Cascaded Dimension Reduction for Effective Anomaly Detection

1st Xiaoyan Zhuo
UMass Lowell
xiaoyan_zhuo@student.uml.edu

2nd Aekyeung Moon
ETRI
akmoon@etri.re.kr

3rd Jialing Zhang
UMass Lowell
jialing_zhang@student.uml.edu

4th Seung Woo Son
UMass Lowell
seungwoo_son@uml.edu

***Abstract*—Recent years have witnessed the growth of the size and dimensionality of data from various applications at an unprecedented rate. Detecting anomalies in these high-dimensional data has a great significance yet remains challenging due to the sparsity, noise, and irrelevant features in the high-dimensional data. Principle Component Analysis (PCA) and AutoEncoder (AE) are the two most widely used dimension reduction (DR) methods where it reduces the number of features while capturing the most necessary information of the original data. PCA alone is, however, less effective for complex data though it is fast and has explained variance to measure the preserved information of reduced data. While combining PCA with AE can be more effective, determining optimal latent representation is a challenging problem. In this work, we propose a cascaded dimension reduction method (CDR) for effective anomaly detection. In CDR, we combine AE and PCA to reduce dimension size significantly while incorporating a knee point detection algorithm to automatically select optimal dimension size k that maximizes the anomaly detection accuracy. Our extensive evaluation of various datasets and anomaly detection models demonstrate that our proposed CDR can significantly reduce dimension size while preserving the most necessary information for effective anomaly detection. As a result, CDR achieves 80%∼98% of reduction ratios and 4∼21x of speedup and outperforms state-of-the-art anomaly detection methods.**

***Index Terms*—dimension reduction, anomaly detection, high-dimensional data, knee point detection**

I. Introduction

Anomaly detection, also known as outlier detection or novelty detection, refers to the process of identifying anomalies that significantly deviate from the normal data instances [1]. Anomaly detection is important as it has been widely used in real-world applications, such as network intrusion detection [2], medical diagnosis [3], or senor networks [4]. Anomaly Detection has been studied for decades, and numerous anomaly detection models have been proposed, such as [5]–[10]. While there has been significant progress in improving detection accuracy, effective anomaly detection in data remains challenging due to several reasons. First, recent years have observed the growing size and dimensionality of data from various applications at an unprecedented rate. Second, as the dimension increases, the space for data points increases proportionally. As a result, data points become more sparse, making many conventional algorithms, such as nearest-neighbor or clustering algorithms, suffer from the curse of dimensionality [11]. Moreover, high-dimensional data usually contain irrelevant or noisy features that conceal the evident features that can be used for discriminating anomalies from normal instances [12]. Lastly, high-dimensional data requires more memory and incurs a heavy computational burden.

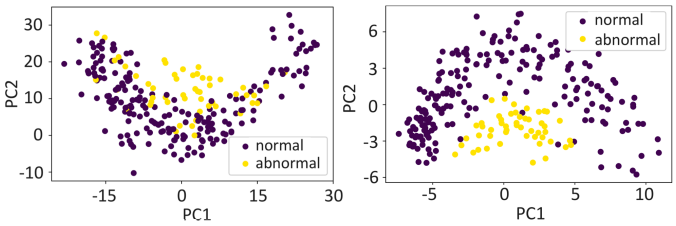
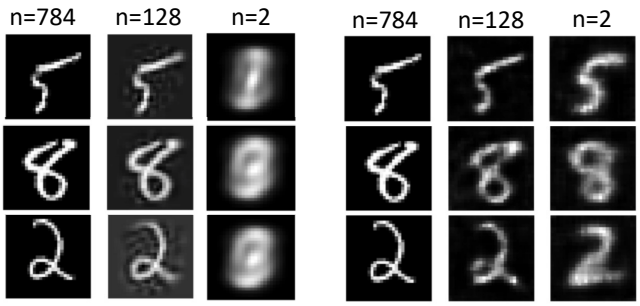


Fig. 1. An example of uncertainties in the high-dimensional and low-dimensional representation of the MNIST dataset. The dimension of the original image is 784, and the dimension size of high-dimensional representation and low-dimensional representation is 512 and 64, respectively.

Dimension reduction (DR) is a natural fit to tackle these problems associated with high dimensionality. DR aims to minimize the number of features such that a smaller set of feature can capture the most crucial information from original high-dimensional data [13]. Also, DR techniques help eliminate the irrelevant or noisy features, such that the uncertainties in high-dimensional data become smaller, which is beneficial to discriminate anomalies from normal instances [14]. Figure 1 illustrates such an example of uncertainties in high-dimensional representation (i.e., dim 512) and low-dimensional representation (i.e., dim 64) of MNIST dataset [15]. We can observe that uncertainties (e.g., irregularities in hand-written digits) in low-dimensional data are smaller than those in high-dimensional data. In other words, classifiers can easily discriminate between anomalies and normal instances using a low-dimensional representation that captures most essential information while irrelevant and noisy features are removed.

Principal Component Analysis (PCA), one of the most commonly used DR methods, projects data into lower dimensional space [16], [17]. The underlying assumption of this approach is that a smaller set of principal components can capture most information (i.e., most variance) [1]. In other words, we can eliminate principal components with low variance or little information to reduce the number of features. In Figure 2a, we present examples of PCA dimension reduction for the

MNIST [15] data, where the original image size is 32×32. In image processing, each pixel can be considered as a feature for an image. In other words, the dimension of original image data is 784. After applying PCA on the original images, we use only 128 principal components to reconstruct images. As shown in the second column of Figure 2a, the reconstructed images are almost identical to the original images, which indicates the low-dimensional representation could capture the most necessary information. However, when the dimension size gets reduced significantly, such as n=2 in the third column of Figure 2a, the reconstructed images appear blurred or distorted, making the evident characteristics of original images unnoticeable. In other words, some important information in the original images is missing. The reason behind this disproportionally higher information loss with higher dimensional reduction is that PCA is linear. Therefore, it is difficult to represent complex data, such as MNIST in Figure 2a when the dimension gets small.



(a) PCA (b) AE/VAE

Fig. 2. Examples of commonly-used dimension reduction methods on MNIST.

Another widely used type of DR method is AutoEncoder (AE) [18] [19], Variational AutoEncoder (VAE) [20], or their variants. AE/VAE usually consists of an encoder and a decoder, where it encodes original high-dimensional data into lower dimensions (bottleneck layer or latent space). Then the decoder uses the compressed data to reconstruct the original input. Contrary to PCA, in most AE/VAE networks, the activation function is non-linear, such as ReLU [21]. AE/VAE learns the representation z with lower dimension by extracting important features and discarding noise and redundancy. As an example, Figure 2b shows the original images with 784 dimensions are compressed into AE representation with 128 dimensions. The second column shows the reconstructed images from those 128 dimensions of compressed data. The results also show the compressed representation z can preserve the most critical information from the original images. While the AE/VAE has eliminated certain redundancy, there is still space to minimize the redundancy [22].

While decreasing the dimension of latent representation in AE/VAE is straightforward, selecting an optimal dimension k for z is a challenging problem, and greedy search or exhaustive searching is time-consuming and impractical. Typically, the dimension of AE/VAE is set heuristically, whereas in PCA, selecting the number of principal components (PCs) based on variance is measurable and controllable. Thus, we explore the case where PCA is applied on the AE/VAE representation z. To demonstrate the information preserved in the same data size, we select the first two PCs that capture around 63% of the variance of z to reconstruct images. As shown in the third column of Figure 2b, the reconstructed images are clear enough to be distinguished from other classes (i.e., other digits in MNIST). Moreover, with the same small dimension (e.g., n=2), the reconstruction quality of PCA on z is much better than directly using PCA on original images.

Motivated by the advantages and disadvantages of PCA and AE and the potential of exploiting benefits of both, we propose a cascaded dimension reduction (CDR) method that combines AE and PCA to reduce the dimension of high-dimensional data to the potential maximum while capturing the most useful information for effective anomaly detection. To achieve this, we propose two techniques to select a proper reduced dimension of k automatically. Our key contributions are summarized as follows:

- We propose a simple yet effective dimension reduction method called CDR for efficient anomaly detection, which can achieve 80%∼98% of reduction ratios and 4∼21x speedup compared with using original image data or AE/VAE compressed data.
- We incorporate a knee point detection algorithm in CDR to select a proper reduced dimension k automatically. We also propose a measurable method where users can decide how much information (i.e., variance) to preserve for optimal k selection.
- We conduct extensive experiments on various datasets and anomaly detection models. The evaluation results demonstrate that CDR can significantly reduce the dimension size of various high-dimensional data and preserve sufficient essential information for making anomaly detection effective. Moreover, CDR for anomaly detection consistently outperforms the state-of-the-art techniques on various datasets.

II. Preliminaries

A. Combination of PCA and AE

To assess the impact of dimensional reduction from PCA and AE, especially the significant of reduction order, we first explore the combination of PCA and AE: AE only, PCA only, PCA on AE (AE + PCA), and AE on PCA (PCA + AE). As shown in Figure 3, AE generally performs better than PCA to represent the original data when the dimension is small. AE generates better results because it is non-linear and incorporates a more sophisticated mechanism so that the model can generalize more complex relationships of the data. Moreover, the combination of PCA on AE (the 4th column) obtains the best reconstruction quality, outperforming the PCA only (the 3rd column) or the AE only (the 2nd column), while the reconstruction quality of PCA on AE (the 5th column) is the worst. The reason behind this is that after AE has extracted the most useful information, PCA further compacts the dimension

to remove noisy or irrelevant features. On the other hand, when we apply PCA first on the original image, compared to AE with the same dimension size, less information is preserved in PCA. Therefore, more useful information could be lost further in the AE compressor due to the insufficient information fed as input.

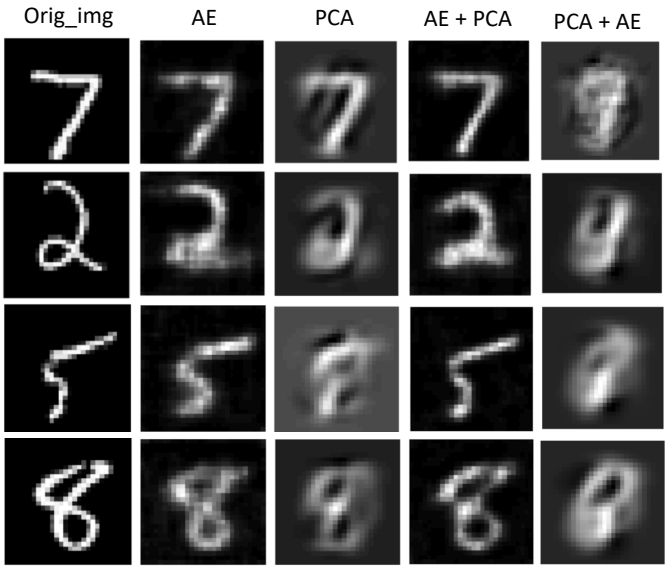


Fig. 3. Examples of using the combination of PCA and AE/VAE for dimension reduction on MNIST. The dimension of the original image (Ori_img) is 784, and the reduced dimensions is 16. Specifically, the dimension of representation is set as 16 in AE only (the 2nd column), and 16 principal components (PCs) are selected in PCA only (the 3rd column). In AE+PCA (4th column), we first use AE to obtain representation z with 128 dimensions and then apply PCA on z to select 16 PCs. In PCA+AE, we use PCA for original images first and select 128 PCs. Then, we apply AE to extract 16 dimension features from the selected 128 PCs.

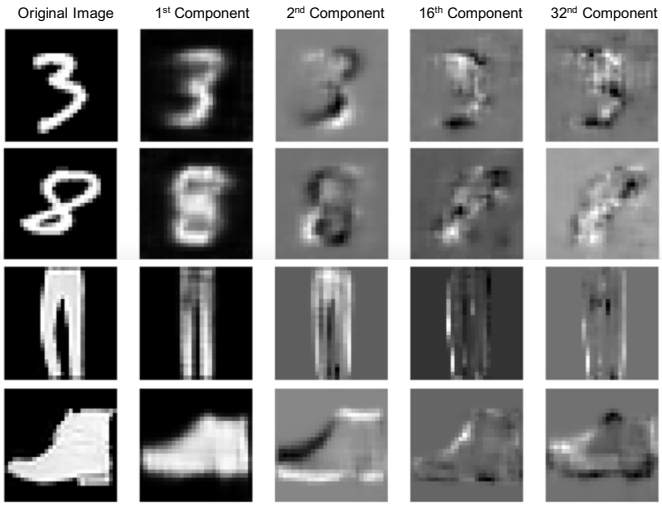


Fig. 4. Reconstructed images using different PCA components in MNIST and Fashion MNIST: the first column shows original images, the second to fifth columns are reconstructed images using 1st, 2nd, 16th, and 32nd PCA components, respectively.

B. Essential Information Compacted in Few Components

Since AE+PCA has the potential, we next investigate if most essential information can be compacted in a small set of PCs after applying PCA on z. Figure 4 demonstrates images reconstructed from the 1st, 2nd, 16th, and 32nd components of AE representation z, respectively. We can observe that the first two components (the 2nd and 3rd columns) capture the most essential information while the higher components (the 4th and 5th columns) contain irregularities or noises. For example, in the MNIST dataset of hand-written digits, the tilt in the 16th or 32nd component of '8' may make it look like '1' or '7' instead of '8'. We obtain similar observations on the Fashion MNIST dataset [23]. For instance, the higher components of the trouser (the 3rd and 4th columns of row 3) add noisy information that looks like a coat, while the first two components capture the most useful information of the trouser.

Our preliminary investigation shows the potential of combining PCA and AE to reduce the dimension of high-dimensional data. Among the combination of PCA and AE, we obtain the best reconstruction results when we apply AE first and then apply PCA on AE learned representation. Also, we can observe that a small number of PCs can contain the most essential information in original high-dimensional data (i.e., original images). However, different datasets exhibit varied characteristics, so does the number of PCs required to contain sufficient information for anomaly detection. In this paper, we propose two techniques to select a proper reduced dimension of k automatically.

III. Our Methodology

Figure 5 depicts an overview of our proposed method: cascaded dimension reduction (CDR) for efficient and effective anomaly detection. First, we apply the AE/VAE network to extract the latent representation z. We then apply PCA on z and calculate the cumulative explained variance (CEV) on the PCA components, where the cumulative proportion of variance of z is calculated. Next, we use the user-defined percentage of CEV to select reduced dimension k as needed or auto-select the dimension size k via knee point detection on CEV. Then the selected k components are used as inputs for anomaly detection algorithms, such as a neural network classifier.

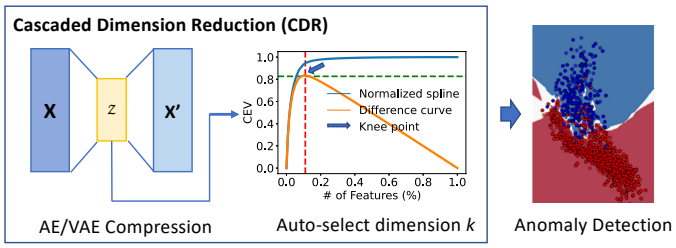


Fig. 5. Overview of our proposed method.

A. AE Compression

AE is an unsupervised neural network with an encoder-decoder architecture, where the encoder compresses high-

dimensional data into low-dimensional data (i.e., compressed data), and the decoder uses the compressed data to reconstruct the original high-dimensional data. The AE network is usually trained to minimize the reconstruction loss functions, such as mean square error (MSE), cross-entropy (CE), or structural similarity index measure (SSIM). To illustrate how CDR works, let us denote the x as original inputs, e as encoder, d as decoder, and ϵ as reconstruction error. We formulate the first step of CDR using AE to learn encoded representation z as follows:

$$(e^*, d^*, z) = \arg\min \epsilon(x, d(e(x))). \tag{1}$$

Besides AE, we also use VAE in CDR. VAE assumes the original data follow a certain underlying probability distribution, such as Gaussian distribution. The loss function of VAE, besides the reconstruction loss like AE, also includes Kullback–Leibler (KL) divergence to constraint the latent distribution. Usually, we assume the latent vectors are produced from a Gaussian distribution $p_\theta(z)$ with a zero mean and unit variance, and a VAE network tries to learn an approximated distribution $q_\phi(z|x)$ based on the input data. Similarly, we can formulate the first step of CDR using VAE to obtain z as follows:

$$(e^*, d^*, z) = \arg\min\{\epsilon(x, d(e(x))) + KL(q_\phi(z|x)||p_\theta(z))\}. \tag{2}$$

B. Auto-Select Dimension k

After obtaining the representation z from an AE or VAE network, we apply PCA on z. In PCA, the relationship among features is identified through a covariance matrix, and we can obtain the eigenvectors and eigenvalues λ via eigendecomposition of the covariance matrix. We then transform the data into principal components with the identified eigenvectors. Meanwhile, we can use eigenvalues to calculate CEV to select the most important k principal components. CEV is calculated as:

$$CEV = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{n} \lambda_i}, \tag{3}$$

where λ_i is the eigenvalue of i-th principle component, n is the dimension number of z, and $k \leq n$. We use CEV to measure how much variance is captured in the selected k components, such as 90%, 95% of the variance. In other words, we can set a threshold of CEV that represents how much variance is required to preserve and then decide how many components we need to select. Note that the variance in PCA components is sorted. Therefore, the first component contains the most variance, the second component captures the second most variance, and so on.

In addition to selecting k components via a pre-defined CEV threshold, we propose a knee point detection technique on CEV to automatically decide the k. In this paper, we define knee point as the optimal point that can best balance between the dimension size and preserved information (i.e., variance). In other words, the cost of increasing dimension sizes is no longer worth the expected benefit of preserving useful information beyond the knee point. Therefore, the degree of increase in CEV starts to decrease when the dimension size is larger than the knee point. In Figure 6, we demonstrate an example of how to find the optimal k via our knee point detection algorithm. First, we fit the CEV of latent represent into a smoothing spline so that it can preserve the overall behavior of the latent distribution. Then we normalize the CEV and feature number into a unit square. The knee point on the normalized CEV (z) curve (in blue color) is the point that has maximum curvature. That is the maximum distance between the normalized CEV (z) curve and the line of $y = x$. Mathematically, the knee point is a local maximum that can be calculated via the first and second derivatives of the spline curve (i.e., CEV curve in our case). We calculate the knee point using the equation denoted as:

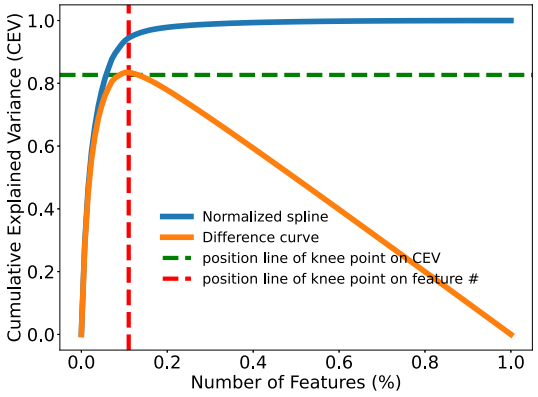


Fig. 6. Example of applying knee point detection algorithm on CEV.

$$K_{CEV}(z) = \frac{CEV''(z)}{(1 + CEV'(z)^2)^{1.5}}, \tag{4}$$

where K is the knee point of the CEV curve of z (i.e., $CEV(z)$). For example, in Figure 6, after applying the knee point detection algorithm on normalized $CEV(z)$, we obtain that the knee point is 0.11. That means 11% of the total number of features of z (i.e., 128) is selected for the optimal k dimension, which is $0.11 \times 128 \approx 14$. The complete training procedure of our proposed method CDR is illustrated in Algorithm 1.

IV. Evaluation

A. Experiment Setup

a) Datasets: We evaluate our proposed method on five benchmark datasets with high dimensions: MNIST [15], Fashion MNIST (FMNIST) [23], CIFAR-10 [24], KDDCUP99 (KDDCUP) [25], and Arrhythmia [26]. Table I shows the statistics of these evaluated datasets.

- **MNIST, FMNIST, CIFAR-10**: Each dataset has ten classes, and we create ten one-class classification setups, similar to the setup in DeepSVDD [27]. That is, we create one class from the entire class as a normal class, and the remaining classes are processed as anomalies. For

Algorithm 1 Cascaded Dimension Reduction (CDR).

Input: Dataset X with m features.
f_θ: AE/VAE network.
z: latent representation obtained (dimension size n).
λ: eigenvalues of PCA transformation of z.
CEV: cumulative explain variance.
K: knee point detection model.
Method 1: knee point detection; Method 2: pre-defined CEV threshold η.
Output: k selected components.

$z \leftarrow f_\theta(X)$.
PCs, eigenvalues $\lambda \leftarrow$ apply PCA on z.
$CEV(z) \leftarrow \lambda$ via Equation 3.
if choose to use method 1 **then**
  $k \leftarrow K(CEV(z))$ via Equation 4.
else if choose to use method 2 **then**
  for $i = 1, 2, ...n$ **do**
    if $CEV(z_i) \geq \eta$ **then**
      $k \leftarrow i$.
    end if
  end for
end if

TABLE I
STATISTICS OF THE EVALUATED DATASETS.

Dataset	# Dimensions	# Instances	Anomaly ratio
MNIST	784	70,000	0.1
FMNIST	784	70,000	0.1
CIFAR10	3072	60,000	0.1
KDDCUP	120	494,021	0.2
Arrhythmia	274	452	0.15

example, samples of classes '1-9' are anomalies for the data of normal class '0'. We adopt the original training and test split of these datasets and use the respective normal class only for the AE/VAE training step of CDR. In this way, the train set size is $n \approx 6,000$ for each class of MNIST and FMNIST, and $n = 5,000$ for CIFAR-10. All the test sets have 10,000 samples, including samples from the remaining nine abnormal classes in each setup.

- **KDDCUP**: The KDDCUP dataset consists of 494,021 network logs. Each log has 41 features, where seven features are categorical and the others are continuous. Though there are different ways to encode the features, for a fair comparison, we adopt the preprocess mechanism used in DAGMM [28]: we apply one-hot representation for the categorical features, and eventually, obtain the KDDCUP dataset with 120 dimensions for evaluation.
- **Arrhythmia**: The Arrhythmia dataset is used to determine the type of arrhythmia via the ECG recordings [26]. It contains 452 recordings and 16 classes, and each recording contains 274 features. Similarly, we adopt the process of Arrhythmia in DAGMM [28] for anomaly detection: we combine classes 3, 4, 5, 7, 8, 9, 14, and 15, which have the smallest amount of recordings, to form the anomaly class, and the remaining classes are combined as the normal class.

b) Network Architecture:

- **MNIST, FMNIST, CIFAR-10**: For a fair comparison, we adopt a similar encoder-decoder network setting used in DeepSVDD for the AE/VAE part in CDR. Specifically, each convolutional neural network (CNN) block consists of a convolutional layer followed by an activation function of leaky ReLU and 2×2 max-pooling. For MNIST and FMNIST, we use two CNN blocks with 8× (5×5×1)-filters, 4× (5×5×1)-filters, and a dense layer of 128 units for latent vectors. For CIFAR-10, we use three CNN blocks with 32× (5×5×3)-filters, 64× (5×5×3)-filters, 128× (5×5×3)-filters, and a dense layer of 512 units for latent vectors. For anomaly detection, we use scikit-learn [29] MLPClassifier with a hidden layer size of 100. The batch size is 512, and the training epoch is 100 for MNIST and FMNIST and 200 for CIFAR-10. The initial learning rate is 0.001 and gradually decreases during training, and the weight decay is set to $1e-6$.
- **KDDCUP**: Unlike the above-mentioned image datasets, KDDCUP data is sequential, thus we use fully connected (FC) layers for AE/VAE part of CDR. Specifically, the network consists of FC(120, 60)-FC(60, 30)-FC(30, 10)-FC(10, 30)-FC(30, 60)-FC(60, 120) and the activation function of tanh is used for each FC layer.
- **Arrhythmia**: Similar to KDDCUP, data in Arrhythmia is sequential and we use (FC) layers to build the AE/VAE network in CDR. The network comprises FC(274, 64)-FC(64, 16)-FC(16, 64)-FC(64, 274), and also we use the activation function of tanh for each FC layer.

c) Evaluation Schemes: First, we analyze the low-dimensional representation learned via CDR. We then evaluate the anomaly detection performance using the low-dimensional representation and compare it with the detection performance of using original images and using AE/VAE only, and state-of-the-art methods, including DeepSVDD [27] and DAGMM [28]. Next, we calculate the data reduction ratios by CDR and its speedup compared with other methods. Also, we evaluate eleven additional commonly used detection models to verify the applicability of CDR.

B. Representation learned by CDR

As CDR consists of two cascaded compressors, AEs and PCAs, we evaluate various combinations of AEs (AE and VAE) and PCAs (PCA and its non-linear variant Kernel PCA (KPCA) [30]). Specifically, The combination includes PCA on AE (AE + PCA), KPCA on AE (AE + KPCA), PCA on VAE (VAE + PCA), and KPCA on VAE (VAE + KPCA), and we also compare with PCA/KPCA on original images (Orig + PCA/KPCA). As shown in Figure 7, CDR using AE (rows 3 and 4) and VAE (rows 5 and 6) performs much better than using original images (rows 1 and 2). Regarding VAE and AE, VAE performs better than AE. AE + KPCA performs slightly better than AE + PCA. VAE + PCA obtains the best reconstruction quality, especially when the dimension is small (e.g., n=2).

Furthermore, we analyze the CEV and MSE of original images (Orig_img), AE encoded representation (AE-z), and VAE latent representation (VAE-z) to verify the reasons behind the observations in Figure 7. As the ranges of CEV and

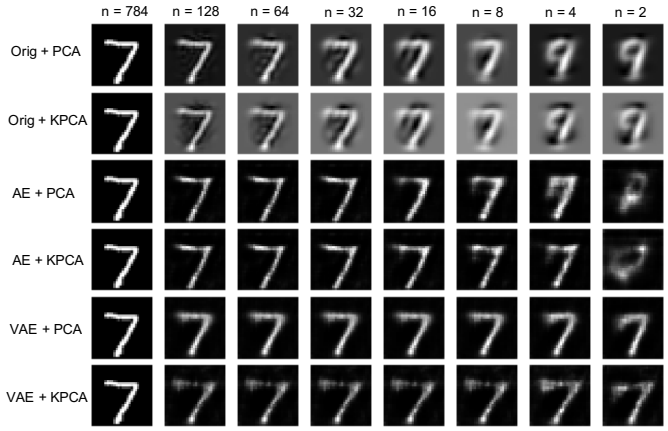


Fig. 7. Comparison of various combinations of evaluated dimension reduction schemes.

dimension sizes are different for Orig_img and AE/VAE-z, we normalize the CEV range to $[0, 1]$ and normalize the dimension (i.e., number of PCA components) to 0% to 100%. In Figure 8, the primary (left) y-axis represents the normalized CEV that evaluates the percentage of information preserved, and the secondary y-axis (right) is MSE that is used to evaluate the reconstruction error, and the x-axis represents the normalized number of PCA components used. As shown in Figure 8, the CEV of VAE-z is most concentrated in the smallest number of PCA components. For VAE-z, less than 10% of PCA components can preserve 95% of CEV while AE-z requires around 20% of PCA components, and Orig_img requires even more number of PCA components to maintain the same amount of CEV. Also, VAE-z has the least reconstruction error when the dimension size is small ($\leq$ 35% of total number PCA components). In other words, CDR using VAE-z can capture the most useful information (in CEV) and have the lowest reconstruction error (MSE) using the least amount of PC components.

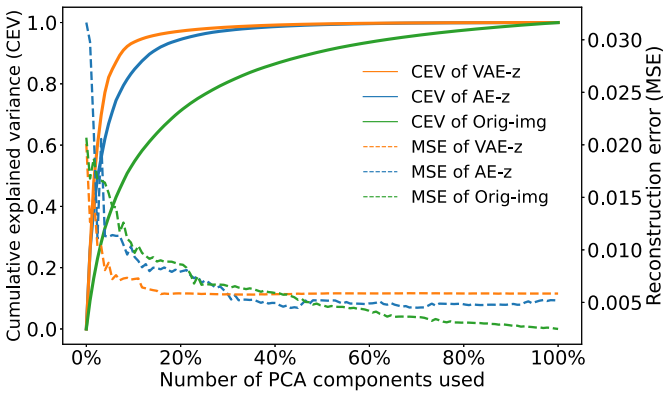


Fig. 8. Cumulative Explained Variance and MSE of original images, AE, and VAE compressed data.

C. Anomaly Detection Performance

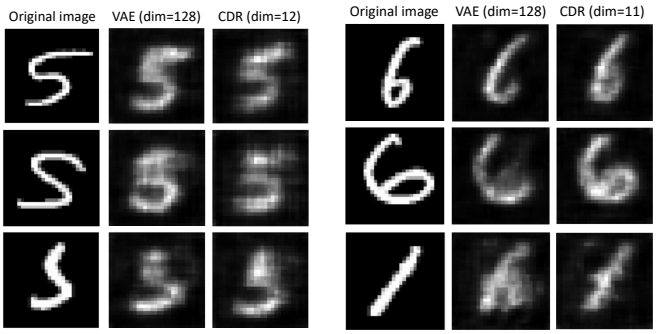
We next evaluate the anomaly detection performance using the low-dimensional representation of CDR and compare it with using the original image and VAE latent representation. Also, we compare the anomaly detection performance with the state-of-the-art deep learning anomaly detection methods: Deep-SVDD [27] and DAGMM [28]. As described in Section III, CDR has two schemes to select the reduced dimension size k: fixed CEV threshold and knee point detection. For the fixed CEV scheme, we use $\geq$ 95% of CEV (CDR-095). Note that the knee point detection (CDR-k) selects dimension size k automatically. Table II shows the evaluation results using MNIST. We can observe that our proposed method, CDR-095 or CDR-k, outperforms the state-of-the-art methods and the methods using original images or VAE only. Between our two methods, CDR-k performs slightly better than CDR-095, which indicates our knee point detection algorithm can adaptively select a more appropriate k than using a fixed CEV threshold.

TABLE II
COMPARISON OF ANOMALY DETECTION PERFORMANCE ON MNIST USING ORIGINAL IMAGES, VAE COMPRESSED DATA, STATE-OF-THE-ART METHODS (DEEP-SVDD AND DA-GMM), AND OURS (CDR-095 AND CDR-K). AVERAGE AUCS IN % WITH STDDEVS (OVER 10 SEEDS) PER METHOD.

Normal Class	Orig_img	VAE-z	DEEP-SVDD	DA-GMM	CDR-095 (ours)	CDR-k (ours)
0	**98.2±0.6**	97.3±0.6	98.0±0.7	98.0±0.5	96.0±0.5	97.3±0.5
1	99.1±0.2	99.1±0.1	**99.7±0.1**	99.4±0.2	98.3±0.2	98.9±0.2
2	91.5±0.6	**96.7±0.8**	91.7±0.8	90.2±2.8	96.5±0.8	96.3±0.8
3	92.8±0.7	92.9±0.7	91.9±1.5	92.2±1.7	93.4±0.6	**93.9±0.6**
4	93.3±0.5	93.1±0.5	**94.9±0.8**	90.0±2.0	92.2±0.7	92.8±0.6
5	89.4±0.9	90.7±0.8	88.5±0.9	87.4±2.9	90.9±0.8	**91.1±0.7**
6	96.9±0.4	97.6±0.5	98.3±0.5	96.5±0.9	**98.3±0.4**	**98.3±0.4**
7	94.6±0.5	**95.7±0.5**	94.6±0.9	93.2±1.2	94.8±0.5	95.5±0.5
8	95.0±1.3	93.8±0.9	93.9±1.6	92.6±1.2	**95.8±1.1**	95.5±1.1
9	95.7±0.4	95.4±0.3	96.5±0.3	95.3±0.8	95.3±0.3	**95.8±0.2**

In Figure 9, we demonstrate examples of using different representation for anomaly detection: original images (dimension size n=784), VAE latent representation (n=128) and CDR representation using knee point detection (n=12 for '5' and n=11 for '6'). The normal classes are 5 and 6 as shown in Figure 9a and Figure 9b, respectively. First, we can observe CDR can effectively extract the most important information from original images like VAE. The visual outcome from both VAE and CDR are similar, but CDR requires a much smaller dimension size, 128 vs. 11 or 12. Moreover, CDR could capture more useful information than VAE and remove noisy or irrelevant features, improving the anomaly detection performance accordingly. For example, in the first two rows of Figure 9b and the second row of Figure 9a, the reconstructed images using the representation by CDR are visually more evident than using VAE, which would make the classifier easier to classify them correctly. Also, we present examples in which the classifier failed to classify samples correctly by using original image or VAE representation but succeed by using CDR representation. The last row in Figure 9a shows

the case where normal class samples are misclassified as anomalies. The original image of '5' is significantly irregular. As the representation with higher dimensions tends to keep the original irregular information, the reconstructed image from the VAE representation looks more like '3' instead of '5'. While the reconstructed image from CDR representation still can be recognized as '5' by a classifier. Also, in the last row in Figure 9a, the number '1' is an anomaly for the normal class of '6', and the classifier misclassifies it as normal by using VAE representation, as it includes some noises that make it look like '6'. However, by using CDR representation, the classifier correctly detects it as an anomaly. This behavior is because CDR reduces the irrelevant noise and preserves the most essential information with lower dimension sizes.



(a) Normal Class 5 (b) Normal Class 6

Fig. 9. Examples of VAE and CDR for images dimension reduction.

Figure 10 demonstrates visualization of learned representation using different approaches, which helps explain why our proposed method, CDR, outperforms the others. We can observe that anomalies and normal instances are more separated in Figure 10c and Figure 10f. In other words, the uncertainties between normal and abnormal samples are smaller, such that the classifiers are easier to determine samples are anomalies or normal, thereby achieving higher detection performance. Also, compared Figure 10c (i.e., CDR-k) with Figure 10f (i.e., CDR-095), the separation between normal and abnormal instances in CDR-k is slightly more than that of CDR-095, which explains why CDR-k performs slightly better than CDR-095 in anomaly detection.

Lastly, Table III, Table IV, and Table V present the comparison results of anomaly detection performance on CIFAR-10, FMNIST, KDDCUP, and Arrhythmia, respectively. Note that there is no comparison with Deep-SVDD in Table IV as Deep-SVDD [27] did not evaluate the performance on FMNIST, and in Table V, we adopt the same metrics (i.e., precision, recall, and F1 score) used in DAGMM to evaluate KDDCUP and Arrhythmia. As we can see, CDR for anomaly detection consistently outperforms state-of-the-art techniques. Overall, the extensive evaluations on various datasets have demonstrated that CDR can effectively extract the most important information while requiring a smaller dimension size.

TABLE III
COMPARISON OF ANOMALY DETECTION PERFORMANCE ON CIFAR-10 USING ORIGINAL IMAGES, VAE COMPRESSED DATA AND STATE-OF-THE-ART METHODS. AVERAGE AUCS IN % WITH STDDEVS (OVER 10 SEEDS) PER METHOD.

Normal Class	Orig_img	VAE-z	DEEP-SVDD	DA-GMM	CDR-095 (ours)	CDR-k (ours)
AIRPLANE	65.3±3.4	64.9±3.5	61.7±4.1	61.0±3.8	**65.8±3.6**	65.2±3.8
AUTOMOBILE	**69.5±1.2**	69.3±1.3	65.9±2.1	63.2±2.4	63.9±1.7	68.7±1.3
BIRD	58.1±0.8	58.4±0.9	50.8±0.8	52.7±1.3	58.4±0.9	**58.9±0.9**
CAT	58.9±1.0	58.5±1.1	59.1±1.4	58.1±3.8	59.3±1.2	**59.7±1.1**
DEER	64.2±0.5	63.4±0.5	60.9±1.1	64.5±1.5	**65.9±0.7**	63.8±0.8
DOG	65.4±1.9	**65.8±2.6**	65.7±2.5	62.7±2.6	63.0±2.4	62.4±2.4
FROG	74.7±1.5	73.1±1.6	67.7±2.6	72.7±1.8	68.4±1.8	**75.4±1.5**
HORSE	65.8±0.9	66.0±1.0	**67.3±0.9**	64.5±1.9	65.0±1.0	66.1±0.8
SHIP	76.0±0.8	76.2±0.8	75.9±1.2	73.6±1.0	75.7±0.8	**77.4±0.8**
TRUCK	72.2±1.2	70.3±1.1	**73.1±1.2**	72.6±1.1	67.5±1.3	69.2±1.2

TABLE IV
COMPARISON OF ANOMALY DETECTION PERFORMANCE ON FASHION MNIST USING ORIGINAL IMAGES, VAE COMPRESSED DATA AND STATE-OF-THE-ART METHODS. AVERAGE AUCS IN % WITH STDDEVS (OVER 10 SEEDS) PER METHOD.

Normal Class	Orig_img	VAE-z	DA-GMM	CDR-095 (ours)	CDR-k (ours)
T-SHIRT/TOP	86.9±0.5	**87.0±0.5**	83.1±0.8	86.4±0.6	86.6±0.6
TROUSER	93.6±0.1	94.8±0.2	92.5±0.4	94.6±0.2	**95.7±0.1**
PULLOVER	90.2±0.7	89.4±0.7	88.4±1.2	89.6±0.7	**90.4±0.7**
DRESS	**94.3±0.6**	94.1±0.9	91.6±1.1	92.9±0.8	93.4±0.7
COAT	84.0±0.9	84.1±1.0	84.5±1.5	**84.8±0.9**	84.1±1.0
SANDAL	95.7±0.4	**96.3±0.4**	94.2±0.7	95.3±0.4	95.2±0.4
SHIRT	80.1±0.4	79.8±0.5	78.7±1.2	78.7±0.5	**82.0±0.5**
SNEAKER	92.8±0.5	93.4±0.6	94.5±0.9	93.4±0.6	**94.4±0.3**
BAG	96.5±0.3	**97.7±0.5**	95.6±1.0	97.4±0.5	97.5±0.5
ANKLE BOOT	97.1±0.2	97.5±0.3	92.7±0.5	97.6±0.3	**97.7±0.2**

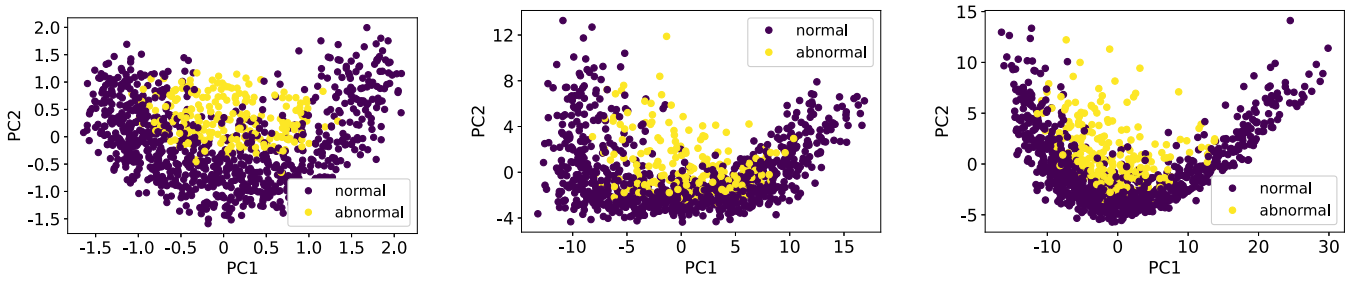
D. Dimension Reduction and Speedup

In Table VI, we present the reduced dimension size from CDR representation (i.e., the selected k components) using two schemes: CDR-095 and CDR-k on the three datasets: MNIST, FMNIST, CIFAR-10. Overall, CIFAR-10 requires more components for the representation than FMNIST and MNIST as CIFAR-10 is more complex than the other two. Similarly, more complex classes in a given dataset require more components. For example, in MNIST, number '5' needs more components than number '1'. The selected k in each dataset is relatively stable using knee point detection while using a pre-defined CEV threshold (i.e., 95%) obtains more fluctuated values. In other words, our knee point detection adaptively preserves the necessary information (i.e., variance) across the classes compared with a fixed threshold. Also, on average, the selected k via knee point detection algorithm is slightly larger than using the pre-defined threshold for each dataset.

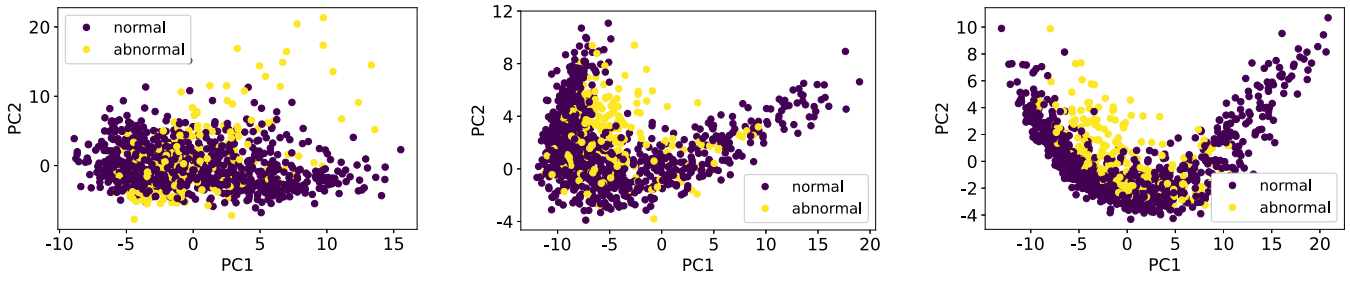
Next, we evaluate the reduction ratio (RR) by CDR and the speedup as a result. The reduction ratio is defined as:

$$RR = \frac{|D| - |D'|}{|D|}, \tag{5}$$

where D is the dimension size of original data and $|D'|$ is the dimension size of compressed data. Higher RR indicates



(a) Orig_img (b) DeepSVDD (c) CDR-k



(d) VAE-z (e) DAGMM (f) CDR-095

Fig. 10. Visualization of representation using different schemes for anomaly detection.

TABLE V
COMPARISON OF ANOMALY DETECTION PERFORMANCE ON KDDCUP AND ARRHYTHMIA USING ORIGINAL IMAGES, VAE COMPRESSED DATA, AND STATE-OF-THE-ART METHODS.

Method	KDDCUP			Arrhythmia		
	Precision	Recall	F1	Precision	Recall	F1
Orig	0.8679	0.92	0.8932	0.4875	0.5016	0.4944
VAE-z	0.8943	0.9402	0.9167	0.5029	0.5113	0.5071
DEEPSVDD	0.9024	0.9487	0.9250	0.5160	0.5027	0.5092
DAGMM	0.9297	0.9442	0.9369	0.4909	0.5078	0.4983
CDR (ours)	0.9341	0.9615	**0.9476**	0.5246	0.5815	**0.5516**

TABLE VI
THE REDUCED DIMENSION SIZE IN CDR REPRESENTATION (I.E., THE SELECTED k COMPONENTS) USING TWO METHODS: PRE-DEFINED CEV THRESHOLD 95% (CDR-095) AND KNEE POINT DETECTION (CDR-K) ON THREE DATASETS: MNIST, FMNIST, CIFAR10.

Class	MNIST		FMNIST		CIFAR10	
	CDR-095	CDR-k	CDR-095	CDR-k	CDR-095	CDR-k
0	11	12	13	14	43	35
1	6	7	12	15	30	36
2	11	12	11	15	22	21
3	12	11	12	14	29	39
4	10	11	12	13	34	38
5	11	12	17	16	51	37
6	11	11	13	14	32	36
7	10	11	9	12	30	35
8	10	11	14	16	35	34
9	12	11	16	15	32	40
Average	10	11	13	14	34	35

greater dimension reduction. In CDR, we evaluate two types of RR: RR with respect to original image size (RR_orig) and RR with respect to the dimension size of AE/VAE latent representation (RR_ae). Specifically, we use the dimension size of original images as D in Equation 5 to calculate RR_orig and use the dimension size of AE/VAE latent representation for RR_ae. D' is the dimension size of CDR representation for both RR_orig and RR_ae. The dimension size of different types of representation and corresponding CR are shown in Table VII. Note that, for CDR-095 and CDR-k, we use the average reduced dimension size for each dataset (the last row in Table VI). As we can observe from this table, CDR significantly reduces the dimension size, achieving an 80% to 93% reduction ratio with respect to AE/VAE representation sizes and over 98% reduction ratio with respect to original image sizes. In other words, CDR needs only around 10% of the AE/VAE representation size and less than 2% of original image sizes to extract the most useful information.

TABLE VII
DIMENSION REDUCTION AND REDUCTION RATIO OF CDR.

Dataset	Orig_img	AE/VAE-z	CDR	RR_ae	RR_orig
MNIST	784	128	11	91.4%	98.6%
FMNIST	784	128	14	89.1%	98.2%
CIFAR10	3072	512	35	93.2%	98.9%
KDDCUP	120	10	2	80.0%	98.3%
Arrhythmia	274	16	3	81.3%	98.9%

Lastly, Figure 11 presents the speedup brought by CDR on inference time, compared with using the original image, only AE/VAE representation, and the state-of-the-art methods. Compared with other techniques, our proposed method achieves 4∼8x speedups across various datasets and achieves

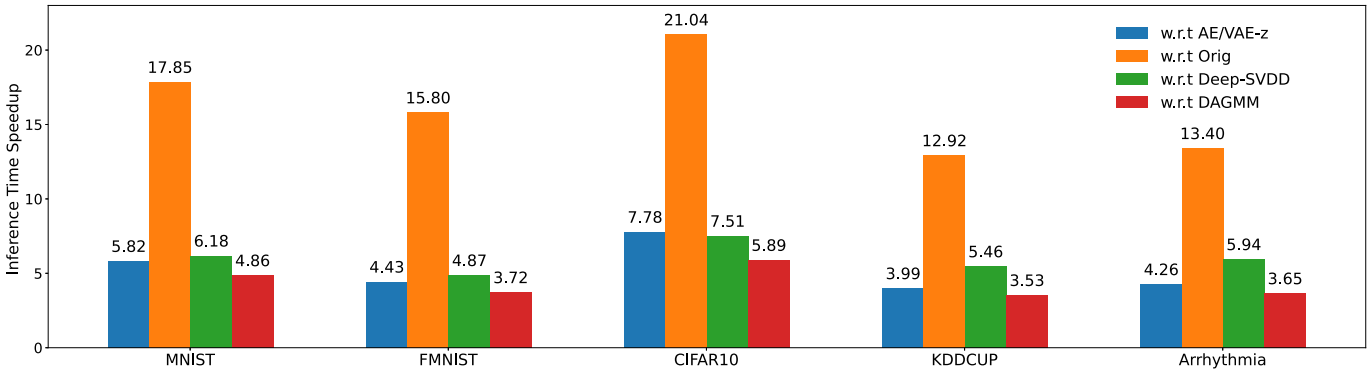


Fig. 11. Speedup brought by CDR for the five evaluated datasets: MNIST, FMNIST, CIFAR10, KDDCUP, and Arrhythmia.

13∼21x speedups compared with one using original images. We also evaluate the PCA transformation and knee point detection time and observe that those times are less than 0.1% of AE/VAE training time. In other words, the overhead of applying PCA on AE/VAE representation is negligible, and we obtain much more benefit for the downstream task of anomaly detection: less training and inference time and improvement of anomaly detection performance.

E. Verification on More Detection Methods

We thus far have demonstrated that our proposed CDR can obtain informative representation with minimal dimension size. We further showed that the neural network classifier performs well on the dimension-reduced representation via CDR. In this subsection, we verify if CDR is applicable to other anomaly detection models. Specifically, we evaluate the CDR representation for eleven additional detection models, including Logistic Regression (LR), Nearest Neighbors (NN), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), AdaBoost (AB), Naive Bayes (NB), Quadratic Discriminant Analysis (QDA), KMeans, Gaussian Mixture Model (GMM), and Kernel Density Estimation (KDE). The evaluation results are shown in Table VIII. For most detection models, CDR consistently outperforms the AE/VAE compressor with a smaller dimension size and achieves better anomaly detection performance. CDR reduces the dimension size and scarcity among data, which is beneficial for the distant-based model such as NN and SVM, or the model that prefers small dimension size, such as NB and QDA. Overall, supervised classification models (i.e., the first 8 rows in Table VIII) perform better than unsupervised clustering models (i.e., the last 3 rows in Table VIII) since the labeled data in supervised models provide useful prior knowledge that can help improve the detection performance. Among three clustering methods, more sophisticated models like GMM and KDE perform better than KMeans.

TABLE VIII
COMPARISON OF ANOMALY DETECTION PERFORMANCE BY CDR WITH ELEVEN ADDITIONAL DETECTION MODELS.

Method	MNIST		FMNIST		CIFAR10	
	VAE-z	CDR-k	VAE-z	CDR-k	VAE-z	CDR-k
LR	94.4±0.4	94.2±0.5	63.4±1.5	62.5±1.4	90.5±0.5	90.0±0.5
NN	90.2±0.6	**90.5±0.6**	63.0±1.3	**63.3±1.3**	89.8±0.7	**90.1±0.6**
SVM	92.4±0.5	**92.5±0.5**	63.8±1.5	62.9±1.5	86.0±0.7	84.7±0.7
DT	91.7±0.5	91.5±0.5	58.3±1.8	**59.0±1.7**	84.2±0.6	**84.5±0.6**
RF	93.5±0.5	**93.5±0.4**	59.5±1.6	60.1±1.4	88.3±0.5	87.7±0.6
AB	92.2±0.5	**93.2±0.6**	65.4±1.4	**65.6±1.4**	90.3±0.4	**91.3±0.3**
NB	92.1±0.4	**92.6±0.4**	63.7±1.5	**64.3±1.3**	86.2±0.6	**90.2±0.5**
QDA	90.2±0.6	**94.6±0.5**	64.3±1.6	**66.0±1.5**	88.1±0.5	**91.5±0.3**
KMeans	87.3±0.1	85.4±0.2	61.4±0.2	61.2±0.2	79.5±0.1	**80.7±0.2**
GMM	92.1±0.1	91.2±0.1	58.1±0.2	**61.4±0.3**	82.9±0.2	**83.0±0.3**
KDE	83.3±0.1	**86.8±0.1**	56.2±0.2	**60.6±0.1**	86.3±0.1	83.4±0.1

V. RELATED WORK

a) High-dimension Data Anomaly Detection.: Anomaly detection (AD) has been actively studied and been widely used in real-world applications, such as network intrusion detection [2], medical diagnosis [3], or senor networks [4]. Prior studies have proposed numerous anomaly detection techniques, including classification-based [5] [6] [7] [31] [32] [33], distance-based [10] [34] [35], probabilistic or spectral techniques [16] [8] [9]. However, as application data are growing rapidly in both size and dimensionality, most conventional AD methods are increasingly becoming less effective due to sparsity and irrelevant features, noise emerging in high-dimensional data [17]. Numerous methods have been proposed to reduce the dimension of the data, such as subspace-based [19] [36] [37] [38] and feature selection-based methods [39] [40] [41]. While these methods are effective for reducing dimension size, they cannot guarantee whether proper and sufficient information has been preserved in the subspace or the selected features for the downstream task of detecting anomalies accurately [42].

In recent years, deep learning has demonstrated promising results in learning expressive representation from complex data, such as using autoencoder to learn the representation of high-dimensional data (e.g., image data). Several deep learning anomaly detection methods have been proposed to address the problem of high-dimensional data anomaly detection. For example, [27] proposed a deep one-class classification model (Deep-SVDD) in which a deep learning network is trained to obtain a hypersphere where normal data is mapping inside the hypersphere whereas anomalies fall outside the hyper-

sphere. In [28], a deep autoencoding Gaussian mixture model (DAGMM) is proposed, which combines deep autoencoder and Gaussian mixture model to learn compressed representation and anomaly detection. In [43], a deep autoencoder and density estimation are combined to enable the simultaneous detection of both global anomaly and local anomaly. However, these deep learning anomaly detection models are sophisticated with complex designs that require long training time and inference time, as they mainly focus on the accuracy of detection performance, not the efficiency. In this work, we propose a simple yet effective method called CDR, which focuses on both efficiency and accuracy of detection performance.

b) Dimension Reduction.: A common strategy to handle high-dimensional anomaly detection problems is dimension reduction (DR). Dimension reduction is a technique that reduces the number of features to obtain lower-dimensional data that captures essential information in original data. There are two main approaches to perform dimension reduction: feature selection (or feature elimination) and feature extraction (or feature transformation) [17]. Feature selection techniques preserve only some important existing features and discard insignificant ones. Feature extraction techniques generate a reduced number of new features from the original features via the transformation techniques such as PCA [16], AutoEncoder [18], and t-SNE [44]. PCA projects data into the direction of high variance, transforming the original data into principal components. The principal components (PCs) with the most important information (i.e., highest variance) are kept and the PCs with lower variance are removed. Kernel PCA (KPCA) [30] is an extension of PCA for non-linear cases. KPCA uses a kernel function, such as polynomial or Gaussian kernel, to project data into a higher dimensional feature space where the data become linearly separable. Due to the relatively complex computation involved in KPCA, it is slower than PCA.

AutoEncoder (AE), on the other hand, consists of an encoder and decoder, where the encoder compacts the high-dimensional data to lower-dimension data, and the decoder uses the compressed data to reconstruct back the original data. Intuitively, AE creates a bottleneck that only the main information of the data can go through. Variational AutoEncoder (VAE) [20] is a generative model that learns an approximated probability distribution where the data are sampled from. It has similar architectures for dimension reduction like AE but also adds the regularization of latent distribution, which helps push the representation far from irrelevant or insignificant factors. However, as there is no general way to select a proper reduced dimension size k for AE or VAE, the k is set heuristically and tends to be larger than necessary. In other words, there is still an opportunity to minimize the redundancy [22] as the results we have shown in Figure 2b. In our proposed method CDR, we incorporate a knee point detection algorithm to select k automatically for the latent representation and minimize k while preserving necessary and sufficient information for anomaly detection. DPZ [45] proposed a multi-stage method that combines discrete cosine transform, PCA, quantization, and encoding to compress scientific datasets. While CDR is also a multi-stage method, it combines AE and PCA in series to extract the most significant features for anomaly detection.

VI. Conclusion

In this work, we propose a cascaded dimension reduction (CDR) technique for effective anomaly detection. In CDR, we serially apply two popular DR techniques, AE/VAE and PCA, and incorporate knee point detection to select the optimal k dimension automatically. Specifically, we use AE/VAE first to obtain representation z and then apply PCA on z to reduce dimension further. By utilizing the knee point detection algorithm for the cumulative explained variance (CEV) on z, we can auto-select the optimal reduced dimensional k for effective anomaly detection. The extensive evaluation using various datasets and detection models has demonstrated that CDR can significantly reduce the dimension size while preserving most essential information. Our results have also shown that CDR for anomaly detection consistently outperforms state-of-the-art techniques. Moreover, compared with using AE/VAE compressed data or original images, CDR can achieve 80%$\sim$98% reduction ratio and 4$\sim$21x speedup.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant No.1751143 and the NVIDIA hardware grant. The authors acknowledge the MIT SuperCloud, MGHPCC, and Lincoln Laboratory Supercomputing Center for providing (HPC, database, consultation) resources that have contributed to the research results reported within this paper.

References

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, Jul. 2009. [Online]. Available: https://doi.org/10.1145/1541880.1541882

[2] X. Zhuo, J. Zhang, and S. W. Son, "Network intrusion detection using word embeddings," in *2017 IEEE International Conference on Big Data (Big Data)*, 2017, pp. 4686–4695.

[3] F. Zhang, L. Luo, X. Sun, Z. Zhou, X. Li, Y. Yu, and Y. Wang, "Cascaded generative and discriminative learning for microcalcification detection in breast mammograms," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12 570–12 578.

[4] A. Moon, X. Zhuo, J. Zhang, S. W. Son, and Y. Jeong Song, "Anomaly detection in edge nodes using sparsity profile," in *2020 IEEE International Conference on Big Data (Big Data)*, 2020, pp. 1236–1245.

[5] A. Y. Ng and M. I. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes," in *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, ser. NIPS'01. Cambridge, MA, USA: MIT Press, 2001, p. 841–848.

[6] P. Cichosz, "Naïve bayes classifier," 2015.

[7] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[8] D. Reynolds, *Gaussian Mixture Models*. Boston, MA: Springer US, 2009, pp. 659–663.

[9] Z. I. Botev, J. F. Grotowski, and D. P. Kroese, "Kernel density estimation via diffusion," *The Annals of Statistics*, vol. 38, no. 5, pp. 2916 – 2957, 2010. [Online]. Available: https://doi.org/10.1214/10-AOS799

[10] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," 2000.

[11] R. E. Bellman, *Dynamic Programming*. USA: Dover Publications, Inc., 2003.

[12] D. L. Donoho, "High-dimensional data analysis: The curses and blessings of dimensionality," in *AMS CONFERENCE ON MATH CHALLENGES OF THE 21ST CENTURY*, 2000.

[13] T. Huang, H. Sethu, and N. Kandasamy, "A new approach to dimensionality reduction for anomaly detection in data traffic," *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 651–665, 2016.

[14] L. H. Nguyen and S. Holmes, "Ten quick tips for effective dimensionality reduction," *PLOS Computational Biology*, vol. 15, no. 6, pp. 1–19, 06 2019. [Online]. Available: https://doi.org/10.1371/journal.pcbi.1006907

[15] Y. LeCun and C. Cortes, "MNIST handwritten digit database," 2010. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[16] I. Jolliffe, *Principal Component Analysis*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 1094–1096. [Online]. Available: https://doi.org/10.1007/978-3-642-04898-2_455

[17] S. Thudumu, P. Branch, J. Jin, and J. J. Singh, "A comprehensive survey of anomaly detection techniques for high dimensional big data," *Journal of Big Data*, vol. 7, no. 1, p. 42, Jul 2020. [Online]. Available: https://doi.org/10.1186/s40537-020-00320-x

[18] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608014002135

[19] M. Sakurada and T. Yairi, "Anomaly detection using autoencoders with nonlinear dimensionality reduction," in *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis*, ser. MLSDA'14. New York, NY, USA: Association for Computing Machinery, 2014, p. 4–11. [Online]. Available: https://doi.org/10.1145/2689746.2689747

[20] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2014.

[21] A. F. Agarap, "Deep learning using rectified linear units (relu)," 2019.

[22] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "A survey of model compression and acceleration for deep neural networks," 2020.

[23] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," 2017.

[24] A. Krizhevsky, V. Nair, and G. Hinton, "Cifar-10 (canadian institute for advanced research)." [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html

[25] "KDD Cup 1999 Data." [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[26] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[27] L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, and M. Kloft, "Deep one-class classification," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 4393–4402. [Online]. Available: http://proceedings.mlr.press/v80/ruff18a.html

[28] B. Zong, Q. Song, M. R. Min, W. Cheng, C. Lumezanu, D. Cho, and H. Chen, "Deep autoencoding gaussian mixture model for unsupervised anomaly detection," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=BJJLHbb0-

[29] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[30] B. Schölkopf, A. J. Smola, and K.-R. Müller, *Kernel Principal Component Analysis*. Cambridge, MA, USA: MIT Press, 1999, p. 327–352.

[31] V. Podgorelec, P. Kokol, B. Stiglic, and I. Rozman, "Decision trees: An overview and their use in medicine," *J. Med. Syst.*, vol. 26, no. 5, p. 445–463, Oct. 2002. [Online]. Available: https://doi.org/10.1023/A:1016409317640

[32] L. B. Statistics and L. Breiman, "Random forests," in *Machine Learning*, 2001, pp. 5–32.

[33] K.-L. Li, H.-K. Huang, S.-F. Tian, and W. Xu, "Improving one-class svm for anomaly detection," in *Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.03EX693)*, vol. 5, 2003, pp. 3077–3081 Vol.5.

[34] P. Cunningham and S. J. Delany, "k-nearest neighbour classifiers - a tutorial," *ACM Computing Surveys*, vol. 54, no. 6, p. 1–25, Jul 2021. [Online]. Available: http://dx.doi.org/10.1145/3459665

[35] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: Identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '00. New York, NY, USA: Association for Computing Machinery, 2000, p. 93–104. [Online]. Available: https://doi.org/10.1145/342009.335388

[36] F. Keller, E. Muller, and K. Bohm, "Hics: High contrast subspaces for density-based outlier ranking," in *2012 IEEE 28th International Conference on Data Engineering*, 2012, pp. 1037–1048.

[37] T. Pevný, "Loda: Lightweight on-line detector of anomalies," *Mach. Learn.*, vol. 102, no. 2, p. 275–304, Feb. 2016. [Online]. Available: https://doi.org/10.1007/s10994-015-5521-0

[38] A. Lazarevic and V. Kumar, "Feature bagging for outlier detection," in *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, ser. KDD '05. New York, NY, USA: Association for Computing Machinery, 2005, p. 157–166. [Online]. Available: https://doi.org/10.1145/1081870.1081891

[39] G. Pang, L. Cao, L. Chen, and H. Liu, "Learning homophily couplings from non-iid data for joint feature selection and noise-resilient outlier detection," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, 2017, pp. 2585–2591. [Online]. Available: https://doi.org/10.24963/ijcai.2017/360

[40] G. Pang, L. Cao, L. Chen, D. Lian, and H. Liu, "Sparse modeling-based sequential ensemble learning for effective outlier detection in high-dimensional numeric data," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, Apr. 2018. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/11692

[41] F. Azmandian, A. Yilmazer, J. G. Dy, J. A. Aslam, and D. R. Kaeli, "Gpu-accelerated feature selection for outlier detection using the local kernel density ratio," in *2012 IEEE 12th International Conference on Data Mining*, 2012, pp. 51–60.

[42] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection," *ACM Computing Surveys*, vol. 54, no. 2, p. 1–38, Apr 2021. [Online]. Available: http://dx.doi.org/10.1145/3439950

[43] L. Nie, L. Zhao, and K. Li, "Glad: Global and local anomaly detection," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*, 2020, pp. 1–6.

[44] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008. [Online]. Available: http://jmlr.org/papers/v9/vandermaaten08a.html

[45] J. Zhang, J. Chen, X. Zhuo, A. Moon, and S. W. Son, "Dpz: Improving lossy compression ratio with information retrieval on scientific data," in *2021 IEEE International Conference on Cluster Computing (CLUSTER)*, 2021, pp. 320–331.