Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2016) 24: 2374 – 2386

doi:10.3906/elk-1312-141

Ship-radiated noise feature extraction using multiple kernel graph embedding and auditory model

Xinzhou XU*, Xinwei LUO, Chen WU, Li ZHAO
Key Laboratory of Underwater Acoustic Signal Processing of Ministry of Education, Southeast University, Nanjing, P.R. China

Received: 20.12.2013 • **Accepted/Published Online:** 12.08.2014 • **Final Version:** 15.04.2016

Abstract: The analysis of underwater acoustic signals, especially ship-radiated noise received by passive sonar, is of great importance in the fields of defense, military, and scientific research. In this paper, we investigate multiple kernel learning graph embedding using auditory model features in the application of ship-radiated noise feature extraction. We use an auditory model to get auditory model features for each signal sample. In order to have more effective features, iterative multiple kernel learning methods are adopted to conduct dimensionality reduction. Validated by experiments, the proposed method outperforms ordinary kernel-based graph embedding methods. The experiments show that the multiple kernel learning method can automatically choose relatively appropriate kernel combinations in dimensionality reduction for ship-radiated noise using auditory model features. In addition, some worthwhile conclusions can be drawn from our experiments and analysis.

Key words: Graph embedding, multiple kernel learning, ship-radiated noise, feature extraction, auditory model

1. Introduction

Ship-radiated noise comes from propellers, machinery and hydrodynamic components, etc. Therefore, the noise is complex in both artificial and machine analysis. As a necessary step of signal analysis and target recognition, feature extraction of ship-radiated noise is important in both civilian and military fields. Many methods have been proposed to solve the problem [1–4], but they are either based on analyzing the spectrum of signals [1,2] or underwater physical characteristics [3,4]. Most parts of those algorithms relied on a priori information to a great extent, which could lead to difficulties in signal detection, feature extraction, and target recognition when the conditions change or the methods improve. For those reasons, learning methods and models should be conducted.

Some algorithms on manifold learning have appeared in academic research work in recent years. Taking locally linear embedding (LLE) [5], Laplacian eigenmaps [6], isomaps [7], and Hessian LLE [8] for example, we can find that those algorithms could solve a number of dimensionality reduction problems based on keeping structures of manifolds, which play a role in linking high-dimension and low-dimension space. Extensions connecting the stages of training and testing are then raised, such as neighborhood preserving embedding [9], locality preserving projections (LPPs) [10], and so on. Those methods were proved effective by experiments, which mostly appeared in the field of image and speech signal feature extraction and recognition. A graph embedding framework was built in [11]. This framework combined manifold learning with discriminant analysis

*Correspondence: radar256@gmail.com

methods. Marginal Fisher analysis (MFA) was proposed in the meantime to introduce marginal supervised factors and to overcome some faults (the number of projecting directions, etc.) in cultural discriminant analysis. Some algorithms related to graph embedding frameworks have been used in papers, such as locally discriminant embedding (LDE) [12] and semisupervised discriminant analysis (SDA) [13].

Kernel methods were originally adopted in support vector machines [14,15] to adapt the application of nonlinear classification. The utility of the kernel method was then extended to linear discriminant analysis (LDA) or Fisher discriminant analysis (FDA) [16–19] and principal component analysis (PCA) [18,19], named kernel Fisher discriminant analysis or kernel linear discriminant analysis (KLDA) [20,21] and kernel principal component analysis [19,22], respectively. Multiple kernel learning [23–26] is known as an improved form of kernel methods, enlarging the application region of ordinary algorithms. The optimization of determining the forms and parameters of kernels has turned into a promising field in machine learning.

In modeling the human auditory system [27–30], the modeling of the cochlea, including its basilar membrane and hair cells, can be regarded as a critical part of auditory modeling. Yang et al. [30] proposed a detailed model, which similarly achieved the simulation of the auditory mechanism, involving the modeling of the modules above and the lateral inhibition of the human nervous system.

A novel algorithm based on multiple kernel graph embedding and an auditory model is proposed to implement feature extraction and recognition of ship-radiated noise. We use an auditory model to achieve feature acquisition. Methods related to graph embedding with mapping according to multiple kernel learning are then adopted to carry out feature extraction. Compared with the works in [31–34], our method aims to extract effective factors with existing auditory model features, rather than exploring new categories of features.

The main contributions of this paper are as follows. An auditory model is used in underwater acoustic signal processing and effective features are generated; algorithms for a graph embedding framework are adopted to solve feature extraction of ship-radiated noise; and multiple kernel learning and its optimization are embedded in the mapping form of graph embedding, which can solve feature extraction of ship-radiated noise by dimensionality reduction.

2. Related work

2.1. Ship-radiated noise and auditory model

Studies of ship-radiated noise, as well as its feature extraction and target recognition, appeared when underwater crafts and ships were widely used in military and other aspects. Experts are able to deal with the signals with significant features; however, it is still difficult to analyze signals in complex conditions. These methods were mostly based on experiences of experts, though some of the methods are based on classic signal processing methods [1–4]. For these reasons, the applicability and automation of these methods could be relatively modest. Methods based on auditory perception mechanisms [31–34] were then proposed to solve the problems with the development of underwater acoustic signal processing and speech signal processing.

The auditory model can be seen as the modeling of the cochlea, including cochlea filters on the basilar membrane, the transduction of hair cells, and processing of auditory nerves. Those modules were researched respectively. Yang et al. proposed an integral computational model of the auditory system in [30]. Teolis et al. [31] adopted this auditory model and artificial neural networks to solve transient signal classification, which was proved effective. Some other researchers also focused on feature extraction and classification of underwater signals by auditory modeling from different points of view.

As to the employment of auditory mechanisms in analyzing underwater acoustic signal, Tucker et al.

[32,33] found the perceptual characteristics affecting underwater signals. They then proposed an algorithm to classify underwater signals by considering the features of timbre, material, and multiscale analysis.

2.2. Manifold learning and multiple kernel graph embedding

A manifold can be considered as a bridge connecting high-dimensional space and low-dimensional space. The goal of manifold learning is principally to find a manifold or manifolds, a fixed structure, between the two feature spaces. LLE [5] maintains a local linear structure, while isomapping [7] holds the global geodesic distances. Some other methods [8] keep different local models and build up final models by combining those local ones. Graph embedding was first proposed by Yan et al. [11], the form of which is similar to discriminant analysis or Rayleigh quotients.

The framework of graph embedding can be expressed as in Eq. (1):

$$\begin{aligned} &\arg\min_{y^T By=d} \sum_{i\neq j} \|y_i - y_j\|^2 W_{ij} = \arg\min_{y^T By=d} y^T Ly \\ &\left(B = L^p = D^p - W^p \text{ or } B = \Lambda,\ L = D - W,\ D^p_{ij} = \begin{cases} \sum\limits_{k=1}^{N} W^p_{ik}, & i = j \\ 0, & i \neq j \end{cases} \right., \\ &D_{ij} = \left. \begin{cases} \sum\limits_{k=1}^{N} W_{ik}, & i = j \\ 0, & i \neq j \end{cases} \right) \end{aligned} \tag{1}$$

where y_i is the feature vector of sample i after dimensionality reduction. W and W^p respectively represent the intrinsic graph and penalty graph adjacency matrix. L and L^p respectively represent the Laplacian matrix of W and W^p. Λ is a diagonal scaling matrix. N is the number of training samples. d is a positive constant value.

By graph embedding, PCA, LDA, LPP, SDA, etc. can be unified or transformed into this framework. The linearization, kernelization, and tensorization of them are also included in the framework. The differences of the graph embedding-related algorithms typically depend on the designing of graphs, including intrinsic graphs and penalty graphs. Therefore, supervising or other information, which can reflect the relationship between training samples, can be added by constructing proper graphs.

Multiple kernel learning [23–26] is often considered as the problem of how to choose the kernel or kernels that best fit current samples. The optimization results change when initial kernels or composite structures of kernels change. The linear combination of kernels [23–25] is now being focused on. As a kind of mapping, multiple kernel learning plays an important role in dimensionality reduction of feature space, for its nonlinear characteristics and wide applicability.

3. Algorithm

The goal of the algorithm is to get features that are optimum for classifying ship-radiated noise signals and corresponding objects of them. As represented in Figure 1, first of all, preprocessing is designed to be applied to underwater acoustic signals. An auditory model that can reconstruct input signals is adopted next. A processed 'frequency-discrete' short-time spectrum is then achieved, by which one can easily obtain different kinds of features to gain sufficient information for recognition. To complement dimensionality reduction as the dimensionality of feature space grows large, methods of graph embedding are also discussed here. By means of using training samples, mapping of dimensionality reduction will be generated for testing data. The whole

process is divided into two steps, an experience-dependent step and sample-dependent step, respectively on behalf of feature acquisition and dimensionality reduction.

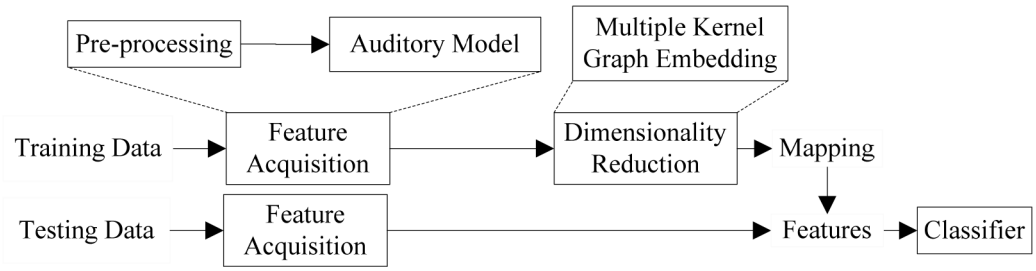

Figure 1. The process of the whole recognition system.

3.1. Feature acquisition

3.1.1. Auditory model

As in [30], the first step of the auditory model is to comply with the basilar membrane filters, which can simulate frequency decomposition of the human basilar membrane. Frequency distribution of the signal is in approximately logarithmic form, which can be simulated by a series of band-pass filters, constructed by wavelet or Gammatone filters.

$$y_1(t,s) = h(t,s) * x(t) \tag{2}$$

where $h(t,s)$represents band-pass constant-Q basilar membrane filters, with separate filters by different band characteristics. We use Gammatone filters [29] here to construct the membrane filters. Due to the manual auditory procedures in underwater target recognition, parameters in the filters are the same as in common auditory models.

The next step, as in Eq. (3), is energy conversion by inner hair cells. The step includes three parts, which are the output of stereocilia, nonlinear saturation characteristics, and the low-pass filter representing the process of the signal passing the hair cell membrane.

$$y_2(t,s) = g(\frac{\partial y_1(t,s)}{\partial t}) * w(t) \tag{3}$$

where $\frac{\partial y_1(t,s)}{\partial t}$ indicates the output of stereocilia. $g(\bullet)$ indicates a nonlinear function. The mapping $g(u) = \frac{1}{1+e^{-\gamma u}} - \frac{1}{2}$ is employed here. γ is the gain parameter. $w(t)$ represents the low-pass filter.

Processed by the steps above, a lateral inhibition network (LIN) is applied to be the later section of the auditory model. The LIN modeling achieves enhancement of transition and effective information (spatial differential term $\frac{\partial}{\partial s}$ and smoothing term $v(s)$ in Eq. (4)).

$$y_3(t,s) = \frac{\partial g(\frac{\partial y_1(t,s)}{\partial t})}{\partial s} *_t w(t) *_s v(s) = \frac{\partial g(\frac{\partial y_1(t,s)}{\partial t})}{\partial s} * w(t,s) \tag{4}$$

Eq. (5) successively represents the half-wave rectifier and the long-time constant integrator in the stage of LIN.

$$y_5(t,s) = \frac{1}{T}\int_{t-T}^{t} y_4(r,s)dr = \frac{1}{T}\int_{t-T}^{t} \max(y_3(r,s),0)dr \tag{5}$$

where T is the time length of integration. The entire model is shown as in Eq. (6).

$$y(t,s) = \frac{1}{T}\int_{t-T}^{t} \max(\frac{\partial g(\frac{\partial[h(r,s)*x(r)]}{\partial r})}{\partial s} *_r w(r) *_s v(s), 0)dr \tag{6}$$

The auditory model can be used as a process of acoustic signal reconstruction, by which signal enhancement and denoising, as well as feature acquisition, are able to be achieved. Similarly, the processing is useful in the conditions of underwater acoustic environments.

3.1.2. Feature acquisition

We first put the signal into the preprocess stage. The preprocessing procedure includes preemphasis and framing. A high-pass filter is utilized in the process of preemphasis. After preemphasis, a Hamming window is adopted in framing. The data are then normalized. After that, the preprocessed signal is input into the auditory model, which approximately imitates the human auditory mechanism, to achieve features.

Since a signal is usually composed of frames by framing, frame features belonging to every channel group are collected to do averaging. However, the output of the auditory model here is not the real spectrum because of the limited number of channels and those filters simulating the basilar membrane being of different frequency response characteristics of basilar membrane filters. Consequently, the output y_5 from each channel is adopted as final generated features by time integration.

3.2. Feature extraction based on multiple kernel graph embedding

Since the dimensionality of the initial feature space could be so high that it may cause the 'dimensionality curse', dimensionality reduction methods must be used. Having completed the procedures above, we bring methods based on graph embedding framework and multiple kernel learning [23–25] into dimensionality reduction.

3.2.1. Graph embedding framework

Graph embedding is a framework involving most existing dimensionality reduction methods, including some manifold learning methods. It is obvious from Eq. (1) that the formula can be represented as in Eq. (7) when B is not diagonal:

$$\arg\min_a \frac{S_c}{S_p} = \arg\min_a \frac{a^T X L X^T a}{a^T X B X^T a} = \arg\min_a \frac{a^T X (D - W) X^T a}{a^T X (D^p - W^p) X^T a} \tag{7}$$

where a stands for the mapping direction of the data. For training samples, the original sample set is $X = (x_1, x_2, ..., x_N)$. The mapping adopted in Eq. (7) is linear, while the form of kernelization is shown in Eq. (8):

$$\arg\min_\alpha \frac{S_c^\phi}{S_p^\phi} = \arg\min_\alpha frac(\phi(X)\alpha)^T \phi(X) L \phi^T(X)(\phi(X)\alpha)(\phi(X)\alpha)^T \phi(X) B \phi^T(X)(\phi(X)\alpha) = \arg\min_\alpha \frac{\alpha^T K L K \alpha}{\alpha^T K B K \alpha} \tag{8}$$

where $\phi(X) = (\phi(x_1), \phi(x_2), ..., \phi(x_N))$. Its column vector $\phi(x_i)$ means x_i mapping to reproducing kernel Hilbert space (RKHS), which is commonly high-dimensional.

One of the most indispensable parts of graph embedding is the designing of embedding graphs. Some methods' graphs are shown in Appendix A, with supervised or unsupervised information. It is worth noting

that an algorithm in accordance with graph embedding might have different forms, owing to optimization or some other reasons.

3.2.2. Multiple kernel graph embedding

Kernelization methods with a single kernel are sometimes not appropriate to describe data with complex distributions. Additionally, how to choose a kernel or kernel parameters could be another problem. Thus, multiple kernel learning [25] is utilized here. The linear type of kernel combination is frequently seen in different multiple kernel learning algorithms, because of its simple and direct form. When kernel $m \quad (m = 1, 2, ..., M)$ is adopted, we assume that the weighted high-dimensional feature vector of sample $x_i \quad (i = 1, 2, ..., N)$ is $\phi'_m(x_i) = \sqrt{\theta_m}\phi_m(x_i)$, where the parameter $\theta_m \geq 0$ and M is the number of different categories of kernels. The element of row i and column j in the Gram matrix is:

$$K_{ij} = k(x_i, x_j) = \sum_{m=1}^{M} \phi_m^{'T}(x_i)\phi'_m(x_j) = \sum_{m=1}^{M} \theta_m k_m(x_i, x_j) \tag{9}$$

This form looks clearly more acceptable than the average combination of kernels, every θ_m being equal to 1 or $1/M$. For sample x_i, its new feature space by multiple kernels is represented as:

$$a^T\phi(x_i) = \alpha^T\phi^T(X)\phi(x_i) = \alpha^T\Omega^{(i)}\Theta \quad (\Theta = (\theta_1, \theta_2, ..., \theta_M)^T) \tag{10}$$

where a is the linear mapping direction for RKHS samples. It can be represented by columns of $\phi(X)$. Suppose $A = (\alpha_1, \alpha_2, ..., \alpha_{n_r})$ is represented by vectors α, where n_r is the number of projection directions. For sample x_i, its multiple kernel matrix $\Omega^{(i)}$ is represented as in Eq. (11).

$$\Omega^{(i)} = \begin{bmatrix} K_1(1,i) & K_2(1,i) & \cdots & K_M(1,i) \\ K_1(2,i) & K_2(2,i) & \cdots & K_M(2,i) \\ \vdots & \vdots & \ddots & \vdots \\ K_1(N,i) & K_2(N,i) & \cdots & K_M(N,i) \end{bmatrix} \tag{11}$$

Hence, according to the primal form of graph embedding in Eq. (1), the optimization form of multiple kernel graph embedding can be expressed as in Eq. (12) [24,25], with multiple mapping directions.

$$\begin{aligned} &\arg\min_{A,\Theta} \sum_{i=1}^{N}\sum_{j=1}^{N} \left\|A^T\Omega^{(i)}\Theta - A^T\Omega^{(j)}\Theta\right\|^2 W_{ij} \\ &s.t. \quad \sum_{i=1}^{N}\sum_{j=1}^{N} \left\|A^T\Omega^{(i)}\Theta - A^T\Omega^{(j)}\Theta\right\|^2 W^p_{ij} = 1 \ , \ \theta_m \geq 0 \quad (m = 1, 2, ..., M) \end{aligned} \tag{12}$$

where W and W^p respectively represent intrinsic and penalty graphs. When the penalty graph is replaced by a scale transformation, the equality constraint in Eq. (12) changes, as Eq. (13), with L^p replaced by $B = \Lambda$.

$$\begin{aligned} &\arg\min_{A,\Theta} \sum_{i=1}^{N}\sum_{j=1}^{N} \left\|A^T\Omega^{(i)}\Theta - A^T\Omega^{(j)}\Theta\right\|^2 W_{ij} \\ &s.t. \quad \sum_{i=1}^{N} \left\|A^T\Omega^{(i)}\Theta\right\|^2 B_{ii} = \left\|A^T(\sum_{i=1}^{N}\rho_i\Omega^{(i)})\Theta\right\|^2 = 1 \ , \ \theta_m \geq 0 \ (B = \Lambda = diag(\rho_1, \rho_2, ..., \rho_N)) \end{aligned} \tag{13}$$

where $m = 1, 2, ..., M$. A method was proposed in [24,25] based on the alternative optimization of the two variables, A and Θ, in the graph embedding framework.

Optimization of A:

When $B = L^p$, the optimization is shown as in Eq. (14).

$$\begin{aligned} &\arg\min_{A} \; tr[A^T(\sum_{i=1}^{N}\sum_{j=1}^{N}(\Omega^{(i)}-\Omega^{(j)})\Theta\Theta^T(\Omega^{(i)}-\Omega^{(j)})^T W_{ij})A] \\ &s.t. \quad tr[A^T(\sum_{i=1}^{N}\sum_{j=1}^{N}(\Omega^{(i)}-\Omega^{(j)})\Theta\Theta^T(\Omega^{(i)}-\Omega^{(j)})^T W_{ij}^p)A] = 1 \end{aligned} \tag{14}$$

The solution of Eq. (14) is easy to draw, as Eq. (15).

$$\begin{aligned} &(\sum_{i=1}^{N}\sum_{j=1}^{N}(\Omega^{(i)}-\Omega^{(j)})\Theta\Theta^T(\Omega^{(i)}-\Omega^{(j)})^T W_{ij})\alpha \\ &= \lambda(\sum_{i=1}^{N}\sum_{j=1}^{N}(\Omega^{(i)}-\Omega^{(j)})\Theta\Theta^T(\Omega^{(i)}-\Omega^{(j)})^T W_{ij}^p)\alpha \end{aligned} \tag{15}$$

When $B = \Lambda$, similarly, the solution is according to Eq. (16).

$$(\sum_{i=1}^{N}\sum_{j=1}^{N}(\Omega^{(i)}-\Omega^{(j)})\Theta\Theta^T(\Omega^{(i)}-\Omega^{(j)})^T W_{ij})\alpha = \lambda(\sum_{i=1}^{N}\Omega^{(i)}\Theta\Theta^T\Omega^{(i)T}B_{ii})\alpha \tag{16}$$

Optimization of Θ:

When $B = L^p$, the optimization of Θ is shown as in Eq. (17).

$$\begin{aligned} &\arg\min_{\Theta} \; \Theta^T(\sum_{i=1}^{N}\sum_{j=1}^{N}(\Omega^{(i)}-\Omega^{(j)})^T AA^T(\Omega^{(i)}-\Omega^{(j)})W_{ij})\Theta \\ &s.t. \quad \Theta^T(\sum_{i=1}^{N}\sum_{j=1}^{N}(\Omega^{(i)}-\Omega^{(j)})^T AA^T(\Omega^{(i)}-\Omega^{(j)})W_{ij}^p)\Theta = 1 \;\; , \quad \theta_m \geq 0 \quad (m = 1, 2, ..., M) \end{aligned} \tag{17}$$

Since Eq. (17) is a quadratically constrained quadratic program nonconvex problem, by adding an auxiliary part [24,25], the optimization of Eq. (17) changes into Eq. (18), as the semidefinite programming (SDP) form, which can be solved according to the common SDP method. The semidefinite relaxations of this problem were given in [35].

$$\begin{aligned} &\arg\min_{\Theta,P} \; tr[P(\sum_{i=1}^{N}\sum_{j=1}^{N}(\Omega^{(i)}-\Omega^{(j)})^T AA^T(\Omega^{(i)}-\Omega^{(j)})W_{ij})] \\ &s.t. \quad tr[P(\sum_{i=1}^{N}\sum_{j=1}^{N}(\Omega^{(i)}-\Omega^{(j)})^T AA^T(\Omega^{(i)}-\Omega^{(j)})W_{ij}^p)] = 1, \;\; \theta_m \geq 0, \; \begin{bmatrix} P & \Theta \\ \Theta^T & 1 \end{bmatrix} \succeq 0 \end{aligned} \tag{18}$$

where $m = 1, 2, ..., M$. The same as in the discussion above, when B is diagonal, the optimization is easy to draw and thus is not mentioned here. The SDP problems of Eq. (18) can be solved by either the existing toolbox or by implementing the module by programming.

4. Experiments

4.1. Preparations

According to the algorithm above, features are generated by the auditory model and its auxiliary courses. The number of the filters of the auditory model here is fixed as 64, which means that the dimensionality of generated features is 63. The filers are selected as Gammatone filters [29], using the toolbox of [36].

For SDA, the parameter τ in Appendix A is chosen as c/N, where c is the number of classes. For LPP, MFA (LDE), and SDA, the number of neighbors is determined according to the number of training samples. It is worth noting that neighbors in kernelization, average multiple kernels, and multiple kernel learning are determined by distances of each two feature vectors in RKHS with high dimensionality, for example, in the kernelization of MFA, LPP, and SDA. The distances can be represented by elements of Gram matrices.

The kernels of the multiple kernel methods are three Gaussian kernels with different parameters, as was proposed in [23]. In multiple kernel learning, the kernelization and average kernelization separately means a single Gaussian kernel and several kernels combined with the same weight. Multiple kernel learning graph embedding aims to achieve a relatively appropriate linear combination of different kinds of kernels in the graph embedding framework. The generalized eigenvalue problems for kernelization and the multiple kernel condition can be solved according to Appendix B.

The iteration process of the optimization by multiple kernel learning can start from either the mapping of dimensionality reduction, $A = (\alpha_1, \alpha_2, ..., \alpha_{n_r})$, or the linear combination coefficients, $\Theta = (\theta_1, \theta_2, ..., \theta_M)^T$. The initial values of A and Θ are chosen merely as was described in [24,25]. The classifiers adopted in the experiments are k-nearest neighbor [18,19] classifiers. The performances of feature extraction algorithms can be roughly described in that way.

4.2. Experiments on simulation data

The simulation database contains 3 categories' ship-radiated signal segments. It simulates radiated noise from different kinds of ships, with differences in propeller speed modulation, propeller blades, propeller cavitation noise, and some other aspects. Additionally, the difference between every two kinds of ship signals is set to be small in this database. The number of training and testing samples, which are randomly selected from the original database, are respectively 120 and 900. Both in training and testing data sets, every class shares the same number of samples. We repeat the experiments with random choices of training and testing samples.

Figure 2 shows recognition rates of KLDA and KSDA when the dimensions change, as well as their average multiple kernel combination form and multiple kernel learning form. For convenience, the three kinds of mapping are separately written as K-Map (kernel mapping), AMK-Map (average multiple kernel mapping), and MKL-Map (multiple kernel learning mapping). The two subfigures show the contrast of K-Map, AMK-Map, and MKL-Map according to different embedding graphs above. Table 1 shows the best low-dimensional recognition rates of LPP, MFA, LDA, and SDA in the conditions of different combinations of kernels.

It can be concluded from Figure 2 and Table 1 that the graph embedding algorithms with multiple kernel learning perform comparatively better than ordinary kernel graph embedding methods in most conditions, especially for low-dimensional conditions of KLDA and KSDA. From Figure 2, when the kernels have different parameters, it is obvious that MKL-Map methods usually 'trail' the comparatively better kernels.

Nevertheless, sometimes it is not obvious to find the differences of performance amount for K-Map, AMK-Map, and MKL-Map. Disturbances when the generalized eigenvalue problems are solved, as well as calculation accuracy, the objective function of optimization, and other factors, could lead to the problem.

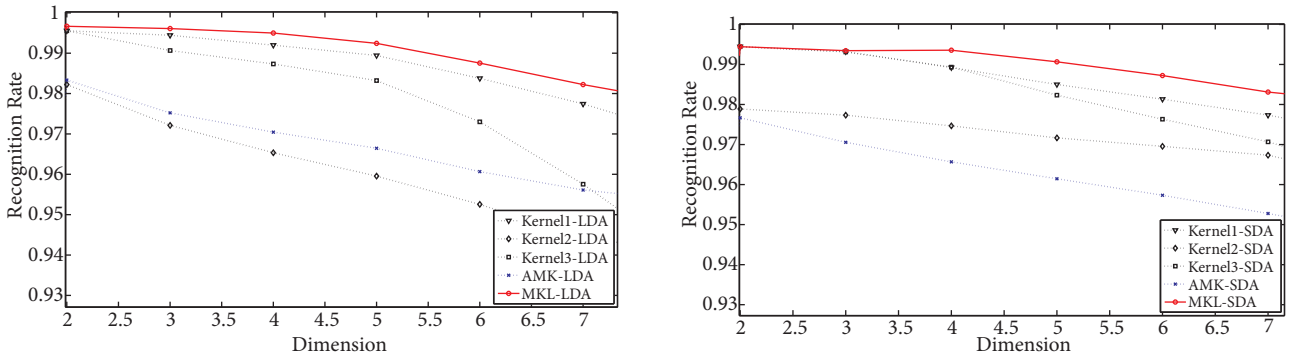

Figure 2. Recognition rates on simulation database in low-dimension space. (a) KLDA, AMK-LDA, and MKL-LDA. (b) KSDA, AMK-SDA, and MKL-SDA.

Table 1. The best recognition rates (in bold) in low-dimension space of simulation data (%).

Methods	Kernel 1	Kernel 2	Kernel 3	AMK	MKL
LPP	81.96	**89.60**	76.46	88.71	89.21
MFA (LDE)	97.78	95.66	96.93	94.79	**98.96**
LDA (FDA)	99.56	98.22	99.54	98.33	**99.67**
SDA	**99.44**	97.89	99.41	97.67	**99.44**

4.3. Experiments on real data

The real database includes 6 categories' ship propeller-radiated signals, which are achieved according to underwater measurement. The samples are from different ships and different underwater acoustic source conditions. The measurements are conducted in a lake by passive sonar. We obtain the data from various types of ships in nearly the same outside conditions of wind speed, temperature, weather, etc. In the experiments on simulation data, the numbers of training samples and testing samples are respectively 189 and 747 with random selections here, also with repeated experiments

Figures 3a and 3b show 6 frames' original signal and auditory model features respectively from those 6 categories' samples, which are collected in an underwater environment and have already been preprocessed as well as normalized. The horizontal axis of Figure 3b shows the corresponding feature marks in the horizontal axis. Figure 4 is similar to Figure 2, showing the unweighted recognition rates of different dimensions for different embedding graphs and kernelization mappings.

It can be learned from Figure 4 and Table 2 that the close recognition rates between supervised and unsupervised methods show the validity of auditory model features from a certain point. The baseline recognition rate with original auditory model features is only 78.20%, which means that dimensionality reduction can improve the performance.

Table 2. The best recognition rates (in bold) in low-dimension space of real data (%).

Methods	Kernel 1	Kernel 2	Kernel 3	AMK	MKL
LPP	85.23	83.99	85.74	83.94	**86.52**
MFA (LDE)	85.58	85.41	86.21	85.39	**86.32**
LDA (FDA)	85.77	85.72	85.97	84.81	**86.59**
SDA	84.93	84.67	**85.89**	84.42	85.53

To combine Figure 2, Figure 4, Table 1, and Table 2, the multiple kernel learning graph embedding methods track the relatively better mapping directions as the number of dimensions changes in most instances. However, the methods are not always pleasing, owing to the objective functions in the iteration process and the number of alternative kernels, as well as the number of training samples' classes. The upgrades of effects for some methods are not so obvious.

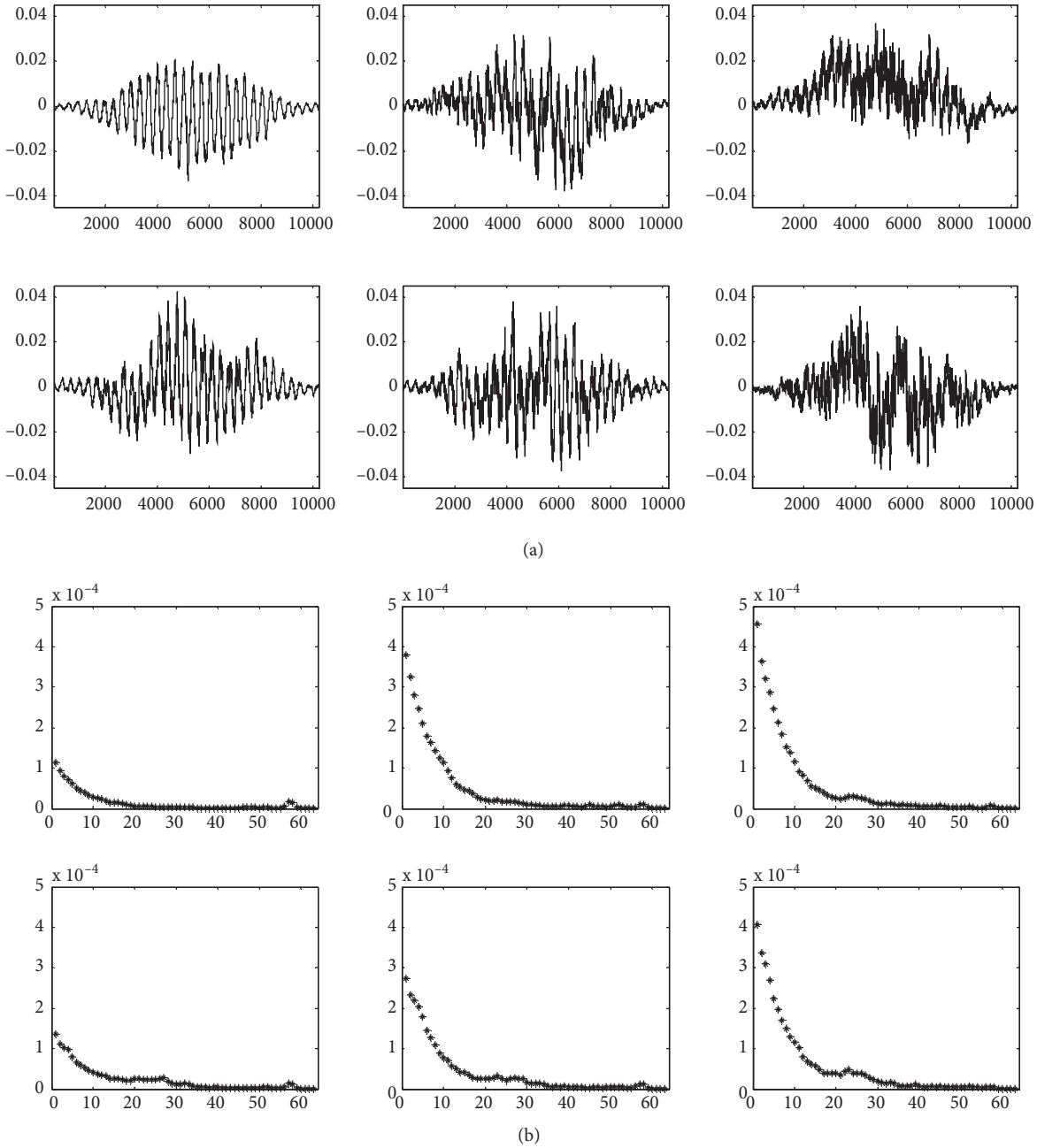

Figure 3. (a) Original ship-radiated signal frames from 6 categories. (b) Auditory model features respectively from 6 categories of ship-radiated noise.

According to [25], for image signal processing, the performance of multiple kernel learning graph embedding algorithms is able to exceed that of all the alternative kernels' methods. However, the performance of multiple kernel learning could not achieve that level here for underwater acoustic signals. Concluding the possible reasons for this, we deduce that the computation process of the objective function, the choices of kernels, and characteristics of different kinds of signals could be the key causes.

From the view of optimization, the objective functions are often changed when the generalized eigenvalue problems are solved to constrain the small eigenvalues in kernel conditions according to Appendix B. This could affect the performance of multiple kernel learning. Therefore, new optimization designs should be adopted when the mapping directions are solved in multiple kernel graph embedding. In addition, the mapping directions are not originally orthogonal between each other. This is another factor influencing the performance of feature extraction.

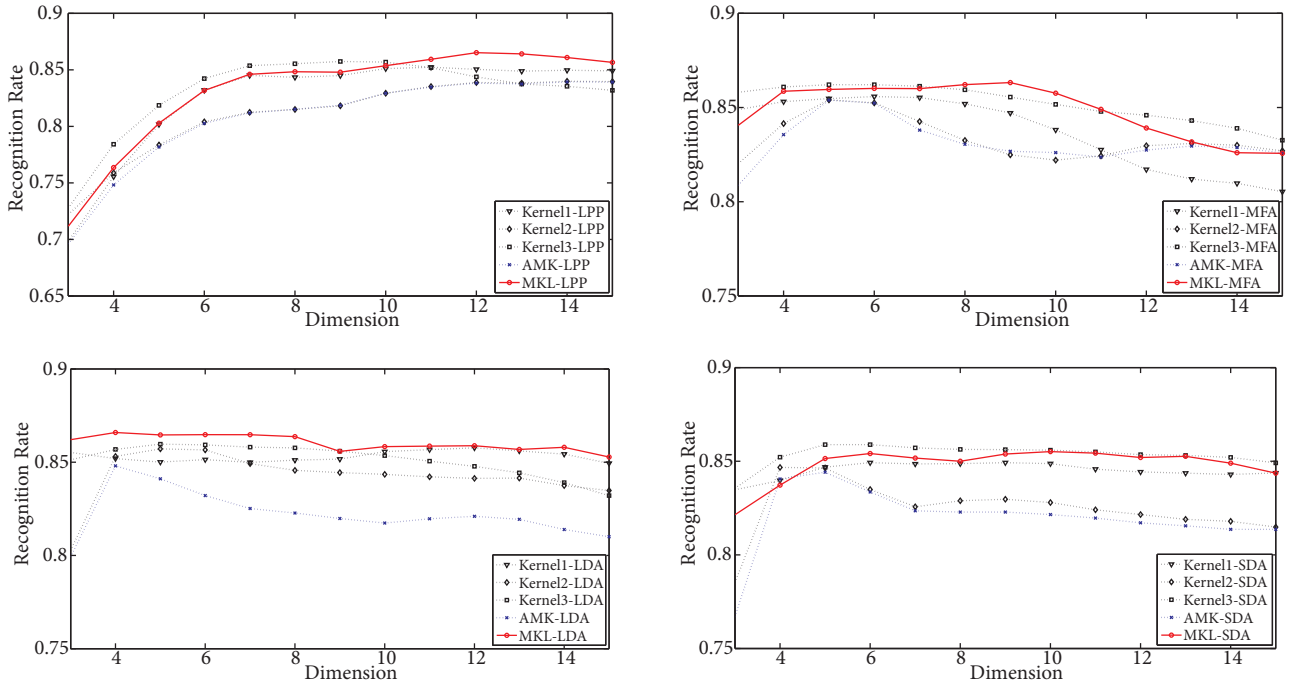

Figure 4. Recognition rates on real database in low-dimension space. (a) KLPP, AMK-LPP, and MKL-LPP. (b) KMFA, AMK-MFA, and MKL-MFA. (c) KLDA, AMK-LDA, and MKL-LDA. (d) KSDA, AMK-SDA, and MKL-SDA.

5. Conclusions and future work

We use multiple kernel learning and an auditory model to solve the feature extraction problem in underwater ship-radiated noise analysis in this paper. For each sample, original features are first generated by the auditory model. We then adopt multiple kernel learning in a graph embedding framework to achieve dimensionality reduction for the features. By these procedures, we can extract effective features for ship-radiated noise. Compared with the existing spectrum analysis-based methods, the proposed method is valid, especially when the underwater conditions change, and it can be wholly automatic in the task of underwater acoustic signal analysis and classification of categories of ships. The experiments on both simulation and real data also validate our algorithm by comparing it with the ordinary kernelized graph-based methods, which only solve optimization problems by controlling one set of parameters.

However, some aspects below, which may affect the performance of the method in this paper, are worth being considered. First, we only choose a small number of Gaussian kernels in the experiments by our experience. Choosing appropriate kernels for the multiple kernel matrix can greatly raise the performance. Second, the objective functions in this paper are usually not so proper to describe data relations due to the inaccurate embedding graphs. Selecting a reasonable graph will lead the iterative optimization into a correct direction. Finally, the optimization goal keeps changing in each iterative step in multiple kernel learning, because of the computational errors in solving the generalized eigenvalue problem when kernelization is adopted. We can change optimization methods or reduce the computational errors in our future research.

Acknowledgments

This work was supported by the Natural Science Foundation of China under Grant No. 61231002 and No. 61273266, and the Doctoral Fund of the Ministry of Education of China under Grant No. 20110092130004.

References

[1] Lourens J, Coetzer M. Detection of mechanical ship features from underwater acoustic sound. In: IEEE Conference on Acoustics, Speech and Signal Processing; 6–9 April 1987; Dallas, TX, USA. New York, NY, USA: IEEE. pp. 1700-1703.

[2] Lourens J. Classification of ships using underwater radiated noise. In: IEEE Conference on Communication and Signal Processing; 24 June 1988; Pretoria, South Africa. New York, NY, USA: IEEE. pp. 130-134.

[3] Seol H, Suh J, Lee S. Development of hybrid method for the prediction of underwater propeller noise. J Sound Vib 2005; 288: 345-360.

[4] Arveson P, Vendittis D. Radiated noise characteristics of a modern cargo ship. J Acoust Soc Am 2000; 107: 118-129.

[5] Roweis S, Saul L. Nonlinear dimensionality reduction by locally linear embedding. Science 2000; 290: 2323-2326.

[6] Belkin M, Niyogi P. Laplacian eigenmaps and spectral techniques for embedding and clustering. In: Advances in Neutral Information Processing Systems; 3–8 December 2001; Vancouver, Canada. pp. 585-591.

[7] Tenenbaum J, de Silva V, Langford J. A global geometric framework for nonlinear dimensionality reduction. Science 2000; 290: 2319-2323.

[8] Donoho D, Grimes C. Hessian eigenmaps: locally linear embedding techniques for high-dimensional data. P Natl Acad Sci USA 2003; 100: 5591-5596.

[9] He X, Cai D, Yan S, Zhang H. Neighborhood preserving embedding. In: IEEE Conference on Computer Vision; 15–21 October 2005; Beijing, China. New York, NY, USA: IEEE. pp. 1208-1213.

[10] He X, Niyogi P. Locality preserving projections. In: Advances in Neural Information Processing Systems; 13–18 December 2004; Whistler, Canada. pp. 153-160.

[11] Yan S, Xu D, Zhang B, Zhang H, Yang Q, Lin S. Graph embedding and extensions: a general framework for dimensionality reduction. IEEE T Pattern Anal 2007; 29: 40-51.

[12] Chen H, Chang H, Liu T. Local discriminant embedding and its variants. In: IEEE Conference on Computer Vision and Pattern Recognition; 20–26 June 2005; San Diego, CA, USA. New York, NY, USA: IEEE. pp. 846-853.

[13] Cai D, He X, Han J. Semi-supervised discriminant analysis. In: IEEE Conference on Computer Vision; 14–20 October 2007; Rio de Janeiro, Brazil. New York, NY, USA: IEEE. pp. 1-7.

[14] Cortes C, Vapnik V. Support vector networks. Mach Learn 1995; 20: 273-297.

[15] Smola A. Learning with kernels. PhD, GMD, Birlinghoven, Germany, 1998.

[16] Fisher R. The use of multiple measurements in taxonomic problems. Ann Hum Genet 1936; 7: 179-188.

[17] Fisher R. The statistical utilization of multiple measurements. Ann Hum Genet 1938; 8: 376-386.

[18] Theodoridis S, Koutroumbas K. Pattern Recognition. New York, NY, USA: Elsevier, 2006.

[19] Bishop C. Pattern Recognition and Machine Learning. New York, NY, USA: Springer, 2007.

[20] Mika S, Rätsch G, Weston J, Schölkopf B, Müller K. Fisher discriminant analysis with kernels. In: IEEE Neural Networks for Signal Processing Workshop; 25 August 1999; Madison, WI, USA. New York, NY, USA: IEEE. pp. 41-48.

[21] Yang J, Frangi A, Yang J, Zhang D, Jin Z. KPCA plus LDA: a complete kernel Fisher discriminant framework for feature extraction and recognition. IEEE T Pattern Anal 2005; 27: 230-244.

[22] Yang M, Ahuja N, Kriegman D. Face recognition using kernel eigenfaces. In: IEEE Conference on Image Processing; 10–13 September 2000; Vancouver, Canada. New York, NY, USA: IEEE. pp. 37-40.

[23] Kim S, Magnani A, Boyd S. Optimal kernel selection in kernel fisher discriminant analysis. In: IEEE Conference on Machine Learning; 25–29 June 2006; Pittsburgh, PA, USA. New York, NY, USA: IEEE. pp. 465-472.

[24] Lin Y, Liu T, Fuh C. Dimensionality reduction for data in multiple feature representations. In: Advances in Neural Information Processing Systems; 6–8 December 2009; Vancouver, Canada. pp. 961-968.

[25] Lin Y, Liu T, Fuh C. Multiple kernel learning for dimensionality reduction. IEEE T Pattern Anal 2011; 33: 1147-1160.

[26] Wang Z, Chen S, Sun T. MultiK-MHKS: a novel multiple kernel learning algorithm. IEEE T Pattern Anal 2008; 30: 348-353.

[27] Quatieri T. Discrete-Time Speech Signal Processing: Principles and Practice. New York, NY, USA: Pearson Education, 2008.

[28] Lyon R, Mead C. An analog electronic cochlea. IEEE T Acoust Speech 1988; 36: 1119-1134.

[29] Patterson R, Nimmo-Smith I, Holdsworth J, Rice P. An efficient auditory filterbank based on the gammatone function. In: A Meeting of the IOC Speech Group on Auditory Modelling at RSRE; 1987.

[30] Yang X, Wang K, Shamma S. Auditory representations of acoustic signals. IEEE T Inform Theory 1992; 38: 824-839.

[31] Teolis A, Shamma S. Classification of the Transient Signals via Auditory Representations. Tech Rep TR91-99. College Park, MD, USA: Institute for Systems Research, 1991.

[32] Tucker S, Brown G. Classification of transient sonar sounds using perceptually motivated features. IEEE J Oceanic Eng 2005; 30: 588-600.

[33] Tucker S. Auditory analysis of sonar signals. PhD transfer report. Sheffield, UK: University of Sheffield, 2001.

[34] Parks T, Weisburn B. Classification of whale and ice sounds with a cochlear model. In: IEEE Conference on Acoustics, Speech and Signal Processing; 23–26 March 1992; San Francisco, CA, USA. New York, NY, USA: IEEE. pp. 481-484.

[35] d'Aspremont A, Boyd S. Relaxations and Randomized Methods for Nonconvex QCQPs. Tech. Rep. Stanford, CA, USA: Stanford University, 2003.

[36] Slaney M. Auditory Toolbox: A MATLAB Toolbox for Auditory Modeling Work (Version 2). Tech Rep 1998: #1998-010. West Lafayette, IN, USA: Purdue University, 1998.

Appendix A

The choices of embedding graphs are provided below, with the adjacency matrices W^{LPP}, W^{LDA}, W^{MFA}, W^{LDE}, and W^{SDA} for LPP, LDA, MFA, LDE, and SDA respectively.

LPP [6,10]:

$$W_{ij}^{LPP} = \begin{cases} 1 \text{ or } e^{-\frac{\|x_i - x_j\|^2}{t}}, & i \in N_{k_1}(j) \text{ or } j \in N_{k_1}(i) \\ 0, & otherwise \end{cases}, \quad B^{LPP} = D^{LPP} \left(D_{ij}^{LPP} = \begin{cases} \sum_{k=1}^{N} W_{ik}^{LPP}, & i = j \\ 0, & i \neq j \end{cases}\right) \tag{19}$$

where the parameter $t > 0$. $N_{k_1}(j)$ is the k_1-neighboring of sample j.

LDA [11,16–19]: Here we talk about a multiclass situation instead of a two-class situation.

$$\begin{cases} L^{LDA} = I - \sum_{c=1}^{N_C} \frac{1}{n_c} e^c e^{cT} \Leftarrow W^{LDA} = \sum_{c=1}^{N_C} \frac{1}{n_c} e^c e^{cT} \\ L^{pLDA} = B^{LDA} = I - \frac{1}{N} e e^T \Leftarrow W^{pLDA} = \frac{1}{N} e e^T \end{cases} \tag{20}$$

where e^c represents the column vectors with the elements corresponding to class c being equal to 1. e is a column vector with all elements equal to 1. n_c is the number of samples in class c. N_c is the number of classes.

MFA [11]:

$$W_{ij}^{MFA} = \begin{cases} 1, & i \in N_{k_1}(j) \quad or \quad j \in N_{k_1}(i) \\ 0, & otherwise \end{cases}, \quad W_{ij}^{pMFA} = \begin{cases} 1, & i \in N_{k_2}^{-}(j) \quad or \quad j \in N_{k_2}^{-}(i) \\ 0, & otherwise \end{cases} \tag{21}$$

where $N_{k_1}(i)$ is the set of x_i's k_1 nearest neighbors. $j \in N_{k_2}^{-}(i)$ represents the set for which x_j belongs to x_i's k_2 nearest neighbors, while x_i and x_j are of two different classes.

LDE [12]: The optimization form of LDE is equal to that of MFA by using $W_{ij}^{LDE} = W_{ij}^{MFA} - W_{ij}^{pMFA}$ only when $k_1 = k_2$.

SDA [13,25]: For the special circumstance that all of the training samples are labeled, being similar to the form of LDA, the supervised form of SDA is represented as in Eq. (22).

$$\begin{aligned} &\arg\max_a \frac{a^T S_b a}{a^T (S_t + \tau X L X) a} = \arg\min_a \frac{a^T (S_w + \tau X L X^T) a}{a^T (S_t + \tau X L X^T) a} \\ &= \arg\min_a \frac{a^T X[(I + \tau D) - (\sum_{c=1}^{N_C} \frac{1}{n_c} e^c e^{cT} + \tau S)] X^T a}{a^T X[(I + \tau D) - (\frac{1}{N} e e^T + \tau S)] X^T a} \end{aligned} \tag{22}$$

where S_w is a within-class scatter matrix, while S_b is a between-class scatter matrix. $S_t = S_b + S_w$. Parameter $\tau \geq 0$. L here is Laplacian matrix of S:

$$\begin{aligned} &L = D - S \\ &S_{ij} = \begin{cases} 1, & i \in N_k(j) \quad or \quad j \in N_k(i) \\ 0, & otherwise \end{cases}, \quad D_{ij} = \begin{cases} \sum_{k=1}^{N} S_{ik}, & i = j \\ 0, & i \neq j \end{cases} \end{aligned} \tag{23}$$

From Eqs. (22) and (23), τ can be considered as the parameter connecting neighboring and label information.

It can be concluded from Eq. (22) that the graph is as shown in Eq. (24).

$$W^{SDA} = \sum_{c=1}^{N_C} \frac{1}{n_c} e^c e^{cT} + \tau S, \quad W^{pSDA} = \frac{1}{N} e e^T + \tau S \tag{24}$$

Suppose there are l labeled samples. Then we generalize the method to a semisupervised condition, as is stated in Eq. (25).

$$\tilde{W}^{SDA} = \begin{bmatrix} (\sum_{c=1}^{N_C} \frac{1}{n_c} e^c e^{cT})_{l\times l} & 0 \\ 0 & 0 \end{bmatrix}_{N\times N} + \tau S_{N\times N}, \quad \tilde{W}^{pSDA} = \begin{bmatrix} (\frac{1}{N} e e^T)_{l\times l} & 0 \\ 0 & 0 \end{bmatrix}_{N\times N} + \tau S_{N\times N} \tag{25}$$

Appendix B

The solution method of Eqs. (15) and (16) can be written as $KLK = \lambda KBK\alpha$. K of the equation is just like in Eq. (9) of the body part, as well as in ordinary kernelization methods. Since K is a symmetric matrix, it can be decomposed into $U_K D_K U_K^T$ by SVD decomposition. The matrix D_K can be seen as a partitioned matrix including large eigenvalue part D'_K and small eigenvalue part D''_K. The corresponding eigenvectors' matrix is U'_K and U''_K.

$$K = U_K D_K U_K^T \approx U'_K D'_K U_K^{'T} = \tilde{K} \quad (D_K = \begin{bmatrix} D'_K & 0 \\ 0 & D"_K \end{bmatrix}, U_K = \begin{bmatrix} U'_K & U"_K \end{bmatrix}) \tag{26}$$

The generalized eigenvalue problem is converted into:

$$U'_K D'_K (U_K^{'T} L U'_K) D'_K U_K^{'T} \alpha = \lambda U'_K D'_K (U_K^{'T} B U'_K) D'_K U_K^{'T} \alpha \tag{27}$$

The two sides are simultaneously multiplied by $D_K^{'-1} U_K^{'T}$:

$$(U_K^{'T} L U'_K)(D'_K U_K^{'T} \alpha) = \lambda (U_K^{'T} B U'_K)(D'_K U_K^{'T} \alpha) \tag{28}$$

Suppose $\alpha = U'_K D_K^{'-1} \beta$. The equation can be written as:

$$(U_K^{'T} L U'_K)\beta = \lambda (U_K^{'T} B U'_K)\beta \Rightarrow \tilde{L}\beta = \lambda \tilde{B}\beta \tag{29}$$

Since the ranks of $\tilde{L}$ and $\tilde{B}$ can be approximately considered as full rank matrices, $\tilde{B}$ can be decomposed as: $\tilde{B} = U_{\tilde{B}} D_{\tilde{B}}^{\frac{1}{2}} D_{\tilde{B}}^{\frac{1}{2}} U_{\tilde{B}}^T$. Then $D_{\tilde{B}}^{-\frac{1}{2}} U_{\tilde{B}}^T \tilde{L} \beta = \lambda D_{\tilde{B}}^{\frac{1}{2}} U_{\tilde{B}}^T \beta$.

Suppose $\beta = U_{\tilde{B}} D_{\tilde{B}}^{-\frac{1}{2}} \gamma$. Then $D_{\tilde{B}}^{-\frac{1}{2}} U_{\tilde{B}}^T \tilde{L} U_{\tilde{B}} D_{\tilde{B}}^{-\frac{1}{2}} \gamma = \lambda \gamma$.

The mapping of kernelization problems is consequently $\alpha = U'_K D_K^{'-1} U_{\tilde{B}} D_{\tilde{B}}^{-\frac{1}{2}} \gamma$