This is the peer reviewed version of the following article:

One-class tensor machine with randomized projection for large-scale anomaly detection in high-dimensional and noisy data

International Journal of Intelligent Systems

First published: **01 Jan 2021**

which has been published in final form at

https://onlinelibrary.wiley.com/doi/10.1002/int.22729

This article may be used for non-commercial purposes in accordance with Wiley Terms and Conditions for Use of Self-Archived Versions.

This article may not be enhanced, enriched or otherwise transformed into a derivative work, without express permission from Wiley or by statutory rights under applicable legislation. Copyright notices must not be removed, obscured or modified. The article must be linked to Wiley's version of record on Wiley Online Library and any embedding, framing or otherwise making available the article or pages thereof by third parties from platforms, services and websites other than Wiley Online Library must be prohibited.

One-class tensor machine with randomized projection for large-scale anomaly detection in high-dimensional and noisy data

Imran Razzak[1] **Nour Moustafa**[2] **Shahid Mumtaz**[3] **Guangdong Xu**[4]

[1]School of Information Technology, Deakin University, Geelong, Australia

[2]School of Engineering and Information Technology (SEIT), University of New South Wales (UNSW) at ADFA, Canberra, Australia

[3]Instituto de Telecomunicações, Aveiro, Portugal

[4]School of Computer Science, University of Technology, Sydney, Australia

Correspondence
Imran Razzak, School of Information Technology, Deakin University, KA5.201. W26, Geelong Waurn Ponds Campus, Geelong 3216, Australia.
Email: imran.razzak@deakin.edu.au

Abstract

The modern industrial sector generates enormous amounts of high-dimensional heterogeneous data daily. However, mostly the vectored data (rank-one tensor) have been considered for anomaly detection, whereas the data in real-life is high dimensional. The expressive power of methods based on vector data is restrictive as they may destroy the structural information embedded in data and lead to the curse-of-dimensionality and over-fitting. In this paper, we present a novel anomaly detection approach for large-scale tensor data. We first present novel one-class support tensor machines (OCSTM) with bounded loss function. We further extend it by leveraging the randomness to design a scalable approach that can also be used for large-scale anomaly detection. To solve the corresponding optimization of the objective function, we utilize half-quadratic optimization followed by solving it like a traditional OCSTM optimization at each iteration. We demonstrate the proposed randomized OCSTM with bounded hinge loss through experiments on 14 benchmark data sets. Experimental results demonstrate the effectiveness of the proposed approach against anomalies and a significant reduction in the computational complexity.

KEYWORDS
high-dimensional data, randomized, STM

1 | INTRODUCTION

Anomalies are rare events; however, are serious and threatening. For example, anomalous transactions indicate stolen credit cards. Hence, accurate identification of anomalous behavior is very important and has been widely used in several application areas, such as financial forecasting,[1] health-care,[2] intrusion detection,[3,4] industrial damage,[5,6] sensor networks,[7] robot behavior,[8] astronomical data,[9] fraud detection,[10] and fault diagnosis.[11,12] Synonymously, anomaly detection is also termed as novelty, adverse behavior or deviation detection and exception mining. Recently, one-class support tensor machine (OCSTM) has shown itself to be very effective approach for unsupervised outliers detection for high-dimensional data. Unlike one-class support vector machine (OCSVM), direct classification of tensorial data is able to preserve higher-order correlation; thus it has significantly outperformed traditional OCSTM methods both in terms of computational (space and time) and accuracy. However, traditional OCSTM is still sensitive in dealing outliers as well as not suitable for larger data sets.

To further improve the effectiveness of OCSTM in the presence of anomalies and overcome the computational challenge, recent work considers improving the robustness of loss function against outliers. Structure-preserving kernel mapping of features is utilized for nonlinear tensor.[13,14] To reduce the computational performance for larger data, nonlinear randomized projection is utilized[15]; however, experiments on large-scale corrupted data showed the sensitivity against outliers. Sparse representations-based tensor decomposition is also used to improve the challenges mentioned above, and it showed significant improvement.[16,17] Although recent support tensor machine (STM) variants showed significantly better performance than traditional STM, however, are still not efficient in the case of anomalous rich data. It may be due to the unsoundness of hinge loss that causes the large loss in the presence of outliers which results in deviation of the decision boundary.[18,19]

Recently, many works have focused on the development of nonconvex methods with rescaled hinge loss to decrease the impact of outliers. However, there is no work done for the advancement of OCSTM. Furthermore, the computational complexity of traditional support tensor machines is high and increases with training samples. Thus, it limits the application of STM for larger data sets. This study presents an efficient approach by introducing a scalable algorithm for the larger data set. Instead of using traditional hinge loss, we introduced hinge loss by bounding it and utilized randomized linear projection that not only helps improve the performance in the presence of outliers but also reduces the run time significantly. Extensive experiments on the benchmark data set showed that the bounding the loss function and addition of randomized linear kernel considerably improve the performance for outliers rich data and reduce the run time significantly compared with benchmark methods. We can describe the *key contributions* as

- Novel OCSTM by bounding the hinge loss and randomized projection, hence the proposed method is nonconvex, bounded, and monotonic, which considerably reduces the effect of the outliers.
- Considered randomized nonlinear feature set,[20] which eliminates the need to deal with larger kernel matrices required for larger data, improving space and time complexity.
- The objective function is nonconvex and challenging to optimize; thus, we presented half-quadratic optimization to solve it.
- Performed run time as well as asymptotic analysis to validate the computational performance.
- Extensive experiment on 14 benchmark data sets showed significant improvement in the detection of outliers with significantly better computational performance.

2 | PRELIMINARIES AND NOTATIONS

First, we describe the basic notations used throughout this paper followed by introducing some preliminary knowledge of tensor algebra. Vector, scalar, matrix, and tensor are represented by lowercase bold letters (e.g., $\boldsymbol{x}$), lowercase letters (e.g., x), uppercase bold letters (e.g., $\boldsymbol{X}$), and calligraphic uppercase letter (e.g., $\mathcal{X}$), respectively. For example, let $\mathcal{X} = [x_1, x_2, x_3, ..., x_N]$ be the $M \times N$ tensor that consists of n training subjects such that $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_M}$ (means $\mathcal{X}$ is the real Mth-order tensor and numbers $N_1, N_2, ..., N_M$ are called the dimensions of the tensor). $y_i \in \{1, -1\}$ are the class labels. Table 1 lists the basic symbols used throughout this paper.

Definition 1 (Tensor)**.** Just as vectors (are n-dimensional represented by one-dimensional array), a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_M}$ is a multidimensional array of real numbers that is a higher-order generalization of vectors (first-order tensors) and matrices (second-order tensors). Tensor is a geometric object that maps in a multilinear manner geometric vectors, scalars, and other tensors to a resulting tensor. Let $\mathcal{X} = [x_1, ..., x_N]$ be the $M \times N$ tensor consisting of n training subjects such that $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_M}$ (means $\mathcal{X}$ is real Mth-order tensor and numbers $I_1, ..., I_M$ are called the dimensions of the tensor). Their elements are represented by indices ranging from 1 to N, that is, an element of tensor is denoted by $\boldsymbol{x}_{i_1, ..., i_n}$ where $1 \leq n \leq N$ and $1 \leq i_n \leq I_n$.

Definition 2 (Tensor product)**.** The outer/tensor product of two tensors $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_P}$ and $\mathcal{Z} \in \mathbb{R}^{I'_1 \times \cdots \times I'_M}$ can be represented as

$$(\mathcal{X} \otimes \mathcal{Z})_{i_1, ..., i_P, i'_1, ..., i'_M} = x_{i_1, ..., i_P} z_{i'_1, ..., i'_M} \tag{1}$$

for all the values indices.

Definition 3 (Inner produce of tensor)**.** The inner produce/scalar product of two same-size tensors ($\mathcal{X}, \mathcal{Z} \in \mathcal{R}^{I_1 \times \cdots \times I_M}$) is the sum of products of their entries

TABLE 1 Notations

Notations	Description
$\boldsymbol{X}$	Boldface uppercase letter represents a matrix
x	Scalar is represented by a lowercase letter
$\mathcal{X}$	Tensor is represented by a calligraphic letter
$\boldsymbol{x}$	Boldface lowercase letter represents a vector
$\mathcal{R}$	Rank of tensor
y_i	corresponding class labels $y_i \in \{1, -1\}$
$[1 : M]$	Set of integers ranging from 1 to M
$\langle \cdot, \cdot \rangle$	Inner product of tensors
$\mathcal{K}(\cdot, \cdot)$	Kernel function
$\otimes$	Tensor product
$vec(\cdot)$	Column stacking operation
δ	Denotes delta function

$$\langle \mathcal{X}, \mathcal{Z} \rangle = \sum_{i_1=1}^{I_1} \cdots \sum_{i_M=1}^{I_M} x_{i_{1,\ldots,M}} z_{i_{1,\ldots,M}}. \tag{2}$$

Definition 4 (Rank-one tensor)**.** An Mth-order tensor $\mathcal{X}$ is the first-order tensor if it is the product of N vectors $\mathbf{u}_i \in \mathbb{R}^{I_i}$, where $1 \leq i \leq M$

$$\mathcal{X} = \mathbf{u}^i \otimes \cdots \otimes \mathbf{u}^M = \prod_{n=1}^{N} \otimes \mathbf{u}^n. \tag{3}$$

The rank R of the Mth-order tensor $\mathcal{X}$ can be determined by the minimum number of rank-one tensors that produce $\mathcal{X}$ in the linear combination. Instead of the whole tensor, storage of component vectors u^1, ..., u^M reduces the storage elements significantly, however, in real-world applications, rank-one tensor is rare.

Definition 5 (Tensor factorization)**.** A tensor decomposition represents an h-way tensor $\mathcal{X}$ as an h third-order tensor. It can be factorized if it can be decomposed as a rank-one tensor of length R.

$$\mathcal{X} = \sum_{r=1}^{R} x_r^1 \otimes \cdots \otimes x_r^M. \tag{4}$$

Definition 6 (Frobenius norm-tensor)**.** The Frobenius norm $\mathcal{X} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ can be defined as

$$\|\mathcal{X}\|_F = \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle}. \tag{5}$$

Definition 7 (Randomized nonlinear projection)**.** Suppose ϕ is the feature map $\mathcal{X} \longrightarrow \mathcal{H}$ such that dot product in $\mathcal{H}$ can be computed using the kernel function

$$\mathcal{K}(x, x') = \left\langle \phi(\mathcal{X}), \phi(\mathcal{X}') \right\rangle,$$

whereas $\mathcal{X}$ is mapped from input $\mathbb{R}^M$ to the feature space $\mathbb{R}^H$ through nonlinear projection function $\phi(\mathcal{X}) = \mathbb{R}^M \longrightarrow \mathbb{R}^{\mathcal{H}}$.

2.1 | One-class support tensor machines

Consider input samples in the data set $D = \{\mathcal{X}_i, y_i\}_{i=1}^N$ are the Mth-order tensors $\mathcal{X}_i \in \mathbb{R}^{I_1 \times \cdots \times I_M}$ with $y_i \in \{1, 0\}$ corresponding class labels for $i = 1, 2, ..., N$. OCSTMs can be formulated using quadratic optimization as

$$\min_{\mathcal{W}, p, \zeta} \quad \frac{1}{2}\|\mathcal{W}\|_F^2 + \frac{1}{N\nu}\sum_{i=1}^{N} \zeta_i - p \tag{6}$$

$$\text{s.t. } (\langle \mathcal{W}, \phi(\mathcal{X}_i)\rangle + b) \geq p - \zeta_i,$$

$$\zeta_i \geq 0, \quad \forall i = 1, ..., N,$$

where $\mathcal{W}$ tensor is the weight of separating hyperplane, $v \in (0, 1]$ is the regularizer that controls the fraction of anomalies and fraction of support vectors. Let ϕ is the mapping function that maps the data set into Hilbert space H and can be formulated as $\phi : \mathcal{X} \longrightarrow \phi(\mathcal{X}) \in \mathbb{R}^{H_1 \times H_2 \times \cdots \times H_{M'}}$. ζ_i are the slack variables that allow some of the data points on the other side of the hyperplane. By applying the Lagrange multiplier and solving Equation (6), we arrive at the following quadratic problem:

$$\min_{\alpha_1, ..., \alpha_N} \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j \mathcal{K}(\mathcal{X}_i, \mathcal{X}_j) \tag{7}$$

$$\text{s.t. } 0 \leq \alpha - i \leq \frac{1}{Nv}, \quad \sum_i \alpha_i = 1.$$

We can write the decision function for tensor as

$$f(\mathcal{X}) = sgn\left(\frac{1}{2}\sum_i^N \alpha_i \mathcal{K}(\mathcal{X}_i, \mathcal{X}_j) - p\right). \tag{8}$$

The solution equation (8) is characterized by parameter v that sets lower bound on the number of training samples used as support vectors and upper bound on the fraction of anomalies. Using the Karush–Kuhn–Tucker optimality condition, the input tensor data can be classified based on its projection below, above, or on the hyperplane boundary in the feature space based on the support tensors.

3 | THE PROBLEM

Traditional hinge loss function (Equation 6) is unbounded that causes larger loss in the presence of outliers. Furthermore, methods based on it work well for small data sets; however, are not scalable and computationally complex for larger data sets. Thus, it limits the applicability of OCSTM for outliers detection for the larger data set, especially when the data set is heavily corrupted. Unlike traditional hinge loss, bounding it could help decrease the loss in the presence of anomalies. Similarly, the nonlinear randomized feature map using random projection can be used to overcome the computational and space complexity challenge. Thus, this paper aims to develop a robust support tensor machine (R1STM-BH) for anomaly detection at a large scale.

4 | RANDOMIZED KERNEL BOUNDED ONE-CLASS STM

While OCSTM has shown itself a practical approach for detecting anomalous behavior of data to some extent, their ability to deal with large-scale corrupt data is still limited. The traditional hinge loss of STM results in large loss due to the existence of outliers. In addition

to this, finding the support vectors is computationally complex thus is not efficient for larger data sets. Unlike the traditional OCSTM loss function and search in high-dimensional space, this study presents STM by bounding the loss function and using the randomized set of features that result in significant improvement for outliers detection while considerably reducing the computational time. We first describe the support tensor machines by bounding the hinge loss followed by the utilization of nonlinear randomized features.

4.1 | Bounding loss function

To best segregate the data from outliers with maximal margin, OCSTM finds optimal hyperplane in high-dimensional data space. However, the unboundedness of loss function causes immense loss due to the existence of outliers, thus, considerably impacting the performance. By limiting the hinge loss, we can overcome the influence of outliers. We can rewrite the objective function of OCSTM (given in Equation 6) as

$$\begin{aligned} \max_{\mathcal{W},p} J(\mathcal{W},p) &= \frac{1}{2}\|\mathcal{W}\|_F^2 - \frac{1}{\nu N}\sum_{i=1}^{N}\aleph_i - p \\ \text{subject to} \quad & \langle \mathcal{W}.(\mathcal{X}_i)\rangle \geq p - \aleph_i, \\ & \aleph_i \geq 0 \quad \forall i = 1, ..., N, \end{aligned} \tag{9}$$

where $\aleph_i = \max\{0, p - \mathcal{Z}_i\}$ is the hinge loss with $\mathcal{Z}_i = \mathcal{W}\phi(\mathcal{X}_i)$.

Note that the traditional hinge loss function shown in Equation (9) is unbounded that can increase the loss due to the outliers. The traditional OCSTM (Equation 10) can be rewritten in bounded form (Equation 11) as

$$\begin{aligned} \max_{\mathcal{W},p} J(\mathcal{W},p) &= \frac{1}{2}\|\mathcal{W}\|_F^2 - p + \frac{1}{\nu N}\sum_{i=1}^{N}\wp_i \\ \text{subject to} \quad & \langle \mathcal{W}.(\mathcal{X}_i)\rangle \geq p - \aleph_i, \\ & \aleph_i \geq 0 \quad \forall i = 1, ..., N, \end{aligned} \tag{10}$$

$$\wp_i = \beta\left[1 - e^{-\eta\aleph_i}\right], \tag{11}$$

where $\eta \geq 0$ and $\beta = \frac{1}{1-e^{-\eta}}$ are the scale constant and normalization constant, respectively. η controls the upper bound and β ensures that $\wp_i = 1$. When $\eta = 0$ the hinge loss ($\wp$) degenerates to traditional ($\aleph$), thus, we can say that traditional STM (Equation 9) is a special case of R1STM-BH (10).

Equation (9) shows that the proposed objective is also nonconvex and monotonic. By simplifying Equations (9) and (10), we can rewrite the objective function as

$$\max_{\mathcal{W},p} J(\mathcal{W},p) = \frac{\beta}{\nu N}\sum_{i=1}^{N} e^{-\eta\zeta_i} + p - \frac{1}{2}\|\mathcal{W}\|_2^2. \tag{12}$$

4.2 | Optimization

The proposed objection function in Equation (12) is nonconvex, which makes it difficult to optimize and traditional optimization methods cannot be directly applied. Hence, we devised half-quadratic optimization.

$$\mathcal{R}(u) = -u\log(-u) + u, \quad u < 0. \tag{13}$$

By conjugate function theory, the above equation can be rewritten

$$e^{-\eta\aleph} = \sup_{u<0} \eta\aleph u - g(u). \tag{14}$$

We can achieve the supermum of $e^{-\eta\aleph}$ at $u = -e^{-\eta\aleph} < 0$.
Now, we can rewrite Equation (12) as

$$\max_{\mathcal{W},p} J(\mathcal{W}, p) = \frac{\beta}{\nu N}\sum_{i=1}^{N}\sup_{u_i<0}\{\eta\aleph_i u_i - g(u_i)\} + p - \frac{1}{2}\|\mathcal{W}\|_F^2, \tag{15}$$

$$\max_{\mathcal{W},p} J(\mathcal{W}, p) = \frac{\beta}{\nu N}\sup_{u<0}\left\{\sum_{i=1}^{N}\eta\aleph_i u_i - g(u_i)\right\} + p - \frac{1}{2}\|W\|_F^2, \tag{16}$$

$$\max_{\mathcal{W},p} J(\mathcal{W}, p) = \sup_{u<0}\left\{\frac{\beta}{\nu N}\sum_{i=1}^{N}\eta\aleph_i u_i - g(u_i) + p - \frac{1}{2}\|W\|_F^2\right\}. \tag{17}$$

We can simplify Equation (16) as

$$\max_{\mathcal{W},u,p} J(\mathcal{W}, u, p) = \frac{\beta}{\nu N}\sum_{i=1}^{N}\eta\aleph_i u_i - g(u_i)\} + p - \frac{1}{2}\|\mathcal{W}\|_F^2. \tag{18}$$

Iteratively solving Equation (17) using alternating methods to compute $\mathcal{W}$, u, and p.
Finally, we can write Equation (16) as

$$\max_{\mathcal{W},p} J(\mathcal{W}, p) = \frac{\beta}{\nu N}\sum_{i=1}^{N}\eta\aleph_i u_i + p - \frac{1}{2}\|\mathcal{W}\|_F^2. \tag{19}$$

We can rewrite Equation (19) as

$$\min_{\mathcal{W},p} J_o(\mathcal{W}, p) = \frac{1}{2}\|\mathcal{W}\|_F^2 + \frac{\beta}{\nu N}\sum_{i=1}^{N}\eta\aleph_i u_i - p. \tag{20}$$

We can apply the Lagrange multiplier to solve Equation (20). By applying the Lagrange multiplier on the above optimization problem, we get

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j \mathcal{K}(\mathcal{X}_i, \mathcal{X}_j) \tag{21}$$

s.t. $\sum_{i=1}^{N} \alpha_i = 1$ and $0 \geq \alpha_i \leq \frac{1}{\nu N} s_i$ for $i = 1, ..., N$, where k is the kernel matrix and $\alpha = [\alpha_1, ..., \alpha_N]^T$ is the vector of Lagrange multipliers.

We can compute the weight tensor $\mathcal{W}$ as

$$\mathcal{W} = \sum_{i=1}^{N} \alpha_i \phi(\mathcal{X}_i). \tag{22}$$

Finally, we can define the decision function as

$$f(x) = sgn(w\phi(x) - p), \tag{23}$$

$$f(x) = sgn\left(\sum_{i=1}^{N} \alpha_i (x_i, x) - p\right). \tag{24}$$

The quadratic problem in Equation (24) is characterized by the parameter ν that sets the lower and upper bounds on the fraction of anomalies and number of training subjects used as support vectors, respectively, thus it limits the loss due to the outliers.

To apply the kernel methods for tensor data, it has been converted into vectors or matrices,[21–23] which results in high dimensionality, overfitting, and destroying the structural information embedded in the tensor data. Thus, kernel learning is essential for tensor data to keep the structural information embedded in the tensor data by sets of essential structural features and design kernel on such sets. CANDECOMP/PARAFAC (CP) factorization has been employed to tensor to foster the use of kernel methods by extracting a structure-preserving kernel in tensor product feature space.[13] It provides an excellent approximation to the original tensor data. More specifically, in this way, each tensor can be represented as a sum of rank-one tensors in its original space, following by mapping them to tensor product features space for kernel learning.

Let $\mathcal{X} = \sum_{r=1}^{R} \prod_{n=1}^{M} \otimes X_r^m$ be the CP factorization of tensor $\mathcal{X}$ such that $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_M}$. The kernel of two same-size tensors can be written as $\mathcal{K}(\mathcal{X}, \mathcal{Y}) = \prod_{m=1}^{M} \mathcal{K}(x^m, y^m)$. Tensor data can be factorized in the feature space, similar to the original space. Feature space mapping on rank $R = 1$ feature mapping of a tensor can be defined as

$$\phi : \mathcal{X}^m \longrightarrow \phi(\mathcal{X}^m) \in \mathbb{R}^{H_1 \times \cdots \times H_M}, \tag{25}$$

$$\phi : \prod_{m=1}^{M} \otimes x^{(n)} \longrightarrow \prod_{m=1}^{M} \otimes \phi(x^{(m)}). \tag{26}$$

The CP factorization of tensor in the feature space is similar to the original space. The CP factorization of tensor $\mathcal{X}$ and $\mathcal{Y} \in \mathbb{R}^{I_1 \times \cdots \times I_m}$ is given as

$$\mathcal{X} = \prod_{m=1}^{N} \otimes x^{(m)} \text{ and } \mathcal{Y} = \prod_{n=1}^{M} \otimes y^{(m)}. \tag{27}$$

The kernel function of two same-size tensors $\mathcal{X}$ and $\mathcal{Y}$ can be written as

$$\mathcal{K}(\mathcal{X}, \mathcal{Y}) = \prod_{m=1}^{M} \mathcal{K}(x^{(m)}, y^{(m)}). \tag{28}$$

The feature mapping of tensors $\mathcal{X}$ and $\mathcal{Y}$ can be derived as

$$\phi : \sum_{r=1}^{R} \prod_{m=1}^{M} \otimes x^{(m)} \longrightarrow \sum_{r=1}^{R} \prod_{m=1}^{M} \otimes \phi(x^{(m)}). \tag{29}$$

This transformation corresponds to mapping the tensor data to high-dimensional tensorial feature space and performing the factorization in the high-dimensional space. Then the kernel in the high-dimensional space is the standard inner product of the tensor data in that feature space.[13] We can directly drive the naive tensor products kernels as

$$\mathcal{K}\left(\sum_{r=1}^{R} \prod_{m=1}^{M} \otimes \phi(x^{m}), \sum_{r=1}^{R} \prod_{m=1}^{M} \otimes \phi(y^{m})\right) \sum_{i=1}^{R} \sum_{j=1}^{R} \prod_{m=1}^{M} \mathcal{K}\left(x_i^m, y_j^m\right). \tag{30}$$

Although the objective function limits the effect of outliers, its computational complexity increases quadratically with the increase in the number of training samples. This issue can be solved using the linear kernel; however, it introduces biases to the origin. Another alternative is the radial basis function (RBF) kernel; however, it results in high computational complexity for high-dimensional kernels, making it inefficient for the larger data set. The use of randomization such as linear random projection showed itself a substitute to overcome the computational burden of kernel matrix construction.[24] Thus, to deal with the challenge above of computational and space complexity, we proposed to use randomized nonlinear projections that serve as a good approximation of the nonlinear kernel (Table 2).

4.3 | Randomized feature embedding

The complexity of the proposed OCSTM (objective function in Equation 9) grows quadratically with the increase in training samples. Thus, the proposed objective function is not efficient for larger data sets. To solve this, we presented the embedding of nonlinear randomized features into robust OCSTMs. Randomized projection is a prevalent approach to deal with overfitting and curse-of-dimensionality. We can randomly sample the parameters from a data-independent distribution and construct a d-dimensional randomized feature map. Thus, we applied OCSVMs with bounded loss function on the randomized nonlinear projection,[15] which reduces the computational complexity by eliminating the need for large kernel matrices for larger data sets. Consequently, reducing the space and computational complexity considerably while outperforming anomaly detection performance compared with conventional nonlinear machines. Here, our aim is to find the optimal $f(x)$ (fitting function) to minimize R_{Emp}.

TABLE 2 OCSTM-BH Algorithm

Input: Train-set: $\mathcal{X}_{i=1}^{N}$
where $X_j \in \mathbb{R}^{m\times n}$; $j = 1, ..., N$,
Kernel function $\mathcal{K}(\mathcal{X}_i, \mathcal{X}_j$
Scale constant $T_{\max}$, η,
Trade-off parameter τ
Output: margin parameter p
Lagrange multiplier α
Step-I: Initialization of parameters
Auxiliary variable $u \in \mathbb{R}^M \ni u_i < 0$, $T = 0$
While $T \leq T_{\max}$ do
Step-II: Calculate p and α^{T+1} by solving Equation (21),
Step-III: Calculate $u^{T+1} = -e^{-\eta\aleph}$.
Step-IV: Increase T by 1 and repeat step II to step III until it converges.
end while
Step-VI: Return p and α

$$f(\mathcal{X}) = \min \frac{1}{N}\sum_{i=1}^{N} c(f(\mathcal{X}_i), y_i) \quad \text{such that } \; y_i = 1, \tag{31}$$

where $c(f(\mathcal{X}_i), y_i)$ is the bounded hinge loss which penalizes the deviation between predictive and label values.

$f(\mathcal{X})$ can be computed through minimization of regularized risk as

$$R_{\text{Reg}}[f(\mathcal{X})] = R_{\text{Emp}}[f(\mathcal{X})] + \frac{1}{2}||f(\mathcal{X})||_F^2, \tag{32}$$

where $\frac{1}{2}||f(x)||_2^2$, $R_{\text{Emp}}[f(x)]$, and $R_{\text{Reg}}[f(x)]$ are the regularizer, empirical risk, and regularizer risk (average loss), respectively. We can compute the empirical risk as

$$R_{\text{Emp}}[f(\mathcal{X})] = \frac{1}{N}\sum_{i-1}^{N} L_{\text{B}}(f(\mathcal{X}_i), y_i), \tag{33}$$

where $L_{\text{B}}(f(\mathcal{X}_i), y_i)$ is the bounded hinge loss (described in Section 4.2) which penalizes the deviation between prediction and labels.

We can solve the fitting function by generating d-dimensional features and randomized sampling $s_i \in \mathbb{R}^d$ from independent distribution

$$Z(\mathcal{X}) = [(\mathcal{Z}_1, \mathcal{Z}_2, ... \; \mathcal{Z}_d)n],$$

where $\mathcal{Z}_i = [\cos(s_i^T, x_1 + b_i), ..., \cos(s_i^T, x_N + b_i)]$ and $e_j = [\cos(s_j^T, y_1 + b_j), ..., \cos(s_j^T, y_N + b_j)]$ are the Fourier-based random features.

Now, replacing the nonlinear kernels with randomized features kernel by unitizing the randomized rank-one tensor and CP factorization. We can rewrite the kernel in Equation (30) as

$$\mathcal{K}\left(\sum_{r=1}^{R}\prod_{m=1}^{M}\otimes\phi(x^m),\sum_{r=1}^{R}\prod_{m=1}^{M}\otimes\phi(y^m)\right)=\sum_{i=1}^{R}\sum_{j=1}^{R}\prod_{m=1}^{M}\left(\mathcal{Z}_i^{(m)}\right)^2 e_j^{(m)}. \tag{34}$$

Equation (34) randomized kernels,

$$\min_{\alpha\in\mathbb{R}d^d}\frac{1}{N}\sum_i\left(\alpha^T z_i,y_i\right)\quad\text{s.t. }\|\alpha\|_\infty\le\beta, \tag{35}$$

where β is a regularization constant.

Thus, utilizing nonlinear randomized features, the above formalization remarkably simplifies the computation. Theorem 1 justifies this claim.

Theorem 1. *Let D is the distribution on Ω and $\phi(x;s)\le 1$. Let $\mathcal{F}=\{f(x)=\int_\delta\alpha(s)\phi(x;s)\,ds:\alpha(s)\le\beta D(s)\}$. Let l be the L-Lipschitz loss function and $\lambda>0$. Draw $s_1,...,s_i$ iid from distribution D. We can write $\{f^*(x)=\sum_{j=1}^{i}\alpha_j\phi(x;s_j)$ minimizes the empirical risk*

$$E_D[l(f^*(x),y)]-\min_{f\in F}E_D[l(f(x),y)]\le O\left(\left(\frac{LB}{\sqrt{N}}+\frac{LB}{\sqrt{d}}\right)\sqrt{\log\frac{1}{\delta}}\right) \tag{36}$$

with a probability of at least $1-2\delta$.

4.4 | Convergence

As we have used half-quadratic optimization to transform the objective function to traditional OCSTM, thus, the R1STM-BH objective function can be related to traditional one-class STM. The convergence theorem can be described as

Theorem 2. *For a given input $\mathcal{X}\in\mathbb{R}^{N\times M}$, the kernel matrix $\mathcal{K}_{i,j}=\mathcal{K}(x_i,x_j)$ and its approximation $\hat{\mathcal{K}}$ using d random features, the following condition holds*:

$$E\|\hat{\mathcal{K}}-\mathcal{K}\|\le\sqrt{\frac{3N^2\log N}{d}}+\frac{2N\log N}{d}. \tag{37}$$

Proof. $\hat{\mathcal{K}}:\frac{1}{d}\sum_{i=1}^{d}\mathcal{Z}_i\mathcal{Z}_i^T$ is $N\times N$ kernel matrix $\ni\mathbb{E}[\hat{\mathcal{K}}]=\mathcal{K}$ where $\hat{\mathcal{K}}=\frac{1}{d}\sum_{i=1}^{d}\mathcal{K}^i$. Since, d (randomized features) are sampled based on independent and identical distribution, X is constant. Individual error matrices can be written as

$$E=\hat{\mathcal{K}}-\mathcal{K},$$

$$E = \sum_{i=1}^{d} E_i \quad \text{s.t. } \mathbb{E}[E_i] = 0 \quad \forall E_i, \;\; i = 1, ..., d, \tag{38}$$

whereas

$$E_i = \frac{\hat{\mathcal{K}}^{(i)} - \mathcal{K}}{d}.$$

As we have utilized the bounded loss function, there exists a constant B such that $||Z||^2 \leq B$. In this experiment, we are considering bounded hinge loss, thus, B is constant such that $||Z||^2 \leq B$. Finally, based on triangle inequality and Jensen's inequality, we can write

$$||E_i|| = \frac{\mathcal{Z}_i\mathcal{Z}_i^T - \mathbb{E}[\mathcal{Z}\mathcal{Z}]}{d} \leq \frac{||\mathcal{Z}_i||^2 - \mathbb{E}[||\mathcal{Z}||^2}{d} \leq \frac{2B}{d}.$$

To bound variance of E, we bound variance of every E_i

$$\mathbb{E}[E_i^2] = \frac{\mathbb{E}\left[\left(\mathcal{Z}_i\mathcal{Z}_i^T - \mathcal{K}\right)^2\right]}{d^2},$$

whereas $\mathcal{K} = \mathbb{E}[\mathcal{Z}_i\mathcal{Z}_i^T]$

$$\begin{aligned}\mathbb{E}[E_i^2] &= \frac{\mathbb{E}\left[||\ \mathcal{Z}_i^2\ ||\ \mathcal{Z}_i\mathcal{Z}_i^T - \mathcal{Z}_i\mathcal{Z}_i^T\mathcal{K} - \mathcal{K}\mathcal{Z}_i\mathcal{Z}_i^T + \mathcal{K}^2\right]}{d^2} \\ &\succcurlyeq \frac{1}{d^2}[B\mathcal{K} - 2\mathcal{K}^2 + \mathcal{K}^2] \\ &\succcurlyeq \frac{B\mathcal{K}}{d^2}.\end{aligned}$$

Now, taking all summands together, we get

$$||\mathbb{E}[E^2]|| \leq ||\sum_{i=1}^{d} \mathbb{E}E_i^2|| \leq \frac{B||\mathcal{K}||}{d}. \tag{39}$$

Thus, we can say,

$$E||\hat{\mathcal{K}} - \mathcal{K}|| \leq \sqrt{\frac{3B||K||\log N}{d}} + \frac{2B\log N}{d}. \tag{40}$$

Observe that kernel evaluation and randomized features ($||\mathcal{Z}||^2 \leq B$, where $B \leq 1$) are upper bounded by 1, thus, both $||\mathcal{K}||$ and B are bounded by N, resulting Equation (37). □

5 | EXPERIMENTS

In this section, we evaluate and compare the performance of bounded OCSTMs and the effect of randomized feature selection for the task of anomaly detection. To validate the gain in performance, we have performed k-fold ($k = 10$) validation on both vector and tensor data sets

and compared the performance with state-of-the-art vector and tensor-based methods. As our core objective is the detection of the outliers in large-scale data, thus, to validate the effectiveness of the proposed approach against the larger data set, we have corrupted the data sets with outliers and performed several experiments with the various number of dimensions and records.

5.1 | Data set

In this experiment, we have considered both tensored and vectored data sets and conducted several experiments. In our first experiment, we have used vector data and transformed it into tensor form. For this purpose, we have used publicly available data sets mostly from UCI repository that are Iris, Import, Ionosphere, Lungs, Sonar,[25] Delftpump AR, Breast Cancer,[26] USPS,[27] Daily and Sport Activity (DSA),[28] Gas Sensor Array (GSA), and PAMAP2 physical activity monitoring data set (PAMAP). Most of these data sets are originally vector-based. Thus, these data sets are transformed to tensorial representation[29] and select the tensor size based on Reference [30]. In our second experiment, we have used tensor data and considered the CASIA gait recognition data set (data set-A[31]), which includes 19,139 images (about 2.2 GB). Furthermore, we have also used the face recognition data set (The ORL Database of Faces) and handwritten digits database (MNIST[32]).

Each record is normalized between $[0, 1]$. Each data set is segmented into train-set and test-set for evaluation purposes by randomly selecting 80% and 20% records, respectively. To observe the effectiveness of the proposed R1STM-BH, we have added 5% anomalies drawn from $\mathbb{U}(0, 1]$. As our approach is unsupervised outliers detection; thus, labels are omitted during training. However, we have used it in the testing phase.

5.2 | Results and discussion

The focus of this study is to improve the robustness of outliers detection and overcome the computational challenge for the larger data set. We have performed several experiments on both vectored and tensored data sets and performed k-fold cross-validation. As described in Section 5.1, we transformed the vector data into tensorial representation. Initially, we performed k-fold cross-validation for both vector (Breast Cancer) and tensor (MNIST) data set to find the optimal range of parameters, followed by an experiment on the rest of the data sets within that optimal range. We have corrupted all data sets with outliers to validate the robustness of the proposed R1STM-BH against anomalies.

To visualize the effect of the bounded hinge loss function and randomized features projection, we performed cross-validation (10 times) for all data sets. Results showed that some methods are comparatively better for a larger data set; however, they are poor for smaller data sets. Similarly, some methods showed better performance comparatively for smaller data sets. However, they provided poor performance for the large data set. Computational complexity is another major challenge that depends on the size of the data set, that is, kernel-based methods can be grouped quadratically based on complexity. To generalize the performance for both larger and smaller data sets, we have performed several experiments by varying the number of records and dimensions on different data sets. The following discussion provides the experimental results on both vectored (syntactically transformed to second- and third-order tensors)

and tensored data sets. Figures 1 and 2 and Tables 3 and 4 show the results on real and corrupted data sets with different subjects.

We compared the proposed R1STM-BH with benchmark methods approaches, such as OCSTM,[30] LOCSTM,[30] R1STM,[15] vector methods (OCSVM[33] LOCSVM,[34] and R1SVM[20]), and deep methods (One-Class Deep SVDD[35] and SB Deep SVDD[35]) on *14* publicly available benchmark data sets. Table 5 shows the comparison of results on the different number of training samples. We can notice that R1STM-BH showed better performance for the small number of training samples (2) and better performance for the larger number of samples (8) per individual, whereas the computational complexity remains very attractive. Table 6 shows the comparison of results on all data sets. We can notice that R1STM-BH showed comparatively better performance concerning OCSVM, LOCSVM, LOCSTM, R1SVM, OCSTM, and LOCSTM; however, results are comparable to deep SVDD and SB deep SVDD. Figures 1–3 show the results on Iris, Lungs, and ORL data sets, respectively. We can notice that R1STM significantly outperforms all for small sample size (2); however, results are comparable to deep SVD and SB deep SVDD for large sample size. This shows that R1STM-BH is scaleable and works for both small sample and larger sample data sets. We can

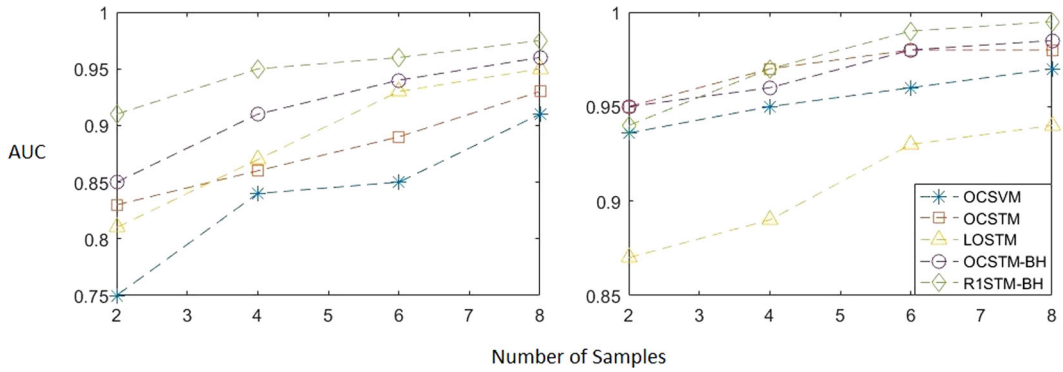

FIGURE 1 Performance comparison of proposed R1STM-BH with state-of-the-art methods on Iris data set (corrupted: left; original: right). AUC, area under curve; OCSTM, one-class support tensor machine; OCSVM, one-class support vector machine; R1STM-BH, robust support tensor machine; STM, support tensor machine [Color figure can be viewed at wileyonlinelibrary.com]

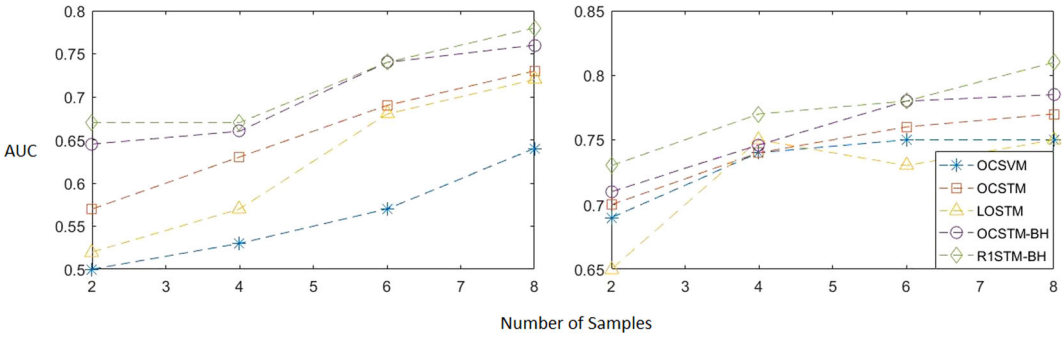

FIGURE 2 Performance comparison of proposed R1STM-BH with state-of-the-art methods on Lungs data set (corrupted: left; original: right). AUC, area under curve; OCSTM, one-class support tensor machine; OCSVM, one-class support vector machine; R1STM-BH, robust support tensor machine; STM, support tensor machine [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 3 Average AUC (%) on MNIST data set

Class	OCSVM-SVDD	SB Deep SVDD	OC-Deep-SVDD	OCSTM	R1OCSTM	OCSTM-BH	R1STM-BH
0	96.75 ± 0.5	97.8 ± 0.7	98.5 ± 0.7	97.87 ± 0.9	97.90 ± 1.1	98.1 ± 0.5	98.29 ± 0.72
1	99.15 ± 0.4	99.6 ± 0.1	99.7 ± 0.08	99.65 ± 0.6	99.60 ± 0.7	99.6 ± 0.07	99.33 ± 1.17
2	79.4 ± 0.9	89.5 ± 0.2	91.7 ± 0.8	90.20 ± 0.4	90.43 ± 0.9	92.1 ± 0.5	92.22 ± 1.07
3	86.1 ± 0.6	90.3 ± 0.1	91.9 ± 0.5	91.1 ± 0.3	91.00 ± 0.7	92.1 ± 0.4	92.22 ± 1.02
4	94.21 ± 0.3	93.8 ± 0.5	95.32 ± 0.8	93.21 ± 0.1	92.8 ± 0.54	95.2 ± 0.9	95.21 ± 0.79
5	73.1 ± 0.8	85.8 ± 0.5	89.23 ± 0.9	86.22 ± 0.4	97.1 ± 0.3	89.12 ± 0.9	89.23 ± 1.16
6	95.5 ± 0.2	98.0 ± 0.4	98.3 ± 0.5	98.0 ± 0.7	98.10 ± 0.1	98.60 ± 0.2	98.69 ± 0.19
7	92.16 ± 0.1	92.7 ± 0.4	94.6 ± 0.9	92.7 ± 0.3	93.85 ± 0.7	95.00 ± 0.5	95.09 ± 0.59
8	89.09 ± 0.4	94.2 ± 0.4	93.9 ± 0.6	93.6 ± 0.2	92.96 ± 0.4	94.1 ± 0.4	94.14 ± 0.54
9	92.71 ± 0.2	94.9 ± 0.6	96.35 ± 0.3	95.7 ± 0.3	95.71 ± 0.2	96.5 ± 0.5	96.61 ± 0.56

Abbreviations: AUC, area under curve; OCSTM, one-class support tensor machine; OCSVM, one-class support vector machine; R1STM-BH, robust support tensor machine; STM, support tensor machine; SVDD, Support Vector Data Description.

TABLE 4 Average AUC (%) Corrupted MNIST data set

Class	OCSVM-SVDD	SB Deep SVDD	OC-Deep-SVDD	OCSTM	R1OCSTM	OCSTM-BH	R1STM-BH
0	91.75 ± 0.22	92.11 ± 0.16	93.32 ± 1.1	93.05 ± 1.22	93.55 ± 1.32	95.22 ± 1.28	95.29 ± 1.11
1	92.45 ± 0.9	93.46 ± 1.4	93.32 ± 0.34	93.05 ± 1.4	92.02 ± 0.5	93.87 ± 0.22	93.91 ± 1.03
2	72.43 ± 1.45	78.32 ± 1.2	82.54 ± 1.45	83.08 ± 1.21	82.01 ± 1.43	84.20 ± 1.32	84.29 ± 0.96
3	78.2 ± 1.54	84.34 ± 1.56	84.19 ± 0.4	82.43 ± 1.06	84.41. ± 1.6	85.11 ± 1.1	85.43 ± 2.03
4	84.43 ± 2.1	86.47 ± 0.9	85.32 ± 1.44	84.61 ± 1.05	84.78 ± 1.00	85.13 ± 1.22	85.10 ± 2.17
5	65.43.1 ± 3.1	77.98 ± 1.3	80.11 ± 0.9	79.11 ± 3.6	83.79 ± 1.22	85.00 ± 1.54	85.19 ± 2.19
6	80.89 ± 1.70	86.76 ± 2.2	88.54 ± 2.5	86.54 ± 1.21	87.76 ± 2.10	88.67 ± 1.43	88.73 ± 1.28
7	80.22 ± 0.1	83.57 ± 0.84	85.55 ± 2.44	86.27 ± 1.23	85.43 ± 1.27	86.81 ± 0.76	86.93 ± 1.22
8	78.65 ± 2.4	86.4 ± 1.54	84.43 ± 2.60	82.46 ± 2.23	84.76 ± 2.43	87.43 ± 0.80	87.66 ± 1.83
9	81.89 ± 1.6	85.55 ± 1.96	84.45 ± 1.75	80.54 ± 4.5	84.79 ± 1.29	86.76 ± 1.1	86.83 ± 0.65

Abbreviations: AUC, area under curve; OCSTM, one-class support tensor machine; OCSVM, one-class support vector machine; R1STM-BH, robust support tensor machine; STM, support tensor machine; SVDD, Support Vector Data Description.

notice that R1STM-BS accuracy is slightly better in most cases; however, computationally, R1STM-BH is significantly better.

To validate the robustness against outliers, we have evaluated corrupted data. Results on corrupted data set are shown in Figures 1 and 2, and Tables 3 and 4. We can notice that R1STM-BH showed significantly better performance than the state-of-the-art methods for corrupted data. This shows that bounding the hinge loss overcame the larger loss that occurred due to the outliers, which affects the performance of anomaly detection. Notice that with the increased intensity of anomalies, OCSTM-BS showed significantly better performance (Table 7).

TABLE 5 Averaged Breast Cancer AUC (area under curve) and accuracy (%) with standard deviations

Num	Class	Metrics	OCSVM	LOCSVM	SB-Deep SVDD	OC Deep SVDD	OCSTM	LOCSTM	OCSTM-BH	R1STM-BH
2	Class 1	Accuracy	43.92 ± 9.82	68.43 ± 13.42	70.21 ± 10.21	71.24 ± 12.2	63.64 ±15.08	73.74 ± 14.21	74.43 ± 10.13	74.41 ± 9.21
		AUC	99.48 ± 0.13	99.48 ± 0.05	99.02 ± 0.04	99.11 ± 0.03	99.32 ±0.16	99.51 ± 0.03	99.63 ± 0.03	99.64 ± 2.22
	Class 2	Accuracy	65.20 ± 0.00	65.83 ± 8.75	67.11 ± 10.11	68.41 ± 9.23	69.40 ±5.17	65.85 ± 16.75	70.22 ± 8.77	70.28 ± 10.11
		AUC	80.93 ± 28.50	76.64 ± 31.88	80.23 ± 12.12	80.21 ± 21.10	84.62 ±23.68	77.55 ± 30.05	85.11 ± 19.21	85.63 ± 13.21
4	Class 1	Accuracy	59.80 ± 13.51	79.58 ± 10.81	83.22 ± 12.19	84.23 ± 13.70	75.67 ±13.02	84.16 ± 10.49	85.87 ± 8.74	85.96 ± 7.94
		AUC	99.43 ± 0.29	99.46 ± 0.06	98.950 ± 0.0	99.21 ± 0.01	98.52 ±1.92	99.48 ± 0.11	98.51 ± 0.0	98.84 ± 1.31
	Class 2	Accuracy	70.59 ± 4.89	63.92 ± 18.67	76.43 ± 6.7	78.21 ± 9.43	78.57 ±7.83	64.80 ± 24.57	79.69 ± 11.43	79.49 ± 10.96
		AUC	89.59 ± 15.31	71.35 ± 31.82	86.32 ± 8.7	90.43 ± 10.93	92.13 ±10.49	70.19 ± 30.92	93.21 ± 14.21	93.44 ± 10.54
6	Class 1	Accuracy	71.68 ± 13.16	84.47 ± 9.38	88.32 ± 12.43	89.21 ± 10.08	82.47 ±10.39	87.03 ± 10.61	88.45 ± 12.98	88.75 ± 11.40
		AUC	99.16 ± 1.10	99.27 ± 0.98	99.10 ± 0.01	99.43 ± 0.06	98.31 ± 1.84	99.02 ± 3.41	99.45 ± 0.11	99.64 ± 2.13
	Class 2	Accuracy	78.16 ± 5.51	68.13 ± 18.7	82.47 ± 4.40	83.86 ± 5.32	83.88 ± 5.95	65.88 ± 26.02	84.01 ± 2.4	83.89 ± 2.31
		AUC	93.82 ± 6.84	75.58 ± 26.79	94.71 ± 3.2	94.45 ± 7.2	92.96 ± 9.81	71.55 ± 29.03	94.21 ± 2.7	94.44 ± 2.75
8	Class 1	Accuracy	76.45 ± 11.65	86.00 ± 8.64	90.21 ± 10.34	90.65 ± 12.65	83.26 ± 10.26	89.02 ± 8.17	90.18 ± 14.51	90.33 ± 9.41
		AUC	99.33 ± 0.73	99.30 ± 0.62	99.51 ± 0.09	99.50 ± 0.11	98.50 ± 1.82	99.47 ± 0.12	99.54 ± 0.08	99.64 ± 2.18
	Class 2	Accuracy	80.90 ± 5.73	70.34 ± 21.13	85.91 ± 16.67	86.21 ± 19.21	84.96 ± 6.65	70.89 ± 23.37	86.43 ± 15.12	86.88 ± 10.19
		AUC	93.81 ± 7.86	75.91 ± 26.61	94.45 ± 3.66	94.51 ± 4.32	92.21 ± 10.69	75.93 ± 25.09	93.43 ± 14.55	93.713 ± 12.11

Abbreviations: OCSTM, one-class support tensor machine; OCSVM, one-class support vector machine; R1STM-BH, robust support tensor machine; STM, support tensor machine; SVDD, Support Vector Data Description.

TABLE 6 Average AUC (area under curve) (%) on benchmark data sets

	AUC								
Data set	**OCSVM**	**LOCSVM**	**SB-Deep SVDD**	**OC Deep SVDD**	**OCSTM**	**LOCSTM**	**R1STM**	**1STM-BH**	**R1STM-BH**
Breast Cancer	90.17	87.65	95.32	99.22	98.29	89.74	99.02	99.25	99.64
Sonar	58.43	66.21	72.13	72.23	61.88	67.87	69.43	72.11	74.43
Lungs	56.88	61.49	78.68	82.56	67.43	66.70	73.45	78.76	81.80
Iris	92.66	94.65	98.74	98.42	94.43	95.11	96.65	98.16	98.47
Delftpump AR	76.77	79.43	94.76	96.67	85.66	87.22	90.43	92.68	96.60
Ionosphere	70.45	73.45	86.70	88.22	75.43	77.43	81.20	84.76	88.44
Import	59.32	64.54	87.19	88.43	67.65	71.43	78.91	86.44	88.32
USPS	99.43	99.61	99.91	99.85	99.75	97.81	99.87	99.91	99.95
UHAD	83.42	89.41	98.67	99.13	95.12	97.11	98.47	99.06	99.23
ORL	96.12	73.87	97.21	97.58	96.43	69.43	96.89	97.58	98.01
DSA	79.43	83.47	98.57	99.12	98.24	98.12	99.17	99.2	99.30
PAMAP2	89.43	91.23	98.47	98.21	94.45	95.11	97.45	98.77	98.85
CASIAA	75.21	79.11	98.34	98.11	94.54	95.11	97.78	98.21	98.19
MNIST	74.67	81.43	94.76	94.79	90.32	90.21	93.47	94.54	95.16

Abbreviations: AUC, area under curve; DSA, Daily and Sport Activity; OCSTM, one-class support tensor machine; OCSVM, one-class support vector machine; PAMAP, physical activity monitoring data set; R1STM-BH, robust support tensor machine; STM, support tensor machine; SVDD, Support Vector Data Description.

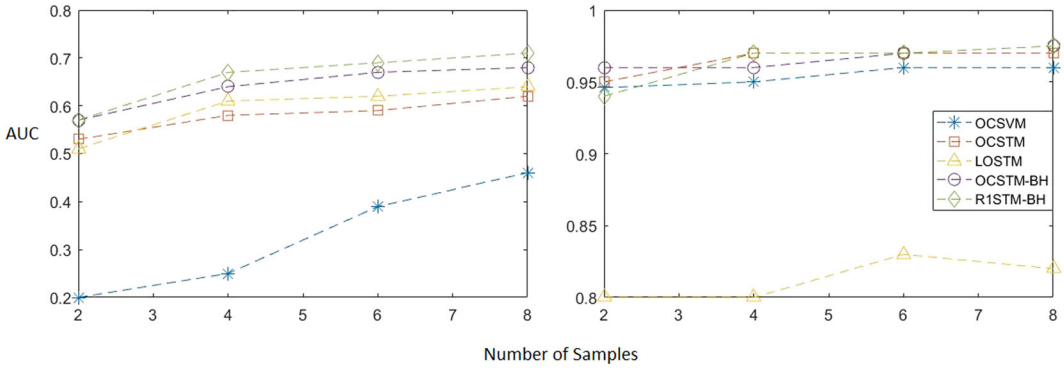

FIGURE 3 Performance comparison of proposed R1STM-BH with state-of-the-art methods on the task of face recognition (corrupted: left; original: right). AUC, area under curve; OCSTM, one-class support tensor machine; OCSVM, one-class support vector machine; R1STM-BH, robust support tensor machine; STM, support tensor machine [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 7 Space and time complexity analysis

Approach	Computational complexity	Space complexity
OCSVM[36]	$O(dN^3)$	$O(d+N^2)$
SVDD[15]	$O(dN^2)$	$O(dN^2)$
Autoencoder[15]	$O(dmN)$	$O(dq)$
ROCSVM[37]	$O(dN^3)$	$O(d+N^2)$
R1SVM[20,38]	$O(kn)$	$O(kn)$
RSVM-RHHQ[39]	$O(IN^3)$	$O(IN^3)$
OCSTM-BH (RBF)	$O(Bkn^2)$	$O(Bkn^2)$
R1STM-BH	$O(Bkn)$	$O(Bkn)$

Abbreviations: OCSTM, one-class support tensor machine; OCSVM, one-class support vector machine; R1STM-BH, robust support tensor machine; RBF, radial basis function; ROCSVM, robust one-class support vector machine; SVDD, Support Vector Data Description.

Table 8 compares the training time, test time, and the number of iterations to converge. We only compared the performance with methods based on support vector machines, such as OCSVM and OCSTM. Results show that R1STM-BH is much faster both in terms of training and testing as compared with other methods. Furthermore, R1STM-BH converges with a low number of iterations as compared with OCSVM and OCSTM.

On the basis of experiments on different benchmark data sets, we observed

- R1STM-BH (Equation 9) is monotonic, nonconvex, and bounded. We proposed half-quadratic to transform the R1STM-BH to original OCSTM.
- Equation (9) degenerates to traditional OCSTMs, thus, OCSTM (6) is a special case of R1STM-BH (10).
- Randomized linear projection with smaller kernel matrices showed significant space and time complexity for large data set, thus, we considered smaller size randomizing features.

TABLE 8 Comparative evaluation of training time (s), test time (s), and number of iterations on Breast Cancer data set

	Training time			Test time			Number of iterations		
Sample size	**OCSVM**	**OCSTM**	**R1STM-BH**	**OCSVM**	**OCSTM**	**R1STM-BH**	**OCSVM**	**OCSTM**	**R1STM-BH**
2	0.0963 ± 0.082	0.0483 ± 0.0674	0.0372 ± 0.047	0.0632 ± 3.32	0.024 ± 2.22	0.014 ± 1.35	16.43 ± 5.27	11.54 ± 3.96	9.65 ± 4.21
4	0.1759 ± 0.098	0.0968 ± 0.087	0.0502 ± 0.065	0.067 ± 4.11	0.054 ± 1.89	0.019 ± 1.43	14.53 ± 3.45	10.32 ± 4.11	8.71 ± 2.76
6	0.2154 ± 0.076	0.1043 ± 0.089	0.0614 ± 0.092	0.081 ± 3.76	0.065 ± 2.02	0.021 ± 1.39	14.59 ± 3.46	8.93 ± 3.65	6.43 ± 3.27
8	0.2334 ± 0.065	0.1232 ± 0.0932	0.0698 ± 0.099	0.087 ± 2.68	0.084 ± 2.32	0.026 ± 1.33	12.43 ± 4.79	7.43 ± 3.76	4.87 ± 3.29
10	0.2782 ± 0.104	0.1365 ± 0.108	0.0783 ± 0.078	0.096 ± 3.11	0.089 ± 2.43	0.037 ± 1.74	12.40 ± 3.78	5.87 ± 2.76	4.11 ± 2.56

Abbreviations: OCSTM, one-class support tensor machine; OCSVM, one-class support vector machine; R1STM-BH, robust support tensor machine.

5.3 | Parameter setting

To find the best parameter range of the proposed R1CSTM-BH, we used two larger data sets with different parameter values initially. R1CSTM-BH requires four parameters required to be optimal, such as trade-off parameter ν, width parameter σ, scale constant η, and k dimension of random features. Once we had a range for the best parameters, we performed 10-fold validation within the selected range for all 14 data sets. Experiments showed that R1STM-BH showed high computational complexity when k is larger; thus, smaller randomizing features are recommended.

Similarly, we have achieved the best performance of proposed anomaly detection at $\sigma = \{10, 14, 14, 27, 20, 15, 9, 21, 24, 28, 43, 31, 36\}$, $\eta = \{0.3, 0.4, 0.2, 0.\ 25, 0.25, 0.25, 0.2, 0.5, 0.3, 0.4, 1.45, 1.65, 1.25\}$, and $\nu = \{0.2, 0.25, 0.2, 0.3, 0.2, 0.25, 0.3, 0.3, 0.25, 0.2, 0.3, 0.25, 0.35\}$ for Breast Cancer, Iris, Import, Ionosphere, Lungs, Sonar, Delftpump AR, USPS, DSA, GSA, and PAMAP2 PAMAP, CASIA, ORL, and MNIST data set, respectively.

5.4 | Computational complexity

This section described the time complexity of a Randomized OCSTM with bounded hinge loss. Considering N is the number of training samples and d is the dimension of features. The computational complexity of solving the dual optimization problem imposed by one-class SVM is $O(dN^3)$, and the computational complexity of OCSVM with RBF kernel function is $O(dN^2)$. Similarly, the computational complexity of OCSTM with RBF kernel is $O(N^2 d_1^2 d_2)$,[30] whereas d_1 and d_2 denote the second-order tensor such that $d = d_1 \times d_2 \approx d$. The computational complexity of OCSTM with bounded loss function is the complexity for dual optimization with RBF kernel $O(N^2 d_1^2 d_2)$ and OCSTM with randomized linear projection is $O(kN)$, where N is the training data set size. The complexity of the auxiliary variable and margin parameter p is N, and α is the complexity of the Lagrange multiplier in every iteration. Thus, the computational complexity of OCSTM-BH is $O(H_{\mathrm{BH}}((N^2 + N + N)d_1^2 d_2)$ and $O(H_{\mathrm{BH}}((kN + N^2 + N)d_1^2 d_2)$ with RBF and randomized kernel, respectively. Neglecting the lower-order terms, we get $O(H_{\mathrm{BH}}((kN^2))$ and $O(H_{\mathrm{BH}}((kN))$ for RBF and randomized kernel, respectively, where H_{BH} is the complexity of the half-quadratic optimization.

To further observe the run time complexity, compare the performance in terms of training, testing time on Breast Cancer data set, as shown in Table 8. We can observe that computational and space complexity (both train and test) are much better as compared with the state-of-the-art methods. Furthermore, it requires much less number of iterations to converge.

6 | CONCLUSION

Traditional support tensor machines and their variants are still sensitive in the presence of outliers. To overcome the challenge mentioned above, we presented a randomized OCSTM for outliers detection in a larger data set. The proposed framework helps improve the robustness against outliers detection and results in improving the time complexity. Instead of utilizing the traditional loss function, We proposed using the bounded hinge loss function and randomized linear projection. Extensive experiments on 14 benchmark data sets showed the robustness of the proposed randomized bounded support tensor machine over support tensor machines and their variants.

Furthermore, the computational and space complexity is beautiful not only for large data sets but also for small ones that validate the proposed approach's scalability.

ORCID

Imran Razzak http://orcid.org/0000-0002-3930-6600
Nour Moustafa https://orcid.org/0000-0001-8258-9020
Shahid Mumtaz https://orcid.org/0000-0001-6364-6149
Guangdong Xu https://orcid.org/0000-0003-4493-6663

REFERENCES

1. Calvi GG, Lucic V, Mandic DP. Support tensor machine for financial forecasting. In: *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2019)*. IEEE; 2019:8152-8156.
2. Tang Y-X, Tang Y-B, Han M, Xiao J, Summers RM. Deep adversarial one-class learning for normal and abnormal chest radiograph classification. In: *Medical Imaging 2019: Computer-Aided Diagnosis*. Vol 10950. International Society for Optics and Photonics; 2019:1095018.
3. Moustafa N, Slay J. The evaluation of network anomaly detection systems: statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. *Inf Secur J: Global Perspect*. 2016;25(1-3):18-31.
4. Karami A, Guerrero-Zapata M. A fuzzy anomaly detection system based on hybrid PSO-kmeans algorithm in content-centric networks. *Neurocomputing*. 2015;149:1253-1269.
5. Moustafa N, Hu J, Slay J. A holistic review of network anomaly detection systems: a comprehensive survey. *J Network Comput Appl*. 2019;128:33-55.
6. Ma Z, Yang LT, Zhang Q. Support multi-mode tensor machine for multiple classification on industrial big data. *IEEE Trans Ind Inf*. 2020;17(5):3382-3390.
7. van Wyk F, Wang Y, Khojandi A, Masoud N. Real-time sensor anomaly detection and identification in automated vehicles. IEEE Trans Intell Transp Syst. 2019.
8. Rettig O, Müller S, Strand M, Katic D. Which deep artificial neural network architecture to use for anomaly detection in mobile robots kinematic data? In: *Machine Learning for Cyber Physical Systems*. Springer; 2019: 58-65.
9. Twomey N, Chen H, Diethe T, Flach P. An application of hierarchical Gaussian processes to the detection of anomalies in star light curves. Neurocomputing. Vol. 342, 2019.
10. Lin H, Liu G, Wu J, Zuo Y, Wan X, Li H. Fraud detection in dynamic interaction network. *IEEE Trans Knowl Data Eng*. 2019;32(10):1936-1950.
11. Zhang L, Lin J, Karim R. An angle-based subspace anomaly detection approach to high-dimensional data: with an application to industrial fault detection. *Reliab Eng Syst Saf*. 2015;142:482-497.
12. Dong L, Shulin L, Zhang H. A method of anomaly detection and fault diagnosis with online adaptive learning under small training samples. *Pattern Recognit.* 2017;64:374-385.
13. He L, Kong X, Yu PS, Yang X, Ragin AB, Hao Z. Dusk: a dual structure-preserving kernel for supervised tensor learning with applications to neuroimages. In: *Proceedings of the 2014 SIAM International Conference on Data Mining*. SIAM; 2014:127-135.
14. Razzak I, Blumenstein M, Xu G. Multi-class support matrix machines by maximizing the inter-class margin for single trial EEG classification. *IEEE Trans Neural Syst Rehabil Eng*, 27(6):1117-1127.
15. Erfani SM, Baktashmotlagh M, Rajasegarad S, et al. R1STM: one-class support tensor machine with randomised kernel. In: *Proceedings of the 2016 SIAM International Conference on Data Mining*. SIAM; 2016: 198-206.
16. Anaissi A, Lee Y, Naji M. Regularized tensor learning with adaptive one-class support vector machines. In: *International Conference on Neural Information Processing*. Springer; 2018:612-624.
17. Razzak I, Khan TM. One-class support tensor machines with bounded hinge loss function for anomaly detection. In: *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE; 2020:1-8.
18. Razzak I, Saris RA, Blumenstein M, Xu G. Integrating joint feature selection into subspace learning: a formulation of 2DPCA for outliers robust feature selection. *Neural Networks.* 2020;121:441-451.

19. Razzak I, Zafar K, Imran M, Xu G. Randomized nonlinear one-class support vector machines with bounded loss function to detect of outliers for large scale IoT data. *Future Gener Comput Syst*. 2020;112:715-723.
20. Erfani S, Baktashmotlagh M, Rajasegarar S, Karunasekera S, Leckie C. R1SVM: a randomised nonlinear approach to large-scale anomaly detection. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1; 2015.
21. Signoretto M, De Lathauwer L, Suykens JA. A kernel-based framework to tensorial data analysis. *Neural networks.* 2011;24(8):861-874.
22. Signoretto M, Olivetti E, De Lathauwer L, Suykens JA. Classification of multichannel signals with cumulant-based kernels. *IEEE Trans Signal Process*. 2012;60(5):2304-2314.
23. Zhao Q, Zhou G, Adalı T, Zhang L, Cichocki A. Kernel-based tensor partial least squares for reconstruction of limb movements. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE; 2013:3577-3581.
24. Hamid R, Xiao Y, Gittens A, DeCoste D. Compact random feature maps. In: *International Conference on Machine Learning*; 2014:19-27.
25. Gorman RP, Sejnowski TJ. Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks.* 1988;1(1):75-89.
26. Street WN, Wolberg WH, Mangasarian OL. Nuclear feature extraction for breast tumor diagnosis. In: *Biomedical Image Processing and Biomedical Visualization*. Vol 1905. International Society for Optics and Photonics; 1993:861-871.
27. Xu L, Krzyzak A, Suen CY. Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE Trans Syst Man Cybern*. 1992;22(3):418-435.
28. Barshan B, Yüksek MC. Recognizing daily and sports activities in two open source machine learning environments using body-worn sensor units. *Comput J*. 2014;57(11):1649-1667.
29. Cai D, He X, Han J. *Learning with Tensor Representation*. Technical report, 2006.
30. Chen Y, Wang K, Zhong P. One-class support tensor machine. *Knowl-Based Syst*. 2016;96:14-28.
31. Wang L, Tan T, Ning H, Hu W. Silhouette analysis-based gait recognition for human identification. *IEEE Trans Pattern Anal Mach Intell*. 2003;25(12):1505-1518.
32. LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. *Proc IEEE*. 1998;86(11):2278-2324.
33. Chang C-C, Lin C-J. LIBSVM: a library for support vector machines. *ACM Trans Intell Syst Technol (TIST)*. 2011;2(3):27.
34. Chen Y, Lu L, Zhong P. One-class support higher order tensor machine classifier. *Appl Intell.* 2017;47(4): 1022-1030.
35. Ruff L, Vandermeulen R, Görnitz N, et al. Deep one-class classification. In: *International Conference on Machine Learning*. PMLR; 2018:4390-4399.
36. Li K-L, Huang H-K, Tian S-F, Xu W. 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. IEEE; 2003:3077-3081.
37. Xiao Y, Wang H, Xu W. Ramp loss based robust one-class SVM. *Pattern Recognit Lett*. 2017;85:15-20.
38. Weerasinghe PS, Alpcan T, Erfani SM, Leckie C. Unsupervised adversarial anomaly detection using one-class support vector machines. 2018.
39. Tian Y, Mirzabagheri M, Bamakan SMH, Wang H, Qu Q. Ramp loss one-class support vector machine; a robust and effective approach to anomaly detection problems. *Neurocomputing*. 2018;310:223-235.