Anomaly detection (AD) aims to distinguish the data points that are inconsistent with the overall pattern of the data. Recently, unsupervised anomaly detection methods have aroused huge attention. Among these methods, feature representation (FR) plays an important role, which can directly affect the performance of anomaly detection. Sparse representation (SR) can be regarded as one of matrix factorization (MF) methods, which is a powerful tool for FR. However, there are some limitations in the original SR. On the one hand, it just learns the shallow feature representations, which leads to the poor performance for anomaly detection. On the other hand, the local geometry structure information of data is ignored. To address these shortcomings, a graph regularized deep sparse representation (GRDSR) approach is proposed for unsupervised anomaly detection in this work. In GRDSR, a deep representation framework is first designed by extending the single layer MF to a multilayer MF for extracting hierarchical structure from the original data. Next, a graph regularization term is introduced to capture the intrinsic local geometric structure information of the original data during the process of FR, making the deep features preserve the neighborhood relationship well. Then, a L1-norm-based sparsity constraint is added to enhance the discriminant ability of the deep features. In order to demonstrate the effectiveness of the proposed approach, we conduct extensive experiments on ten datasets. Compared with the state-of-the-art methods, the proposed approach can achieve the best performance.

1. Introduction

Anomaly detection (AD) aims at finding the part of data that do not conform with the expected behavior [1]. These data are usually called outliers, anomalies, and so on. The anomalies sometimes naturally represent the abnormal events, e.g., damage to the sensors, cyberattack, and black swan events in the financial sector. Therefore, a series of AD-based methods have been proposed to remove these outliers from the original data and applied in many application fields such as fraud detection, wireless sensor networks, medical diagnosis, and so on [2, 3].

AD-based methods can be roughly divided into the following three categories: supervised anomaly detection (SAD), semisupervised anomaly detection (SSAD), and unsupervised anomaly detection (UAD). SAD-based methods, e.g., support vector machine (SVM) [4, 5], can be regarded as a one-class classification (OCC) problem under the unbalanced samples. SSAD-based methods, e.g., one-class random forest [6], often use partially labeled data to train the model. Since these approaches depend on the labeled data to train the model, the insufficient labeled data will limit their performance. However, the unlabeled data are often enough and easy to obtain, so some researchers proposed UAD-based methods, which utilize unlabeled data to build model and classify the anomaly data points. For instance, local outlier factor (LOF) [7] defines a metric to calculate outlier score for every data point directly.

In UAD-based methods, the data are often collected from the high-dimensional space, which leads to the high computational cost and storage space. In this case, the distance-based methods [8, 9] cannot perform efficiently. Although some accelerate techniques [10] have been proposed to deal with the aforementioned issue, they are still not
suitable for handling the complex data. Furthermore, the “Distance Concentration” phenomenon, as well as called “Curse of Dimensionality” problem, is prone to occur in the complex data, which leads to the distances among data points tend to become almost the same [11]. Under this circumstance, it is very hard to use the deviation to distinguish abnormal values from the normal values. Besides, the high-dimensional data always have a lot of irrelevant noise data, which interfere with the detection of outliers [12]. In order to overcome these problems, some scholars proposed some clustering-based approaches for anomaly detection. In these methods, feature representation methods such as subspace learning are used to transform the original high-dimensional data into the low-dimensional feature space. Then, the clustering algorithms are performed on the new feature representation of the original data to discover outliers [13, 14]. Although these methods can achieve better detection results, their performances may be greatly affected by both the quality of feature representation methods and the stability of the clustering algorithms. To reduce the influence of clustering algorithms, reconstruction error-based methods have been proposed in which the error can be regarded as the outlier score for anomaly detection [15, 16].

Learning more useful feature representation from the original data for detecting outliers is very important and also attracts too much attention. Matrix factorization (MF) is a brilliant framework for FR, which has been widely used for anomaly detection such as principal component analysis (PCA) [17] and nonnegative matrix factorization (NMF) [18]. Compared with PCA, NMF has obtained a more meaningful feature representation due to the fact that the nonnegative constraints are added during the procedure of MF. NMF aims to decompose the original matrix into the inner product of a nonnegative basis matrix and a nonnegative coefficient matrix. Therefore, the original samples can be represented as the linear combination of the basis matrix’s column vector and the combination coefficient is the corresponding row of the coefficient matrix. Due to the nonnegative constraints, the learned components can be linearly added to represent the original samples, which make NMF be widely used in anomaly detection [19–22]. Tong et al. [23] propose a nonnegative residual matrix factorization (NRMF) framework, which finds misbehavioral IP sources and abnormal users. Kannan et al. [24] employ NMF to search the outliers from the text data. In addition, Alshammari et al. [25] do the similar work on wireless sensor networks’ data. However, since the abovementioned methods ignore the structural information of data, their performances will be affected. To overcome this problem, some variants’ NMF methods have been proposed. For example, Cai et al. [26] introduce the manifold learning into original NMF and propose graph regularized NMF (GNMF). GNMF regularizes the original NMF formulation by using a Laplacian matrix and the structural information can be preserved well. Kuang et al. [27] propose symmetric NMF (SNMF), which can not only takes the structure information into considered but also obtains a low-rank result. Recently, Ahmed et al. [28] consider the neighborhood structure similarity information and propose neighborhood structure-assisted NMF (NS-NMF). NS-NMF uses minimum spanning tree (MST) to characterize the structural information, which shows good performance in anomaly detection.

Different from NMF-based methods, sparse representation (SR) [29] is another MF-based approach and has received growing attention in many applications, e.g., denoising [30, 31], classification [32, 33], and pattern recognition [34, 35]. In the field of anomaly detection, SR-based methods also show powerful performances. For example, Cong et al. [36] propose the sparse reconstruction cost (SRC) over the normal dictionary and apply it to detect abnormal events. Similar to some density-based anomaly detection methods, Xiao et al. [37] introduce sparsity measurement on the original NMF to detect anomalies in surveillance video. Based on low rank (LR) and SR, Xu et al. [38] propose an anomaly detection method for hyperspectral images. Different from [38], Ling et al. [39] impose the sum-to-one and nonnegativity constraints to get physically meaningful result. Pilastre et al. [40] propose a method based on SR and dictionary learning (DL) which can handle multivariate telemetry time series described by mixed continuous and discrete parameters.

Since the original SR-based methods only focus on the approximation representation of the original data and ignore the intrinsic structure of the data, it can hardly deal with the complex data well. In other words, the new feature representation loses the local geometric structure of the original high-dimensional data. Actually, a pair of adjacent data in a high-dimensional feature space should maintain the same relationship in a new feature space. To achieve this goal, Zheng et al. [41] introduce the manifold learning into SR and design graph regularized sparse coding (GRSC). GRSC uses the Laplace matrix to measure the features so that the features can preserve the local geometric structure. Previous studies [26–28, 42] have also shown that the geometric structure of the data can help to detect abnormal points.

In addition, original SR-based methods belong to shallow feature representation framework, which can only extract the shallow representation of the data. To remedy this limitation, He et al. [43] propose a deep sparse coding (DSC) method which extends a single layer sparse coding to a three-layer deep network architecture model. Moreover, in order to learn more discriminative feature representation, Sharma et al. [44] added a dense layer between two sparse layers. Tariyal et al. [45] and Singh et al. [46] propose a deep dictionary learning (DDL) framework for image classification and nonintrusive load monitoring. Cheng et al. [47] propose a deep sparse representation (DSR) method, which integrates a two-layer convolutional neural network (CNN) for extracting the high-level features and a sparse representation classifier (SRC) for face recognition. In addition, deep neural network (DNN) approaches including AutoEncoder (AE) [48] and Generative Adversarial Net (GAN) [49] have also been used in anomaly detection, but these approaches are easy to fall into overfitting, and the results are hard to interpret.
Inspired by the works of [26, 41, 43], we propose a novel deep representation framework based on SR named as graph regularized deep sparse representation (GRDSR) for detecting anomaly data in the high-dimensional space, as shown in Figure 1. Similar to the residual block on the residual net [50], we introduce the graph regularization to the deep features on each layer to maintain the local geometric structure. Furthermore, the $L_1$-norm is applied to learn the deep sparse representations to avoid overfitting. Unlike DNN-based anomaly detection methods, there are fewer parameters in our proposed approach. More importantly, the proposed approach is simpler and more straightforward, which can obtain interpretable results. The experiments are carried out on ten benchmark datasets and the experimental results verify the effectiveness of the proposed approach.

The main contributions of the proposed approach are given as below:

(1) This paper employs a deep feature representation framework to detect anomalies. Different from the traditional single layer SR-based methods, the proposed framework performs deep representation on the coefficient matrix so that the obtained hierarchical deep feature representations are more discriminative.

(2) Unlike the DNN-based methods, the proposed SR-based deep representation framework has a multi-layer linear structure. Therefore, the extracted deep feature representations have stronger interpretability.

(3) To make the deep feature representation preserve the intrinsic geometry of the original high-dimensional data, the graph regularization term is integrated into the deep feature representation framework by constructing a nearest neighbor graph to model the manifold structure. Besides, we impose a sparse constraint on the deep feature representations which makes the features be more sparsity and discriminative.

The rest of paper is organized as follows. In Section 2, there is a brief introduction of sparse coding and the graph regularization term. Section 3 introduces the proposed method in detail. In Section 4, we conduct extensive experiments on public datasets to test the performance of the proposed method. And finally, we conclude our study in Section 5.

2. Related Works

In this section, we will make a brief introduction of sparse coding and the graph regularization.

2.1. Sparse Coding. Suppose that the $m$-dimensional data $X$ has $n$ samples (i.e., $X = \{x_1, \ldots, x_n\} \in \mathbb{R}^{m \times n}$); spares’ coding aims to find a dictionary matrix constructed by a set of basis vectors that capture high-level semantics from the original high-dimensional data. Let $W = [w_1, \ldots, w_k] \in \mathbb{R}^{m \times k}$ be the over-complete dictionary matrix in which the $k$ columns are called as atoms. $H = [h_1, \ldots, h_n] \in \mathbb{R}^{k \times n}$ is the representation coefficient matrix. With the usage of dictionary $W$, the data sample $x_j$ can be reconstructed as $x_j \approx \sum_{i=1}^{k} w_i h_{ij}$. Therefore, $x_j$ can be regarded as a sparse linear combination of new basis and $H_{ij}$ is the combination coefficient.

Usually, spares’ coding can be seen as an optimization problem and the objection function is defined as

$$O_{SC} = \min_{W,H} \|X - WH\|^2_F + \beta \sum_{i=1}^{n} f(h_i),$$

where $\|\cdot\|^2_F$ represents the Frobenius norm and $f(\cdot)$ is the function to measure the sparse. For convenience, $f(\cdot)$ can be chosen as the $L_0$-norm, which counts the nonzero entries. Unfortunately, the optimization problem of equation (1) has been proven to be an NP-hard problem. Therefore, we use the $L_1$-norm to replace the $L_0$-norm so that it becomes a convex relaxation of the original problem and the objective function can be rewritten as

$$O_{SC} = \min_{W,H} \|X - WH\|^2_F + \beta \sum_{i=1}^{n} \|h_i\|_1,$$

where $\|\cdot\|_1$ is the $L_1$-norm.

2.2. Graph Regularization. For given two data points $x_i$ and $x_j$, $h_i$ and $h_j$ are the corresponding feature representations with respect to the learned new basis. If $x_i$ and $x_j$ are close in the intrinsic geometry of the data distribution, then $h_i$ and $h_j$ are also close to each other, which is called locality assumption. To achieve locality assumption, the manifold structure of the high-dimensional data is introduced, which can be represented by a Laplacian matrix.

Firstly, we define a graph $G = (V, E, S)$, where $V$ is the set of nodes, $E$ is the set of edges, and $S$ is a weight matrix of $E$. Generally, some methods like $k$-NN firstly judge whether a pair of points is connected, and then, the weights on the edges are computed. There are many ways to compute the weight matrix. Here, three most commonly used methods are introduced as follows:

(1) 0-1 weighting:

$$S_{ij} = \begin{cases} 1, & \text{if nodes } i \text{ and } j \text{ are connected,} \\ 0, & \text{others.} \end{cases}$$

(2) Heat kernel weighting:

$$S_{ij} = \begin{cases} e^{-\|x_i - x_j\|^2_2 / \sigma}, & \text{if nodes } i \text{ and } j \text{ are connected,} \\ 0, & \text{others,} \end{cases}$$

where $\sigma$ is the hyperparameter.
Dot-product weighting:

\[ S_{ij} = \begin{cases} x_i^T x_j, & \text{if nodes } i \text{ and } j \text{ are connected,} \\ 0, & \text{others.} \end{cases} \]  \tag{5}

Equation (5) can be equivalent to cosine similarity if \( x \) is normalized to 1. \( W^+ \) the weight matrix is also called the similarity matrix.

Finally, the smoothness of the feature representation is measured by the similarity matrix, which is defined as follows:

\[
R = \frac{1}{2} \sum_{i,j} S_{ij} = \text{Tr}(HDHT) - \text{Tr}(HSHT) = \text{Tr}(HLHT),
\]

where \( D \) is a diagonal matrix, \( D_{ii} = \sum_j S_{ij} \), \( L \) is Laplacian matrix and \( L = D - S \), and \( \text{Tr}(\bullet) \) denotes the trace of a matrix.

### 3. The Proposed Method

In this section, the objection function of the proposed approach is introduced first. Next, an iteration scheme is proposed to solve the objection function. Then, a criterion for anomaly detection is provided. At last, convergence analysis of the proposed optimization algorithm is given.

#### 3.1. The Objection of GRDSC.

Firstly, similar to MF, we represent \( X \) into the inner product of matrixes \( W \) and \( H \); therefore, the process can be represented by

\[
O_{\text{MF}} = \min_{W,H} \|X - WH\|_F^2. \tag{8}
\]

Since the traditional MF method only contains a single layer structure, it just extracts the shallow features so that the learned basis may contain complex hierarchical information. To address this disadvantage, the deep representation framework is proposed. Different from the existing methods, we further decompose the learned basis to get a better higher-level feature representation from the original data. Moreover, the multilayer structure can also learn multiple hidden basis of the original data. The objective function of deep representation framework can be represented as

\[
O_{\text{DeepMF}} = \min_{W_l,H_l} \|X - \Lambda_{l-1} W_l H_l\|_F^2, \tag{9}
\]

where \( l \) is the layer number, \( \Lambda_{l-1} = W_1 W_2 \cdots W_{l-1} \), \( \Lambda_0 = I_m \) is the identity matrix. and \( W_i \) and \( H_i \) are temporary variables generated in the calculation process.

Next, the aforementioned deep representation framework in equation (9) does not take the geometrical information of data into consideration, which may lead to the poor feature representation when the data have complex manifold structures. Therefore, in order to preserve the local geometric structure information, the graph regularize term is introduced to guide the feature representation, i.e., similar samples are grouped into the same cluster. The graph regularize term can be defined as follows:

\[
O_{\text{Graph}} = \min_{H_i} \sum_{r=1}^{l} \text{Tr}(H_i L H_i^T). \tag{10}
\]

Then, in order to enhance the discriminant ability of the deep feature, a sparsity constraint of the deep feature representation is added, which can be defined as

\[
O_{\text{sparsity}} = \min_{H_i} \sum_{r=1}^{n} \|H_i\|_{1}, \tag{11}
\]

where \( \|\bullet\|_1 \) denote the \( L_1 \)-norm of vector.
At last, taking equations (9)–(11) into consideration, the objective function of the proposed method can be summarized as

\[
O_{\text{GRDSC}} = \min_{W_l, H_l} \left\{ \|X - \Lambda_{l-1} W_l H_l\|_F^2 + \alpha \sum_{i=1}^{l-1} \text{Tr} \left( H_i L H_i^T \right) + \beta \sum_{i=1}^{k_l} \| h_i \|_1 \right\},
\]

where \(\alpha \geq 0\) and \(\beta \geq 0\) are two tradeoff parameters and \(w_i\) and \(h_i\) are the vectors of the final dictionary matrix \(W_l\) and the coefficient matrix \(H_l\), respectively.

### 3.2. The Optimization of GRDSC

Since the objective function in equation (12) is not convex in both \(W_l\) and \(H_l\), it is very hard to get the globally optimal solution. To deal with this problem, this paper proposes an iterative updating algorithm to achieve the local optimal solution. Similar to the expectation maximization algorithm, we update one variable and fix the rest variables, and all variables are alternately updated. Additionally, a layer-by-layer processing strategy is applied to simplify the algorithm flow. Since the last layer is different from others, we will deal with it separately.

#### 3.2.1. Update Rule for \(i\)th Layer \((i < l)\)

Because of the objective function for each layer is similar, we just take the \(i\)th layer for instance. The optimal problem can be represented as

\[
O^i_{\text{GRDSC}} = \min_{W_i, H_i} \left\{ \|X - \Lambda_{i-1} W_i H_i\|_F^2 + \alpha \sum_{i=1}^{l-1} \text{Tr} \left( H_i L H_i^T \right) \right\}.
\]

First, using the correlation properties of the matrix, we rewrite the objective function as

\[
J_i = \text{Tr} \left( (X - \Lambda_{i-1} W_i H_i)^T (X - \Lambda_{i-1} W_i H_i) \right) + \alpha \text{Tr} \left( H_i L H_i^T \right)
= \text{Tr} \left( X^T X - 2X^T \Lambda_{i-1} W_i H_i + H_i^T W_i^T \Lambda_{i-1} H_i \right) + \alpha \text{Tr} \left( H_i L H_i^T \right).
\]

Then, the Lagrange function is

\[
\mathcal{L} = \text{Tr} \left( X^T X \right) - 2\text{Tr} \left( X^T \Lambda_{i-1} W_i H_i \right) + \text{Tr} \left( H_i^T W_i^T \Lambda_{i-1} H_i \right) + \alpha \text{Tr} \left( H_i L H_i^T \right).
\]

Taking partial derivation of \(\mathcal{L}\) with respect to \(W_i\) and \(H_i\), respectively, we have

\[
\frac{\partial \mathcal{L}}{\partial W_i} = -2 \Lambda_{i-1}^T X H_i^T + 2 \Lambda_{i-1}^T \Lambda_{i-1} W_i H_i H_i^T,
\]

\[
\frac{\partial \mathcal{L}}{\partial H_i} = -2 W_i^T \Lambda_{i-1}^T X + 2 W_i^T \Lambda_{i-1}^T \Lambda_{i-1} W_i H_i + 2 \alpha H_i L.
\]

Setting \((\partial \mathcal{L}/\partial W_i) = 0\) and \((\partial \mathcal{L}/\partial H_i) = 0\), we have

\[
W_i^* = \left( \Lambda_{i-1}^T \Lambda_{i-1} \right)^{+} \Lambda_{i-1}^T X H_i H_i^T \left( H_i L H_i^T \right)^{+},
\]

\[
W_i^T \Lambda_{i-1}^T \Lambda_{i-1} W_i H_i + \alpha H_i L = W_i^T \Lambda_{i-1}^T X.
\]

where \((\cdot)^{+}\) denotes the pseudoinverse. See from equation (18), it is the Sylvester equation, and the optimal solution of \(H_i\) can be solved by Matlab function lyap.

#### 3.2.2. Update Rule for \(l\)th Layer

The optimization in the last layer is different from other layers because of the sparse regularization term. The objective function of the last layer can be represented as follows:

\[
O^l_{\text{GRDSC}} = \min_{W_l, H_l} \left\{ \|X - \Lambda_{l-1} W_l H_l\|_F^2 + \alpha \sum_{i=1}^{l-1} \text{Tr} \left( H_i L H_i^T \right) + \beta \sum_{i=1}^{k_l} \| h_i \|_1 \right\},
\]

where \(\| \cdot \|_1 \leq 1\). Next, we will discuss how to solve \(W_l\) and \(H_l\).

Computation of \(W_l\): when \(H_l\) is fixed, the dictionary \(W_l\) needs to be learned at first, and the problem of \(W_l\) can be described as
\( \phi(W_i) = \min \|H_{t-1} - W_i H_i\|_F^2, \)
\[
s.t. \|u\|^2_i \leq 1, \quad i \in \{1, \ldots, k_t\}. \tag{20} \]

Suppose that \( \Gamma = \{y_1, \ldots, y_{k_t}\} \) is the Lagrange multiplier corresponding to \( \|u\|^2_1 \geq 1 \). Then, we can get the Lagrange dual function as
\[
g(\Gamma) = \inf_{W_i} L_w(W_i, \Gamma)
= \inf_{W_i} \left(\|H_{t-1} - W_i H_i\|_F^2 + \sum_{i=1}^{k_t} y_i (\|w_i\|^2_1 - c)\right). \tag{21} \]

And, \( L_w(W_i, \Gamma) \) can be written as
\[
L_w(W_i, \Gamma) = \|H_{t-1} - W_i H_i\|_F^2 + \text{Tr}(W_i^T W_i A) - \text{Tr}(A)
= \text{Tr}(H_{t-1}^T H_{t-1}) - 2 \text{Tr}(W_i H_i H_{t-1}^T) + \text{Tr}(W_i^T W_i H_{t-1}^T)
+ \text{Tr}(W_i^T W_i A) - \text{Tr}(A). \tag{22} \]

where \( A \) is a diagonal matrix and \( A_{ij} = y_i \). Then, partial derivation of equation (22) with respect to \( W_i \) is
\[
\frac{\partial L_w(W_i, \Gamma)}{\partial W_i} = W_i H_i H_{t-1}^T - H_{t-1} H_{t-1}^T + W_i A. \tag{23} \]

Let equation (23) be equal to zero, and we have
\[
W_i^* = H_{t-1} H_{t-1}^T (H_{t-1} H_{t-1}^T + A)^{-1}. \tag{24} \]

Then, substituting equation (24) into equation (22), we have
\[
g(\Gamma) = \text{Tr}(H_{t-1}^T H_{t-1}) - \text{Tr}(H_{t-1} H_{t-1}^T (H_{t-1} H_{t-1}^T + A)^{-1} H_{t-1}^T) - \text{Tr}(A). \tag{25} \]

From equation (25), we can get the following Lagrange dual function:
\[
\min_{W_i} \frac{\sum_{i=1}^{k_t} (\|H_i\|_F^2 + \alpha \text{Tr}(H_i^2 H_i^T + A) + \beta \sum_{i=1}^{k_t} \|H_i\|_1^2)}{W_i = H_{t-1} H_{t-1}^T (H_{t-1} H_{t-1}^T + A)^{-1}}. \tag{26} \]

We can see that equation (28) is convex, but is non-differentiable because of the \( l_1 \)-regularization. Following the work of [41], we will adopt an optimization method based on coordinate descent to solve this issue.

The vector \( h_i^j \) should be updated individually and other vectors are fixed unchanged. So, we rewrite equation (26) as
\[
\min_{h_i^j} \left\{ \|H_i - W_i h_i^j\|_F^2 + \alpha \sum_{i,j=1}^{n} L_n(h_i^j)^2 + \beta \sum_{i=1}^{k_t} \|h_i^j\|_1 \right\}. \tag{29} \]

And, the optimization problem about \( h_i^j \) is
\[
\min_{h_i^j} f(h_i^j) = \|h_i^j - W_i h_i^j\|_2^2 + \alpha L_n(h_i^j)^2 + 2 \alpha (\sum \Lambda_i h_i^j)^T h_i^j + \beta \sum_{i=1}^{k_t} \|h_i^j\|_1. \tag{30} \]

where \( (h_i^j)^j \) is the \( j \)th coefficient of \( h_i^j \).

We use subgradients of \( f(h_i^j) \) to deal with the non-differentiable points; therefore, equation (30) can be rewritten as
\[
f(h_i^j) = v(h_i^j) + \beta \sum_{j=1}^{n} \left| (h_i^j)^j \right|, \tag{31} \]

where \( v(h_i^j) = \|h_i^j - W_i h_i^j\|_2^2 + \alpha (h_i^j)^T h_i^j + 2 \alpha \sum \Lambda_i (h_i^j)^T h_i^j. \) This problem of equation (31) can be solved by feature-sign search algorithm proposed in [51].

The optimization algorithm of GRDSC is summarized in Algorithm 1, and algorithm flowchart is shown in Figure 2.

3.3 Anomaly Detection. In this section, we will give the description of the anomaly detection using the proposed

GRDSC approach. Similar to SR-based methods for anomaly detection, the reconstruction error is employed to distinguish anomalies. Because the anomalies and normal data belong to different distribution and the number of anomalies is much less than normal data, the model is easily learned from classes with a large number of samples and ignores classes with a small number of samples. In other words, the reconstruction quality of anomalies is poor that has a higher anomaly score. Once the optimal \( W_i^*, H_i^* \), and \( \Lambda_i^* \) are obtained, the reconstruction error between the original data and the reconstruction data is measured as follows:
\[
\text{RE}(X, X') = \|X - X'\|_F^2 = \|X - \Lambda_{t-1} W_i H_i^T\|_F^2, \tag{32} \]

where \( X' \) denotes as the reconstruction data.
Input training samples and parameters

Initialize variables and set $t=1$

Compute matrix $L$

Compute matrix $\Lambda_{i-1}$

Update matrix $W_i$

Update matrix $H_i$

Reach max iteration

No

Yes

Output matrices

$\{H_1, H_2, \ldots, H_l\}$

$\{W_1, W_2, \ldots, W_l\}$

Figure 2: A flowchart of the proposed optimization algorithm.

Algorithm 1: The optimization algorithm of GRDSC.
Furthermore, for every single sample, the reconstruction error can be computed as
\[
O_j = \|x_j - \hat{x}^2\|_F = \|x_j - \sum_{i=1}^{k} (\Lambda_i^j W_i^* \Lambda_i^j) x_j\|_F^2, \tag{33}
\]
where \((W_i^*)\) is the \(i\)th column in the \(W_i^*\). Then, we rank the score set \(\{O_j, j = 1, \ldots, n\}\) in descending order and those samples with high anomaly scores are marked as anomalies. The anomaly detection process is summarized in Algorithm 2.

3.4. Convergence Analysis. We will discuss the convergence of the proposed algorithm in this section. The optimization process can be divided into two subproblems as formulated in equations (13) and (19). Then, each subproblem can be divided into two subproblems. Thus, four subproblems can be solved iteratively. Let \(\phi(W_t, H_t, W_t, H_t)\) be the objective function value of GRDSR, and we have the following theorem.

Theorem 1. The objective function value is nonincremental if Algorithm 1 is used to solve \(\phi(W_t, H_t, W_t, H_t)\).

Proof. Let \(\phi(W_t, H_t, W_t, H_t)\) denote the value of objective function in the \(t\)th iteration. We first can solve the subproblem \(\min_{W_t} \phi(W_t, H_t, W_t, H_t)\) when \(H_t, W_t,\) and \(H_t\). The optimal solution \(W_t^{t+1}\) in the \(t + 1\)th iteration can be obtained via equation (17). Since the subproblem is convex, we can obtain
\[
\phi(W_t^{t+1}, H_t^{t+1}, W_t^{t+1}, H_t^{t+1}) \leq \phi(W_t^{t+1}, H_t^{t+1}, W_t^{t+1}, H_t^{t+1}). \tag{34}
\]
Next, by fixing \(W_t, H_t, W_t,\) and \(H_t\), we can solve the subproblem \(\min_{H_t} \phi(W_t, H_t, W_t, H_t)\). The optimal value of \(W_t^{t+1}\) can be obtained by solving equation (18). Since this subproblem is a convex problem, we then have
\[
\phi(W_t^{t+1}, H_t^{t+1}, W_t^{t+1}, H_t^{t+1}) \leq \phi(W_t, H_t, W_t, H_t). \tag{35}
\]
Then, we fix \(W_t^{t+1}, W_t^{t+1},\) and \(H_t^{t+1}\) to solve the subproblem \(\min_{H_t} \phi(W_t, H_t, W_t, H_t)\) as depicted in equation (24) according to literature [41], so this subproblem is convex, and we can obtain
\[
\phi(W_t^{t+1}, H_t^{t+1}, W_t^{t+1}, H_t^{t+1}) \leq \phi(W_t^{t+1}, H_t^{t+1}, W_t^{t+1}, H_t^{t+1}). \tag{36}
\]
Similarly, we solve the subproblem \(\min_{W_t} \phi(W_t, H_t, W_t, H_t)\) as depicted in equation (28) by fixing \(W_t, H_t, W_t,\) and \(H_t^{t+1}\). Then, we can obtain
\[
\phi(W_t^{t+1}, H_t^{t+1}, W_t^{t+1}, H_t^{t+1}) \leq \phi(W_t, H_t, W_t^{t+1}, H_t^{t+1}). \tag{37}
\]
Combining equations (34)–(37), we can obtain
\[
\phi(W_t^{t+1}, H_t^{t+1}, W_t^{t+1}, H_t^{t+1}) \leq \phi(W_t^{t+1}, H_t^{t+1}, W_t^{t+1}, H_t^{t+1}). \tag{38}
\]
Therefore, Theorem 1 is proved.

4. Experiment Results and Analysis

To evaluate the performance of the proposed method, we conduct extensive experiments on real-world anomaly detection datasets and compare it with the state-of-art methods. The results show that the proposed method achieves better performance on most of the evaluated datasets.

4.1. Datasets’ Descriptions. The datasets are chosen randomly from the study of Campose et al. [52]. Follow the work of [28], the missing values are removed and categorical variables are converted into numerical format. Besides, all of the data are normalized. The detail descriptions of the datasets are given below, and a brief summary of the datasets is also shown in Table 1.

Arrhythmia is a medical dataset about hypothyroidism, which contains three classes as normal (not hypothyroid), hyperfunction, and subnormal functioning. For anomaly detection, we treat hyperfunction and subnormal classes as abnormal.

Spambase is a dataset representing emails categorized as spam (outliers) or nonspam. The spam emails come from postmaster and individuals who had filed spam.

Wisconsin Prognostic Breast Cancer (WPBC) is collected from patients seen by Dr. Wolberg since 1984. Each sample represents follow-up data for one breast cancer case. The class R (recur) is marked as anomaly and the class N (nonrecur) is marked as normal.

Cardiotocography is a medical dataset which consists of measurements of fetal heart rate (FHR) and uterine contraction (UC) features on cardiotograms. It is classified into normal, suspect, and pathologic by experts. For anomaly detection, the suspect class is discarded.

Ionosphere contains signals’ data from good radars and bad radars in ionosphere where the ‘bad’ class is treated as anomaly and ‘good’ class is regarded as normal.

WBC records the measurements for breast cancer cases including benign and malignant two classes, where the malignant is considered as anomaly.

Arrhythmia is a multiclass classification dataset which contains 15 type of cardiac arrhythmia. The healthy people are treated as normal data and patients are marked as anomaly.

Pen digits collected 250 samples from 44 writers which are classified into 10 classes (0 . . . 9). In the experiment, Class 4 is defined as anomaly.

Stamps contain genuine stamps and forged stamps. The genuine stamps are using ink to print and treated as normal data. The forged data are photocopied or scanned and treated as anomaly.

Heart is an image dataset which describes diagnosing of cardiac Single Proton Emission Computed Tomography (SPECT). The original data are downsampled and affected patients are considered anomaly.
4.2. Score Metrics. As mentioned before, we compute the reconstruction error for each sample and obtain an anomaly score set \( \{ O_j, j = 1, \ldots, n \} \). The higher score the observations associated with, the higher probability it be flagged as anomalies. However, the cut-off threshold is hard to selection. A common and widely used approach in practice is to select the top \( N \) instances and mark these as potential anomalies. In this paper, we follow this approach to mark the top \( N \) samples as anomalies and treat the rest as normal instances. For better evaluating the performance of the proposed method, we set \( N \) as the number of total anomalies in corresponding datasets.

Furthermore, the metric called precision at \( N(P@N) \) [52] is adopted to evaluate the performance of all of the methods. \( P@N \) is a straightforward metric and defined as the proportion of true outliers in the detected values which can be flagged as anomalies. Considering a dataset DB with \( n \) instances, \( O \subset DB \) is the anomaly set and \( O \cap DB \) is the normal data set, \( DB = O \cup I \). \( P@N \) is defined as

\[
P@N = \frac{\# \{ o \in O | \text{rank}(o) \leq N \}}{N},
\]

where \( N = |O| \) is the number of anomaly samples.

4.3. Visualization Results and Analysis. To more directly show the results of all of the detection processes, we plot the reconstruction error. Considering the proposed GRDSR method is based on MF in essence, we also choose the MF-based approaches for comparison. The selected comparison approaches include graph regularized sparse coding (GraphSC) [41], sparse representation (SR) [33], offline neighborhood structure-assisted NMF (Offline NS-NMF) [28], Online NS-NMF [28], graph regularized NMF (GNNMF) [26], and symmetric NMF (SNMF) [27]. Besides, the ionosphere dataset is selected as representative for simplify. Considering the visualization results of compared algorithms may be affected by sample imbalance, we randomly select normal samples to balance the abnormal samples. The results are shown in Figure 3.

Form Figure 3, we can easily see that the reconstruction error calculated by the proposed GRDSR method is naturally divided into two parts, which means that the anomalies have bigger reconstruction error. Meanwhile, most of the normal data have smaller reconstruction error and distributed at the bottom right of the figure. In contrast to GRDSR, other MF-based methods have more difficulties to distinguish anomalies and normal data by the reconstruction error.

4.4. Comparison with the State-of-the-Art Methods. To further explore the performance of the proposed method, in this section, we first test the MF-based methods on all of the datasets. Since initialization is very important for MF-based methods, we fix the initialize method for all of the approaches. For GraphSC, SR, and GRDSR, we turn graph regularization parameter \( \alpha \) and sparse regularization parameter \( \beta \) from a set \( \{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3\} \) and report the best result. Following the work of [28], for all single-layer methods, the number of latent features or clusters \( d \) is set as 5. This is because changing \( d \) in the range of [5, 15] does not affect MF-based methods when \( d = 5 \) makes most of the methods perform well. For Offline NS-NMF, we set \( \alpha = 0.8 \) and \( \gamma = 0.2 \); for Online NS-NMF, we set \( \alpha = 0.8 \) and \( \varepsilon = 20 \). In GNNMF, we set the neighborhood graph construction

\[
\text{Input:} \text{ Origin data } X \text{ and the number of anomaly samples } N \\
\text{Factorization matrices } \{ W_i \} \text{ and } \{ H_j \}, i = 1, \ldots, I \\
\text{Output:} \text{ The selected } N \text{ anomaly samples} \\
(1) \text{ Compute } A_{l:1} \text{ by } \{ W_i \} \text{ and } \{ H_j \} \\
(2) \text{ For } j = 1: n \\
(3) \text{ Compute reconstruction error by } O_j = \| x_j - \sum_{i=1}^k (A_{l:1} W_i) (H_j)_l \|_F^2 \\
(4) \text{ End for} \\
(5) \text{ Sort the score set } \{ O_j, j = 1, \ldots, n \} \text{ in a descending order.} \\
(6) \text{ Mark the samples associated with top } N \text{ scores as anomaly samples.} \\

\text{Algorithm 2: The procedure of anomaly detection}

\begin{table}[h]
\centering
\begin{tabular}{|l|c|c|c|}
\hline
Dataset & Number of sample & Feature dimension & Outliers (%) \\
\hline
Annthyroid & 7200 & 21 & 347(4.82) \\
SpamBase & 4601 & 57 & 280(6.09) \\
WPBC & 198 & 33 & 47(23.73) \\
Cardiotocography & 2126 & 21 & 86(4.05) \\
Ionosphere & 351 & 32 & 126(35.90) \\
WBC & 454 & 9 & 10(2.20) \\
Arrhythmia & 450 & 259 & 12(2.67) \\
Pen digits & 9868 & 16 & 20(4.71) \\
Stamps & 340 & 9 & 10(2.20) \\
Heart & 270 & 13 & 7(2.59) \\
\hline
\end{tabular}
\caption{Public benchmark datasets collected from real world.}
\end{table}
Figure 3: Continued.
parameter \( k \) in \( k\)NN as 5. Besides, we use 0-1 weighting as the weighting method. For SNMF, Gaussian similarity measure is utilized to construct the input similarity matrix.

Additionally, for fair comparison, the similarity matrix is constructed in our model identical to GraphSC and GNMF. In this experiment, the number of the hidden features for each layer, i.e., the layers’ size, is set as \([ (m/2), (m/4) ]\), where \( m \) means the dimension of the dataset. In practice, \( m/2 \) and \( m/4 \) are rounded to the nearest integer values. The settings for all of the MF-based methods are summarized in Table 2. The results are shown in Table 3. From Table 3, we can draw the following conclusions. Firstly, GNMF, Online NS-NMF, and Offline NS-NMF perform better than NMF and SNMF. Moreover, GraphSC performs better than SR. These results demonstrate that the graph regularization is helpful to preserve the intrinsic geometry during the process of feature representation. Secondly, GraphSC can achieve better performance than GNMF, which proves that the sparse representation with sparsity constraint can improve the discriminant ability of feature representation. Finally, compared with all MF-based methods,

![Figure 3: Reconstruction error for MF-based methods on ionosphere dataset. (a) GRDSR, (b) SNMF, (c) GraphSC, (d) SR, (e) Offline NS-NMF, (f) Online NS-NMF, (g) NMF, and (h) GNMF.](image-url)
Table 4: Results of non-NMF-based methods.

| Dataset       | GRDSR | COF  | LDF  | kNN  | ODIN | LOF  | kNNW | SLOF | LoOP | INFLO | LDOF | FABOD | AER | DAGMM | Total anomalies |
|---------------|-------|------|------|------|------|------|------|------|------|-------|------|-------|-----|-------|-----------------|
| Annthyroid    | 30    | 5    | 5    | 3    | 3    | 3    | 3    | 3    | 3    | 3     | 3    | 3     | 10  | 94    | 347             |
| SpamBase      | 77    | 55   | 48   | 76   | 65   | 55   | 76   | 44   | 55   | 55    | 55   | 55    | 64  | 31    | 61              |
| WPBC          | 14    | 11   | 13   | 10   | 12   | 10   | 9    | 10   | 11   | 13    | 10   | 10    | 9   | 47    |                 |
| Cardiotocography | 32    | 28   | 28   | 30   | 18   | 22   | 31   | 27   | 26   | 23    | 25   | 29    | 29  | 27    | 86              |
| Ionosphere    | 108   | 107  | 104  | 99   | 105  | 108  | 104  | 101  | 101  | 101   | 108  | 63    | 79  | 126   |                 |
| WBC           | 9     | 8    | 8    | 4    | 8    | 4    | 8    | 6    | 4    | 4     | 4    | 7     | 6   | 2     | 10              |
| Arrhythmia    | 5     | 5    | 5    | 3    | 3    | 3    | 3    | 3    | 3    | 3     | 3    | 3     | 2   | 2     | 12              |
| Pen digits    | 3     | 3    | 1    | 0    | 0    | 1    | 1    | 1    | 1    | 0     | 1    | 1     | 0   | 5     | 20              |
| Stamps        | 4     | 5    | 4    | 3    | 3    | 4    | 4    | 4    | 4    | 4     | 5    | 6     | 3   | 0     | 16              |
| Heart         | 5     | 4    | 4    | 4    | 4    | 4    | 4    | 4    | 4    | 4     | 2    | 2     | 2   | 2     | 7               |

The bold values mean the best result for each dataset.
the proposed method either performs better or achieves the same best performance on all datasets except ionosphere dataset. This proves the effectiveness of the proposed GRDSR method under the deep framework based on SR for anomaly detection.

Then, in order to fully evaluate the performance of the proposed GRDSR method, we compare the proposed GRDSR method with other non-MF-based methods. Hence, 12 nearest neighborhood-based methods are chosen for comparison. These methods are $k$NN [8], $k$NN weight ($k$NNW) [53], local outlier factor (LOF) [7], outlier detection using indegree number (ODIN) [8], local distance-based outlier factor (LDOF) [9], connectivity-based outlier factor (COF) [54], local outlier probabilities (LoOP) [55], influenced outlierness (INFLO) [56], local density factor (LDF) [57], fast angle-based outlier detection (FastABOD) [58], and kernel density estimation outlier detection (KDEOD) [59]. Among them, kNN, ODIN, and $k$NNW can be seen as global methods. Another large category is derived from LOF, which can be seen as local methods. Besides, we also employ two DNN-based methods for comparison. They are autoencoder with an embedding regularizer (AER) [60] and deep autoencoding Gaussian mixture model (DAGMM) [61].

The number of the nearest neighbors ($k$) is required to be set in non-NMF-based methods. According to the guideline of [62], this paper tunes the values of $k$ from 1 to 100 and the best value will be chosen. In our experiment, we only report the true positive detection number of all of the test methods. The results compared with non-NMF-based methods and DNN-based methods are shown in Table 4. The bolded entries mean the best performance in the corresponding datasets. Seen from Table 4, our proposed GRDSR method performs better in most cases except for the DANGMM and FABOD methods and achieves

| Datasets           | $\alpha$ | $\beta$ |
|--------------------|----------|---------|
| Annthyroid         | $10^{-2}$| 1       |
| SpamBase           | $10^{-3}$| $10^{-3}$|
| WPBC               | $10^{-3}$| 1       |
| Cardiotocography   | $10^{-2}$| $10^{-1}$|
| Ionosphere         | $10^{-2}$| 1       |
| WBC                | 1        | $10^{-3}$|
| Arrhythmia         | $10^{-2}$| 1       |
| Pen digits         | $10^{-3}$| 1       |
| Stamps             | $10^{-3}$| $10^{-3}$|
| Heart              | $10^{-3}$| 10      |

Figure 4: The performance of the GRDSR method with varied $\alpha$ on different datasets. (a) Ionosphere, (b) SpamBase, (c) Cardiotocography, and (d) WBC.
Figure 5: The performance of GRDSR method with varied $\beta$ on different datasets. (a) Ionosphere, (b) SpamBase, (c) Cardiotocography, and (d) WBC.

Figure 6: Continued.
Figure 6: Continued.
the best results on the Annthyroid, Pen digits, and Stamps datasets, respectively. In additional, the DANGMM method performs too much better than other methods on the Annthyroid dataset. Generally speaking, the proposed GRDSR method has made great progress than most of NMF-based methods and all non-NMF-based methods.

4.5. Parameter Sensitive Analysis. The proposed method has two trade off parameters, \( \alpha \) and \( \beta \), which are needed to be set at the beginning. In order to explore the settings of these parameters on each dataset, we conduct extensive experiments. As mentioned above, these parameters are set varied in the range of \( \{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3\} \), and we use a grid-search strategy to find the best parameter settings. The combinations of optimal parameters on different datasets are reported on Table 5. From this table, we can see that these parameters need to be set at a small value to reach a good performance in most cases. Compared with \( \beta \), \( \alpha \) often behaves smaller. This phenomenon shows that all datasets have a strong local structure.

| Datasets        | Running time (s) |
|-----------------|------------------|
| Annthyroid      | 163.32           |
| SpamBase        | 61.35            |
| WPBC            | 2.55             |
| Cardiotocography| 23.85            |
| Ionosphere      | 4.33             |
| WBC             | 2.34             |
| Arrhythmia      | 15.21            |
| Pen digits      | 232.84           |
| Stamps          | 3.16             |
| Heart           | 1.85             |

In order to further visualize the influence of these two parameters, we randomly select four datasets. In order to make the visualization results more intuitively, we show one parameter and keep another fixed at the best. The results are reported on Figures 4 and 5. From Figure 4, it can be observed that, for Ionosphere, Cardiotocography, and SpamBase, the performances are first improved with the increase of the values of \( \alpha \). However, when the performances reached at the best, the performances begin to reduce or keep stable. However, for WBC, the trend is converse. This may be the characteristic of this datasets, that is to say, the sparsity is weaker than others so that the penalty factor needs to be set at a bigger value. From Figure 5, we can see that the trend seems to be identical for all datasets. The performance is stable when \( \beta \) is small. However, when \( \beta \) exceeds a certain value, performance will be decreased until it is stable again. The thresholds are different on different datasets.

4.6. Convergence Evaluation. The updating rules of GRDSR are essentially iterative and the convergence for the objective function value is theoretically guaranteed. Now, we investigate how fast the rules can reach convergence. We conduct the experiments at all datasets, and the results are shown on Figure 6. For each figure, \( x \)-axis is the iteration number and \( y \)-axis denotes the objective function value. It shows that the proposed GRDSR method can reach convergence after 100 iterations at most of the datasets.

4.7. Running Time. In order to show the efficiency of the proposed algorithm more intuitively, we test the running time of our proposed GRDSR method on each dataset. Our algorithm is implemented by MATLAB, and these experiments are carried out on PC machine with Intel i9 9900K 3.60 GHz and 32 GB memory. We record the running time when the iteration
number is set to be 100 as report in Table 6. From the results, the running time of the proposed method is acceptable.

5. Conclusions

Different from the traditional MF-based methods, we propose a deep representation framework based on sparse representation named graph regularized deep sparse representation (GRDSR) to learn the deep feature representation for anomaly detection. In GRDSR, we first apply multilayers’ factorization to extend the single matrix factorization. Next, we add the graph regularize term into each layer factorization to capture the intrinsic geometric structure information of the original data. Then, we introduce a sparsity constraint-based l1-norm to avoid the overfitting problem and extract more discriminative deep feature representations. Last, we utilize a criterion-based reconstruction error to detect anomaly data. The experiments are carried out on ten widely used datasets. According to the experimental results, we can learn that the proposed method outperforms the state-of-the-art approaches.

Data Availability

The data used to support the findings of the study are derived from public domain resources.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the National Natural Science Foundation of China, under Grant nos. 62062040, 62102270, 62006174, and 61967010, the Chinese Postdoctoral Science Foundation (Grant nos. 2019M661117), Science and Technology Research Project of Jiangxi Provincial Department of Education (Grant nos. GJJ191709 and GJJ191689), Fundamental Research Funds for the Central Universities, under Grant no. 2412019FZ049, Graduate Innovation Foundation Project of Jiangxi Normal University, under Grant no. YJS2020045, Scientific Research Fund Project of Liaoning Provincial Department of Education (no. JYT19040), and Young Talent Cultivation Program of Jiangxi Normal University.

References

[1] A. Boukerche, L. Zheng, and O. Alfandi, “Outlier detection,” ACM Computing Surveys, vol. 53, no. 3, pp. 1–37, 2020.
[2] Y. Yang Zhang, N. Meratnia, and P. Havinga, “Outlier detection techniques for wireless sensor networks: a survey,” IEEE Communications Surveys & Tutorials, vol. 12, no. 2, pp. 159–170, 2010.
[3] M. Ahmed, A. Naser Mahmood, and J. Hu, “A survey of network anomaly detection techniques,” Journal of Network and Computer Applications, vol. 60, pp. 19–31, 2016.
[4] Y. Yi, W. Zhou, Y. Shi, and J. Dai, “Speedup two-class supervised outlier detection,” IEEE Access, vol. 6, pp. 63923–63933, 2018.
[5] Y. Yi, Y. Shi, W. Wang, G. Lei, J. Dai, and H. Zheng, “Combining boundary detector and SND-SVM for fast learning,” International Journal of Machine Learning and Cybernetics, vol. 12, no. 3, pp. 689–698, 2021.
[6] C. Désir, S. Bernard, C. Petitetjean, and L. Heutte, “One class random forests,” Pattern Recognition, vol. 46, no. 12, pp. 3490–3506, 2013.
[7] M. M. Breunig, H. P. Kriegel, R. T. Ng et al., “LOF: identifying density-based local outliers,” in Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, pp. 93–104, Dallas, TX, USA, May 2000.
[8] V. Hautamaki, I. Karkkinen, and P. Franti, “Outlier detection using k-nearest neighbour graph,” vol. 3, pp. 430–433, in Proceedings of the 17th International Conference on Pattern Recognition, 2004 (ICPR 2004), vol. 3, pp. 430–433, IEEE, Cambridge, UK, 2004.
[9] K. Zhang, M. Hutter, and H. Jin, “A new local distance-based outlier detection approach for scattered real-world data,” in Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 813–822, Springer, Bangkok, Thailand, April 2009.
[10] S. D. Bay and M. Schwabacher, “Mining distance-based outliers in near linear time with randomization and a simple pruning rule,” Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 29–38, 2003.
[11] A. Zimek, E. Schubert, and H.-P. Kriegel, “A survey on unsupervised outlier detection in high-dimensional numerical data,” Statistical Analysis and Data Mining, vol. 5, no. 5, pp. 363–387, 2012.
[12] H. P. Kriegel, M. Schubert, and A. Zimek, “Angle-based outlier detection in high-dimensional data,” in Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 444–452, Las Vegas, NV, USA, August 2008.
[13] J. Li, H. Izakian, W. Pedrycz, and I. Jamal, “Clustering-based anomaly detection in multivariate time series data,” Applied Soft Computing, vol. 100, p. 106919, 2021.
[14] G. Pu, L. Wang, J. Shen et al., “A hybrid unsupervised clustering-based anomaly detection method,” Tsinghua Science and Technology, vol. 26, no. 2, pp. 146–153, 2020.
[15] J. Lei, S. Fang, W. Xie, Y. Li, and C.-I. Chang, “Discriminative reconstruction for hyperspectral anomaly detection with spectral learning,” IEEE Transactions on Geoscience and Remote Sensing, vol. 58, no. 10, pp. 7406–7417, 2020.
[16] W. Luo, W. Liu, D. Lian et al., “Video anomaly detection with sparse coding inspired deep neural networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 3, pp. 1070–1084, 2021.
[17] Y. J. Lee, Y. R. Yeh, and Y. C. F. Wang, “Anomaly detection via online oversampling principal component analysis,” IEEE Transactions on Knowledge and Data Engineering, vol. 25, no. 7, pp. 1460–1470, 2012.
[18] X. Guan, W. Wang, and X. Zhang, “Fast intrusion detection based on a non-negative matrix factorization model,” Journal of Network and Computer Applications, vol. 32, no. 1, pp. 31–44, 2009.
[19] W. Zhang, X. Lu, and X. Li, “Similarity constrained convex nonnegative matrix factorization for hyperspectral anomaly detection,” IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 7, pp. 4810–4822, 2019.
[20] L. Xiong, X. Chen, and J. Schneider, “Direct robust matrix factorization for anomaly detection,” in Proceedings of the
H. Du, Y. Zhang, L. Ma, and F. Zhang, "Structured discriminant analysis dictionary learning for pattern classification," Knowledge-Based Systems, vol. 216, p. 106794, 2021.

[36] Y. Cong, J. Yuan, and J. Liu, "Abnormal event detection in crowded scenes using sparse representation," Pattern Recognition, vol. 46, no. 7, pp. 1851-1864, 2013.

[37] T. Xiao, C. Zhang, and H. Zha, "Learning to detect anomalies in surveillance video," IEEE Signal Processing Letters, vol. 22, no. 9, pp. 1477–1481, 2015.

[38] Y. Xu, Z. Wu, J. Li et al., "Anomaly detection in hyperspectral images based on low-rank and sparse representation," IEEE Transactions on Geoscience and Remote Sensing, vol. 54, no. 4, pp. 1990–2000, 2015.

[39] Q. Ling, Y. Guo, Z. Lin et al., "A constrained sparse representation model for hyperspectral anomaly detection," IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 4, pp. 2358–2371, 2018.

[40] B. Palastr, L. Bousof, S. D’Escrivan, and J.-Y. Tournier, "Anomaly detection in mixed telemetry data using a sparse representation and dictionary learning," Signal Processing, vol. 168, p. 107320, 2020.

[41] M. Zheng, J. Bu, C. Chen et al., "Graph regularized sparse coding for image representation," IEEE Transactions on Image Processing, vol. 20, no. 5, pp. 1327–1336, 2010.

[42] Y. Yi, J. Wang, W. Zhou, C. Zheng, J. Kong, and S. Qiao, "Non-negative matrix factorization with locality constrained adaptive graph," IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 2, pp. 427–441, 2020.

[43] Y. He, K. Kavukcuoglu, Y. Wang et al., "Unsupervised feature learning by deep sparse coding," in Proceedings of the 2014 SIAM International Conference on Data Mining, pp. 902–910, Philadelphia, PA, USA, April 2014.

[44] P. Sharma, V. Abrol, and A. K. Sao, "Deep–sparse-representation-based features for speech recognition," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 25, no. 11, pp. 2162–2175, 2017.

[45] S. Tariyal, A. Majumdar, R. Singh, and M. Vatsa, "Deep dictionary learning," IEEE Access, vol. 4, pp. 10096–10109, 2016.

[46] S. Singh and A. Majumdar, "Deep sparse coding for non-intrusive load monitoring," IEEE Transactions on Smart Grid, vol. 9, no. 5, pp. 4669–4678, 2017.

[47] E.-J. Cheng, K.-P. Chou, S. Rajora et al., "Deep sparse representation classifier for facial recognition and detection system," Pattern Recognition Letters, vol. 125, pp. 71–77, 2019.

[48] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 665–674, Halifax, Canada, August 2017.

[49] D. Li, D. Chen, J. Goh et al., "Anomaly detection with generative adversarial networks for multivariate time series," 2018, https://arxiv.org/abs/1809.04758.

[50] K. He, X. Zhang, S. Ren et al., "Deep residual learning for image recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778, Las Vegas, NV, USA, June 2016.

[51] H. Lee, A. Battle, R. Raina et al., "Efficient sparse coding algorithms," Advances in Neural Information Processing Systems, pp. 801–808, 2007.

[52] G. O. Campos, A. Zimek, J. Sander et al., "On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study," Data Mining and Knowledge Discovery, vol. 30, no. 4, pp. 891–927, 2016.

[53] F. Angiulli and C. Pizzuti, "Outlier mining in large high-dimensional data sets," IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 2, pp. 203–215, 2005.
[54] J. Tang, Z. Chen, A. W. C. Fu et al., “Enhancing effectiveness of outlier detections for low density patterns,” in Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 535–548, Springer, Taipei, Taiwan, May 2002.
[55] H. P. Kriegel, P. Kröger, E. Schubert et al., “LoOP: local outlier probabilities,” in Proceedings of the 18th ACM Conference on Information and Knowledge Management, pp. 1649–1652, Hong Kong, China, November 2009.
[56] W. Jin, A. K. H. Tung, J. Han et al., “Ranking outliers using symmetric neighborhood relationship,” in Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 577–593, Springer, Singapore, April 2006.
[57] B. Tang and H. He, “A local density-based approach for outlier detection,” Neurocomputing, vol. 241, pp. 171–180, 2017.
[58] H. P. Kriegel, M. Schubert, and A. Zimek, “Angle-based outlier detection in high-dimensional data,” in Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 444–452, Las Vegas, NV, USA, August 2008.
[59] E. Schubert, A. Zimek, and H. P. Kriegel, “Generalized outlier detection with flexible kernel density estimates,” in Proceedings of the 2014 SIAM International Conference on Data Mining, pp. 542–550, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, April 2014.
[60] W. Yu, G. Zeng, P. Luo et al., “Embedding with autoencoder regularization,” in Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 208–223, Springer, Prague, Czech Republic, September 2013.
[61] B. Zong, Q. Song, M. R. Min et al., “Deep autoencoding Gaussian mixture model for unsupervised anomaly detection,” in Proceedings of the International Conference on Learning Representations, Vancouver, Canada, April-May 2018, https://openreview.net/forum?id=BJJLHbb0.
[62] I. Ahmed, A. Dagnino, and Y. Ding, “Unsupervised anomaly detection based on minimum spanning tree approximated distance measures and its application to hydropower turbines,” IEEE Transactions on Automation Science and Engineering, vol. 16, no. 2, pp. 654–667, 2018.