Applied Intelligence (2023) 53:23628–23640
https://doi.org/10.1007/s10489-023-04767-2

Latent feature reconstruction for unsupervised anomaly detection

Jinghuang Lin[1] · Yifan He[1] · Weixia Xu[1] · Jihong Guan[2] · Ji Zhang[3] · Shuigeng Zhou[1]

Accepted: 5 June 2023 / Published online: 13 July 2023

Abstract

Anomalies (or outliers) indicate a minority of data items that are quite different from the majority (inliers) of a dataset in a certain aspect. Unsupervised anomaly detection (UAD) is an important but not yet extensively studied research topic. Recent deep learning based methods exploit the reconstruction gap between inliers and outliers to discriminate them. However, it is observed that the reconstruction gap often decreases rapidly as the training process goes. And there is no reasonable way to set the training stop point. To support effective UAD, we propose a new UAD framework by introducing a Latent Feature Reconstruction (LFR) layer that can be applied to recent UAD methods. The LFR layer acts as a regularizer to constrain the latent features in a low-rank subspace from which inliers can be reconstructed well while outliers cannot. We develop two new UAD methods by implementing the proposed framework with autoencoder architecture and geometric transformation scheme. Experiments on five benchmarks show that our proposed methods can achieve state-of-the-art performance in most cases.

Keywords Unsupervised anomaly detection · Latent feature reconstruction · Autoencoder · Geometric transformation

1 Introduction

Anomaly detection (AD), sometimes also referred to as outlier detection or novelty detection [1], is to identify a relatively small number of special data points (outliers) from a noisy dataset that deviates from the majority (inliers) of the dataset. It has various applications such as financial fraud detection [2], intrusion detection [3], anomalous behavior discovery in social networks [4] etc. Anomalies exist ubiquitously in various types of data. For example, searching for novel techniques from patent databases, detecting cancers in medical images, and identifying accidents in traffic monitoring videos. Recently, a number of deep neural network based methods have been proposed for anomaly detection, including reconstruction-based [5, 6], GAN-based [7, 8], discrimination-based [9, 10], and density-based [11].

In the context of machine learning, anomaly detection can be supervised (SAD), semi-supervised (SSAD), and unsupervised (UAD), depending on how many labeled data are available [12]. Note that in some previous works [9, 13], "unsupervised anomaly detection" refers to the setting where the training set consists of only normal samples, which is actually SSAD, rather than UAD. Differently, UAD in this paper refers to that the training set is completely unlabeled, and normal data are the majority, but mixed up with some outliers. This paper addresses the UAD problem.

Currently, autoencoders (AEs) and convolutional autoencoders (CAEs) are widely-used for anomaly detection. They seek a low-dimensional latent feature space, from which the input can be reconstructed. The intuition behind these methods is that the inliers (normal data) are reconstructed better from the latent space than the outliers (abnormal data).

✉ Shuigeng Zhou
sgzhou@fudan.edu.cn

Jinghuang Lin
jhlin18@fudan.edu.cn

Yifan He
yfhe20@fudan.edu.cn

Weixia Xu
xuweixia@fudan.edu.cn

Jihong Guan
jhguan@tongji.edu.cn

Ji Zhang
ji.zhang@zhejianglab.com

[1] School of Computer Science, amd Shanghai Key Lab
of Intelligent Information Processing, Fudan University,
Shanghai 200438, China

[2] Department of Computer Science and Technology,
Tongji University, Shanghai 201804, China

[3] Nanhu Headquarters, Zhejiang Lab, Hangzhou 311121, China

However, in the UAD setting, it is observed that AEs/CAEs usually reconstruct outliers as well as inliers, and the reconstruction gap between inliers and outliers decreases as the training process goes. To illustrate this phenomenon, we give an example in Fig. 1, which shows the inlier and outlier reconstruction errors of a CAE trained on Fashion-MNIST. When the number of epochs reaches 1000, the two curves coincide, which means that the trained model can no longer discriminate outliers from inliers.

Though some existing works have tried to handle this problem to some extent, they also have their own limitations. For example, RSRAE [14] proposes a robust subspace recovery (RSR) layer for AEs to regularize inliers into a low-rank subspace, from which the outliers stay far away. However, RSRAE is designed specifically for AEs, and AEs are ineffective in handling high-dimensional and complex datasets like CIFAR10. To do SSAD over complex datasets, GEOM [9] employs ResNet for powerful feature representation and geometric transformations for data augmentation. And E^{3}Outlier extends the transformations to RSRAE for UAD, it can retard the reduction of the loss gap between inliers and outliers. But both of them are applicable only to images, and the additional transformations incur much computational cost in training/testing.

In this paper, we propose a new and more general framework for UAD by introducing a latent feature reconstruction (LFR) layer as a plug-in module that can be embedded in the two types of existing UAD methods: autoencoder based methods (e.g. RSRAE) and geometric transformation based methods (e.g. GEOM and E^{3}Outlier) to effectively handle the above-mentioned problem. In the training phase, the LFR layer linearly maps the latent features into a low-dimensional subspace that keeps the significant information, and from which the latent feature space can be reconstructed so that for inliers the reconstructed features are close to the original features while for outliers are not. We implement the proposed framework based on both AE and geometric transformations, and consequently develop two new UAD methods, which are called AE-LFR and GT-LFR, respectively. We also propose a novel yet simple anomaly scoring strategy by connecting the LFR layer and the backbone network in testing. We show that this strategy can get a large gap in anomaly scores between inliers and outliers.

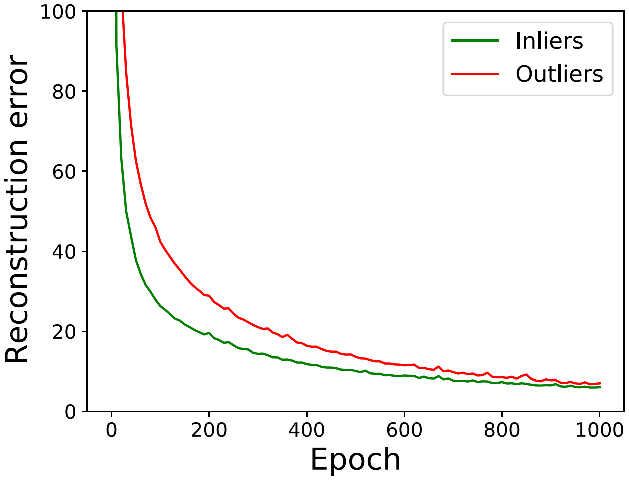


Fig. 1 Averaged inliers and outliers reconstruction errors of a CAE trained on Fashion-MNIST. Inliers (green) are sampled from class “T-shirt", and outliers (red) are sampled from the rest classes. The ratio of outliers over inliers is 0.1. As training goes on, the error gap between inliers and outliers steadily decreases, and two curves coincide at around the 1000-th epoch

In summary, our contributions include

1 We propose a new UAD framework with a latent feature reconstruction (LFR) layer that can be applied to two major types of existing UAD methods. The LFR layer regularizes the latent features to a low-rank subspace for inliers by back-propagation while outliers stay far away from this subspace. We design a novel anomaly scoring function that can maintain a score gap large between inliers and outliers.
2 We develop two new UAD methods by implementing the proposed framework based on AE and geometric transformations.
3 We conduct extensive experiments on five datasets to validate the proposed framework and methods, which achieve state-of-the-art performance in most cases.

The most related work to our paper is the RSRAE method [14]. It should be pointed out that our LFR framework is different from the RSRAE method in at least three aspects: (1) Our LFR framework employs different structures for training and testing, and in training the LFR layer is separated from the backbone network, while RSRAE has a similar structure for both training and testing, which is like that in our testing phase. (2) Our LFR framework is more general and can serve as a plug-in component to be applied to both AE based methods and geometric transformation (GT) based methods, while RSRAE is only a typical AE based method. (3) Our methods clearly outperform RSRAE in most cases.

The rest of this paper is organized as follows: Section 2 reviews the related works. Section 3 presents our methods in details. Section 4 is performance evaluation. Section 5 concludes this paper.

2 Related work

Most traditional works on anomaly (or novelty) detection consider that the training set consists of only normal data (inliers), so they treat the problem as one-class classification, and propose SVM based method [15] and principle component analysis (PCA) based methods [16, 17] etc. They can be subsumed to *supervised anomaly detection* (SAD in short).

Recently, more and more deep neural network based methods are introduced for anomaly detection by exploiting their powerful representations of high-dimensional data (e.g. images and videos). A detailed review of deep learning for anomaly detection can be referred to [18]. The majority of such existing works treat anomaly detection as a semi-supervised learning problem, that is, *semi-supervised anomaly detection* (SSAD in short). Those SSAD methods mainly fall into four types: reconstruction-based [5, 6], GAN-based [7, 8], discrimination-based [9, 10], and density-based [11] methods.

Unsupervised anomaly detection (UAD in short) is a more challenging problem that has not yet been extensively studied, where the challenge lies in that no inlier or outlier labels are provided in the training data. Up to now, only a few deep learning-based methods are proposed for UAD, which can be grouped into two categories: *autoencoder (AE) based* and *geometric transformation (GT) based* methods. In [18], they are also called *reconstruction based* and *discrimination based* methods, respectively.

Among the AE based methods, [5] proposes an autoencoder-based method that identifies the outliers by maximizing the reconstruction loss difference between inliers and outliers with a specifically designed loss function. [6] utilizes *robust principal component analysis* (RPCA) that decomposes the unlabelled input data matrix into a low-rank part and a sparse part to separate the inliers and outliers. And [19] jointly optimizes an AE and an estimation network in an end-to-end manner. The estimation network is used to fit a Gaussian mixture model. Inspired by *robust subspace recovery* (RSR), the RSRAE method [14] introduces an RSR layer within an AE to cope with the situation where a large portion of data points are corrupted by exploiting the latent low-rank subspace structure of the training data. UniAD [20] revisits the formulations of fully-connected layer, convolutional layer, as well as attention layer, and confirms the important role of query embedding in distinguishing normal and abnormal samples. It first proposes a layer-wise query decoder to model the normal distribution, and introduces a feature jittering strategy that urges the model to recover the correct message even with noisy input.

Up to now, the only GT based method is E^{3}Outlier [10], which is based on GEOM [9] by changing the original pre-define self-supervised task in GEOM via extending the regular affine transformation to irregular affine transformation and patch re-arranging. GEOM is an SSAD method, which first applies different geometric transformations to normal training images, and then trains classification models for a pre-defined task (predicting the orientations of rotated images) on the augmented data. At the evaluation phase, the anomaly score of an instance is defined as the average of softmax classification scores of all the corresponding transformed images.

In addition to the advances in model structure and algorithm perspectives, some recent works try to introduce additional auxiliary information to improve the performance of anomaly detection. FCDD [21] collects anomalous samples from 80 millions Tiny Image and ImageNet, and trains a Fully Convolutional Data Description (FCDD), which maps normal samples near to the center c of normal distribution and the anomalous samples away from c. Salehi et al. [22] perform distillation on the expert network pretrained on ImageNet, detect and localize anomalies using the discrepancy between the expert and cloner networks' intermediate activation values. DRAEM [23] takes the auxiliary images as anomaly texture sources to generate anomalous images, then it learns a joint representation of an anomalous image and its anomaly-free reconstruction, while learning a decision boundary between normal and anomalous examples. Elite [24] even introduces some labeled examples as thvalidation set and leverages the gradient of the validation loss to predict if one training sample is abnormal.

3 Method

3.1 Problem statement

Given an unlabeled data set $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$, where $\mathbf{x}_i \in \mathbb{R}^D$ and N is the size of $\mathbf{X}$, $\mathbf{X}$ implicitly consists of a subset of inliers $\mathbf{X}_{in}$ and a subset of outliers $\mathbf{X}_{out}$. Data in $\mathbf{X}_{in}$ and data in $\mathbf{X}_{out}$ are sampled or generated from two completely different distributions (or distribution mixtures). The goal of UAD is to build a detector based on $\mathbf{X}$ such that for any data point $\mathbf{x}_i \in \mathbf{X}$, it can determine whether $\mathbf{x}_i$ belongs to $\mathbf{X}_{in}$ or belongs to $\mathbf{X}_{out}$.

In what follows, we first introduce the LFR framework for UAD, then present two implementations of the LFR framework based on autoencoder and geometric transformation, respectively. These two implementations correspond to two new UAD methods, which are called AE-LFR and GT-LFR.

3.2 The LFR framework

Recent deep learning based methods for UAD learn the feature representations of training data points mainly by a generic feature learning method like autoencoder or ResNet [25]. They pursue an underlying representation to distinguish anomalies from normal data. The process can be formally represented as follows:

$$\begin{aligned} z_i &= \phi(x_i; \theta) \\ \{\theta^*, \omega^*\} &= \underset{\theta,\omega}{\arg\min} \sum_{i=1}^{N} L_{ori}(\psi(z_i; \omega)) \\ s_x &= f(x, z, \phi_{\theta^*}, \psi_{\omega^*}) \end{aligned} \tag{1}$$

where $\phi(\cdot;\theta)$ is the feature extractor that maps $x_i \in \mathbb{R}^D$ to its latent feature $z_i \in \mathbb{R}^d$, $\psi(\cdot;\omega)$ is a surrogate task that takes z_i as input and learns a critical latent feature space for the input, $L_{ori}(\cdot)$ is a loss function depending on the backbone applied, and $f(\cdot)$ is an anomaly scoring function that measures the degree of abnormality s_x. Outliers are usually identified by choosing a proper threshold for s_x.

In UAD, the model is optimized for outliers and inliers simultaneously. Though the property "*inlier priority*" [10] indicates that the model gives priority to reducing the inliers' loss, the loss gap between inliers and outliers will decrease after enough training epochs, as shown in Fig. 1. Usually, the anomaly score is just the loss or a variant of the loss, so the anomaly score gap will decrease as well. To keep the score gap between inliers and outliers large, we introduce a new and general framework for UAD, where the core component is a latent feature reconstruction (LFR) layer embedded in the training and testing phases. We call this new framework LFR, which is illustrated in Fig. 2.

Here, the LFR layer is a plug-in component that can be embedded in existing methods without changing their backbone networks, it regularizes the latent features through back-propagation. The LFR layer takes the latent feature z_i as input and outputs its reconstruction z_i', which can also be used as the input of $\psi(\cdot;\omega)$. In the testing phase, we just simply embed the LFR layer into the backbone.

In the training phase, we regulate the learning of z_i by the LFR layer. Inspired by RSRAE [14], we introduce the robust subspace recovery (RSR) loss to the LFR layer. Specifically, the LFR layer seeks a low-rank latent feature subspace for inliers. It applies a linear transformation $A \in \mathbb{R}^{k\times d}$ that maps the original latent feature z_i into a k-dimensional space, from which we reconstruct it in the original latent feature space by the transpose of A. The loss function is as follows,

$$\begin{aligned} L_{\mathrm{RSR}}(\theta, A) =& \lambda_1 \sum_{i=1}^{N} \left\| z_i - A^T A z_i \right\|_2^1 \\ &+ \lambda_2 \left\| A A^{\mathrm{T}} - I_{k\mathrm{F}} \right\|^2 , \end{aligned} \tag{2}$$

where A^T is the transpose of A, I_d denotes the identity matrix and $\|.\|_{\mathrm{F}}$ denotes the Frobenius norm. As demonstrated in [14], $A^T A$ is close to an orthogonal projector, and the loss will guide the latent features to lie in a low-rank subspace.

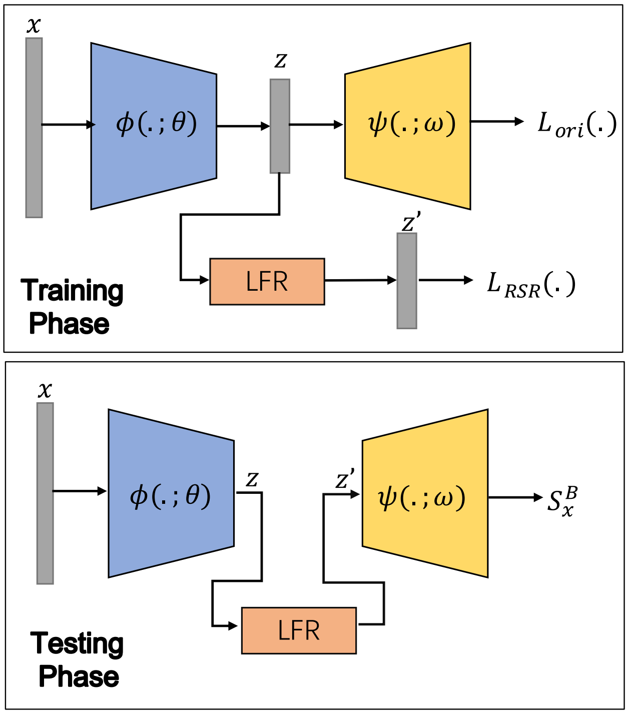


Fig. 2 The framework of LFR. Here, the upper subfigure shows the structure for training, and the lower subfigure shows the structure for testing

The total loss of our framework is the sum of the original loss (the backbone loss) and the RSR loss in (2), i.e.,

$$L(\theta, \omega, A) = L_{ori}(\psi(z_i; \omega)) + L_{RSR}(\theta, A). \tag{3}$$

In (1), $s_x = f(x, z, \phi_{\theta^*}, \psi_{\omega^*})$ is the original anomaly score. Though it can also be used as the scoring function of our framework, it cannot make full use of the learnt LFR layer.

In the testing phase, we embed the LFR layer to the original backbone, as shown in the lower subfigure of Fig. 2, and thus have a new scoring function as follows:

$$s_x^B = f(x, A^T A z, \phi_{\theta^*}, \psi_{\omega^*}) \tag{4}$$

The intuition behind this function is like this: with the loss function of (2), the reconstruction $z' = A^T A z$ for inliers is close to the original latent feature z, but for the outliers, it is not. Therefore, we replace z with z' in the scoring function. The anomaly score gap between inliers and outliers will be enlarged, which is beneficial to discriminating outliers from inliers.

3.3 The AE-LFR method

We first implement the LFR framework by applying it to AE based UAD methods, and get our first new method called AE-LFR. That is, we plug the LFR layer in any AE based UAD method. As AE is used as the backbone, $\phi_e(\cdot)$ and $\psi_d(\cdot)$ are the encoder and the decoder, respectively. So we have,

$$\begin{aligned} z_i &= \phi_e(x_i;\theta_e), \\ \hat{x}_i &= \psi_d(z_i;\omega_d), \\ L_{recon}(\theta_e,\omega_d) &= \sum_{i=1}^{N} \left\| x_i - \psi_d\big(\phi_e(x_i;\theta_e);\omega_d\big) \right\|_2^1, \\ \left\{\theta_e^*,\omega_d^*\right\} &= \underset{\theta_e,\omega_d}{\arg\min}\, L_{recon}(\theta_e,\omega_d), \\ s_x &= \left\| x - \psi_d\big(z;\omega_d^*\big) \right\|^2, \end{aligned} \tag{5}$$

Above, the encoder takes x_i as input and outputs the hidden feature z_i, then the decoder maps z_i to get the reconstruction $\hat{x}_i$ of x_i. As a plug-in layer, the LFR layer can be directly applied to any encoder-decoder architecture as illustrated in Fig. 2. In addition to the reconstruction loss of AE, the RSR loss in (2) is also used as the supervision signal. Accordingly, we have the following loss for the AE-LFR method,

$$L(\theta_e,\omega_d,A) = L_{recon}(\theta_e,\omega_d) + L_{RSR}(\theta_e,A). \tag{6}$$

Then, by replacing z with $z' = A^T Az$ in the scoring function s_x in (5), we get the anomaly score function of AE-LFR as follows:

$$s_x^B = \left\| x - \psi_d\big(A^T Az;\omega_d^*\big) \right\|^2. \tag{7}$$

3.4 The GT-LFR method

Here, we apply our framework to geometric transformation based methods. Concretely, we take GEOM as an example, and get our second new method GT-LFR.

GEOM [9] first applies a set of geometric transformations $\{T_m\}_{m=1}^M$, including rotations, reflections, and translations, to the training images. Then, it sets up a self-supervised task that trains a multi-class classification model on the augmented data to predict the transformations it applied. In the evaluation phase, an image is applied with M given transformations, and its anomaly score is the average of all probability outputs of the learned classification model over the M transformed images. Formally,

$$\begin{aligned} z_i^{T_m} &= \phi_f(T_m(x_i);\theta_f) \\ L_{GEOM}(\theta_f,\omega_g) &= \sum_{i=1}^{N}\sum_{m=1}^{M} CE(\psi_g(z_i^{T_m};\omega_g), y_{T_m}) \\ \left\{\theta_f^*,\omega_g^*\right\} &= \underset{\theta_f,\omega_g}{\arg\min}\, L_{GEOM}(\theta_f,\omega_g) \\ s_x &= \frac{1}{M}\sum_{m=1}^{M} P^{T_m}(z^{T_m};\theta_f^*,\omega_g^*) \end{aligned} \tag{8}$$

where $\phi_f(T_m(\cdot);\theta_f)$ is a deep classification model like ResNet [25] and Wide Resnet (WRN) [26], which extracts the latent representations of input images augmented by the pre-defined geometric transformation T_m. $\psi_g(\cdot;\theta_g)$ is a multi-class classifier and CE denotes the cross-entropy loss. $P^{T_m}(\cdot;\theta_f^*,\omega_g^*)$ is the softmax output of ψ_g on transformation T_m.

Here, the LFR layer also regularizes the latent feature learning of $z_i^{T_m}$. But unlike AE-LFR, there are M distinct subsets of the augmented image set, with which it is hard to find a single low-rank latent feature subspace for the inliers. To tackle this problem, we try to find a separate feature subspace for each transformation. Thus, we assign a linear matrix $A_{(m)} \in \mathbb{R}^{k\times d}$ for each transformation T_m to accommodate the corresponding feature subspace, that is,

$$\begin{aligned} L_{\mathrm{RSR_{GEOM}}}(\theta_f,A) = {}& \lambda_1 \sum_{i=1}^{N}\sum_{m=1}^{M} \left\| z_i^{T_m} - A_{(m)}^T A_{(m)} z_i^{T_m} \right\|_2^1 \\ & + \lambda_2 \sum_{m=1}^{M} \left\| A_{(m)} A_{(m)}^{\mathrm{T}} - I_d \right\|_{\mathrm{F}}^2 . \end{aligned} \tag{9}$$

So the loss function of GT-LFR can be formulated as follows:

$$L(\theta_f,\theta_g,A) = L_{GEOM}(\theta_f,\theta_g) + L_{\mathrm{RSR_{GEOM}}}(\theta_f,A). \tag{10}$$

By replacing the latent feature of each transformed image z^{T_m} with $A_{(m)}^T A_{(m)} z^{T_m}$ in the scoring function s_x of (8), we have the anomaly score function of GT-LFR as follows:

$$s_x^B = \frac{1}{M}\sum_{m=1}^{M} P^{T_m}(A_{(m)}^T A_{(m)} z^{T_m};\theta_f^*,\omega_g^*). \tag{11}$$

4 Performance evaluation

4.1 Experiment setup

We evaluate our methods on five public datasets, including three image datasets: Caltech101 [27], Fashion-MNIST (FMNIST) [28], CIFAR10 [29], and two text datasets: Reuters-21578 (Reuters) [30] and 20 Newsgroups (20News) [31].

For fair comparison, we process the datasets by following the settings of previous UAD methods [5, 6, 10, 14, 32]. For example, we follow the settings in [14] to handle Caltech101: taking 11 classes of Caltech101 and randomly choosing 100 images per class. Each training set with anomalies is constructed as follows: sampling the examples from a certain class as inliers, and combing some samples from each of the other classes as outliers. The ratio c of outliers/inliers is set to $\{0.1, 0.3, 0.5, 0.7, 0.9\}$ respectively. Note that in UAD, all inlier/outlier labels are unknown to the model in the training phase. For a given ratio c, we first evaluate the performance of taking a certain class as inliers, then compute the average of all classes' results as the final performance.

And for each class, we do 5 trials with different random seeds and report the averaged result. The Area under the Receiver Operating Characteristic curve (AUROC) and the Area under the Precision-Recall curve (AUPR) are used as performance metrics. We treat the outliers as "positive" in evaluation.

4.2 Compared methods

We compare our methods with seven existing methods: AE/CAE [33], DRAE [5], RSRAE [14], GEOM [9], E^{3}Outlier [10], LVAD [34], and Elite [24]. AE/CAE, DRAE, and RSRAE are AE-based methods, GEOM and E^{3}Outlier are geometric transformation-based methods, and LVAD is density-based. Elite has two variants, Elite-AE is AE-based, while Elite-SVDD is discrimination-based. Although GEOM was originally proposed for SSAD, it can be extended to UAD. Among these methods, GEOM, E^{3}Outlier, LVAD, and Elite can only handle image data. As for RSRAE, LVAD, and Elite, we use the official code[1] and follow its original setting. For the other methods, we utilize the implementations in the site[2] and adapt them to the settings of datasets used in our paper.

4.3 Implementation detail

We use the same autoencoder structure for the compared AE-based methods and our AE-LFR method. For the image datasets, the encoder in AE consists of three convolutional layers and a fully connected layer with output channels (32, 64, 128, 256) and the kernel sizes ($5 \times 5, 5 \times 5, 3 \times 3$) in convolutional layers, the output of encoder is a 256-dimensional vector.

The decoder has an inverse architecture of the encoder, and replaces the convolutional kernels with deconvolutional kernels. For AE-LFR, we set $k = 10, \lambda_1 = 2, \lambda_2 = 0.1$ in all experiments. The AE-based models are optimized with Adam using a learning rate of 0.00025, a mini-batch size of 128, and 1000 epochs. The activation function is Tanh. All images are normalized into $[-1, 1]$.

For the GT-based methods, GEOM and E^{3}Outlier are implemented with a wide ResNet (WRN) with the widen factor being 4. Our GT-LFR method follows the settings of GEOM and uses its 72 transformations in self-supervised learning. We set $k = 20, \lambda_1 = 0.0002, \lambda_2 = 0.00001$ for GT-LFR in all experiments. As GT-based methods use powerful feature extractors and the change in latent features has significant impact on the downstream classification tasks, so we reduce the values of λ_1 and λ_2.

Our methods are implemented in Pytorch and all experiments are conducted on 8 RTX2080Ti GPUs.

4.4 Performance comparison with existing methods

As GEOM, E^{3}Outlier, LVAD, Elite, GT-LFR can handle only images, we evaluate them only on CIFAR10, FMNIST and Caltech101. The AUROC and AUPR results for different ratio $c \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ are shown in Figs. 3 and 4 respectively.

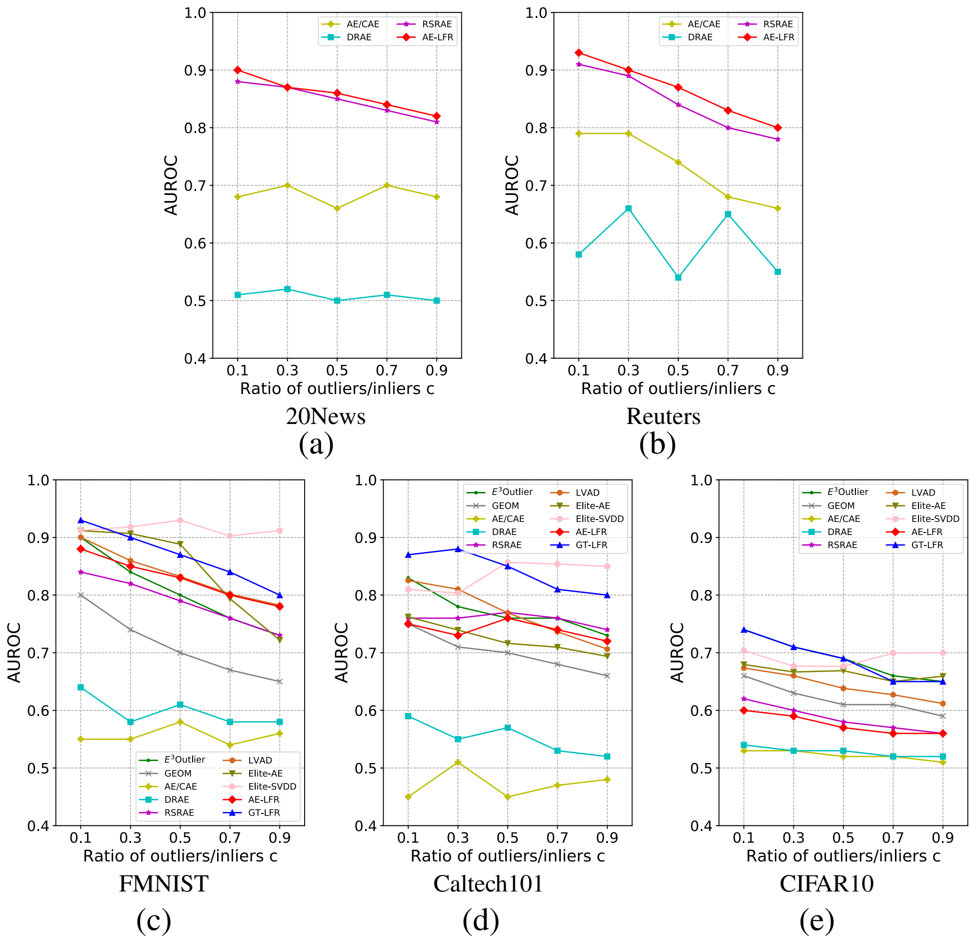
We can see that our methods achieve state-of-the-art performance in most cases, while DRAE and AE/CAE perform worse than the other methods because of the consequence of overfitting to outliers after being trained 1000 epochs. For AE-based methods, our AE-LFR method performs best on the two text datasets 20News and Reuters, and we gets competitive performance against RSRAE in most datasets, and outperforms RSRAE by 4% averaged AUROC on FMNIST.

For the GT-based methods, our GT-LFR method significantly outperforms the others on the two image datasets FMNIST and Caltech101, and is competitive to E^{3}Outlier on CIFAR10. Though GT-LFR is based on GEOM, it performs considerably better than GEOM, which shows the effectiveness of our LFR framework. E^{3}Outlier outperforms GEOM because it uses more geometric transformations. However, E^{3}Outlier consumes more computation than the others because it uses more transformations, while our proposed method needs just an additional matrix A, which consumes a little additional computation cost, so it is much faster than E^{3}Outlier. LVAD is generally better than AE-based methods and worse than GT-based method, because

[1] https://github.com/dmzou/RSRAE

[2] https://github.com/demonzyj56/E3Outlier

Fig. 3 AUROC comparison for different c values (from 0.1 to 0.9)



20News
(a)

Reuters
(b)

FMNIST
(c)

Caltech101
(d)

CIFAR10
(e)

the density estimation method is not robust on the data with complex distribution. Elite introduce some labeled samples as the validation set, which makes its performance insensitive to abnormal proportions. However, even if it uses labeled samples, it is inferior to our method in the case of fewer anomalies, which is also more consistent with the data distribution of real application scenarios. In summary, our proposed method achieves better or competitive performance with additional parameters A and computation cost ($O(kd(k+d))$). As we adopt low-rank reconstruction for latent feature ($k \ll d$), the additional computation cost approximates $O(kd^2)$.

4.5 Ablation study

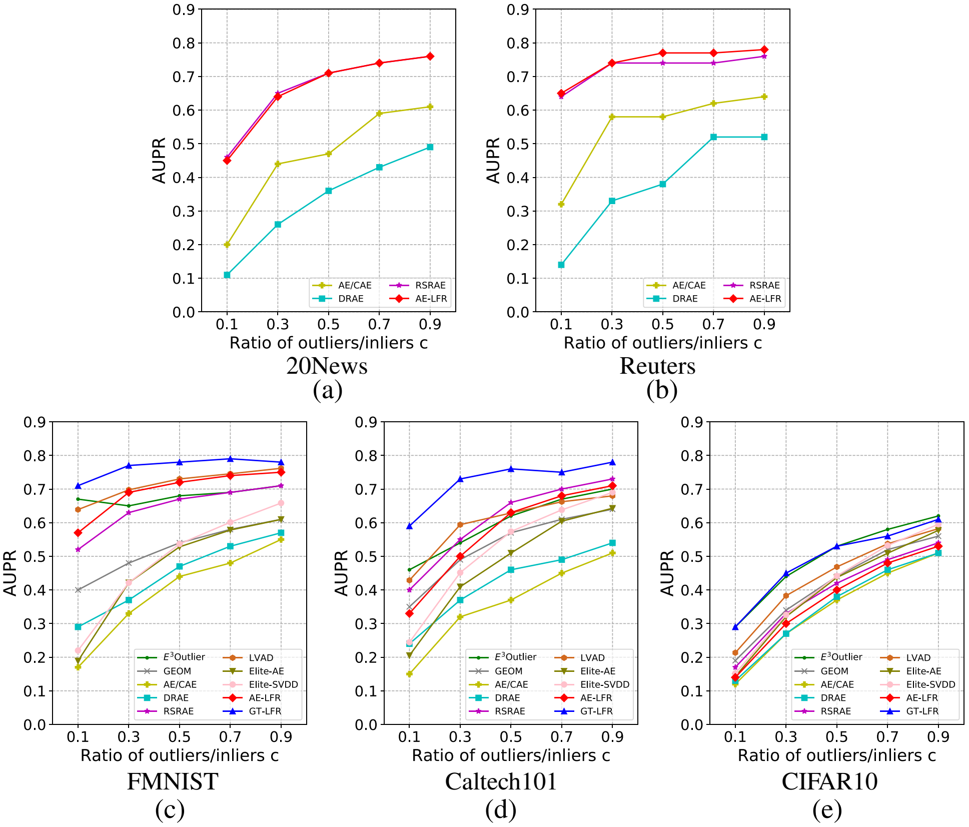
Here, we consider different combination configurations of the loss function and the anomaly scoring function in our methods AE-LFR and GT-LFR, and get different variants of our methods. We then compare these variants with two baselines AE/CAE and GEOM respectively.

For convenience, we use the following notations of the loss and scoring functions:

$$\begin{aligned}
L_A &:= L_{ori}(\psi(z;\omega)) + L_{RSR}(\theta, A) \\
L_B &:= L_{ori}(\psi(A^T Az;\omega)) + L_{RSR}(\theta, A) \\
S_A &:= f(x, z, \phi_{\theta*}, \psi_{\omega*}) \\
S_B &:= f(x, A^T Az, \phi_{\theta*}, \psi_{\omega*}).
\end{aligned} \tag{12}$$

Note that the loss and scoring functions implicitly represent the model architecture. For example, L_A means that the decoder take z as input, corresponding to the architecture of the training phase in Fig. 2, while L_B means that the decoder is fed with $z' = A^T Az$, corresponding to the architecture of the testing phase in Fig. 2. Thus, we can use $L_i S_j$ to represent a combination configuration of loss and scoring functions in the training and testing phases, where $i, j \in \{A, B\}$. For example, $L_A S_A$ indicates that the model is optimized by the loss function L_A in the training phase and evaluated by the scoring function S_A in the testing phase. So our methods can be denoted as $L_A S_B$. Meanwhile, the backbone model

Fig. 4 UAD performance (AUPR) comparison with varying c from 0.1 to 0.9



20News
(a)

Reuters
(b)

FMNIST
(c)

Caltech101
(d)

CIFAR10
(e)

(AE/CAE) can be regarded as a deteriorated model trained with only $L_{ori}(\psi(z; \omega))$, i.e., $L_{RSR}(\theta, A)$ is not used.

Table 1 presents the results of performance comparison between the baseline AE/CAE with AE-LFR ($L_A S_B$) and its three variants ($L_A S_A$, $L_B S_A$ and $L_B S_B$). From Table 1, we can see that

(1) The four variants significantly outperform AE/CAE, which shows that the LFR layer is effective in regularizing the hidden features in the low-rank subspace.

(2) $L_A S_B$ outperforms $L_A S_A$ in all settings, which shows that our proposed scoring function can boost performance.

(3) $L_B S_A$ is better than $L_B S_B$. In $L_B S_A$, the input of the decoder is the output of the LFR layer in training. The

Table 1 Comparison among AE/CAE and our AE-based variants

		0.1	0.3	0.5	0.7	0.9
FMNIST	AE/CAE	0.55	0.55	0.58	0.54	0.56
	$L_A S_A$	0.74	0.70	0.67	0.65	0.63
	$L_A S_B$ (AE-LFR)	0.88	0.85	0.83	0.80	0.78
	$L_B S_A$	0.88	0.86	0.83	0.81	0.79
	$L_B S_B$	0.81	0.75	0.71	0.69	0.67
CIFAR10	AE/CAE	0.53	0.53	0.52	0.52	0.51
	$L_A S_A$	0.54	0.54	0.53	0.52	0.52
	$L_A S_B$ (AE-LFR)	0.61	0.59	0.57	0.56	0.56
	$L_B S_A$	0.60	0.58	0.57	0.56	0.55
	$L_B S_B$	0.55	0.54	0.53	0.53	0.52
Caltech101	AE/CAE	0.45	0.51	0.45	0.47	0.48
	$L_A S_A$	0.61	0.67	0.62	0.61	0.60
	$L_A S_B$ (AE-LFR)	0.75	0.73	0.76	0.74	0.72
	$L_B S_A$	0.75	0.72	0.72	0.71	0.70
	$L_B S_B$	0.61	0.65	0.63	0.60	0.60

Table 2 Comparison among GEOM and our GT-based variants

		0.1	0.3	0.5	0.7	0.9
FMNIST	GEOM	0.81	0.74	0.70	0.67	0.65
	$L_A S_A$	0.83	0.75	0.71	0.69	0.67
	$L_A S_B$ (GT-LFR)	0.93	0.90	0.87	0.84	0.80
CIFAR10	GEOM	0.66	0.63	0.61	0.61	0.59
	$L_A S_A$	0.73	0.68	0.65	0.62	0.60
	$L_A S_B$ (GT-LFR)	0.74	0.71	0.69	0.65	0.65
Caltech101	GEOM	0.75	0.71	0.70	0.68	0.66
	$L_A S_A$	0.80	0.78	0.73	0.69	0.67
	$L_A S_B$ (GT-LFR)	0.87	0.88	0.85	0.81	0.80

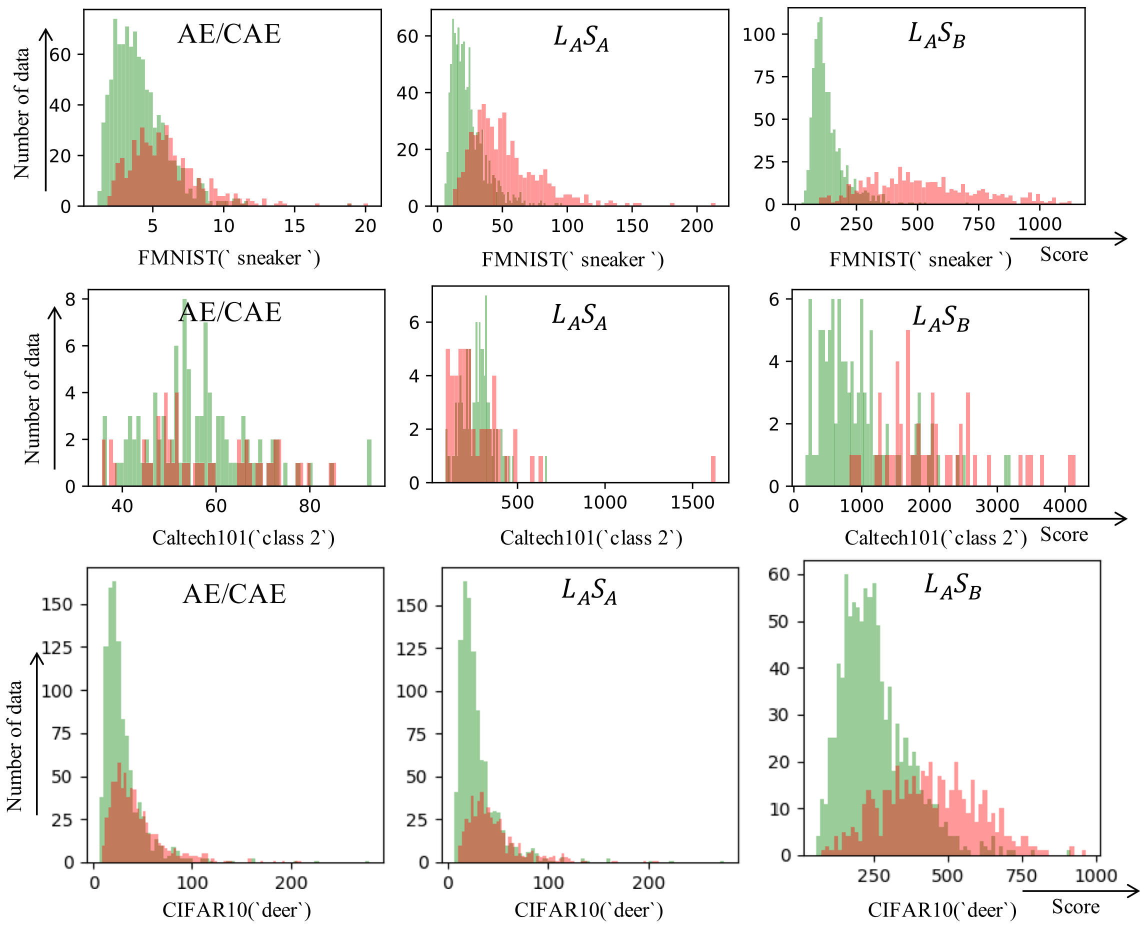


Fig. 5 Anomaly score (reconstruction error) histograms of inliers (green) and outliers (red) for AE-based variants (c=0.5)

learning goal of the LFR layer is to perfectly reconstruct the latent features of inliers, instead of outliers. So even if the autoencoder overfits the outliers after training, it is still hard for the autoencoder to recover the outliers when the LFR layer is removed.

Table 2 presents the results of GT-based variants. As it is difficult for the models to converge when L_B is applied, here we report only L_AS_A and L_AS_B. We can see that L_AS_B (GT-LFR) performs better than L_AS_A.

Figure 5 shows the inlier/outlier anomaly score histograms of AE/CAE, L_AS_A and L_AS_B (AE-LFR) on class *sneaker* of FMNIST, class 2 of Caltech101 and class *deer* of CIFAR10. We can see that with AE-LFR, most inliers have smaller anomaly scores while most outliers have larger ones. On the contrary, we see quite different results with AE/CAE. This conforms to our expectation: our LFR layer reconstructs inliers much better than outliers. Figure 6 shows the anomaly score histograms of GT-based variants, we can see patterns similar to that of AE-based variants in Fig. 5.

Figure 7 shows how the anomaly score changes with the number of training epochs in four methods. As expected, our methods can still keep a large gap between the anomaly scores of inliers and outliers as the number of training epochs increases. However, for AE/CAE and GEOM, the anomaly score gap decreases rapidly with the increase of training epochs. This explains the good performance of our methods.

5 Conclusion

In this paper, we introduce a novel UAD framework with a latent feature reconstruction (LFR) layer and a new anomaly scoring function. The LFR layer is used as a plug-in component to regularize the latent features of samples to a low-rank subspace so that the inliers can be perfectly reconstructed while the outliers cannot. We implement the proposed framework by embedding the LFR layer into two major types of existing UAD methods: AE based methods and GT based methods, consequently deriving two new UAD methods called AE-LFR and GT-LFR. Extensive experiments on five

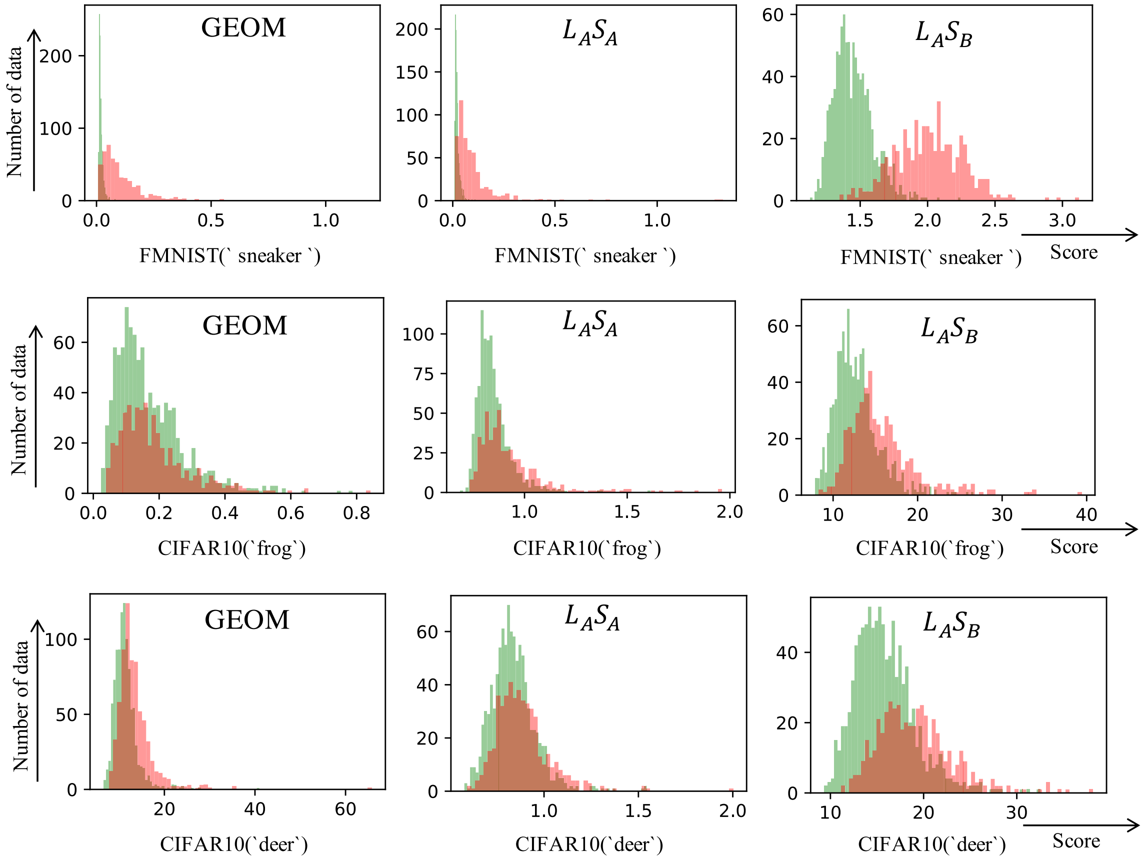


Fig. 6 Anomaly score (*log* of the averaged softmax output) histograms of inliers (green) and outliers (red) for GT-based variants (*c*=0.5)

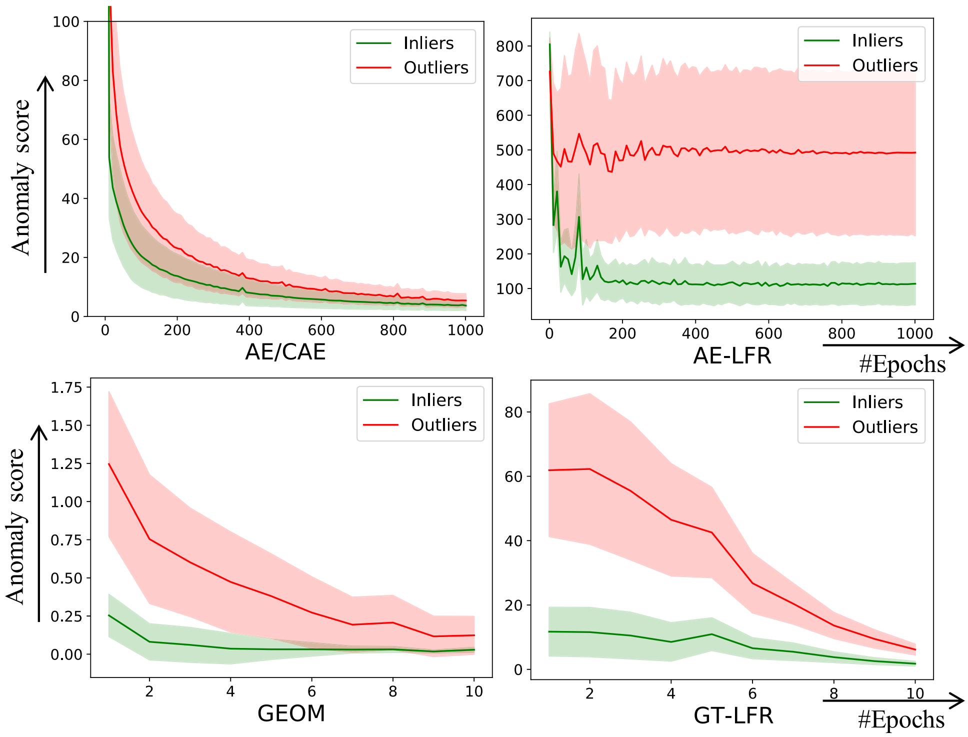


Fig. 7 Anomaly score *vs.* #Epochs for class "sneaker" of FMNIST with *c*=0.5. Shadow regions are the standard deviation

datasets show that the proposed methods outperform the existing methods in most cases. As for future work, on the one hand, we are to apply our methods to more datasets, especially videos. On the other hand, we plan to extend the proposed methods for SAD and SSAD tasks.

Acknowledgements This work was supported by Open Research Program of Zhejiang Lab under Grant No. 2019KB0AB05. We would like to thank Ding Xi for her contribution to the revision phase.

Data Availability Statements All data analysed during this study are public, including three image datasets: Caltech101 [27], Fashion-MNIST (FMNIST) [28], CIFAR10 [29], and two text datasets: Reuters-21578 (Reuters) [30] and 20 Newsgroups (20News) [31].

Declarations

Conflict of Interests The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Chandola V, Banerjee A, Kumar V (2009) Anomaly detection: A survey. ACM Comput Surv (CSUR) 41(3):1–58
2. Phua C, Lee V, Smith K, Gayler R (2010) A comprehensive survey of data mining-based fraud detection research. arXiv preprint arXiv:1009.6119
3. Davis JJ, Clark AJ (2011) Data preprocessing for anomaly based network intrusion detection: A review. computers & security 30(6–7):353–375
4. Portnoff RS (2018) The Dark Net: De-anonymization, Classification and Analysis. University of California, Berkeley, ???
5. Xia Y, Cao X, Wen F, Hua G, Sun J (2015) Learning discriminative reconstructions for unsupervised outlier removal. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1511–1519
6. Zhou C, Paffenroth RC (2017) Anomaly detection with robust deep autoencoders. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 665–674
7. Schlegl T, Seeböck P, Waldstein SM, Langs G, Schmidt-Erfurth U (2019) f-anogan: Fast unsupervised anomaly detection with generative adversarial networks. Med Image Anal 54:30–44
8. Schlegl T, Seeböck P, Waldstein SM, Schmidt-Erfurth U, Langs G (2017) Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: Information Processing in Medical Imaging: 25th International Conference, IPMI 2017, Boone, NC, USA, June 25–30,2017, Proceedings, pp 146–157. Springer
9. Golan I, El-Yaniv R (2018) Deep anomaly detection using geometric transformations. Advances in neural information processing systems 31
10. Wang S, Zeng Y, Liu X, Zhu E, Yin J, Xu C, Kloft M (2019) Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network. Advances in neural information processing systems 32
11. Zhai S, Cheng Y, Lu W, Zhang Z (2016) Deep structured energy based models for anomaly detection. In: International Conference on Machine Learning, pp 1100–1109. PMLR
12. Chandola V, Banerjee A, Kumar V (2007) Outlier detection: A survey. ACM Comput Surv 14:15
13. Kiran BR, Thomas DM, Parakkal R (2018) An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. Journal of Imaging 4(2):36
14. Lai C-H, Zou D, Lerman G (2020) Robust subspace recovery layer for unsupervised anomaly detection. In: Eighth International Conference on Learning Representations
15. Scholkopf B, Williamson R, Smola A, Shawe-Taylor J, Platt J et al (2000) Support vector method for novelty detection. Advances in neural information processing systems 12(3):582–588
16. SHYU M-L (2003) A novel anomaly detection scheme based on principal component classifier. In: Proc. of ICDM Foundation and New Direction of Data Mining Workshop, 2003
17. Hoffmann H (2007) Kernel pca for novelty detection. Pattern Recogn 40(3):863–874
18. Pang G, Shen C, Cao L, Hengel AVD (2021) Deep learning for anomaly detection: A review. ACM Comput Surv (CSUR) 54(2):1–38
19. Zong B, Song Q, Min MR, Cheng W, Lumezanu C, Cho D, Chen H (2018) Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In: International Conference on Learning Representations
20. You Z, Cui L, Shen Y, Yang K, Lu X, Zheng Y, Le X (2022) A unified model for multi-class anomaly detection. In: Advances in Neural Information Processing Systems
21. Liznerski P, Ruff L, Vandermeulen RA, Franks BJ, Kloft M, Muller KR (2021) Explainable deep one-class classification. In: International Conference on Learning Representations
22. Salehi M, Sadjadi N, Baselizadeh S, Rohban MH, Rabiee HR (2021) Multiresolution knowledge distillation for anomaly detection. In: Proceedings of the IEEE/CVF Conf Comput Vis Pattern Recognit, pp 14902–14912
23. Zavrtanik V, Kristan M, Skočaj, D (2021) Draem-a discriminatively trained reconstruction embedding for surface anomaly detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 8330–8339
24. Zhang H, Cao L, VanNostrand P, Madden S, Rundensteiner EA (2021) Elite: robust deep anomaly detection with meta gradient. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, pp 2174–2182
25. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proc IEEE Conf Comput Vis Pattern Recognit, pp 770–778
26. Zagoruyko S, Komodakis N (2016) Wide residual networks. In: British Machine Vision Conference 2016. British Machine Vision Association
27. Fei-Fei L, Fergus R, Perona P (2004) Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In: 2004 Conf Comput Vis Pattern Recognit Workshop, pp 178–178. IEEE
28. Xiao H, Rasul K, Vollgraf R (2017) Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms

29. Krizhevsky A (2009) Learning multiple layers of features from tiny images. Master's thesis, University of Tront
30. Reuters-21578 text categorization test collection. Distribution 1.0, AT&T Labs-Research (1997)
31. Lang K (1995) Newsweeder: Learning to filter netnews. Machine Learning Proceedings 1995, pp 331–339
32. Liu W, Hua G, Smith JR (2014) Unsupervised one-class learning for automatic outlier removal. In: Proc IEEE Conf Comput Vis Pattern Recognit, pp 3826–3833
33. Masci J, Meier U, Cireşan D, Schmidhuber J (2011) Stacked convolutional auto-encoders for hierarchical feature extraction. In: Artificial Neural Networks and Machine Learning, pp 52–59. Springer
34. Lin W-Y, Liu Z, Liu S (2022) Locally varying distance transform for unsupervised visual anomaly detection. In: Computer Vision–ECCV 2022: 17th European Conference on Computer Vision, pp 354–371 Springer

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Jinghuang Lin received the B.S. and M.S. degrees from the School of Computer Science, Fudan University, Shanghai, China, in 2018 and 2021, respectively. He is currently an engineer of Alibaba Group. His research interests include computer vision, deep learning, anomaly detection.

Yifan He received the B.S. and M.S. degrees from the School of Computer Science, Jiangsu University of Science and Technology, Zhenjiang, China, in 2017 and 2020, respectively. He is currently pursuing the Ph.D degree in the School of Computer Science, Fudan University, Shanghai, China. His research interests include data mining, deep learning, anomaly detection, time series processing.

Weixia Xu is a lecturer of School of Information Management, Shanghai Lixin University of Accounting and Finance, and a post-doctoral researcher of School of Computer Science, Fudan University, Shanghai, China. She received the Bachelor degree from East China University of Science and Technology (ECUST) in 2011, and the PhD of Computer Science from Fudan University in 2019. Her research interests include big data management and analytics, statistical learning and machine learning.

Jihong Guan is now a full professor of Department of Computer Science & Technology, Tongji University, Shanghai, China. She received her Bachelor degree from Huazhong Normal University in 1991, her Master degree from Wuhan Technical University of Surveying and Mapping (merged into Wuhan University in Aug. 2000) in 1991, and her PhD from Wuhan University in 2002. Before joining Tongji University, she served in the Department of Computer, Wuhan Technical University of Surveying and Mapping from 1991 to 1997, as an assistant professor and an associate professor (since August 2000) respectively. She was an associate professor (Aug. 2000-Oct. 2003) and a professor (Since Nov. 2003) in the School of Computer, Wuhan University. Her research interests include spatial databases, artificial intelligence and bioinformatics. She has published more than 200 research papers in domestic and international journals and conferences.

Ji Zhang is currently a full professor in Computer Science at the University of Southern Queensland (UniSQ), Australia. He is an IET Fellow, BCS Fellow, RSA Fellow, IEEE Senior Member, Australian Endeavour Fellow, Queensland International Fellow (Australia) and Izaak Walton Killam Scholar (Canada), and a visiting scholar of Zhejiang Lab, China. His research interests are big data analytics, knowledge discovery and data mining (KDD) and computational intelligence. He has published over 270 papers in major peer-reviewed international journals and conferences including TKDE, TKDD, TIST, AAAI, IJCAI, VLDB, CIKM, SIGKDD, ICDE, ICDM and WWW.

Shuigeng Zhou is a full professor of School of Computer Science, and the director of Shanghai Key Lab of Intelligent Information Processing, Fudan University, Shanghai, China. He received his Bachelor degree from Huazhong University of Science and Technology (HUST) in 1988, his Master degree from the University of Electronic Science and Technology of China (UESTC) in 1991, and his PhD of Computer Science from Fudan University in 2000. He served in Shanghai Academy of Spaceflight Technology from 1991 to 1997, first as an engineer and then as a senior engineer (since August 1995) respectively. He was a post-doctoral researcher in the State Key Lab of Software Engineering, Wuhan University from 2000 to 2002. His research interests include big data management and analytics, artificial intelligence, and bioinformatics. He has extensively published in international journals (including ACM TITS, IEEE TKDE, IEEE TPDS, VLDB Journal, ACM/IEEE TCBB, Nature Communications, Bioinformatics etc.) and conferences (including SIGMOD, VLDB, ICDE, SIGKDD, AAAI, IJCAI, ICCV, CVPR, SODA, ACM-MM, ISMB and RECOMB etc.). Currently he is a fellow of China Computer Federation (CCF), a senior member of IEEE and a member of ACM.