Robust Kernel Principal Component Analysis With $\ell_{2,1}$-Regularized Loss Minimization

DUO WANG AND TOSHIHISA TANAKA, (Senior Member, IEEE)
Department of Electronic and Information Engineering, Tokyo University of Agriculture and Technology, Tokyo 184-8588, Japan
Corresponding author: Toshihisa Tanaka (tanakat@cc.tuat.ac.jp)

This work was supported in part by JSPS KAKENHI Grant Number 17H01760.

ABSTRACT Principal component analysis (PCA) is a widely used unsupervised method for dimensionality reduction. The kernelized version is called kernel principal component analysis (KPCA), which can capture the nonlinear data structure. KPCA is derived from the Gram matrix, which is not robust when outliers exist in the data. This may yield the principal axis in the feature space deviated by outliers, leading to misinterpretation of the principal components. In this paper, we propose a robust method for KPCA with a reformulation in Euclidean space to construct a robust KPCA method, where an error measurement is introduced into the loss function, and $\ell_{2,1}$-regularization is added to the loss function. The idea of $\ell_{2,1}$-regularization of the proposed method is motivated by sparse PCA via variable projection. However, because orthogonality is not satisfied in the proposed method, orthonormal bases are obtained by using the Gram-Schmidt orthonormalization process. In the experiments, a toy example and real data are used for outlier detection to verify the method’s performance and effectiveness. In the toy example, the proposed method reduces the influence of outliers and detects more outliers than KPCA. For the real data, the proposed method improves detection in comparison to other existing methods.

INDEX TERMS Kernel principal component analysis, $\ell_{2,1}$ regularization, outlier detection, robustness.

I. INTRODUCTION
Principal component analysis (PCA) [1] is a linear feature extraction method for reducing dimensions from possibly high-dimensional data. A kernel extension of PCA for capturing the nonlinear feature is kernel principal component analysis (KPCA) [2], where data are mapped into the feature space through nonlinear mapping. Instead of replacing nonlinear mapping, the kernel trick is used to calculate the inner product. KPCA has been successfully applied in many applications, such as fault detection [3], novelty detection [4], outlier detection [5], and so on. It is known that PCA can be obtained from the maximum variance criterion or the mean squared error (MSE) criterion. However, no matter what derivation is used, PCA is sensitive to, or not robust against, outliers. For instance, outliers with large squared errors can dominate in MSE and lead to misleading interpretations. The same problem occurs in KPCA.

To solve this problem, many researchers have proposed robust KPCA to mitigate this weakness. The robust KPCA method can be divided into five categories. The first category for robustifying KPCA method is based on the robust estimator of the Gram matrix. The rationale for this method is that the zero mean is not a robust measure of the true center when outliers exist in the data. Therefore, this method has been proposed to approximate the robust version of the covariance matrix [6]. In [6], a robust principal component (PC) can be obtained from the eigendecomposition of the robustness covariance matrix. There are many robust estimators for estimating multivariate location and scatter in the presence of outliers from a statistical perspective, such as the M estimator [6], the minimum covariance determinant method [7], and so on. An overview of several robust estimation methods is presented by Rousseeuw and Hubert [8]. Debruyne and Verdonck [9] presented three robust KPCA algorithms: spherical KPCA, kernel projection pursuit (KPP), and kernel ROBPCA, which are the kernel versions of sphere PCA [10], projection pursuit [11], and robust PCA (ROBPCA) [12], respectively. Spherical KPCA calculates the spatial median in the feature space and computes the Gram matrix with the spatial median. KPP uses the Gram matrix obtained from spherical KPCA to obtain
the first PC with the help of the Q estimator [13], and then the remaining PCs are found by implementing the projection pursuit based on the Q estimator. Kernel ROBPCA utilizes the Stahel-Donoho estimator to calculate the outlyingness for each sample and selects three-quarters of the samples with the lowest outlyingness to compute KPCA. However, this method is computationally expensive for a large training set.

The second category for robustifying KPCA is to consider use of a robust loss function. One example of this strategy is PCA-ℓ₁ [14]. The ℓ₁-norm is substituted into the ℓ₂-norm maximum variance objective function of standard PCA. After the first PC is obtained, the subsequent PCs are derived with the greedy search method. Kwak proposed a robust method for KPCA with the nonlinear projection trick [15]. By using the eigendecomposition of the kernel matrix, the input data is mapped into a reduced dimensional kernel space with explicit maps, yielding data with new coordinates, and then the PCA-ℓ₁ algorithm is performed on these data. Xiao et al. [16] proposed ℓ₁-KPCA to robustify KPCA. This method can be seen as a kernel extension of PCA-ℓ₁, but the constraint is different from that of PCA-ℓ₁. The procedure for the ℓ₁-KPCA algorithm includes the upper layer and the lower layer to obtain orthogonal loading vectors. The upper layer is to deflate the Gram matrix to preserve the orthogonal loading vectors, and the lower layer is to obtain the optimal coefficients of the loading vector. Kim and Klabjan [17] proposed a simple and fast algorithm for robustifying KPCA. The advantage of this method is that it avoids implementing eigendecomposition. Although these methods reduce the effect of outliers, one drawback is that the components are obtained sequentially rather than simultaneously. Other methods that include a robust loss function have been proposed [18]–[20]. Alzate and Suykens [18] used an epsilon-insensitive robust loss function and proposed two algorithms. However, the optimization process is complex, and only one component can be obtained at a time. Huang et al. [19] proposed an iterative reweighted algorithm for robustifying KPCA, and small weights are assigned to outliers in the iteration process. He et al. [20] presented robust KPCA based on a maximum correntropy criterion. However, the optimization procedure of these methods is equivalent to the optimization problem of weighted PCA, which leads to expensive computation.

The third category for robustifying KPCA can be considered outlier rejection. Lu et al. [21] introduced robust KPCA by iteratively eliminating outliers according to reconstruction errors from the training sets. However, this method fails to cope with the small sample size problem. Ding et al. [22] considered removing undesirable samples with the largest influence indexes. Based on the idea of the influence index, Duan et al. [23], [24] constructed a weight function by using the influence index for the Gram matrix and computing the weight iteratively. The outlier can be identified by a weight smaller than a threshold. Xu et al. [25] proposed a robust PCA algorithm and extended it to a kernel version. This approach includes a random step to remove a possible outlier at the price of the computational cost for each iteration. Feng et al. [26] proposed a deterministic approach by decreasing the weights of all observations for each iteration to reduce the computation cost of the algorithm in [25]. However, those methods are computationally expensive due to the iterative calculation of the Gram matrix.

The fourth category for robustifying KPCA is based on robust PCA (RPCA) [27], which aims to recover a low-rank matrix and a sparse matrix from the corrupted matrix. Ma et al. [28], [29] extended this model in the feature space and proposed two approaches for optimizing the problem. The first method builds a kernel matrix for a low-rank matrix and a sparse matrix by using the distributive property over the matrix addition. The second approach is constructed by using the Gram matrix as a surrogate of the original low-rank matrix in the feature space and a ℓ₁-norm sparse matrix in the input space. The limitation of these two models is that the former may lose the advantage with a small number of training sets, while the latter is sensitive to the initialization. Fan and Chow [30] proposed to robustify KPCA based on the nuclear norm of a low-rank matrix in the feature space and a ℓ₁-norm sparse matrix in the input space. However, this method involves a derivative matrix, which raises the computational complexity because of the large size of the Gram matrix.

The fifth category uses the regularization technique. Nguyen and Torre [31] proposed a robust method for handling noise, missing data, and outliers with the help of the penalty term. However, in the case of noise and missing data, KPCA is calculated on a clean dataset. The method gives few details for how to deal with outliers. Another option for using the regularization technique is based on the membership values of the input data. The membership value indicates the contribution to the calculation of the covariance matrix. Pang et al. [32] decided the membership values by using the penalty term with discrete values restricted in [0, 1]. The idea of this method is based on a robust technique for PCA [33]. However, the limitation is that the membership value is decided in the context of the first PC. It is not sufficient to reflect the inherent structure learned from the data. To overcome this limitation, Heo et al. [34] considered more than one PC to decide the membership by using entropy regularization. Then, the fuzzy mean and the fuzzy covariance are computed with the membership values. The drawback of this algorithm is its sensitivity to the initial membership value. Tao et al. [35] proposed density-sensitive robust fuzzy KPCA to overcome this drawback. However, the effect of the kernel parameter is not discussed in their research. Mateos and Giannakis [36] considered a low-rank bilinear decomposition model with outlier-sparsity regularization to find outliers. The sparsity parameter must be adjusted according to the percentage of outliers. However, it is difficult to manually set to perfectly identify outliers in practical applications.

The underlying idea of regularization focuses on robust estimates of the Gram matrix or outlier sparsity to reduce the influence of outliers, but pays little attention to mitigating the contribution of some samples to the principal axis.
Because the Gram matrix is not robust when outliers exist in the data, the principal axis in the feature space could be deviated by outliers, leading to misinterpretation of the PCs. In this paper, we propose a robust method for KPCA. The idea of the proposed method comes from sparse PCA via variable projection [37]. In detail, we show that KPCA can be reformulated in Euclidean space. Then, a robust loss function based on the idea of RPCA [27] is introduced, and \( \ell_{2,1} \)-regularization is added to the loss term. Different from the other regularization method, our idea is to restrain the effect of outliers expressed in the principal axis. The goal of the proposed method is analogous to \( \ell_1 \)-norm methods of the second category, but based on \( \ell_{2,1} \)-regularized minimization. In addition, as orthogonality is not satisfied in the proposed method, we use the Gram-Schmidt orthonormalization process to obtain the orthonormal bases.

The rest of the paper is organized as follows. In Section II, we describe KPCA, an expression of KPCA in Euclidean space, and the drawback of KPCA. Then, in Section III, we provide the proposed algorithm, and in Section IV, we show the experimental results. Finally, in Section V, we present the conclusion.

II. KPCA AND ITS DRAWBACK

In this section, we give some basic notations first and then detail KPCA and its drawback.

A. NOTATIONS

For convenience, some notations are used in this paper. The lowercase bold letters denote vectors. The superscript \( T \) represents the transpose of a matrix or vector. The uppercase bold letters denote matrices. The lowercase letters \( i, j, k \) denote the index, and \( t \) represents the iteration. \( D \) denotes the dimensionality, \( N \) denotes the number of samples, and \( M \) stands for the number of indexes. The other uppercase letters represent function for convenience. \( \|x\|_2 \) denotes the \( \ell_2 \)-norm of vector \( x \). For matrix \( E \in \mathbb{R}^{N \times M} \), \( \|E\|_F \) represents the Frobenius norm. The \( \ell_{2,1} \)-norm of matrix \( E \) is defined as
\[
\|E\|_{2,1} = \sum_{i=1}^{N} \left( \sum_{j=1}^{M} E_{ij}^2 \right)^{\frac{1}{2}},
\]
where \( E_{ij} \) is the \((i,j)\)th element of matrix \( E \).

Let \( \{x_i\}_{i=1}^{N} \) be a dataset in the \( D \) dimension of the input space. Denote \( \phi : \mathbb{R}^D \rightarrow \mathcal{H} \) as the kernel mapping maps the input data into the feature space, where \( \mathcal{H} \) is a reproducing kernel Hilbert space (RKHS) [38]. Given the mapping \( \phi \), each \( x_i \) is mapped into the feature space \( \mathcal{H} : x_i \rightarrow \phi(x_i) \), yielding the nonlinear dataset \( \{\phi(x_i)\}_{i=1}^{N} \). Next, we detail KPCA and its drawback.

B. KPCA

Denote \( \Phi = [\phi(x_1), \ldots, \phi(x_N)] \). For simplicity, we suppose \( \Phi \) is a mean-centered matrix. The covariance matrix \( C \) in the feature space can be expressed as:
\[
C = \sum_{i=1}^{N} \phi(x_i)\phi(x_i)^T = \Phi^T\Phi. \quad (1)
\]
The eigenvalue \( \lambda_k \) and the eigenvector \( v_k \) of \( C \) satisfy:
\[
Cv_k = \lambda_k v_k, \quad k = 1, \ldots, M. \quad (2)
\]
Note that \( v_k \) in the feature space can be written as a linear combination of \( \phi(x_1), \ldots, \phi(x_N) \); that is,
\[
v_k = \sum_{i=1}^{N} a_{ik} \phi(x_i) = \Phi a_k, \quad (3)
\]
where \( a_k = (a_{1k}, \ldots, a_{Nk})^T \) is the coefficient vector. Substituting \( v_k \) in (2) yields
\[
\Phi \Phi^T \Phi a_k = \lambda_k \Phi a_k. \quad (4)
\]
As the inner product in the RKHS is given by kernel function \( \phi(x)^T\phi(x) = k(x, x) \), left-multiplying \( \Phi \) on both sides of (4), we obtain \( K^2 a_k = \lambda_k K a_k \), where \( K = \Phi^T \Phi \) is the Gram matrix whose \((i,j)\)th element is \( K_{ij} = k(x_i, x_j) \). Vector \( a_k \) satisfies \( K a_k = \lambda_k a_k \). Note that \( v_k^T v_k = 1 \); we have \( a_k^T K a_k = 1 \). Solution \( a_k \) can be obtained from the \( k \)th eigenvector of \( K \) corresponding to the \( k \)th largest eigenvalue. By using the eigendecomposition, \( K \) can be decomposed into
\[
K = EDE^T.
\]
where \( E = [e_1, \ldots, e_M] \) whose columns are orthonormal, and \( D \) is a diagonal matrix storing the eigenvalues \( \lambda_1, \ldots, \lambda_M \) in descending order. Then, solution \( a_k \) is given as
\[
a_k = e_k \lambda_k^{-\frac{1}{2}}. \quad (5)
\]
Denoting \( V = [v_1, \ldots, v_M] \), we have \( V = \Phi A \) according to (3), where \( A = [a_1, \ldots, a_M] \). From (5), we have
\[
A = [e_1 \lambda_1^{-\frac{1}{2}}, \ldots, e_M \lambda_M^{-\frac{1}{2}}] = E_M D_M^{-\frac{1}{2}}. \quad (6)
\]
where \( E_M = [e_1, \ldots, e_M] \), and \( D_M = \text{diag}(\lambda_1, \ldots, \lambda_M) \). The principal orthogonal axis \( v_k \) can be expressed as \( v_k = \Phi a_k = \lambda_k^{-\frac{1}{2}} \Phi e_k \). Therefore, the projection of \( \phi(x_i) \) onto the \( k \)th principal axis \( v_k \) is given by \( \phi(x)^T v_k = \lambda_k^{-\frac{1}{2}} \Phi^T e_k \), where \( k_i = \Phi^T \phi(x_i) \) is the \( i \)th column vector of \( K \). The projection of \( \Phi \) onto the subspace spanned by the first \( M \) principal axes is given by
\[
V^T \Phi = D_M^{-\frac{1}{2}} E_M^T K. \quad (7)
\]
However, an alternative interpretation of KPCA can be obtained in the least square sense. Jenssen [39] pointed out that the KPCA procedure minimizes the Frobenius norm of \( K - K_M \), namely,
\[
\min_{\lambda_1 e_1, \ldots, \lambda_N e_N} \|K - K_M\|_F, \quad (8)
\]
where
\[
K_M = E_M D_M E_M^T = \left( D_M^{-\frac{1}{2}} E_M^T \right) D_M^{-\frac{1}{2}} E_M^T.
\]
The underlying idea behind this derivation is based on the connection between KPCA and classical metric multidimensional scaling [40]. \( D_M^{-\frac{1}{2}} E_M^T \) are known as the coordinates of \( \Phi \).
in low-dimensional space [15]. Projection (7) can be rewritten as 
\[ V^T \Phi = D_M^{-\frac{1}{2}} E_M^T K = D_M^{-\frac{1}{2}} E_M^T, \]
which is the solution to minimization problem (8) (see [39] for details).

C. DRAWBACK OF KPCA
One of the main drawbacks of KPCA is its sensitivity to outliers. Note that the projection of \( \Phi \) onto \( v_k \) is given by
\[ \Phi^T v_k = \Phi^T \Phi a_k = K a_k = \lambda_k a_k = \frac{1}{\lambda_k} e_k = \frac{1}{\lambda_k} e_k, \]
leading to misinterpretation of the PCs because outliers distort the direction of \( v_k \). Coefficient vector \( a_k \) constructs a relationship between principal axis \( v_k \) and orthonormal basis \( e_k \) from the formulation above. The row index of \( a_k \) corresponding to the sample index coincides with the row index of \( e_k \). This implies that it is possible to reduce the influence of outliers by applying a robust method to \( e_k \). Therefore, we consider optimizing with respect to \( e_k \) instead of \( a_k \). Because each column vector of \( E_M \) comes from \( E, E_M \) meets \( KE_M = E_M D_M \) according to the eigendecomposition. Thus, we have
\[ \| K - K \| _F^2 = \| K - KE_M D_M^T \| _F^2 = \sum_{i=1}^N \| k_i - E_M^T k_i \| _2^2. \]

Note that \( E_M^T E_M = I_{M \times M} \). The equation above can be interpreted as a PCA problem in Euclidean space with dataset \( \{k_i\}_{i=1}^N \). There are various methods for obtaining sparse representation based on objective function (9). However, objective function (9) is formulated in the least square sense, which is not robust to outliers. A method for robustifying the estimation of \( E_M \) must be developed. In the next section, we harness a similar idea of RPCA [37] to illustrate the proposed method.

III. PROPOSED METHOD FOR ROBUST KPCA
Zou et al. [41] proved the minimization problem given as (10) is equivalent to the minimization of (9):
\[ \min_{E_M, F_M} \frac{1}{2} \| K - KE_M F_M^T \| _F^2 + \frac{\beta}{2} \| E_M \| _F^2 \]
\[ \text{s.t. } F_M^T F_M = I_{M \times M}, \]
where \( \beta > 0 \). The factor of 1/2 is included for convenience. Note that in the relationship, between \( E_M \) and \( A \) in (6), Wang and Tanaka [42] considered the application of the \( l_2,1 \)-norm on \( A \) to obtain sparse representation, which means the principal axis can be represented by the same data. However, the robust problem is not considered in their work. Recently, Ericsson et al. [37] presented a robust formulation for sparse PCA. The underlying idea is based on the idea of RPCA [27] in which a robust loss function is introduced, and \( l_1 \)-norm regularization is added to the sparse model. Inspired by this method, we note that the norm \( \| k_i - E_M^T k_i \| _2 \) would be affected by outliers. For this reason, we propose

the following objective function:
\[ J = \min_{E_M, F_M, S} \frac{1}{2} \| K - KE_M F_M^T - S \| _F^2 + \frac{\beta}{2} \| E_M \| _F^2 + \alpha \| E_M \| _{2,1} + \kappa \| S^T \| _{2,1} \]
\[ \text{s.t. } F_M^T F_M = I_{M \times M}. \]

The optimization problem above (11) can be interpreted as follows. The \( l_{2,1} \)-norm is enforced on \( S^T \) to restrict the contribution from samples whose error is affected by outliers. After \( S \) is calculated, the expression \( \| K - KE_M F_M - S \| _F^2 \) can be seen as a robust criterion for dampening the contamination of the outliers. However, to guarantee each principal axis can be expressed by the same data, the \( l_{2,1} \)-norm is imposed on \( E_M \). The objective function (11) involves \( E_M, F_M, \) and \( S \) which need to be optimized. We can optimize them alternatively, and each optimization problem involves only one variable.

A. OPTIMIZING THE E_M SUBPROBLEM
First, we use the Gauss-Seidel method [43], [44] to optimize the \( E_M \) subproblem. Fixing the variable \( F_M \) and \( S \) at the iteration \( t \), and adding a proximal term containing \( E_M \) for the purpose of guaranteeing convergence, we have the following problem for solving \( E_M \):
\[ E_M^{t+1} = \arg \min_{E_M} \frac{1}{2} \| K - KE_M F_M^T - S_t \| _F^2 + \frac{\beta}{2} \| E_M \| _F^2 + \alpha \| E_M \| _{2,1}. \]

Denote the first and second terms of (11) by
\[ H(E_M, F_M, S) = \frac{1}{2} \| K - KE_M F_M^T - S \| _F^2 + \frac{\beta}{2} \| E_M \| _F^2. \]

A linearized approximation of \( H(E_M, F_M, S') \) at the point of \( E_M \) is given by
\[ Q_t(E_M, E_M') = \langle E_M - E_M', \nabla H_{E_M}(E_M', F_M', S') \rangle + H(E_M', F_M', S') \]
where \( \nabla H_{E_M}(E_M, F_M, S) \) denotes the gradient with respect to \( E_M \). For \( H(E_M, F_M, S) \),
\[ \nabla H_{E_M}(E_M, F_M, S) = -K^T W F_M + \beta E_M, \]
where \( W = K - S - KE_M F_M^T \). Thus, problem (12) that uses the proximal algorithm can be addressed by
\[ E_M^{t+1} = \arg \min_{E_M} Q_t(E_M, E_M') + \frac{1}{2\mu} \| E_M - E_M' \| _F^2 + \alpha \| E_M \| _{2,1}. \]

Note that
\[ \frac{1}{2\mu} \| E_M - E_M' \| _F^2 + \langle E_M - E_M', \nabla H_{E_M}(E_M', F_M', S') \rangle = \frac{1}{2\mu} \| E_M - E_M' + \mu \nabla H_{E_M}(E_M', F_M', S') \| _F^2 \]
\[-\frac{\mu}{2} \| \nabla H_{E_M}(E_M', F_M', S') \| _F^2; \]
To optimize problem (18), we first optimize the following decomposition (SVD):

\[
H(E_M^t, F_M^t, S^t) - \frac{\mu}{2} \| \nabla H E_M(E_M^t, F_M^t, S^t) \|_F^2 + \frac{1}{2\mu} \| E_M - E_M^t + \mu \nabla H E_M(E_M^t, F_M^t, S^t) \|_F^2 + \alpha \| E_M \|_{2,1}.
\]  

(14)

The first two terms in (14) are constant values with respect to \( E_M \) and can be omitted. Therefore, the \( E_M \) subproblem is equivalent to the following problem:

\[
E_M^{t+1} = \arg \min_{E_M} \frac{1}{2\mu} \| E_M - \Theta(E_M^t) \|_F^2 + \alpha \| E_M \|_{2,1}.
\]

(15)

where \( \Theta(E_M^t) = E_M^t - \mu \nabla H E_M(E_M^t, F_M^t, S^t) \), and \( e^t \) and \( \theta(e^t) \) are the \( i \)-th row vectors of \( E_M \) and \( \Theta(E_M^t) \), respectively. The solution of each \( e^t \) is given by the vectorial soft-threshold operator [45],

\[
e^t = \theta(e^t) \cdot \max(\|\theta(e^t)\|_2 - \mu \alpha, 0) \cdot \max(\|\theta(e^t)\|_2 - \mu \alpha, 0) + \mu \alpha.
\]

(16)

Thus, the solution of (15) can be written as \( E_M^{t+1} = \{ e^1; \ldots; e^N \} \). Through the vectorial soft-threshold operator, we may see that some rows of the optimal \( E_M \) corresponding to (15) could be close to zero, which dampens the effect of some samples expressed in the principal axis.

**B. OPTIMIZING THE \( F_M \) SUBPROBLEM**

Next, for fixed \( E_M^{t+1} \) and \( S^t \), the \( F_M \) subproblem minimizes the following problem:

\[
F_M^{t+1} = \arg \min_{F_M} \frac{1}{2} \| K - K E_M^{t+1} F_M^T - S^t \|_F^2
\]

s.t. \( F_M^T F_M = I_{M \times M} \).

(17)

It can be solved by the orthogonal Procrustes problem [46]. The solution of \( F_M^{t+1} \) can be obtained from the singular value decomposition (SVD):

\[
(K - S^t)^T K E_M^{t+1} = M \Lambda N^T,
\]

and set \( F_M^{t+1} = MN^T \).

**C. OPTIMIZING THE \( S \) SUBPROBLEM**

Finally, for fixed \( E_M^{t+1} \) and \( F_M^{t+1} \), updating the \( S \) subproblem is optimized as follows:

\[
S^{t+1} = \arg \min_S \frac{1}{2} \| K - K E_M^{t+1} F_M^{t+1} - S \|_F^2 + \kappa \| S^{t+1} \|_{2,1}.
\]

(18)

To optimize problem (18), we first optimize the following problem:

\[
S^{t+1} = \arg \min_{S^{t+1}} \frac{1}{2} \| Z - S^{t+1} \|_F^2 + \kappa \| S^{t+1} \|_{2,1},
\]

(19)

where \( Z = K - K E_M^{t+1} F_M^{t+1} - S^t \). The solution of \( S^{t+1} \) can be obtained by using a vectorial soft-threshold operator on each row of \( Z \), and the optimal \( S^{t+1} \) is the transpose of \( S^{t+1} \), namely,

\[
S^{t+1} = (S^{t+1})^T.
\]

(20)

**Algorithm 1 Robust KPCA With \( \ell_{2,1} \)-Regularized Loss Minimization**

1. **Input**: \( E_M^0, F_M^0, \) and \( S \).  
2. **Initialization**: \( \kappa, tol, t = 0, \alpha \leftarrow \bar{\alpha}, \beta \leftarrow \bar{\beta}, \mu = \frac{1}{\lambda_1 + \beta} \), where \( \lambda_1 \) is the largest eigenvalue in KPCA.  
3. **while** not converged or within the preset iteration do  
4. Compute \( E_M^{t+1} \) according to (15)  
5. Compute the SVD of (17) to update \( F_M^{t+1} \)  
6. Update \( S^T \) using (19) and obtain \( S^{t+1} \) from (20)  
7. \( t \leftarrow t + 1 \)
8. **end while**  
9. **Output**: \( F_M^t \)

Based on the analysis, the proposed robust KPCA method is summarized in Algorithm 1. The algorithm is terminated when either the relative error is smaller than a preset tolerance value or a maximum number of iterations is reached. The relative error at the \( t \) iteration is defined as \( \| S^{t} - S^{t-1} \|_F < tol \), where \( J \) is objective function (11), and \( tol \) is the preset tolerance value. Inputs \( E_M \) and \( F_M \) are matrices obtained from KPCA, and \( S \) is the zero matrix in Step 1 of Algorithm 1.

**IV. EXPERIMENTS AND DISCUSSION**

In this section, a toy example and analysis with real data are performed to verify the effectiveness of the proposed method. The Gaussian radial basis function \( k(x, y) = \exp(-\|x-y\|^2 / 2\sigma^2) \) is used as the kernel function in the experiment.

**A. PARAMETERS IN KERNEL PRINCIPAL COMPONENT ANALYSIS**

Two parameters are associated with KPCA: the Gaussian kernel parameter \( \sigma \) and the number of PCs, \( M \). In the toy example, \( \sigma \) is given in Section IV-C. In the real dataset, parameter \( \sigma^2 \) is estimated by the average pair-wise squared Euclidean distance between the training samples, that is,

\[
\sigma^2 = \frac{2}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \| x_i - x_j \|_2^2.
\]

The choice of parameter \( \sigma \) depends on the nature of the data, and different parameters might have different levels of effectiveness in outlier detection. We complied with the following recommendation [5], [30]: Parameter \( \sigma \) should be of the same order of magnitude as the pair-wise distance between the training samples. However, the pair-wise distance and its squared distance are not essentially different for \( \sigma \) and \( \sigma^2 \). Therefore, for computational simplicity, we use the sum of the pair-wise squared distance to compute \( \sigma^2 \), which can
be calculated by computing the Gram matrix. The number of PCs, \( M \), is determined by the cumulative percent variance (CPV) \[47\],

\[
\text{CPV}(M) = \frac{\sum_{i=1}^{M} \lambda_i}{\sum_{i=1}^{N} \lambda_i} \times 100% ,
\]

where \( \lambda_i \) is the eigenvalue of the Gram matrix sorted in descending order. The CPV measures the percentage of the variance accounted for by the first \( M \) PCs. With this criterion, we select the desired CPV for detecting outliers. We use the same kernel parameter \( \sigma \) and the same number of PCs chosen by KPCA in the proposed method.

**B. OUTLIER SCORE AND THRESHOLD VALUE**

To detect outliers, the outlier score \([5],[48]\) for each sample is utilized. When KPCA is used, the outlier score can be calculated by

\[
s(x_i) = \sum_{j=1}^{M} \left( \phi(x_i)^T \tfrac{\phi(x_j)}{\sqrt{\lambda_j}} \right)^2 \sum_{j=1}^{M} \left( \frac{k_j^T e_j}{\lambda_j} \right)^2 .
\]

The outlier score involves orthonormal basis \( e_j \) and deviation \( \lambda_j \) along the direction of \( e_j \). To compute the outlier score that corresponds to the proposed method, we need to compute orthonormal basis \( \tilde{e}_j \) and deviation \( \tilde{\lambda}_j \) along the direction of each \( \tilde{e}_j \). First, orthogonality can be sacrificed in the proposed method. Second, if the PCs are correlated, the total variance could be too optimistic. For these reasons, we apply the Gram-Schmidt orthonormalization process to \( \tilde{E}_M \) to obtain a matrix \( \tilde{E}_M = [\tilde{e}_1, \tilde{e}_2, \ldots, \tilde{e}_M] \) with orthonormal columns. Then, orthonormal basis \( \tilde{e}_j \) is the \( j \)th column vector of \( \tilde{E} \). Next, we use \( \tilde{E} \) to replace \( E_M \) in (7); namely, the new embedding matrix is \( \tilde{D}_M^{-\frac{1}{2}} \tilde{E}^T \tilde{K} \). For this new embedding matrix, deviation \( \tilde{\lambda}_j \) along the direction of each \( \tilde{e}_j \) is calculated as

\[
\tilde{\lambda}_j = \sum_{i=1}^{N} (\tilde{\lambda}_j \frac{1}{2} \tilde{e}_j^T k_i) \tilde{e}_j = \lambda_j^{-1} \tilde{e}_j^T K K^T \tilde{e}_j = \lambda_j^{-1} \gamma_j^2 ,
\]

where \( \gamma_j = \|K^T \tilde{e}_j\|_2 \). The intuition behind this computation is that \( k_i \) with scaled \( \tilde{\lambda}_j^{-\frac{1}{2}} \) will project onto a new orthonormal basis with a high deviation. Therefore, the outlier score corresponding to the proposed method is computed as follows:

\[
\tilde{s}(x_i) = \sum_{j=1}^{M} \left( \frac{k_j^T \tilde{e}_j \tilde{\lambda}_j^{-\frac{1}{2}}}{\gamma_j} \right)^2 = \sum_{j=1}^{M} \left( \frac{k_j^T \tilde{e}_j}{\gamma_j} \right)^2 . \quad (21)
\]

After the outlier score is calculated for each \( x_i \), outlier threshold value \( \tilde{c} \) is decided based on the percentile of the empirical distributions of these outlier scores.

**C. EXPERIMENT ON THE TOY DATASET**

In the toy example, artificial data are generated from the Gaussian distribution at three clusters \((-0.5, -0.2), (0, 0.6), \) and \((0.5, 0)\) with a standard deviation of 0.1. Each cluster has 100 samples, and a total of 300 samples are deemed the normal data. Twenty outliers are added to the normal data in which the horizontal coordinates are generated by uniform distribution over \([-0.5, 0.5]\), and vertical coordinates are created by uniform distribution over \([1, 2]\). A scatter plot of this dataset is shown in Fig. 1.

![Toy dataset used in the experiment.](image)

Parameter \( \alpha^2 = 0.05 \) is used in KPCA. We calculate the CPV from the candidate set \([50\%, 60\%, 70\%, 80\%, 90\%]\). The parameters used in the proposed method are \( \kappa = 0.1, \bar{\alpha} = 0.001, \bar{\beta} = 0.1, \) and \( tol = 10^{-9} \), and the iteration number is set to 300. The distribution of the outlier scores corresponding to KPCA and the proposed method is shown in Fig. 2, and the first row to the fifth row correspond to 50\%, 60\%, 70\%, 80\%, and 90\% of the CPV in order. In Fig. 2, the green line represents the threshold \( c \), which is used to classify the normal, and the outlier. It is determined based on the distribution of the outlier score. In the case of KPCA, threshold value \( c \) is computed at the 6.1\% percentile of the distribution of the outlier scores except for 80\% and 90\% of the CPV. For 80\% and 90\% of the CPV, threshold value \( c \) is computed at the 94\% percentile of the distribution of the outlier scores. For the proposed method, threshold value \( \tilde{c} \) is computed at the 6.1\% percentile of the outlier score distribution for the entire CPV candidate set.

KPCA can detect outliers correctly at 50\% and 60\% of the CPV. As the CPV increases, the detection performance worsens at 70\% and 80\%. When 90\% of the CPV is used in KPCA, the outliers could be correctly classified with a few normal data wrongly predicted. However, the performance of the proposed method is superior to KPCA especially with the same number of PCs decided by 70\%, 80\%, and 90\% of the CPV. The toy example shows that the distribution of outlier scores in KPCA depends on the number of PCs. Different numbers of PCs used in the computation of outlier scores lead to different distributions, and have an effect on outlier detection. Compared to KPCA, the experimental result reveals that the proposed method reduces the influence of outliers and is more robust than KPCA for outlier detection.

**D. EXPERIMENTS ON REAL DATA**

In this section, six real datasets from the Outlier Detection DataSets (ODDS) Library \([49]\) are used to verify the performance of the proposed method. Information about the dataset is given in Table 1. For each dataset, the whole samples are
FIGURE 2. The outlier score distribution in the context of the different CPVs. Left: KPCA. Right: The proposed method.
used in the experiment and preprocessed with the zero mean and the unit variance.

1) EVALUATION METRICS
The classification result for outlier detection can be expressed in a confusion matrix as shown in Table 2. True positive means the number of outliers correctly classified, and true negative represents normal data correctly predicted. False negative refers to outliers wrongly classified as normal, and false positive means normal data incorrectly classified as outliers. To evaluate the classification performance, we use the following metrics:

\[
\text{Precision} = \frac{TP}{TP + FP}, \\
\text{Recall} = \frac{TP}{TP + FN}, \\
\text{False Positive Rate} = \frac{FP}{FP + TN}.
\]

Recall is also referred to as the detection rate or the true positive rate. As outliers are regarded as rare instances in the dataset, the most reliable evaluations are based on a high detection rate and a low false positive rate. One may obtain fewer false negatives but at the expense of more false positives. To this end, we report the area under the precision-recall curve (AUC-PR) [50]. The AUC-PR is computed with the average precision [51]. For each dataset, we first compute the outlier score under different CPV situations. After the outlier score \(s(x_i)\) is calculated for each \(x_i\), the observation \(x_i\) is a outlier if \(s(x_i) > c\), where \(c\) is determined according to the percentile of the outlier score distribution. We adjust \(c\) by changing the percentile within the outlier score range and record the AUC-PR. The maximum AUC-PR corresponding to the specified CPV is reported in the experiment results. The parameters used in the proposed method for each dataset are shown in Table 3. The value in parentheses represents the iteration number needed to adjust in some context of the CPV. We explain this point in Section IV-E.

2) EXPERIMENTAL RESULTS
We evaluate the proposed method on the six datasets and compare it with several existing methods, \(\ell_1\)-KPCA [16] and simple \(\ell_1\)-KPCA [17]. Figs. 3, 4, 5, 6, 7, and 8 show the AUC-PR for six datasets with a varying number of CPVs, respectively. The PCs are again chosen from the CPV computed using KPCA and the same number of PCs used in the other methods.

For the Wisconsin-Breast Cancer (WBC) dataset, we calculate the outlier score under the candidate CPV set \{40\%, 50\%, 60\%, 70\%, 80\%, 90\%\}. The result is shown in Fig. 3. The performance of KPCA decreases as the CPV increases, while the other methods give a better performance than KPCA.

Fig. 4 shows the AUC-PR values for all methods at different CPVs for the Ionosphere dataset. The proposed method achieves satisfactory results at 90\% of the CPV. Fig. 5, Fig. 6, and Fig. 7 show the results for the BreastW, Cardio, and Musk datasets, respectively. We see that the AUC-PR is high only when a few CPVs are used in the KPCA case. As the CPV increases, the poor performance becomes even more apparent. Note that the different numbers of PCs used in the outlier score yield different distributions. The most relevant principal axes could contain all the information about the distribution of the data, and the remaining axes are considered to be associated with outliers [52]. This is why KPCA gives a high AUC-PR with a few CPVs. However, for the Cardio dataset, the proposed method gives low detection at 40\% and 50\% of the CPV but obtains a satisfactory
D. Wang, T. Tanaka: Robust KPCA With $\ell_{2,1}$-Regularized Loss Minimization

FIGURE 4. The AUC-PR values at different CPVs for the Ionosphere dataset.

FIGURE 5. The AUC-PR values at different CPVs for the BreastW dataset.

FIGURE 6. The AUC-PR values at different CPVs for the Cardio dataset.

FIGURE 8. The AUC-PR values at different CPVs for the Mnist dataset.

FIGURE 7. The AUC-PR values at different CPVs for the Mnist dataset.

TABLE 4. The Comparison of the AUC-PR Results.

| Dataset     | AUC-PR | KPCA | Simple $\ell_{2,1}$-KPCA | $\ell_{2}$-KPCA | Proposed |
|-------------|--------|------|--------------------------|-----------------|----------|
| WBC         | 0.38657| 0.45238| 0.48095                 | 0.52414         |          |
| Ionosphere  | 0.78403| 0.79680| 0.80390                 | 0.80219         |          |
| BreastW     | 0.88784| 0.87581| 0.88140                 | 0.92617         |          |
| Cardio      | 0.63261| 0.62368| 0.60479                 | 0.68184         |          |
| Musk        | 1      | 1    | 0.99002                 | 1               |          |
| Mnist       | 0.25145| 0.25231| 0.25719                 | 0.31316         |          |

E. DISCUSSION

Note that the evaluation of the proposed method is derived from the outlier score extracted from the new embedding matrix $D_M^{-\frac{1}{2}}\hat{E}^TK$, and factor $\lambda_j^{-\frac{1}{2}}$ does not cause any contribution to outlier score $y(x_i)$ in (21). Therefore, we can interpret that the deviation of dataset $[k_j]_{j=1}^N$ along the new orthonormal basis $\hat{e}_j$ is $\gamma_j^2 = \hat{e}_j^TK^T\hat{e}_j$, in other words, the transformed embedding matrix $K^T\hat{E}$ whose outlier score gives the same expression with the new embedding matrix. Therefore, we use $\hat{E}$ to replace $E_M$ in (7) to compute the outlier score. The new set of orthonormal bases is the columns of $\hat{E}$ obtained from the $\ell_{2,1}$-regularized minimization and the Gram-Schmidt orthogonalization process in tandem. However, there is an issue that we still need to consider—the convergence of the proposed method and the parameters used in the proposed method. It is difficult to theoretically

performance gradually in comparison with KPCA. Although the proposed method is poor in the case of only a few PCs, a natural way to compensate this loss is to tune the parameter
prove the convergence of the proposed algorithm. Instead, we follow the analysis of multilinear sparse PCA [53] to show the convergence property. The objective function values and corresponding iteration numbers are drawn for each dataset. Fig. 9 displays the behavior of the objective function value at each iteration. All plots are obtained at 70% of the CPV and under the parameters shown in Table 3. It can be observed from the figure that the algorithm monotonically decreases in (11), and the curve of each dataset becomes flat when either the relative error is smaller than a preset tolerance value or the maximum number of iterations is reached.

Parameters \( \kappa \), \( \tilde{\alpha} \), \( \tilde{\beta} \), and \( \text{tol} \) and the number of iterations in Table 3 are set empirically. Because the number of PCs in KPCA depends on the CPV, a low number of PCs can cause every row of \( \mathbf{E}_M \) to be shrunk to zero during the iteration in the proposed method. In this case, \( \mathbf{E}_M^r \) is the zero matrix, and thus, the Gram-Schmidt orthonormalization process is invalid. To this end, we need to reduce the iteration numbers or change parameters \( \kappa \), \( \tilde{\alpha} \), and \( \tilde{\beta} \), respectively. In contrast, when more PCs are used in the algorithm, if we set the loops to low numbers of iterations, the algorithm does nothing but minimizes the objective function in two consecutive iterations, and no outliers or samples at all would be dampened from the \( \ell_{2,1} \)-regularization. In both cases, we experimentally vary the iteration number or parameters \( \kappa \), \( \tilde{\alpha} \), and \( \tilde{\beta} \) to monitor the performance. It is hard to reconcile with the same iteration number for all CPV cases. As the proposed method is similar to the sparse PCA model [41], it appears to be sensitive to the choice of the number of PCs and all the sparsity parameters [54]. Therefore, for the Musk dataset, the numbers of iterations are 110 and 300 at 80% and 90% of the CPV.
respects, For the Mnist dataset, the iteration number is set to 600 at 90% of the CPV.

V. CONCLUSION

In this paper, we propose a new robust method for KPCA from the perspective of $\ell_{2,1}$-regularized loss minimization and apply this approach to detect outliers. Based on the fact that the coefficient vector expressed in the principal axis builds a bridge between the orthonormal eigenvector of the Gram matrix and the principal axis, we consider optimizing with respect to the orthonormal eigenvector instead of the coefficient with $\ell_{2,1}$-norm regularization in Euclidean space. Note that orthogonality is not satisfied in the proposed method. The Gram-Schmidt orthonormalization process is applied to serve the purpose of the computation of the outlier score. In the experiment, the toy example and real data are used to verify the performance. In the toy example, the proposed method gives more robust results than KPCA. For the real dataset, the results indicate that the proposed method is better than KPCA with the help of $\ell_{2,1}$-norm regularization, yielding a more satisfactory robust performance than KPCA and other existing methods. In future work, we will investigate how the parameters influence the detection performance and aim to apply the proposed method in other fields, such as fault detection and the brain-computer interface. Furthermore, we would like to investigate the relationship between KPCA and the graph-based method [55].

REFERENCES

[1] I. Jolliffe, Principal Component Analysis. Berlin, Germany: Springer, 2011.
[2] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," Neural Comput., vol. 10, no. 5, pp. 1299–1319, Jul. 1998.
[3] J.-M. Lee, C. Yoo, S. W. Choi, P. A. Vanrolleghem, and I.-B. Lee, "Nonlinear process monitoring using kernel principal component analysis," Chem. Eng. Sci., vol. 59, no. 1, pp. 223–234, Jan. 2004.
[4] H. Hoffmann, "Kernel PCA for novelty detection," Pattern Recognit., vol. 40, no. 3, pp. 863–874, Mar. 2007.
[5] C. C. Aggarwal, Outlier Analysis, 2nd ed. Cham, Switzerland: Springer, 2017.
[6] N. A. Campbell, "Robust procedures in multivariate analysis I: Robust covariance estimation," Appl. Statist., vol. 29, no. 3, pp. 231–237, 1980.
[7] J. P. Rousseeuw and M. Hubert, "Robust statistics for outlier detection," Data Mining Knowl. Discovery, vol. 1, no. 1, pp. 73–79, 2011.
[8] M. Debruyne and T. Verdonck, "Robust kernel principal component analysis and classification," Adv. Data Anal. Classification, vol. 4, nos. 2–3, pp. 151–167, Sep. 2010.
[9] N. Locantore, J. Marron, D. Simpson, N. Tripoli, J. Zhang, K. Cohen, G. Boente, R. Fraiman, B. Brumback, and C. Crouch, "Robust principal component analysis for functional data," Test, vol. 8, no. 1, pp. 1–73, 1999.
[10] C. Croux and A. Ruiz-Gazen, "A fast algorithm for robust principal components based on projection pursuit," in COMPSTAT, Berlin, Germany: Springer, 1996, pp. 211–216.
[11] M. Hubert, P. J. Rousseeuw, and K. Vanden Branden, "ROBPCA: A new approach to robust principal component analysis," Technometrics, vol. 47, no. 1, pp. 64–79, Feb. 2005.
[12] P. J. Rousseeuw and C. Croux, "Alternatives to the median absolute deviation," J. Amer. Statist. Assoc., vol. 88, no. 424, pp. 1273–1283, Dec. 1993.
[13] N. Kwak, "Principal component analysis based on L1-norm maximization," IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 9, pp. 1672–1680, Sep. 2008.
**DUO WANG** received the B.S. degree from the Henan University of Technology, China, in 2007, and the M.S. degree from Northeastern University, China, in 2009. He is currently pursuing the Ph.D. degree with the Tokyo University of Agriculture and Technology, Japan. His research interests include kernel methods, outlier detection, signal processing, and machine learning.

**TOSHIHISA TANAKA** (Senior Member, IEEE) received the B.E., M.E., and Ph.D. degrees from the Tokyo Institute of Technology, in 1997, 2000, and 2002, respectively. From 2000 to 2002, he was a JSPS Research Fellow. From October 2002 to March 2004, he was a Research Scientist with the RIKEN Brain Science Institute. He joined the Department of Electrical and Electronic Engineering, Tokyo University of Agriculture and Technology, in April 2004, where he is currently a Professor. He was a Royal Society Visiting Fellow with the Communications and Signal Processing Group, Imperial College London, U.K., in 2005. From June 2011 to October 2011, he was a Visiting Faculty Member with the Department of Electrical Engineering, University of Hawaii at Manoa. His research interests include a broad area of signal processing and machine learning, including brain and biomedical signal processing, brain–machine interfaces, and adaptive systems. He is a member of IEEE, APSIPA, the Society for Neuroscience, and the Japan Epilepsy Society. He served as a Member-at-Large and the Board of Governors (BoG) of the Asia-Pacific Signal and Information Processing Association (APSIPA). He is a Co-Editor of Signal Processing Techniques for Knowledge Extraction and Information Fusion (Mandic, Springer, 2008) and a leading Co-Editor of Signal Processing and Machine Learning for Brain-Machine Interfaces (U.K.: Arvaneh, IET, 2018). He served as an Associate Editor and a Guest Editor of special issues in journals, including Neurocomputing, IEEE Transactions on Fundamentals, and Computational Intelligence and Neuroscience (Hindawi). He serves as the Editor-in-Chief of Signals (MDPI). He also serves as an Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS. Applied Sciences (MDPI), Neural Networks (Elsevier), and Advances in Data Science and Adaptive Analysis (World Scientific). He is a Distinguished Lecturer of APSIPA.

---

**References**

[40] C. K. Williams, “On a connection between kernel PCA and metric multidimensional scaling,” *Mach. Learn.*, vol. 46, nos. 1–3, pp. 11–19, 2002.

[41] H. Zou, T. Hastie, and R. Tibshirani, “Sparse principal component analysis,” *J. Comput. Graph. Statist.*, vol. 15, no. 2, pp. 265–286, Jun. 2006.

[42] D. Wang and T. Tanaka, “Kernel principal component analysis allowing sparse representation and sample selection,” *ECTI Trans. Comput. Inf. Technol.*, vol. 13, no. 1, pp. 9–20, Jun. 2019.

[43] A. Auslender, “Asymptotic properties of the fenchel dual functional and Procrustes Problems,” *Math. Program.*, vol. 146, nos. 1–2, pp. 459–494, Aug. 2014.

[44] J. Bolte, S. Sabach, and M. Teboulle, “Proximal alternating linearized minimization for nonconvex and nonsmooth problems,” *Math. Program.*, vol. 146, nos. 1–2, pp. 459–494, Aug. 2014.

[45] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, “Sparse reconstruction by separable approximation,” *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2479–2493, Jul. 2009.

[46] J. C. Gower and G. B. Dijksterhuis, *Procrustes Problems*. New York, NY, USA: Oxford Univ. Press, 2004.

[47] S. Valle, W. Li, and S. J. Qin, “Selection of the number of principal components: The variance of the reconstruction error criterion with a comparison to other Methods,” *Ind. Eng. Chem. Res.*, vol. 38, no. 11, pp. 4389–4401, Nov. 1999.

[48] M. L. Shyu, S. C. Chen, K. Sarinmapakorn, and L. Chang, “A novel anomaly detection scheme based on principal component classifier,” in *Proc. IEEE Found. New Direct. Data Mining Workshop, 3rd IEEE Int. Conf. Data Mining*, Jan. 2003, pp. 172–179.

[49] S. Rayana, “ODDS library,” Dept. Comput. Sci., Stony Brook Univ., New York, NY, USA: Oxford Univ. Press, 2004.

[50] X. Qi, R. Luo, and H. Zhao, “Sparse principal component analysis by separable approximation,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 10, pp. 1942–1950, Oct. 2014.

[51] Z. Qi, R. Luo, and H. Zhao, “Sparse principal component analysis by choice of norm,” *J. Multivariate Anal.*, vol. 114, pp. 127–160, Feb. 2013.

[52] D. Wang and T. Tanaka, “Kernel principal component analysis allowing sparse representation and sample selection,” *ECTI Trans. Comput. Inf. Technol.*, vol. 13, no. 1, pp. 9–20, Jun. 2019.

[53] Z. Liu, Z. Lai, W. Ou, K. Zhang, and R. Zheng, “Structured optimal graph based sparse feature extraction for semi-supervised learning,” *Signal Process.*, vol. 170, May 2020, Art. no. 107456.