as powers of the diffusion operator or, more conveniently, as powers of the used eigenvalues in the embedding process. In Section 7, we use different diffusion times (also expressed as powers of the used spectrum in the embedding) to obtain ideal data clustering in the resulting embedded space of the LPD-based PTE.

6. Numerical examples

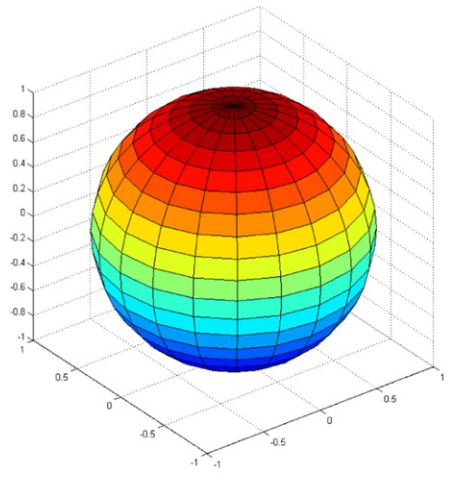
This section presents several numerical results that demonstrate the PTE characteristics on synthetically produced datasets. Specifically, the following demonstration relates Theorem 3.1 and the corresponding corollary (Corollary 5.1) to the LPD super-kernels. Algorithm 1 is used to construct a PTE for the analysis of three exemplary manifolds.

Algorithm 1 Patch-to-Tensor Embedding Construction (PTEC)

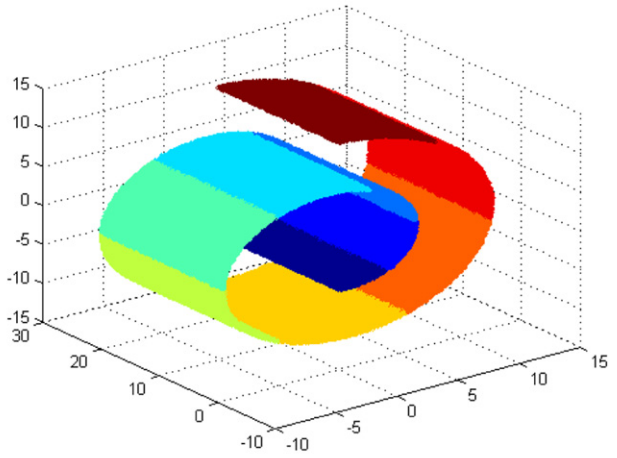
Input: Data points: $x_1, \ldots, x_n \in R^m$ and parameters: Patch size ρ and ℓ

1: For each $x \in M$ estimate an orthonormal basis $O_x \in R^{m\times d}$ of the local tangent space based on ρ points uniformly distributed over a small neighborhood of x
2: Construct a diffusion affinity kernel A according to Eq. (2.2)
3: Construct an LPD super-kernel G according to Definition 5.1
4: Construct spectral map $\Phi(o_x^j)$ for $j = 1, \ldots, d$ according to Eq. (2.7) utilizing the SVD decomposition of the constructed LPD super-kernel G
5: Construct a Tensor $\mathcal{T}_x \in \mathbb{R}^\ell \otimes \mathbb{R}^d$ for each $x \in M$ according to Eq. (2.8)

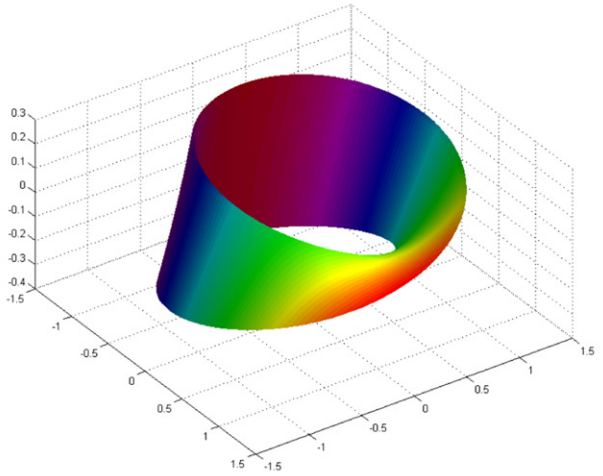
The examined manifolds are illustrated in Fig. 3. They include: the unit sphere S^2, the three-dimensional Swiss roll and the three-dimensional Mobius band. The analyzed datasets were produced using the following steps. We sample 2000 points uniformly from each manifold embedded in $\mathbb{R}^3$. Each set of points was extended to an ambient space of 17 by a linear transformation operator $Q \in \mathbb{R}^{3\times 17}$. The linear operator Q was chosen randomly with uniform distribution under



(a) A Sphere



(b) A Swiss Roll



(c) A Mobius Band

Fig. 3. Examined manifolds.

the constraint $Q^T Q \succ 0$. The positive definiteness constraint guaranty that Q is non-singular. Algorithm 1 with parameters $\rho = 30$ and $\ell = 3$ was utilized to find the LPD super-kernel and the corresponding mapping for each example. The choice of the value $\ell = 3$ was calculated by aggregating all the estimated local dimensions of each tangent space $T_x(\mathcal{M})$ following the footsteps of [17].

Fig. 4 describes the numerical rank of the LPD super-kernel for increasing values of μ. The resulting eigenvalues for all of the examples are decaying for $\mu = 1$ and the decay increases as μ increases.

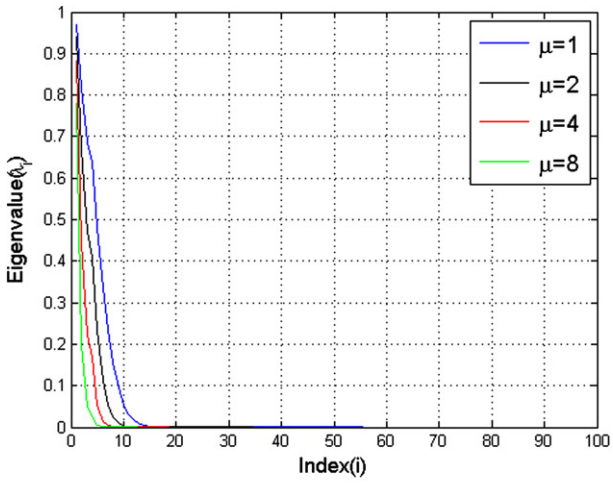
In order to analyze the support interval that the eigenvalues span, we compute the Cumulative Distribution Function (CDF) of the eigenvalues of the LPD super-kernel. The corresponding CDF is the probability that any real-valued eigenvalue of the LPD super-kernel will have a value less than or equal to a threshold τ. More rigorously, the CDF is defined as

$$F\big(\tau, f(\lambda_i)\big) = P \quad (\lambda_i \leqslant \tau), \tag{6.1}$$

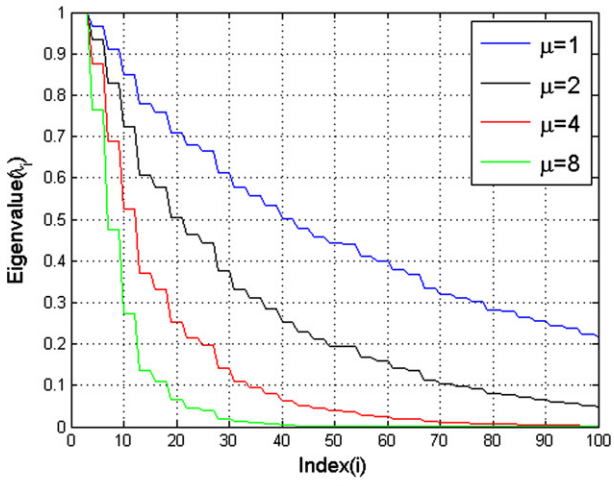
where $f(\lambda_i)$ is the distribution function of λ_i and τ is a given threshold.

The utilization of the CDF enables a compact and informative presentation of the characterization of the relevant eigenvalues. The CDF describes the interval on which there is a positive probability to find eigenvalues and what is the percentage of non-negligible eigenvalues from all the eigenvalues distributions.

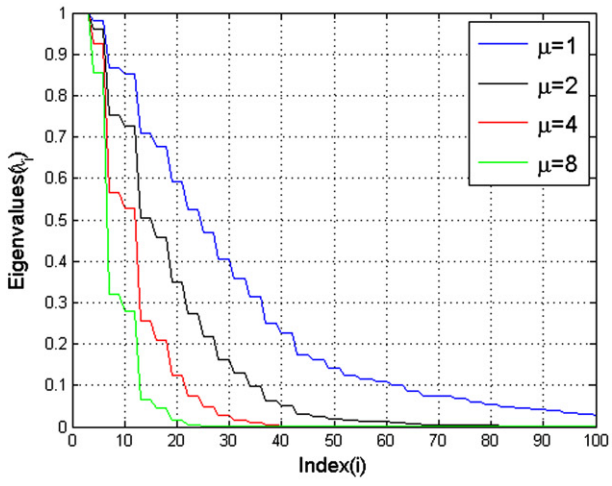
The estimated CDFs of the eigenvalues for all the manifold examples are presented in Fig. 5. For each LPD super-kernel instance, we estimated the distribution function $f(\lambda_i)$ by integrating the corresponding histogram of the resulted eigenvalues.



(a)

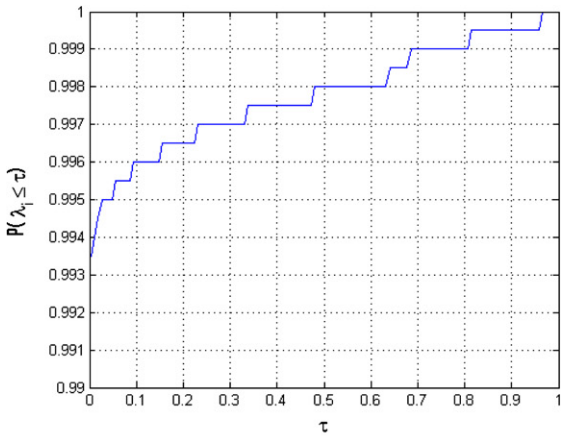


(b)

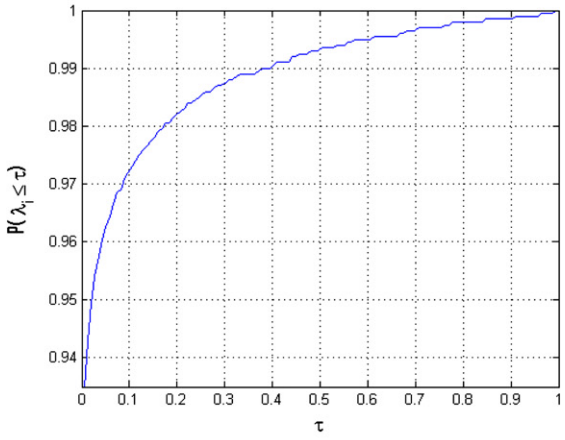


(c)

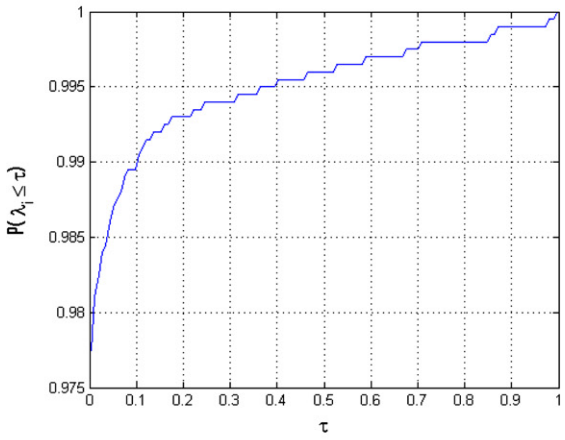
Fig. 4. The 100 largest eigenvalues of the LPD super-kernels that corresponds to (a) Sphere, (b) Swiss roll, and (c) Mobius band.



(a)



(b)



(c)

Fig. 5. The CDFs of the eigenvalues of the LPD super-kernels of (a) Sphere, (b) Swiss roll, and (c) Mobius band.

According to the calculated CDFs the examined LPD super-kernels have positive probability only on the $[0, 1]$ interval as was suggested by Theorem 3.1 and by the corresponding Corollary 5.1 for the LPD super-kernel. Furthermore, the CDFs calculated probabilities, which an eigenvalue will have a value which is less than 0.1 on the Sphere, Swiss roll and the Mobius examples, are 0.996, 0.970 and 0.990, respectively. The CDFs have high probabilities to have a small eigenvalue, hence, only a small number of eigenvalues and their corresponding eigenvectors are required to preserve the structure and variability in the LPD super-kernel matrices.

Table 1
The six classes of tissues that are represented in the analyzed dataset.

	Number of measurements
Normal tissue classes	
Connective tissue (con)	14
Adipose tissue (adi)	22
Glandular tissue (gla)	16
Pathological tissue classes	
Carcinoma (car)	21
Fibro-adenoma (fad)	15
Mastopathy (mas)	18

7. Data analysis using patch-to-tensor embedding

PTE provides a general framework that can be utilized in a wide collection of data analysis tasks such as clustering, classification, anomaly detection and related manifold learning tasks. In this section, we demonstrate the application of the PTE method to two data analysis challenges: 1. Classification of breast tissue impedance measurements. 2. Data clustering that is based on image segmentation.

7.1. Electrical impedance breast tissue classification

Biological tissues have complex electrical impedance related to the tissue dimension, the internal structure and the arrangement of the constituent cells. Therefore, the electrical impedance can provide useful information based on heterogeneous tissue structures, physiological states and functions [26].

Electrical impedance techniques have long been used for tissue characterization [27]. Recently, an interesting dataset of breast tissue impedance measurements was published [28]. The dataset consisted of 106 spectra recorded in samples of breast tissue from 64 patients undergoing breast surgery. Each spectrum consisted of twelve impedance measurements taken at different frequencies ranging from 488 Hz to 1 MHz. Detailed description of the data collection procedure as well as classification of the cases and frequencies used are given in [29,30]. Table 1 shows the six classes of tissue that are represented in the given dataset.

Several extracted features from the impedance measurements for the classification preprocessing step were described in [30]:

- I_0 – impedivity at zero frequency (low frequency limit resistance);
- PA_{500} – phase angle at 500 kHz;
- S_{HF} – high frequency slope of phase angle (at 250, 500 and 1000 KHz points);
- D_4 – impedance distance between spectral ends;
- *AREA* – area under spectrum;
- $AREA_{D_4}$ – area normalized by D_4;
- IP_{Max} – maximum of the spectrum;
- *DR* – distance between I_0 and real part of the maximum frequency point;
- *PERIM* – length of spectral curve.

The computed attributes are given in the dataset. More details are given in [30]. A tissue classification method for the given impedance attributes is given in [30]. The suggested method is based on a hierarchal architecture in which in the first stage a classifier is used to discriminated fatty tissue (Connective and Adipose tissue) from the non-fatty tissue (Carcinoma, Fibroadenoma, Mastopathy and Glandular tissue). At the second stage, additional classifier is used to discriminate Carcinoma tissue from other non-fatty tissue categories. The performances of the algorithm in [30] are 100% success in discriminating the fatty from the non-fatty tissue at the first stage. The Carcinoma can be discriminated from the other non-fatty tissue types with more than 86% success. The success in identifying the FMG class (Fibroadenoma + Mastopathy + Glandular tissue) is 94.5%.

In this section, we follow the footsteps of [30] in classifying the post-prosing attributes into the same tissue categories using PTE. Initially, the given dataset was normalized to have zero mean and a unit standard deviation for each attribute. Then, the PTE construction, detailed in Algorithm 1, was used to construct the LPD super-kernel followed by embedding of the measurements into a tensor space. The affinity kernel is computed by Eq. (2.2) where ε was chosen as the mean Euclidean distance between all the pairs of data points in the given dataset. The parameters in the PTE construction were $\ell = 5$ and $\rho = 66$. They were chosen in an exhaustive search to optimize the classification accuracy.

The classification performance is based on a leave-one-out methodology in which each of the measurements was labeled according to its nearest neighbor in the embedded tensor space. The Frobenius norm was used as the distance metric. The classification performance is described in Table 2.

Table 2
Performance summery of the PTE-based classification algorithm.

Tissue category	Correct detection	False detection	Miss-detection
Fatty	97.2%	0	2.7%
Carcinoma	86.36%	13.6%	9.5%
FMG	93.9%	6.1%	6.1%

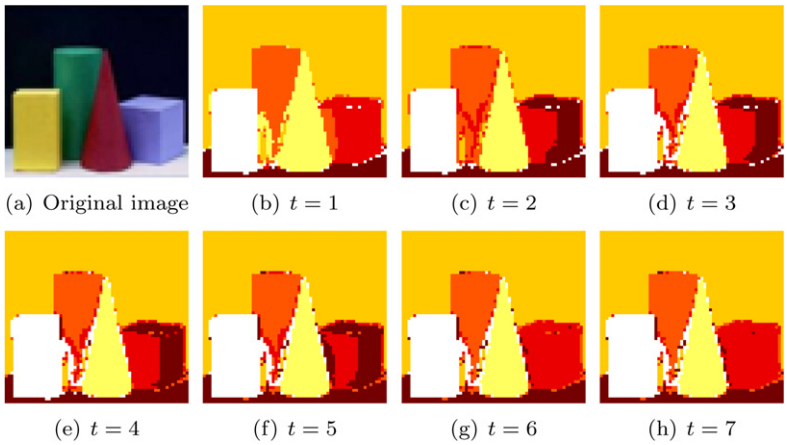


Fig. 6. The PTE segmentation results for the image 'Cubes' when $\ell = 10$ and $d = 10$. The results are shown at several diffusion times t.

The achieved classification performances, which were obtained by PTE with a single classification stage, are competitive to the ones in [30]. The optimization of the classifier was done only with respect to two parameters: ρ the number of points per patch and ℓ the number of eigenvectors from the application of the SVD procedure.

7.2. Image segmentation

Image segmentation clusters pixels into image regions corresponding to individual surfaces, objects, or natural parts of objects. It plays a key role in many computer vision tasks such as object recognition, image compression, image editing and image retrieval. It has been extensively studied in computer vision [31–33] and statistics with a vast number of different algorithms [34–37]. Early techniques utilized region splitting or merging [31,38,39], which correspond to divisive and agglomerative algorithms in the clustering literature [36]. More recent algorithms often optimize some global criterion such as intra-region consistency and inter-region boundary lengths or dissimilarity [40–43].

Graph cut techniques from combinatorial optimization are used for image segmentation [44–46]. Graph cut methods view the image as a graph weighted to reflect intensity changes and performs a max-flow/min-cut analysis to find the minimum-weight cut between the source and the sink. One of the features of this algorithm is that an arbitrary segmentation may be obtained with enough user interaction and it generalizes easily to 3D and beyond.

The PTE framework enables to view the image via a LPD super-kernel that reflects the affinities between pixels and the projection of the related tangent spaces. The PTE construction translates the given pixel-related features into tensors in the embedded space. The image segmentation into similar sets is achieved by clustering the tensors in the embedded space.

For our image segmentation examples, we utilized pixel color information and its spatial (x,y) location multiplied by scaling factor $w = 0.1$. Hence, given an RGB image with $I_x \times I_y$ pixels, we generated a $5 \times (I_x \cdot I_y)$ dataset X.

Algorithm 1 embeds X into a tensor space. The first step in Algorithm 1 constructs local patches. Each generated patch captures the relevant neighborhood and considers both color similarity and spatial similarity. Hence, a patch is more likely to include attributes related to spatially close pixels. It is important to note that the affinity kernel is computed according to Eq. (2.2) where ε equals the mean Euclidean distance between all the pairs in X. The PTE parameters ℓ and ρ were chosen to generate the most homogeneous segments. The k-means algorithm with "sum of square differences" was used to cluster the tensors into similar sets.

Figs. 6–9 present the segmentation results from the application of the PTE algorithm, where for each figure, (a) is the original image. All of the images are of size 60×60 except for the 'Sport' (Fig. 8) image, which is of size 79×42. Each figure describes the segmentation result at several diffusion times t. The impact of the diffusion time on the segmentation quality was significant for the 'Cubes' (Fig. 6) and 'Fabric' (Fig. 9) images. For example, as can be seen in Fig. 9, the first two images (Fig. 6(b) and Fig. 6(c)), which correspond to $t = 1$ and $t = 2$ respectively, show poor segmentation qualities. As t increases, the segmentation becomes more homogeneous and the main structures in the original image can be separated as we see, for example, in (e) where $t = 4$. Another interesting aspect related to the diffusion time parameter t is the smoothing effect it has, when it increases, on the pairwise distances between data points in the embedded space. By increasing t, the pairwise

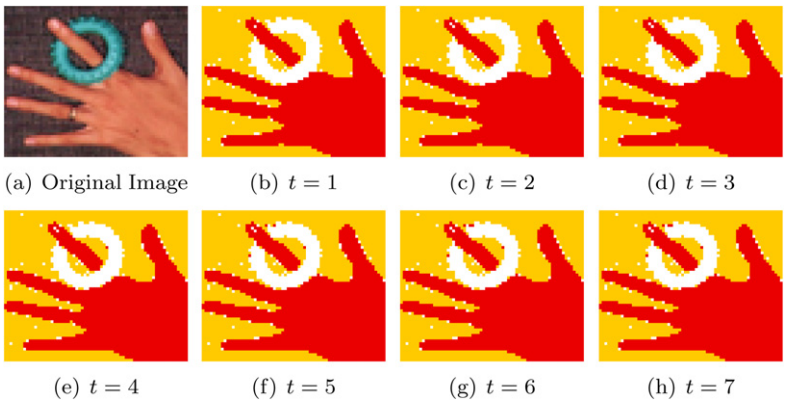


Fig. 7. The PTE segmentation results for the image 'Hand' when $l = 10$ and $d = 10$. The results are shown at several diffusion times t.

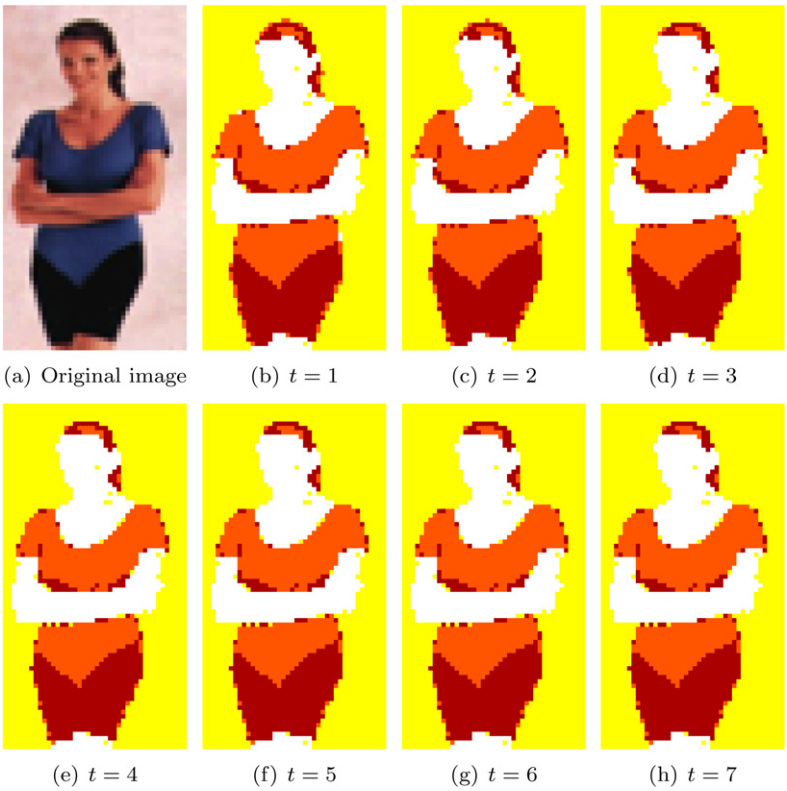


Fig. 8. The PTE segmentation result for the image 'Sport' when $\ell = 10$ and $d = 17$. The results are shown at several diffusion times t.

distances between similar tensors decrease while the distances between dissimilar tensors increase. In the segmentation case, the result will be pixel-label change. For example, Fig. 6 presents the 'Cubes' image segmentation as a function of $t = 1, 2, \ldots, 7$. The rightmost cube in the segmented images becomes more homogeneous as t increases.

8. Conclusion

In this paper, we presented an extension of the scalar-affinity kernels that are used in kernel methods. We used mainly a linear-projection-based construction of this extension, which we call a *super-kernel*. Other constructions such as ones based on orthogonal transformations can also be used. Such constructions will be explored in future works.

The linear-projection diffusion (LPD) super-kernel that was introduced in this paper is further explored in [25]. There, its properties in the continuous case are examined and the generated diffusion process that propagates tangent vectors along the manifold is presented. This LPD process can also be utilized for out-of-sample extensions of vector fields. Future

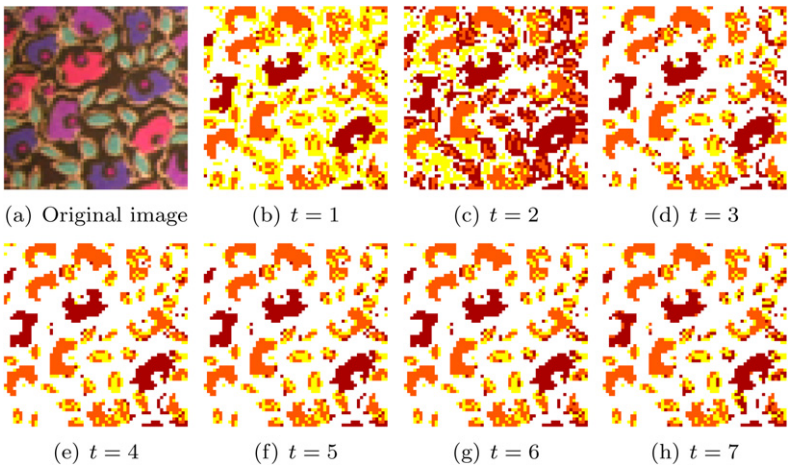


Fig. 9. The PTE segmentation results for the image 'Fabric' when $\ell = 10$ and $d = 10$. The results are shown at several diffusion times t.

works will present this utilization together with a complete patch-processing data-mining framework that combines coarse-graining, dictionary-based subsampling, dimensionality reduction and smooth interpolation techniques.

Among other benefits, the patch-processing approach introduced here will 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.

Acknowledgment

This research was partially supported by the Israel Science Foundation (Grant No. 1041/10).

Appendix A. Technical proofs

Lemma 3.2. *Let G be an LP super-kernel and let $u \in \mathbb{R}^{nd}$ be an arbitrary vector of length nd. Then, $u^T G u = \sum_{i=1}^{m} (\tilde{u}^i)^T \Omega \tilde{u}^i$, where Ω is the affinity kernel, always holds.*

Proof. Let G be an LP super-kernel as was defined in Definition 3.1, and let $u \in \mathbb{R}^{nd}$ be an arbitrary nd vector. The product $u^T G u$ can be expressed block-wise as

$$u^T G u = \sum_{x \in M} \sum_{y \in M} u_x^T G_{xy} u_y. \tag{A.1}$$

By using the definition of the blocks of G, we get

$$u_x^T G_{xy} u_y = \omega(x, y) u_x^T O_x^T O_y u_y = \omega(x, y) \tilde{u}_x^T \tilde{u}_y, \quad x, y \in M, \tag{A.2}$$

therefore,

$$u^T G u = \sum_{x \in M} \sum_{y \in M} \omega(x, y) \tilde{u}_x^T \tilde{u}_y = \sum_{x \in M} \sum_{y \in M} \omega(x, y) \sum_{i=1}^{m} \tilde{u}_x^i \tilde{u}_y^i \tag{A.3}$$

$$= \sum_{i=1}^{m} \left(\sum_{x \in M} \sum_{y \in M} \tilde{u}_x^i \omega(x, y) \tilde{u}_y^i \right) = \sum_{i=1}^{m} (\tilde{u}^i)^T \Omega \tilde{u}^i, \tag{A.4}$$

as stated in the lemma. □

Lemma 3.3. *Let $x, y \in M$ be two points on the manifold and let $\mathcal{T}_x$ and $\mathcal{T}_y$ be their embedded tensors (Eq. (2.8)). If the embedding is done by using the spectral map Φ (Eq. (2.7)) of an LP super-kernel G with the meta-parameter $\mu = \frac{1}{2}$, then*

$$G_{xy} = \mathcal{T}_x^T \mathcal{T}_y, \quad x, y \in M,$$

where the tensors are treated as matrices (i.e., their coordinate matrices).

Proof. Let $x, y \in M$ be two points on the manifold, and let G be the LP super-kernel that is used to embed the points in the lemma. According to Eq. (2.9) (with $\mu = \frac{1}{2}$), the elements of the matrix product $\mathcal{T}_x^T \mathcal{T}_y$ are

$$\left[\mathcal{T}_x^T \mathcal{T}_y\right]_{ij} = \sum_{\xi=1}^{\ell} [\mathcal{T}_x]_{\xi i} [\mathcal{T}_y]_{\xi j} = \sum_{\xi=1}^{\ell} \sqrt{\lambda_\xi} \varphi_\xi^i(x) \sqrt{\lambda_\xi} \varphi_\xi^j(y), \quad i, j = 1, \ldots, d.$$

According to the spectral theorem, Eqs. (2.6) and (2.4), we have

$$\sum_{\xi=1}^{\ell} \lambda_\xi \varphi_\xi^i(x) \varphi_\xi^j(y) = \sum_{\xi=1}^{\ell} \lambda_\xi \phi_\xi\left(o_x^i\right) \phi_\xi\left(o_y^j\right) = g\left(o_x^i, o_y^j\right), \quad i, j = 1, \ldots, d,$$

thus,

$$\left[\mathcal{T}_x^T \mathcal{T}_y\right]_{ij} = g\left(o_x^i, o_y^j\right) = [G_{xy}]_{ij}, \quad i, j = 1, \ldots, d.$$

Therefore, $\mathcal{T}_x^T \mathcal{T}_y = G_{xy}$ as the lemma states. □

Lemma 4.1. *Let G be a diffusion super-kernel and let Φ be a spectral map (Eq. (2.7)) of this kernel with the meta-parameter $\mu = 1$. For every $x, y \in M$ and $j = 1, \ldots, d$,*

$$\left\| \Phi\left(o_x^j\right) - \Phi\left(o_y^j\right) \right\| = \left\| g\left(o_x^j, \cdot\right) - g\left(o_y^j, \cdot\right) \right\|,$$

where $g(o_x^j, \cdot)$ (or $g(o_y^j, \cdot)$) is a vector whose elements are $g(o_x^j, o_z^\xi)$ (or $g(o_y^j, o_z^\xi)$), which are defined in Eq. (2.4) for every $z \in M$ and $\xi = 1, \ldots, d$.

Proof. According to the definition of the spectral map Φ (Eq. (2.7)) with $\mu = 1$,

$$\left\langle \Phi\left(o_x^i\right), \Phi\left(o_y^j\right) \right\rangle = \sum_{\zeta=1}^{\ell} \lambda_\zeta^2 \phi_\zeta\left(o_x^i\right) \phi_\zeta\left(o_y^j\right), \quad x, y \in M,\ i, j = 1, \ldots, d. \tag{A.5}$$

We recall that $\{\lambda_\zeta^2\}_{\zeta=1}^{\ell}$ are the eigenvalues of G^2, thus, according to the spectral theorem,

$$\sum_{\zeta=1}^{\ell} \lambda_\zeta^2 \phi_\zeta\left(o_x^i\right) \phi_\zeta\left(o_y^j\right) = \left\langle g\left(o_x^i, \cdot\right), g\left(o_y^j, \cdot\right) \right\rangle, \quad x, y \in M,\ i, j = 1, \ldots, d, \tag{A.6}$$

since the right side of the equation is a cell in G^2. Therefore,

$$\begin{aligned} \left\| \Phi\left(o_x^i\right) - \Phi\left(o_y^j\right) \right\|^2 &= \left\langle \Phi\left(o_x^i\right), \Phi\left(o_x^i\right) \right\rangle - 2\left\langle \Phi\left(o_x^i\right), \Phi\left(o_y^j\right) \right\rangle + \left\langle \Phi\left(o_y^j\right), \Phi\left(o_y^j\right) \right\rangle \\ &= \left\langle g\left(o_x^i, \cdot\right), g\left(o_x^i, \cdot\right) \right\rangle - 2\left\langle g\left(o_x^i, \cdot\right), g\left(o_y^j, \cdot\right) \right\rangle + \left\langle g\left(o_y^j, \cdot\right), g\left(o_y^j, \cdot\right) \right\rangle \\ &= \left\| g\left(o_x^i, \cdot\right) - g\left(o_y^j, \cdot\right) \right\|^2, \end{aligned}$$

as stated in the lemma. □

References

[1] S. Mika, B. Schölkopf, A. Smola, K. Müller, M. Scholz, G. Rätsch, Kernel PCA and de-noising in feature spaces, in: Proceedings of the 1998 Conference on Advances in Neural Information Processing Systems, vol. II, MIT Press, Cambridge, MA, USA, 1999, pp. 536–542.
[2] B. Schölkopf, A. Smola, K. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Comput. 10 (1998) 1299–1319.
[3] R. Coifman, S. Lafon, Diffusion maps, Appl. Comput. Harmon. Anal. 21 (1) (2006) 5–30.
[4] A. Bermanis, A. Averbuch, R. Coifman, Multiscale data sampling and function extension, in: The 9th International Conference on Sampling Theory and Applications, 2011, Best student paper award in SampTA 2011, Applied and Computational Harmonic Analysis, submitted for publication.
[5] R. Coifman, S. Lafon, Geometric harmonics: a novel tool for multiscale out-of-sample extension of empirical functions, Appl. Comput. Harmon. Anal. 21 (1) (2006) 31–52.
[6] I. Jolliffe, Principal Component Analysis, Springer, New York, NY, 1986.
[7] H. Hotelling, Analysis of a complex of statistical variables into principal components, J. Educ. Psychol. 24 (1933) 417–441, doi:10.1037/h0071325.
[8] T. Cox, M. Cox, Multidimensional Scaling, Chapman and Hall, London, UK, 1994.
[9] J. Kruskal, Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis, Psychometrika 29 (1964) 1–27.
[10] S. Roweis, L. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323–2326.
[11] J. Tenenbaum, V. de Silva, J. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.
[12] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Comput. 15 (6) (2003) 1373–1396.
[13] D. Donoho, C. Grimes, Hessian eigenmaps: new locally linear embedding techniques for high dimensional data, Proc. Natl. Acad. Sci. USA 100 (2003) 5591–5596.

[14] G. Yang, X. Xu, J. Zhang, Manifold alignment via local tangent space alignment, in: International Conference on Computer Science and Software Engineering (1) 2008, pp. 928–931.
[15] Z. Zhang, 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.
[16] F. Chung, Spectral Graph Theory, CBMS, vol. 92, AMS, Providence, RI, 1997.
[17] A. Singer, H. Wu, Orientability and diffusion maps, Appl. Comput. Harmon. Anal. 31 (1) (2011) 44–58.
[18] A. Singer, H. Wu, Vector diffusion maps and the connection Laplacian, arXiv:1102.0075.
[19] A. Singer, Z. Zhao, Y. Shkolnisky, R. Hadani, Viewing angle classification of cryo-electron microscopy images using eigenvectors, SIAM J. Imaging Sci. 4 (2) (2011) 723–759.
[20] S. Agarwal, J. Lim, L. Zelnik-Manor, P. Perona, D. Kriegman, S. Belongie, Beyond pairwise clustering, in: IEEE Computer Society Conference on Computer Vision Pattern Recognition, vol. 2, 2005, pp. 838–845.
[21] A. Shashua, R. Zass, T. Hazan, Multi-way clustering using super-symmetric non-negative tensor factorization, in: A. Leonardis, H. Bischof, A. Pinz (Eds.), Computer Vision, ECCV 2006, in: Lecture Notes in Comput. Sci., vol. 3954, Springer, Berlin/Heidelberg, 2006, pp. 595–608.
[22] B. Nadler, S. Lafon, R. Coifman, I. Kevrekidis, Diffusion maps, spectral clustering and eigenfunctions of Fokker–Planck operators, in: Y. Weiss, B. Schölkopf, J. Platt (Eds.), Adv. Neural Inf. Process. Syst., vol. 18, MIT Press, Cambridge, MA, 2006, pp. 955–962.
[23] B. Nadler, S. Lafon, R. Coifman, I. Kevrekidis, Diffusion maps, spectral clustering and reaction coordinates of dynamical systems, Appl. Comput. Harmon. Anal. 21 (1) (2006) 113–127.
[24] S. Lafon, Diffusion maps and geometric harmonics, PhD thesis, Yale University, May 2004.
[25] G. Wolf, A. Averbuch, Linear-projection diffusion on smooth Euclidean submanifolds, Appl. Comput. Harmon. Anal., submitted for publication.
[26] H. Schwan, Alternating current spectroscopy of biological substances, Proc. Inst. Radio Eng. 47 (1959) 1841–1855.
[27] W. Kubicek, R. Patterson, D. Witsoe, Impedance cardiography as a noninvasive method of monitoring cardiac function and other parameters of the cardiovascular system, Ann. NY Acad. Sci. 170 (1970) 724–732.
[28] A. Frank, A. Asuncion, UCI machine learning repository, 2010; http://archive.ics.uci.edu/ml.
[29] J. Jossinet, Variability of impedivity in normal and pathological breast tissue, Med. Biol. Eng. Comput. 34 (1996) 346–350.
[30] J.E. da Silva, J.M. de Sá, J. Jossinet, Classification of breast tissue by electrical impedance spectroscopy, Med. Biol. Eng. Comput. 38 (2000) 26–30.
[31] C. Brice, C. Fennema, Scene analysis using regions, Artificial Intelligence 1 (3–4) (1970) 205–226.
[32] T. Pavlidis, Structural Pattern Recognition, Springer-Verlag, Berlin, New York, 1977.
[33] R. Haralick, L. Shapiro, Image segmentation techniques, Comput. Vis. Graphics Image Process. 29 (1) (1985) 100–132.
[34] A. Jain, R. Dubes, Algorithms for Clustering Data, Prentice-Hall, Englewood Cliffs, NJ, 1988.
[35] L. Kaufman, P. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis, John Wiley Sons, Hoboken, 1990.
[36] A. Jain, A. Topchy, M. Law, J. Buhmann, Landscape of clustering algorithms, in: International Conference on Pattern Recognition, ICPR, 2004, pp. 260–263.
[37] E. Navon, O. Miller, A. Averbuch, Color image segmentation based on adaptive local thresholds, Image Vis. Comput. 23 (1) (2005) 69–85.
[38] R. Ohlander, K. Price, D. Reddy, Picture segmentation using a recursive region splitting method, Comput. Graphics Image Process. 8 (3) (1978) 313–333.
[39] T. Pavlidis, Y. Liow, Integrating region growing and edge detection, IEEE Trans. Pattern Anal. Mach. Intell. 12 (3) (1990) 225–233.
[40] D. Mumford, J. Shah, Optimal approximations by piecewise smooth and variational problems, Comm. Pure Appl. Math. 42 (5) (1989) 577–685.
[41] M. Meila, J. Shi, Learning segmentation by random walks, in: Advances in Neural Information Processing Systems, NIPS 2000, vol. 13, 2001.
[42] P. Felzenszwalb, D. Huttenlocher, Efficient graph-based image segmentation, Int. J. Comput. Vis. 59 (2) (2004) 167–181.
[43] D. Cremers, M. Rousson, R. Deriche, A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape, Int. J. Comput. Vis. 72 (2) (2007) 195–215.
[44] Y. Boykov, V. Kolmogorov, An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, IEEE Trans. Pattern Anal. Mach. Intell. 26 (9) (2004) 1124–1137.
[45] C. Rother, V. Kolmogorov, A. Blake, "grabcut"—interactive foreground extraction using iterated graph cuts, ACM Trans. Graphics (Proc. SIGGRAPH) 23 (3) (2004) 309–314.
[46] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2004) 888–905.