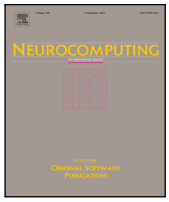
Skip-patching spatial-temporal discrepancy-based anomaly detection on multivariate time series

Neurocomputing 609 (2024) 128428

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom



Skip-patching spatial–temporal discrepancy-based anomaly detection on multivariate time series

Yinsong Xu [a,b,c], Yulong Ding [a,c], Jie Jiang [d], Runmin Cong [e,f], Xuefeng Zhang [i], Shiqi Wang [b], Sam Kwong [g], Shuang-Hua Yang [c,h,*]

[a] Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, Guangdong, China
[b] Department of Computer Science, City University of Hong Kong, Hong Kong, China
[c] Shenzhen Key Laboratory of Safety and Security for Next Generation of Industrial Internet, Southern University of Science and Technology, Shenzhen, Guangdong, China
[d] College of Artificial Intelligence, China University of Petroleum (Beijing), Beijing, China
[e] School of Control Science and Engineering, Shandong University, Jinan 250061, China
[f] Key Laboratory of Machine Intelligence and System Control, Ministry of Education, Jinan, China
[g] Department of Computing and Decision Sciences, Lingnan University, Hong Kong, China
[h] Department of Computer Science, University of Reading, Reading RG6 6AH, UK
[i] College of Sciences, Northeastern University, Shenyang 110819, China

ARTICLE INFO

Communicated by N. Zeng

Keywords:
Anomaly detection
Industrial Internet of Things
Self-supervised learning
Multivariate time series

ABSTRACT

Anomaly detection in the Industrial Internet of Things (IIoT) is a challenging task that relies heavily on the efficient learning of multivariate time series representations. We introduce Skip-patching and Spatial–Temporal discrepancy mechanisms to improve the efficiency of detecting anomalies. Traditional feature extraction is hindered by redundant information in limited datasets. The situation is that feature generation from stable operational processes results in low-quality representations. To address this challenge, we propose the Skip-Patching mechanism. This approach involves selectively extracting features from partial data patches, prompting the model to learn more meaningful knowledge through self-supervised learning. It also effectively doubles the training sample size by creating independent sub-groups of patches. Despite the complex spatial and temporal relationships in IIoT systems, existing methods mainly extracted features from a single domain, either temporal or spatial (sensor-wise), or simply cascaded two features, i.e., one after one, which limited anomaly detection capabilities. To address this, we introduce the Spatial–Temporal Association Discrepancy component, which leverages discrepancies between spatial and temporal features to enhance latent representation learning. Our Skip-Patching Spatial–Temporal Anomaly Detection (SSAD) framework combines these two components to provide a more diverse and comprehensive learning process. Tested across four multivariate time series anomaly detection benchmarks, SSAD demonstrates superior performance, confirming the efficacy of combining Skip-patching and Spatial–Temporal features to enhance anomaly detection in IIoT systems.

1. Introduction

The rapid expansion of the Industrial Internet of Things (IIoT), driven by more affordable connectivity and advanced automation, emphasizes the importance of efficient security measures. The importance of security and safety in IIoT is highlighted by the severe consequences of breaches, which can lead to catastrophic industrial damage and pose significant risks to human safety. To safeguard the wealth and health of industrial processes within the Industrial Internet of Things (IIoT), the detection of anomalies in extensive multivariate time series data is essential. This data, generated by sensors and actuators involved in these processes, requires sophisticated monitoring to prevent losses. Consequently, data-driven approaches, especially deep learning techniques, have been increasingly adopted for the efficient and stable detection of anomalies in such multivariate time series data.

Deep learning approaches are typically categorized into supervised and unsupervised learning. Supervised learning is challenging for IIoT anomaly detection as labeling vast datasets is impractical and requires identifying unseen anomalies. In contrast, unsupervised learning, which

* Corresponding author at: Department of Computer Science, University of Reading, Reading RG6 6AH, UK.
E-mail addresses: yinsongxu2-c@my.cityu.edu.hk (Y. Xu), dingyl@sustech.edu.cn (Y. Ding), jiangjie@cup.edu.cn (J. Jiang), rmcong@sdu.edu.cn (R. Cong), zhangxuefeng@mail.neu.edu.cn (X. Zhang), shiqwang@cityu.edu.hk (S. Wang), samkwong@ln.edu.hk (S. Kwong), shuang-hua.yang@reading.ac.uk (S.-H. Yang).

https://doi.org/10.1016/j.neucom.2024.128428
Received 16 May 2024; Received in revised form 25 July 2024; Accepted 16 August 2024
Available online 20 August 2024

does not require labeled data, is preferred. Unsupervised anomaly detection methods often focus on tasks such as reconstruction or predictions using normal samples, which are much easier to obtain than abnormal data in the regular operation of IIoT systems. These methods assume that anomalies will significantly deviate from the model's reconstructions [1–3] or predictions [4,5], trained with normal data.

Representation learning, which focuses on automatically discovering the representations needed for feature detection from raw data in machine learning, plays a pivotal role in data-driven anomaly detection within the IIoT. Applying representation learning to multivariate time series data in IIoT contexts introduces distinct challenges: **(i) Quality of Latent Feature**: A detailed and comprehensive representation is vital for accurate reconstruction. The absence of explicit labels in unsupervised learning models compounds the difficulty. In the IIoT context, the demand for stable operation results in minimal behavioral variations, leading to repetitive system patterns. This stability, while crucial, inadvertently generates similar data samples, significantly complicating critical feature extraction in unsupervised learning models. **(ii) Complementarity of Features Across Different Perspectives**: Anomalies in multivariate time series can be delineated through both temporal and spatial dependencies [6]. The connections in the time domains lead to the temporal feature, and the cause–effect relation between sensors in IIoT presents the spatial feature. Certain features possess a superior ability to detect specific types of anomalies. However, for anomalies that exhibit both characteristics, it is often difficult for a single approach to identify them effectively. One can significantly enhance the anomaly detection rate by harmonizing the insights from diverse features.

To extract high-quality features from data characterized by simple tasks and redundant information, researchers have developed various strategies to enhance a model's learning capabilities. Self-supervised learning emerges as a key technique for improving representation in deep learning frameworks. Among its strategies, masking stands out by concealing portions of the data, thereby encouraging the model to uncover deeper, more meaningful features to compensate for the missing parts. Traditional masking techniques, however, typically operate at the point level, which might not be fully effective in time series data characterized by extensive redundancy; adjacent time points can often predict missing values, diminishing the learning challenge. Our methodology introduces patch-level skipping, a more sophisticated approach that hides critical segments of data, challenging the model to engage in more complex reconstruction tasks. This technique utilizes the power of self-supervised learning, compelling the model to identify and focus on the extraction of more significant features. This strategy has shown efficacy in fields such as natural language processing (NLP) [7] and computer vision (CV) [8]. Moreover, our technique effectively doubles the size of the training dataset, leading to enhanced generalization, heightened robustness, and improved contextual understanding. The integration of skip-patching and data augmentation techniques collectively results in a marked improvement in feature representation.

There are two main characteristics for anomalies [6]: temporal, which introduces irregular patterns in the time series, and spatial, which affects the coordination among sensors. Certain anomalies exhibit both characteristics, requiring a comprehensive approach that draws on both spatial and temporal insights for detection. Current efforts often concentrate on a single dimension of analysis. The approach is limited in scope, effectively addressing only one type of anomaly and falling short in scenarios where both temporal and spatial anomalies coexist [4,9,10]. While some efforts tried to integrate both spatial and temporal features, their methods simply cascade two features that can hinder the model's ability to balance between two types of information, thus limiting representational effectiveness [11]. In response, we propose a parallel architecture to bridge a strong connection between two distinct feature-extracting branches, each dedicated to mining insights from the spatial and temporal domains. This arrangement leverages the discrepancies between the two features' entropy to enhance the feature extraction quality. This approach provides a more balanced and comprehensive framework for anomaly detection in IIoT.

In this paper, we propose a novel Skip-Patching Spatial–Temporal Anomaly Detection (SSAD) method with greatly enhanced representation learning ability. Our method consists of two key components: Skip-Patching and Spatial–Temporal Association Discrepancy. The Skip-Patching technique not only enriches feature quality but also increases the volume of training samples. Concurrently, the Discrepancy module optimally aligns spatial and temporal features, facilitating a comprehensive representation of learning to capture a broader spectrum of anomalies.

We summarize the contributions as follows:

1. We propose a Skip-Patching mechanism for multivariate time series-related tasks for high-quality representation and more training samples.
2. We propose a Spatial–Temporal discrepancy mechanism that leverages both spatial and temporal perspectives of the data to provide complementary views.
3. We implement a novel model architecture and performed experiments on four multivariate time series datasets, demonstrating that our approach, SSAD, achieves superior anomaly detection performance compared to seven state-of-the-art methods.

2. Related work

Anomaly in the multivariate time series could lead to a considerable loss of human health and property, so researchers have designed many detection methods at the early stage of an anomaly event to limit the influence of anomalies. The common idea of anomaly is they have irregular patterns from standard series. The classic method turns to find outliers during the testing phase, such as based on the distance [12] or density [13]. Along with the system becoming more complex, a simple strategy makes it hard to differentiate the anomaly samples. The machine learning (ML) method has the advantage of finding the inconspicuous relations in the data [14,15]. Traditional ML strategies have a similar way of finding outliers [16]. Technics like Random Forest [17], K-nearest Neighbor [18], and Support Vector Machines [19] have been demonstrated to achieve similar actions with better performance.

Feature quality is a significant element for a model to distinguish between normal and anomalous samples. TimesNet [20] designs a more general way by masking random time points. In the computer vision and the nature language process fields, BERT presents a masked language model to predict the masked words to enhance the representation of the connection within the context [7,8] uses a masked-autoencoder which reconstructs a figure by using limited fragments to increase the semantic knowledge for the representation learning. The idea is also used in multivariate time series-related tasks. According to PatchTST [21], a subseries of a data series achieves a better effective representation learning than one at the point level. DUMA [22] uses masks on the data patches to get a better detection performance but in a random order.

Recent works use deep learning methods to achieve further complex feature extraction [23–26]. Similar to the classic strategies, the Gaussian Mixture Model in DAGMM [27], Long Short Term Memory (LSTM) in USMD [28], and Variational Auto-Encoder (VAE) in VLSTM [9] all use an Auto-Encoder (AE) structure to compress the sample to hidden features which present difference between normal and anomaly samples. To capture the representation of the data more effectively, MemAE [3] and CMAE [29] use a memory component to capture different hidden features in vectors and only use information from them to reconstruct the data. DAEMON [2] and MAD-GAN [30] generate similar data as the normal to train the model in an adversarial way to reserve only the core feature. Anomaly samples are reconstructed with the normal features with higher difficulty and produce a distinct

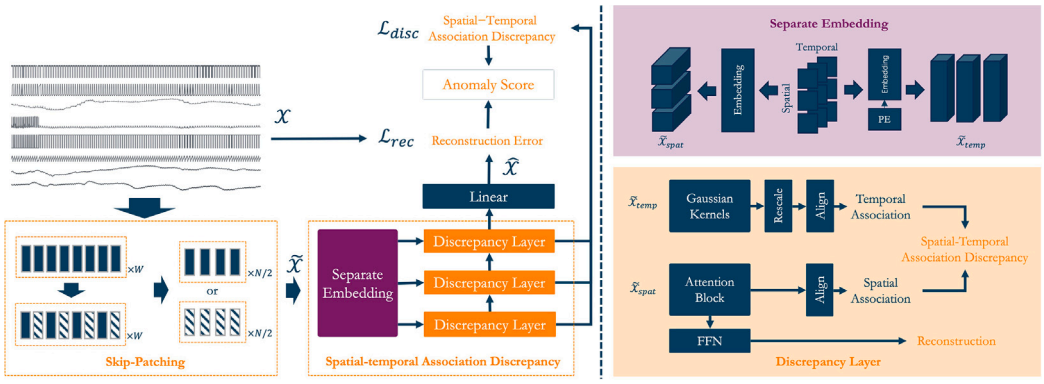


Fig. 1. Skip-Patching Spatial–Temporal Anomaly Detection (SSAD) architecture. The Skip-Patching module splits the data into patches and archives half patches as input. Spatial–temporal association discrepancy contains separate embedding and discrepancy layers. The details of the components are shown on the right. Separate Embedding embeds the data from different domains and keeps the other unchanged. The discrepancy layer uses the tendentious outputs from the Separate Embedding to extract two different features to calculate the spatial–temporal association discrepancy.

anomaly score. To get better representation learning, EncDec-AD [10] and TSMAE [31] use LSTM to aim at the temporal information from the data. On the other hand, besides knowledge of the time domain, there is also a connection between channels. TimesNet [20] finds the connection between periods within the temporal dimension to support the model to extract the hidden information. DLinear [32] uses a linear method to find the connection between characters of data frequency and use for long-term forecasting. GDN [4] and VGCRN [5] focus on finding the spatial representation using Graph Neural Network-based frameworks. To increase the efficiency of digging comprehensive critical information, the Anomaly Transformer [1] produces two different temporal hidden features and uses the discrepancy to support the model to study the data from two temporal views with the Transformer. InterFusion [6] designed a cascaded structure to unite spatial and temporal features simultaneously, letting the model also take care of the anomaly in the spatial domain.

3. Model structure

3.1. Overview

The SSAD architecture, depicted in Fig. 1, processes a time window of multivariate time series data, $X : (x_i, \ldots, x_W)$, where $X \in \mathbb{R}^{W \times E}$, W denotes the window size and each x_i at timestamp i comprises the data points of E sensors. The model employs a Skip-Patching mechanism to divide the data into patches, alternately picking half of the patches to form the input samples. These patches are then concatenated and embedded along temporal and spatial domains, subsequently branching into respective pathways for further analysis. The model leverages the discrepancy between temporal and spatial features to enhance their representation learning mutually. Notably, reconstruction is performed using the spatial feature solely, resulting in an estimated output, $\hat{X} : (\hat{x}_i, \ldots, \hat{x}_W)$. This dual focus on spatial and temporal representations allows the model to discern valuable information effectively throughout the learning phase. A key aspect of this process is calculating an association discrepancy based on these features, which amplifies the differentiation between normal and anomalous samples when combined with reconstruction error. After training, the threshold for the anomaly score is determined with the help of assigned training and threshold dataloaders. This allows for identifying anomalies in the testing phase by comparing the score with the threshold.

3.2. Skip-Patching

Utilizing a time window as the input to assess anomaly status in a system is a widely accepted approach. Traditionally, research has leveraged data reconstruction to enrich models with comprehensive data insights, enhancing representation learning quality. However, the simplicity of using complete information limits the model's ability to uncover the data's underlying relational knowledge. Inspired by BERT [7] and MAE [8], we develop a learning strategy that conceals half of the timestamps, compelling the model to reconstruct the entire dataset. This self-supervised approach enables the model to grasp essential reconstruction information. Nonetheless, the temporal domain's redundancy often allows for easy recovery of time points by their immediate neighbors, thereby impeding the model's acquisition of crucial latent knowledge. To address this challenge, as Fig. 2 shows, we introduce a Skip-Patching mechanism. This technique divides the input data into patches. Subsequently, either odd or even patches are selected and concatenated to form the input for subsequent analysis. This process removes large segments of the series and prevents straightforward reconstruction based on neighboring values, requiring the model to explore deeper data connections to enhance feature quality. Moreover, creating independent sub-groups of patches doubles the available sample size, alleviating the constraints of limited datasets prevalent in anomaly detection research.

Initially, the data X is segmented into patches of a fixed size p. If W represents the total length of the time series, then the total number of patches P is calculated as $P = W/p$. The resulting matrix, X_s, has dimensions $P \times p \times E$, where E denotes the spatial dimension. The patches are divided into two sets based on their sequence (even or odd). After randomly selection, one set is reserved for the further processing. The selected patches are then concatenated along the temporal dimension to form $\widetilde{X} \in \mathbb{R}^{W/2 \times E}$, which is prepared for feature extraction. The model processes the reshaped data through the feature extraction module, which is then used for data reconstruction.

By retaining only half of the data patches, the Skip-Patching strategy brings multiple advantages to the model's learning process. Firstly, this approach requires the model to infer missing time points, enhancing its predictive capabilities. The preserved patches, embodying both localized details and the overarching structure of the time window, compel the model to leverage both local and global semantic information during reconstruction. This enriches the model's comprehension and utilization of the data. Secondly, this selective retention amplifies the impact of anomalous data segments. In scenarios where predominantly anomalous points are preserved, the model's output anomaly score intensifies due to a reduced presence of normal timestamps. Conversely, if anomalous points are predominantly excluded, the model attempts to reconstruct a normal series from an originally anomalous input, thereby increasing the reconstruction error. This duality enhances the model's sensitivity to anomalies. Thirdly, processing a truncated series reduces computational demands, optimizing the learning framework's efficiency. Additionally, employing a randomized selection strategy for

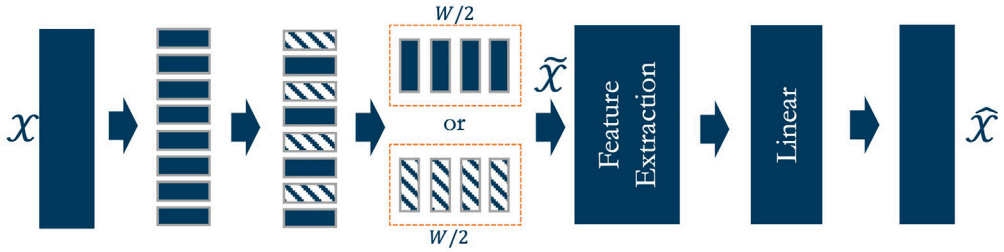


Fig. 2. The structure of the Skip-Patching module.

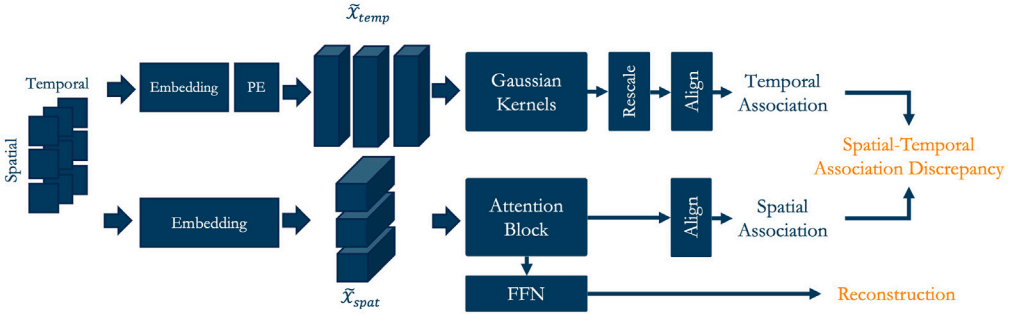


Fig. 3. The structure of the Spatial–Temporal Association Discrepancy module.

the patch groups ensures comprehensive learning from the dataset, preventing model bias towards any specific data subset. Alternatively, two independent patch groups for training effectively double the training sample size, trading off increased training duration for improved model robustness.

3.3. Spatial–temporal association discrepancy

Multivariate time series data encapsulate two pivotal forms of knowledge: temporal and spatial. Temporal information extraction, a conventional practice in time series analysis, emphasizes the sequential interconnectivity of data points. Spatial information, on the other hand, elucidates the interrelations among multiple sensors or channels at a given time, particularly relevant in IIoT datasets where device interactions are common. As Fig. 3 shows, we employ two specialized embeddings to capture these dimensions: temporal and spatial. The temporal embedding maintains the integrity of the time sequence, mapping each point to a higher-dimensional feature space. Spatial embedding follows a parallel strategy, preserving the spatial dimension's original scale but projecting it into an enriched feature space. These embeddings are strategically designed to generate informative matrices optimized for the distinct analytical requirements of subsequent processing stages, ensuring a comprehensive understanding of spatial and temporal dynamics in the data. Our approach leverages distinct features to construct a time-point-wise feature matrix, utilizing the discrepancies between these features to bolster anomaly detection. Initially, data transforms two separate matrices tailored for specific analytical purposes:

$$\begin{aligned} \tilde{X}_{temp} &= TempEmbed(\tilde{X}) + PE(\tilde{X}) \\ \tilde{X}_{spat} &= SpatEmbed(\tilde{X}^T) \end{aligned} \tag{1}$$

Here, $PE()$ denotes position embedding, integrating spatial dimensions into a hidden feature space while incorporating a classic position embedding through $TempEmbed()$ and maintaining the temporal domain unchanged. Conversely, $SpatEmbed()$ expands the temporal dimension of raw data into a latent feature size and keeps the spatial domain unchanged, focusing on the spatial interrelations among data points.

Upon embedding, the dataset undergoes processing through a three-layer structure, each layer l comprising two branches dedicated to temporal and spatial associations. The temporal association branch utilizes $\tilde{X}_{temp}$ to extract temporal features by assessing the relative distances between temporal vectors. Inspired by [1], we use a learnable scale parameter, $\sigma \in \mathbb{R}^{N\times 1}$,

$$\sigma = \tilde{X}_{temp} W_{\sigma}^{l} \tag{2}$$

facilitating the Gaussian kernels' ability to encapsulate the overarching characteristics of channel activities at specific time points:

$$T_f^l = Rescale\left(\left[\frac{1}{\sqrt{2\pi}\sigma_i} exp\left(-\frac{|j-i|^2}{2\sigma_i^2}\right)\right]_{i,j\in\{1,\ldots,N\}}\right) \tag{3}$$

where $Rrescale()$ converts the weights into a normalized distribution across the time domain. Here, the temporal feature matrix is refined by computing the Gaussian kernel weights between the ith metric and all other jth time points, subsequently rescaled to produce a discrete distribution $T_f^l \in \mathbb{R}^{W\times W}$. The result then passes through an $Align()$ function to realign the temporal length to its original dimension. The final temporal association matrix, T^l, is derived after applying a $Softmax()$ layer, setting the stage for comparative analysis with its spatial counterpart:

$$T^l = Softmax(Align(T_f^l)) \tag{4}$$

For spatial feature extraction, we adopt a Transformer architecture utilizing multi-head attention mechanisms to enhance spatial understanding. Within each attention head $h = 1, \ldots, H$ the input undergoes transformation into query Q, key K, and value V matrices:

$$Q, K, V = \tilde{X}_{spat}^{l-1} W_h^Q, \tilde{X}_{spat}^{l-1} W_h^K, \tilde{X}_{spat}^{l-1} W_h^V \tag{5}$$

The computation of spatial attention maps $S_f^l \in \mathbb{R}^{E\times E}$ is achieved by applying a softmax function to the product of Q and K, normalized by the square root of the model's dimension:

$$S_f^l = Softmax\left(\frac{Q_h K_h^T}{\sqrt{d_{model}}}\right) \tag{6}$$

Analogous to the temporal association process, spatial features are projected through an $Align()$ function followed by a $Softmax()$ to derive the spatial association S^l:

$$S^l = Softmax(Align(S_f^l)) \tag{7}$$

Subsequently, the attention matrix, in conjunction with the value matrix V, constructs the layer-specific features. These features are further refined through layer normalization $LN()$, a feedforward network $FFN()$, and a highway network $HW()$, a bypassing for deeper learning, facilitating the reconstruction of this step's input and preparing it for subsequent layers:

$$\tilde{X}^l_{spat} = LN(FFN(LN(S^l_f V^l_h + \tilde{X}^{l-1}_{spat})) + HW(\tilde{X}_{spat})) \tag{8}$$

Through iterative processing across multiple layers, the model delves deeper into spatial detail extraction. At the final step, $\tilde{X}^l_{spat}$ undergoes a linear layer to reconstruct the data, $\hat{X}$, matching the original input's dimensions. This rigorous spatial feature processing significantly enhances the model's capacity for nuanced representation learning.

Upon feature extraction by both branches, the matrices embody distinct representations of the original data from spatial and temporal perspectives. To quantify the disparity between these representations, we employ the Kullback–Leibler (KL) divergence, facilitating a thorough comparison of the spatial and temporal association distributions:

$$AssDis(S,T) = \left[\frac{1}{L}\sum_{l=1}^{L}\left(KL(S^l_{i,:}, T^l_{i,:}) + KL(T^l_{i,:}, S^l_{i,:})\right)\right]_{i=1,\ldots,W} \tag{9}$$

where $S^l, T^l \in \mathbb{R}^{W\times W}$ represent the spatial and temporal association matrices, respectively. Given the inherent asymmetry of the KL divergence — yielding different values when directionally swapped — we opt for a symmetrical approach to KL divergence computation. The achieved association distributions ensure a balanced and comprehensive evaluation of the knowledge encapsulated within the two latent feature matrices, enhancing our understanding of their interrelation and the overall data representation, which provides more information for distinguishing the normal and anomaly samples.

3.4. Loss function and optimization

The model's learning optimization integrates reconstruction error with Association Discrepancy, partitioning the loss function into two primary components. The initial component, the reconstruction loss $\mathcal{L}_{rec}$, is quantified using the Mean Square Error (MSE) between the original input X and its reconstructed counterpart $\hat{X}$.

$$\mathcal{L}_{rec} = \|X - \hat{X}\|^2 \tag{10}$$

The discrepancy loss $\mathcal{L}_{disc}$ accounts for the second component, encapsulating the divergence between spatial and temporal feature representations. This is modulated by a parameter λ facilitating a minimax optimization strategy:

$$\mathcal{L}_{disc} = -\lambda \times AssDis(S,T) \tag{11}$$

Finally, the total loss combines two loss results:

$$\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{disc} \tag{12}$$

Employing a minimax strategy [1], this approach exploits the symmetrized KL divergence discrepancy to refine representation learning. A dynamic learning mechanism further amplifies the benefits of dual-view representation learning, enforcing reciprocal learning between the two feature sets. This is operationalized through separate backpropagation (BP) phases for each feature set, with the alternate feature being temporarily detached:

$$\begin{aligned} MinimizePhase &: \mathcal{L}(\hat{X}, T, S_{detach}, -\lambda, X) \\ MaximizePhase &: \mathcal{L}(\hat{X}, T_{detach}, S, \lambda, X) \end{aligned} \tag{13}$$

This strategy not only optimizes the learning process but also distinctly accentuates discrepancies between normal and anomalous samples, enhancing the model's discriminative capacity.

3.5. Anomaly criterion

The Anomaly Score serves as the criterion for distinguishing between normal and anomalous samples, synthesizing the association discrepancy and the reconstruction error. This score accentuates differences between normal and anomalous inputs by integrating the time point-wise discrepancy with the reconstruction error through element-wise multiplication:

$$\begin{aligned} AnomlayScore(X) = Softmax(-AssDis(S,T)) \\ \odot [\|X - \hat{X}\|^2]_{i,\ldots,W} \end{aligned} \tag{14}$$

This formulation reflects the nuanced interplay between temporal and spatial features within the model. Based on the information from both domains, anomalies that contain characteristics from both types can be identified. In the presence of anomalies, abnormal behavior may be localized to a few devices, causing the spatial feature to exhibit localized discrepancies as well. This condition results in a reduced KL divergence, thereby elevating the Anomaly Score. The product of these components magnifies the effect of each factor, ensuring that any deviation from the norm significantly influences the overall score. Since the model masters the reconstruction with higher quality features, the anomaly samples have a higher chance of failure of any component to align with the expected pattern during normal operations, which precipitates a magnification in the Anomaly Score, facilitating the identification of anomalies.

SSAD is designed to mirror the decision-making process of domain experts in diagnosing anomalies in operational systems. Typically, experts recognize patterns in normal operational data and utilize the relational dynamics between sensors to identify anomalies, despite the inherent challenges in monitoring complex sensor data. In similar fashion, the SSAD framework uses dual feature extraction mechanisms to independently analyze temporal and spatial dimensions of the data. This allows the framework to assess interactions within each dimension as well as across them, much like an expert would check whether trends in one sensor follow those in another. Discrepancies between the spatial and temporal features serve as a metric for evaluating the congruence of data patterns against expected norms. This method of discrepancy measurement effectively pinpoints anomalies by highlighting atypical relationships that deviate from established patterns. Additionally, the Skip-patching mechanism in SSAD plays a crucial role by focusing the model's attention on essential information, emulating an expert's ability to disregard irrelevant data noise and focus on significant signals. This mechanism not only enhances the model's robustness to anomalies but also ensures that it prioritizes critical features indicative of normal operation or potential deviations. The anomaly score combines both feature and data discrepancies, mimicking expert diagnostic processes.

4. Experiments

This section presents a detailed evaluation of our proposed SSAD model, outlining the methodology employed to assess its performance in anomaly detection within multivariate time series data. We introduce the datasets utilized for testing, selected for their relevance to real-world applications, and the challenges they present in anomaly detection. The metrics chosen to measure the model's performance are then described, emphasizing their importance in accurately quantifying detection capabilities. Following this, we detail the implementation specifics of our experiments, ensuring reproducibility and transparency. After that, a comparison between the performance of all the methods is presented. Additionally, an ablation study is performed to elucidate the contribution of each model component to overall performance. Finally, we explore the model's sensitivity to various hyperparameters, identifying optimal configurations that enhance its anomaly detection accuracy. Together, these sections offer a holistic view of the SSAD model's evaluation.

Table 1
Statistical summary of SWaT, WADI, SMAP, and MSL datasets.

Datasets	SWaT	WADI	SMAP	MSL
# Features	51	127	25	55
Training size	495 000	1 048 571	135 183	58 317
Testing size	449 919	172 801	427 617	73 729
Anomaly rate (%)	12.14	5.99	12.80	10.50

4.1. Datasets

In our study, we employ four widely used datasets to evaluate the performance of the proposed SSAD model: MSL (Mars Science Laboratory rover) and SMAP (Soil Moisture Active Passive Satellite): Sourced from NASA's spacecraft monitoring systems, these datasets are compiled from telemetry data recorded in Incident Surprise Anomaly (ISA) reports, with MSL featuring 55 variables and SMAP comprising 25 variables [33]. SWaT (Security Water Treatment): This dataset is derived from a fully operational water treatment testbed, documenting readings from 51 sensors and actuators [34]. WADI (Water Distribution Testbed): As a dataset representing a water distribution system, WADI includes data from 127 industrial devices monitored by a Supervisory Control And Data Acquisition (SCADA) system, simulating a real-world industrial environment [35]. Table 1 summarizes these datasets, highlighting their diverse applications and complexity. Before analysis, all data points are normalized using a standard scaler, optimizing them for deep learning model training and ensuring consistency in model evaluation.

4.2. Evaluation metrics

To assess the performance of our proposed model in comparison with baseline models, we employ three standard metrics: Precision, Recall, and the F1-score. These metrics are calculated as follows: $Precision = TP/(TP + FP)$, $Recall = TP/(TP + FN)$, and $F1 = 2TP/(2TP + FP + FN)$, where TP, FP, and FN denote the counts of true positives, false positives, and false negatives, respectively. $Precision$ is the ratio of correctly predicted positive observations to the total predicted positives. $Recall$ is the ratio of correctly predicted positive observations to all observations in the actual class. $F1 - score$ is the weighted average of Precision and Recall. Consistent with established practices in the field, we implement an adjustment strategy for anomaly identification: a sample is considered correctly identified if at least one of the points within an anomalous segment is detected [1, 36]. This approach acknowledges the real-world application scenario where any segment encompassing an anomalous event indicates an anomaly, thereby adopting a flexible detection criterion that enhances the practical relevance of our model's performance evaluation.

4.3. Implementation details

Our proposed model was developed using PyTorch, version 2.0.1 [37], and leveraged CUDA 11.7 for GPU acceleration, ensuring efficient computation. The experimental environment was configured on an Ubuntu 22.04.2 LTS system equipped with an Intel i7-12700 CPU and an NVIDIA RTX 3090Ti graphics card, providing the necessary computational resources for model training and evaluation. For optimization, the Adam optimizer was selected for its effectiveness in handling sparse gradients on noisy problems, with a learning rate set at 0.0001. The model architecture includes three layers dedicated to extracting hidden features, designed to capture the complex patterns inherent in multivariate time series data. For the architectural parameters, hidden layer channels are set to 512 to allow for a comprehensive feature representation. The number of attention heads is fixed at 8, facilitating the model's ability to attend to different parts of the input sequence for better context understanding. The batch size is configured at 32, balancing the trade-off between training speed and memory usage. The window size for input data is chosen as 256 with a patch size of 16, optimizing the model's ability to process and learn from temporal segments effectively. The anomaly detection threshold is established by designating a specific proportion of the validation datasets as anomalies. This predefined proportion is adjusted for each dataset: 0.1% for SWaT and WADI datasets, 1% for MSL and SMAP datasets. All datasets are scaled by using standardization. Additionally, the hyperparameter λ, which balances the reconstruction loss and association discrepancy in the loss function, was meticulously set to 3.

4.4. Baselines

To evaluate the performance of SSAD, we employ the following 7 baselines: **PCA** uses Principal Component Analysis to transform the representation from high to low dimensions [38]. The reconstruction of the transformation is used as the anomaly score. **AE** uses the classic Autoencoder structure [39]. An encoder digs the hidden variables from the input, and a decoder reconstructs the data based on the hidden information. **LSTM-VAE** concatenates LSTM with a fully connected network in a variational autoencoder [40]. The model leverages LSTM as the major component to explore the temporal knowledge from the data. **MAD-GAN** also uses LSTM but with generative adversarial training [30]. A generator works with a discriminator to squeeze the hidden feature for more details. **InterFusion** introduces a hierarchical VAE to utilize the spatial and temporal features for finding inter-spatial–temporal anomalies [6]. **GDN** designed a Graph Neural network-based model to focus on learning the spatial relationships from normal data [4]. **PatchTST** utilizes an individual channel strategy to process each channel in the multivariate time series independently [21]. **iTransformer** leverages spatial relationships with spatial features to improve detection accuracy [41]. **DLinear** uses a lightweight deep linear neural network-based framework to find connections within the frequency of temporal dimension [32]. **TimesNet** focuses on the intraperiod and interperiod relations [20]. **Anomaly Transformer** provides a new anomaly score formula that appends the KL divergence discrepancy between features to enlarge the difference between normal and anomaly [1]. The baselines' performance will be compared with our proposed method.

4.5. Comparison

4.5.1. Performance

Table 2 presents the comparative performance of the proposed SSAD model against the baseline models, evaluated across the metrics of precision, recall, and F1 score. The performance of TimesNet and DLinear on the WADI dataset, as well as all results for the Anomaly Transformer model, are derived directly from their respective official source codes. All other results are from relevant papers. The results demonstrate the superior performance of our model across all datasets. Specifically, the SSAD model achieves significant improvements in F1 score compared to the best-performing baselines, with an improvement of 5.96% on the MSL dataset, 1.77% on the SMAP dataset, 2.94% on the SWaT dataset, and 5.37% on the WADI dataset.

4.5.2. Computational cost and scalability

Table 3 presents the computational costs of recent methods, measured in seconds per epoch during training. iTransformer and DLinear, which are based on a simple Transformer and a combination of linear layers, respectively, exhibit short training times due to their straightforward architectures [32,41]. However, the limited learning layers in these models result in poor performance when handling more complex data. PatchTST, while also a simple Transformer-based model, employs a channel-independent strategy, which increases processing time as it handles each channel sequentially [21]. TimesNet, Anomaly Transformer, and SSAD are all built upon the Transformer architecture. TimesNet expands one time series into three variations, requiring

Table 2
Performance comparison between the proposed method and the baselines.

Methods	MSL			SMAP			SWaT			WADI		
Metric (%)	P	R	F1	P	R	F1	P	R	F1	P	R	F1
PCA	29.37	24.14	26.50	28.84	19.93	23.57	24.92	21.63	23.16	39.53	5.63	9.86
AE	71.66	50.08	58.96	72.16	79.95	75.86	72.63	52.63	61.03	34.35	34.35	34.35
LSTM-VAE	85.49	79.94	82.62	92.20	67.75	78.10	76.00	89.5	82.20	87.79	14.45	24.82
DAGMM	89.60	63.93	74.62	86.45	56.73	68.51	89.92	57.84	70.40	54.44	26.99	36.09
GDN	91.35	86.12	88.66	89.32	88.72	89.02	**99.35**	68.12	80.82	**97.50**	40.19	56.92
InterFusion	81.28	92.70	86.62	89.77	88.52	89.14	80.59	85.58	83.01	92.31	88.47	90.35
PatchTST	88.33	68.51	77.17	89.53	54.33	67.62	99.16	75.21	85.54	28.11	34.55	31.00
iTransformer	58.29	16.37	25.56	90.46	50.53	64.84	91.47	80.49	85.63	30.59	39.15	34.34
DLinear	84.34	85.42	84.88	92.32	55.41	68.26	80.91	95.30	87.52	33.52	45.23	38.51
TimesNet	83.92	86.42	85.15	92.52	58.29	71.52	86.76	**97.32**	91.74	52.76	93.11	67.36
Anomaly Transformer	91.01	84.50	87.63	92.50	96.71	94.56	94.78	93.86	94.32	87.21	97.97	92.28
SSS-AT (Ours)	**91.81**	**97.60**	**94.62**	**93.83**	**98.97**	**96.33**	97.71	96.81	**97.26**	95.41	**100**	**97.65**

Table 3
Computational cost comparison in 4 datasets.

Datasets	MSL	SMAP	SWaT	WADI
iTransformer	14	25	115	412
PatchTST	167	174	865	4814
DLinear	3	6	25	58
TimesNet	339	4140	5391	13 133
Anomaly Transformer	135	313	1100	1781
SSAD	106	234	865	1532

Table 4
The ablation study of the proposed method. Skip-Patching (SP) and spatial–temporal association discrepancy (ST) are the two components.

Components		F1 Score (as %)			
SP	ST	SMAP	MSL	SWaT	WADI
		94.56	87.63	94.32	92.28
✓		95.23	90.67	95.68	96.25
	✓	95.84	89.65	96.65	97.23
✓	✓	**96.33**	**94.62**	**97.26**	**97.65**

more processing time [20]. SSAD, similar in structure to the Anomaly Transformer, incorporates a Skip-patching mechanism that reduces the amount of data handled, thereby decreasing the overall training cost.

The general utility of iTransformer, PatchTST, DLinear, and TimesNet spans across various tasks in the multivariate time series domain, with anomaly detection being one of the applications. In contrast, both Anomaly Transformer and SSAD are specialized for anomaly detection in multivariate time series, optimizing their structures for this specific challenge.

4.6. Ablation studies

Our ablation study evaluates the individual contributions of Skip-Patching (SP) and spatial–temporal association discrepancy (ST) to the overall performance of our anomaly detection model. As detailed in Table 4, we systematically replaced parts of our framework to assess their impact, using the Anomaly Transformer as a base model for comparison.

For the Skip-Patching replacement experiments, We conducted two sets of experiments for Skip-Patching: one utilizing only half of the input series and the other maintaining the original setup. Incorporation of the Skip-Patching module yielded performance improvements across four datasets by 0.67%, 3.04%, 1.36%, and 3.97%, respectively. This indicates that Skip-Patching significantly enhances model performance by promoting semantic richness, sparsity, and an increased sample count.

For our second contribution, Comparing our model's performance with the base model, which primarily analyzes two temporal knowledge representations, our approach — incorporating hidden features and discrepancies from both spatial and temporal domains — outperformed the baseline by margins of 1.28%, 2.02%, 2.33%, and 4.95%. This underscores the value of integrating spatial features, which enrich the model's knowledge base and improve its ability to detect spatial anomalies.

Table 5
Performance comparison with patching size of 2s, 8s, 16s, 32s, and 64s in 4 datasets.

Patch size	2	8	16	32	64
MSL	95.54	96.55	97.26	96.52	96.76
SMAP	95.40	96.19	96.33	95.91	95.49
SWaT	90.53	91.36	94.62	92.82	91.57
WADI	94.16	97.24	97.65	97.68	97.67

Integrating both Skip-Patching and spatial–temporal association discrepancy components, our model significantly outstripped the baseline, demonstrating improvements of 1.77%, 6.99%, 2.94%, and 5.37%. This comprehensive enhancement validates these components' essential and synergistic roles in elevating the model's anomaly detection capabilities. These ablation experiments conclusively demonstrate the necessity and efficacy of each model component, affirming their collective contribution to superior anomaly detection performance across diverse datasets. Besides performance comparison, the p-value has been calculated to exam the difference between SSAD and the baseline. Using 10 experiment results with MSL datasets, the resulting p-value of $7.494\,907\,185\,472\,132 \times 10^{-6}$ indicates a statistically significant difference in performance, strongly favoring the proposed SSAD method over the baseline.

4.7. Hyperparameter sensitivity

4.7.1. Patch size

The selection of patch size within the Skip-Patching component is a critical parameter in our model, directly influencing the amount of local information preserved for subsequent processing. To ascertain the optimal patch size and its impact on model performance, we conducted tests across varying sizes: 2, 8, 16, 32, and 64. These variations aim to gauge the model's sensitivity to the extent of neighboring information considered during training. Fig. 4 and Table 5 illustrate the maximum F1-score achieved with each patch size across different datasets. The results indicate that a patch size of 16 consistently yields the best performance. Smaller patches have a similar effect as masking on the point level, which makes it easy for the neighbors to recover the missing data points, limiting the model's ability to learn meaningful representations. Conversely, larger patches introduce a barren of information. Too much key information is removed from the time series, which makes it impossible for the model to recover full data with limited information. This exploration into patch size sensitivity underscores its importance in our model's architecture, affirming that a well-chosen patch size can significantly improve anomaly detection performance by ensuring an optimal balance of local information retention.

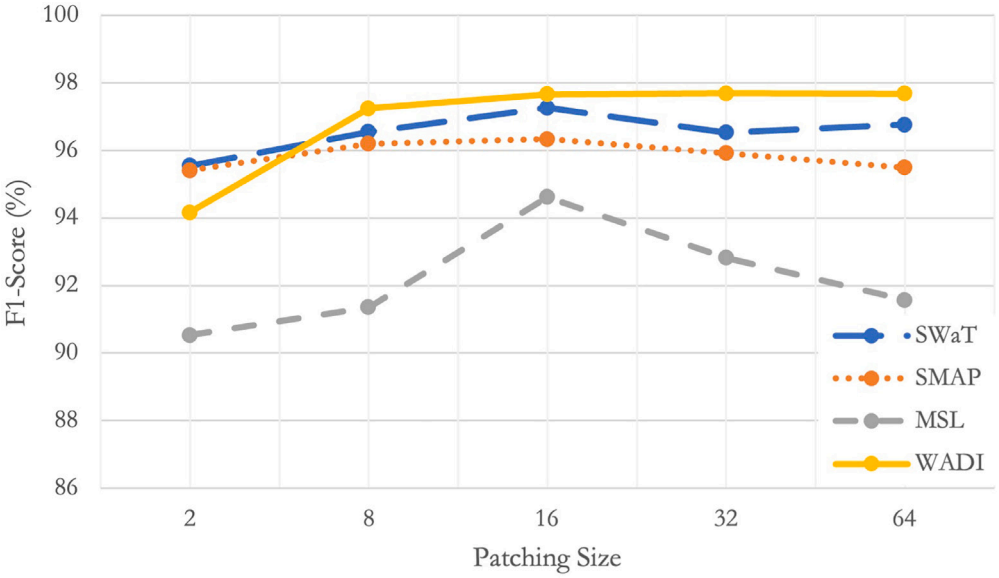


Fig. 4. Performance comparison with patching size of 2s, 8s, 16s, 32s, and 64s in 4 datasets.

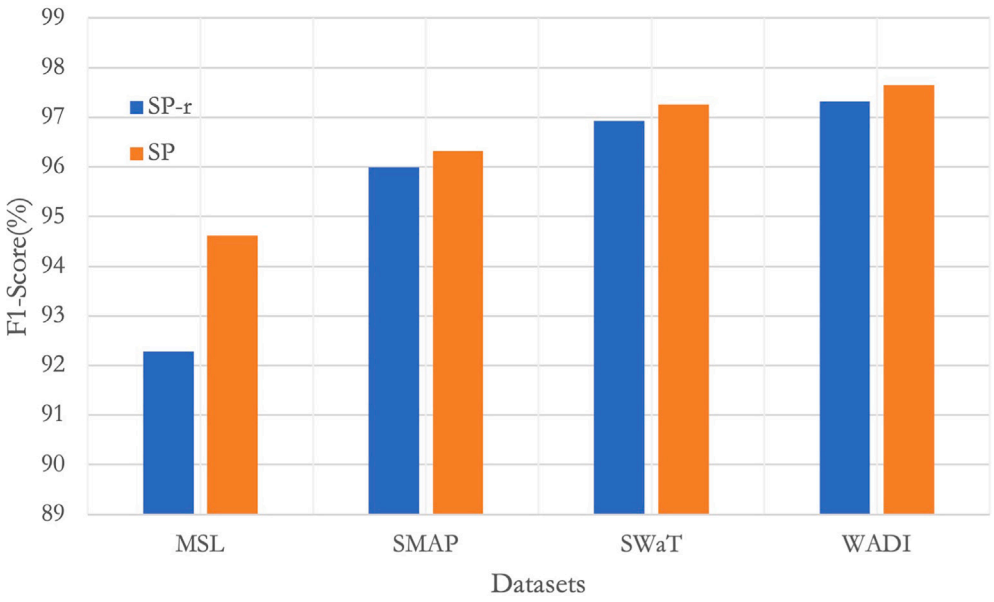


Fig. 5. Performance comparison between SP and SP-r in 4 datasets. SP contains a Skip-Patching component. SP-r replaces the component with a random patches masking module.

4.7.2. Skip-Patching order

The order in which patches are selected and omitted in the Skip-Patching process plays a crucial role in the model's ability to capture and retain meaningful temporal relationships within the data. Our approach involves selectively concealing either odd or even patches, a strategy designed to preserve relational information more effectively than random selection. To validate the efficacy of this ordered patching approach, we conducted an experimental comparison, assessing the impact on model performance. Fig. 5 and Table 6 present a performance comparison, measured in maximum F1-scores, between two configurations: SP, which incorporates the standard Skip-Patching component, and SP-r, which employs a random patch masking module instead. The experiment aimed to determine whether a structured approach to patch omission — focusing on either odd or even patches — offers advantages over random masking. The findings indicate a clear benefit to maintaining a regular order in patch masking. The SP configuration, adhering to an ordered selection process, consistently outperformed the SP-r configuration across all four datasets. This suggests that preserving patches in a sequential manner — thereby maintaining fixed temporal distances between retained patches — facilitates more effective learning of temporal dynamics and relationships. By keeping the temporal structure intact, the regular omission pattern inherent to Skip-Patching enables the model to learn a richer and more efficient representation of the data, underscoring the importance of the Skip-Patching order in enhancing anomaly detection capabilities.

5. Conclusion and future work

In this article, we introduce the novel SSAD method, which significantly enhances representation learning for multivariate time series in IIoT. SSAD method enhances IIoT anomaly detection by improving feature quality and doubling training data volume through innovative half-patch techniques. These techniques create two independent sub-groups, enabling deeper data insights. Alongside this, the Spatial–Temporal Association Discrepancy module aligns spatial and temporal features for more effective representation learning, facilitating precise

Table 6
Performance comparison between SP and SP-r in 4 datasets.

	MSL	SMAP	SWaT	WADI
SP-r	92.28	95.99	96.93	97.32
SP	94.62	96.33	97.26	97.65

anomaly detection in both temporal and spatial domains. The integration of Skip-Patching and spatial–temporal analysis significantly advances the understanding of IIoT data. The model's performance exceeds seven baseline models in four datasets, confirming SSAD's superior capability in anomaly detection.

Deploying the SSAD framework for anomaly detection in real-world IIoT environments is our research's ultimate goal. The quality of training data is crucial for building a robust SSAD model, as it requires data with regular patterns that cover several cycles. The Skip-patching mechanism enhances the model's robustness to missing data, allowing SSAD to handle noisy data points or minor instances of incomplete data. However, it is important to note that a large volume of noisy data points might be misinterpreted as an anomaly event. In extreme conditions, incomplete data could obscure critical information, leading to suboptimal model performance. While SSAD demonstrates strong potential for deployment in real-world IIoT environments, addressing challenges such as handling large volumes of noisy or incomplete data is critical for its successful implementation.

CRediT authorship contribution statement

Yinsong Xu: Writing – review & editing, Writing – original draft, Software, Methodology, Data curation, Conceptualization. **Yulong Ding:** Writing – review & editing. **Jie Jiang:** Writing – review & editing. **Runmin Cong:** Writing – review & editing. **Xuefeng Zhang:** Writing – review & editing. **Shiqi Wang:** Writing – review & editing, Supervision. **Sam Kwong:** Writing – review & editing, Supervision. **Shuang-Hua Yang:** Writing – review & editing, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The datasets used are public datasets.

Acknowledgments

This research is supported in part by the National Natural Science Foundation of China (Grant No. 92067109, 62211530106, 62471278), in part by the Taishan Scholar Project of Shandong Province (Grant No. tsqn202306079), in part by the National Science and Technology Major Project (Grant No. 2021ZD0112100), in part by the Shenzhen Science and Technology Program (Grant No. ZDSYS20210623092007023, GJHZ20210705141808024) and in part by Xiaomi Young Talents Program.

References

[1] J. Xu, H. Wu, J. Wang, M. Long, Anomaly transformer: Time series anomaly detection with association discrepancy, 2021, arXiv preprint arXiv:2110.02642.
[2] X. Chen, L. Deng, F. Huang, C. Zhang, Z. Zhang, Y. Zhao, K. Zheng, Daemon: Unsupervised anomaly detection and interpretation for multivariate time series, in: 2021 IEEE 37th International Conference on Data Engineering, ICDE, IEEE, 2021, pp. 2225–2230.
[3] D. Gong, L. Liu, V. Le, B. Saha, M.R. Mansour, S. Venkatesh, A.v.d. Hengel, Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1705–1714.
[4] A. Deng, B. Hooi, Graph neural network-based anomaly detection in multivariate time series, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 4027–4035.
[5] W. Chen, L. Tian, B. Chen, L. Dai, Z. Duan, M. Zhou, Deep variational graph convolutional recurrent network for multivariate time series anomaly detection, in: International Conference on Machine Learning, PMLR, 2022, pp. 3621–3633.
[6] Z. Li, Y. Zhao, J. Han, Y. Su, R. Jiao, X. Wen, D. Pei, Multivariate time series anomaly detection and interpretation using hierarchical inter-metric and temporal embedding, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 3220–3230.
[7] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018, arXiv preprint arXiv:1810.04805.
[8] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, R. Girshick, Masked autoencoders are scalable vision learners, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 16000–16009.
[9] X. Zhou, Y. Hu, W. Liang, J. Ma, Q. Jin, Variational LSTM enhanced anomaly detection for industrial big data, IEEE Trans. Ind. Inform. 17 (5) (2020) 3469–3477.
[10] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, G. Shroff, LSTM-based encoder-decoder for multi-sensor anomaly detection, 2016, arXiv preprint arXiv:1607.00148.
[11] Z. Chen, D. Chen, X. Zhang, Z. Yuan, X. Cheng, Learning graph structures with transformer for multivariate time-series anomaly detection in IoT, IEEE Internet Things J. 9 (12) (2021) 9179–9189.
[12] F. Angiulli, C. Pizzuti, Fast outlier detection in high dimensional spaces, in: European Conference on Principles of Data Mining and Knowledge Discovery, 2002, pp. 15–27.
[13] M.M. Breunig, H.-P. Kriegel, R.T. Ng, J. Sander, LOF: identifying density-based local outliers, in: Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, 2000, pp. 93–104.
[14] K. Li, S. Kwong, K. Deb, A dual-population paradigm for evolutionary multiobjective optimization, Inform. Sci. 309 (2015) 50–72.
[15] Y. Hong, S. Kwong, Y. Chang, Q. Ren, Consensus unsupervised feature ranking from multiple views, Pattern Recognit. Lett. 29 (5) (2008) 595–602.
[16] C.-H. Tsang, S. Kwong, Ant colony clustering and feature extraction for anomaly intrusion detection, in: Swarm Intelligence in Data Mining, Springer, 2006, pp. 101–123.
[17] J. Li, Z. Zhao, R. Li, H. Zhang, Ai-based two-stage intrusion detection for software defined iot networks, IEEE Internet Things J. 6 (2) (2018) 2093–2102.
[18] F. Zhang, H.A.D.E. Kodituwakku, J.W. Hines, J. Coble, Multilayer data-driven cyber-attack detection system for industrial control systems based on network, system, and process data, IEEE Trans. Ind. Inform. 15 (7) (2019) 4362–4369.
[19] A.N. Jahromi, H. Karimipour, A. Dehghantanha, K.-K.R. Choo, Toward detection and attribution of cyber-attacks in IoT-enabled cyber–physical systems, IEEE Internet Things J. 8 (17) (2021) 13712–13722.
[20] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, M. Long, Timesnet: Temporal 2d-variation modeling for general time series analysis, 2022, arXiv preprint arXiv:2210.02186.
[21] Y. Nie, N.H. Nguyen, P. Sinthong, J. Kalagnanam, A time series is worth 64 words: Long-term forecasting with transformers, 2022, arXiv preprint arXiv:2211.14730.
[22] J. Pan, W. Ji, B. Zhong, P. Wang, X. Wang, J. Chen, DUMA: Dual mask for multivariate time series anomaly detection, IEEE Sens. J. 23 (3) (2022) 2433–2442.
[23] C. Zeng, S. Kwong, Combining CNN and transformers for full-reference and no-reference image quality assessment, Neurocomputing 549 (2023) 126437.
[24] R. Lin, M. Wang, P. Zhang, S. Wang, S. Kwong, Multiple hypotheses based motion compensation for learned video compression, Neurocomputing 548 (2023) 126396.
[25] H. Li, Z. Wang, C. Lan, P. Wu, N. Zeng, A novel dynamic multiobjective optimization algorithm with non-inductive transfer learning based on multi-strategy adaptive selection, IEEE Trans. Neural Netw. Learn. Syst. (2023).
[26] L. Hu, Z. Wang, H. Li, P. Wu, J. Mao, N. Zeng, ℓ-DARTS: Light-weight differentiable architecture search with robustness enhancement strategy, Knowl.-Based Syst. 288 (2024) 111466.
[27] B. Zong, Q. Song, M.R. Min, W. Cheng, C. Lumezanu, D. Cho, H. Chen, Deep autoencoding gaussian mixture model for unsupervised anomaly detection, in: International Conference on Learning Representations, 2018.
[28] A. Alsaedi, Z. Tari, R. Mahmud, N. Moustafa, A. Mahmood, A. Anwar, USMD: Unsupervised misbehaviour detection for multi-sensor data, IEEE Trans. Dependable Secure Comput. 20 (1) (2022) 724–739.
[29] H. Lu, T. Wang, X. Xu, T. Wang, Cognitive memory-guided autoencoder for effective intrusion detection in internet of things, IEEE Trans. Ind. Inform. 18 (5) (2021) 3358–3366.
[30] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, S.-K. Ng, MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks, in: International Conference on Artificial Neural Networks, Springer, 2019, pp. 703–716.

[31] H. Gao, B. Qiu, R.J.D. Barroso, W. Hussain, Y. Xu, X. Wang, Tsmae: a novel anomaly detection approach for internet of things time series data using memory-augmented autoencoder, IEEE Trans. Netw. Sci. Eng. (2022).
[32] A. Zeng, M. Chen, L. Zhang, Q. Xu, Are transformers effective for time series forecasting? in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 37, 2023, pp. 11121–11128.
[33] K. Hundman, V. Constantinou, C. Laporte, I. Colwell, T. Soderstrom, Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 387–395.
[34] A.P. Mathur, N.O. Tippenhauer, Swat: A water treatment testbed for research and training on ICS security, in: 2016 International Workshop on Cyber-Physical Systems for Smart Water Networks (CySWater), IEEE, 2016, pp. 31–36.
[35] C.M. Ahmed, V.R. Palleti, A.P. Mathur, WADI: a water distribution testbed for research in the design of secure cyber physical systems, in: Proceedings of the 3rd International Workshop on Cyber-Physical Systems for Smart Water Networks, 2017, pp. 25–28.
[36] H. Xu, W. Chen, N. Zhao, Z. Li, J. Bu, Z. Li, Y. Liu, Y. Zhao, D. Pei, Y. Feng, et al., Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications, in: Proceedings of the 2018 World Wide Web Conference, 2018, pp. 187–196.
[37] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch, 2017.
[38] M.-L. Shyu, S.-C. Chen, K. Sarinnapakorn, L. Chang, A novel anomaly detection scheme based on principal component classifier, in: Proceedings of the IEEE Foundations and New Directions of Data Mining Workshop, IEEE Press, 2003, pp. 172–179.
[39] O.I. Provotar, Y.M. Linder, M.M. Veres, Unsupervised anomaly detection in time series using lstm-based autoencoders, in: 2019 IEEE International Conference on Advanced Trends in Information Theory, ATIT, IEEE, 2019, pp. 513–517.
[40] D. Park, Y. Hoshi, C.C. Kemp, A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder, IEEE Robot. Autom. Lett. 3 (3) (2018) 1544–1551.
[41] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, M. Long, Itransformer: Inverted transformers are effective for time series forecasting, 2023, arXiv preprint arXiv: 2310.06625.

Yinsong Xu received his B.S. degree in Computer Science from Oregon State University, Corvallis, USA, in 2016. He is currently pursuing a Ph.D. degree with the Department of Computer Science, City University of Hong Kong, Hong Kong. His research primarily focuses on the safety and security of the Industrial Internet of Things.

Yulong Ding received the B.Sc. and M.Sc. degrees in chemical engineering from Tsinghua University, Beijing, China, in 2005 and 2008, respectively, and the Ph.D. degree in chemical engineering from The University of British Columbia, Canada, in 2012. He is currently a Research Associate Professor with the Shenzhen Key Laboratory of Safety and Security for Next Generation of Industrial Internet, and Department of Computer Science and Engineering, Southern University of Science and Technology. His main interests are safety and security of industrial Internet of Things.

Jie Jiang received her B.S. from Chang'an University, China in 2007, M.S. degree from Xi'an Jiaotong University, China in 2010, and Ph.D. degree from Delft University of Technology, the Netherlands in 2015. After finishing her Ph.D., she joined University of Surrey as a Research Fellow. Subsequently, she served as a Research Assistant Professor at Southern University of Science and Technology. Currently, she holds the position of Research Associate Professor in the college of Artificial Intelligence at China University of Petroleum (Beijing). Her research interests primarily lie in sensor-generated data mining, time series analysis, and machine learning.

Runmin Cong (IEEE Senior Member) received the Ph.D. degree in information and communication engineering from Tianjin University, Tianjin, China, in June 2019. He is currently a Professor with the School of Control Science and Engineering, Shandong University (SDU), Jinan, China. His research interests include computer vision, multimedia understanding, content enhancement, machine learning, etc. He has published more than 100 papers in prestigious international journals and conferences, including 2 ESI hot papers (Top 0.1%), 16 ESI highly cited papers (Top 1%). In addition, 32 China patents have been authorized.

Xuefeng Zhang received his Ph.D. degrees in control theory and control engineering from Northeastern University, Shenyang, China. He is currently a professor with the College of Sciences, Northeastern University. He has published more than 200 journal and conference papers and three books. His research interests include fractional order control systems and singular systems. He is "American Mathematical Review" reviewer. He is also the Associate Editors of Information Sciences, IEEE Access, Fractal Fract, IET Electronics Letters and Journal of the Chinese Institute of Engineers and is the Committee Member of Technical Committee on Fractional Systems and Control of Chinese Association of Automation.

Shiqi Wang is an Associate Professor of Computer Science at the City University of Hong Kong. He holds a Ph.D. (2014) from the Peking University. Before his current appointment, from Mar. 2014 to Mar. 2016, he was a Postdoc Fellow with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada. From Apr. 2016 to Apr. 2017, He was with the Rapid-Rich Object Search Laboratory (ROSE), Nanyang Technological University, Singapore, as a Research Fellow. His primary research interests include semantic and visual communication; AI generated content management; information forensics and security; and image/video quality assessment.

Sam Kwong (Fellow, IEEE) received the B.S. degree in electrical engineering from The State University of New York at Buffalo, Buffalo, NY, USA, in 1983, the M.S. degree from the University of Waterloo, Waterloo, ON, Canada, in 1985, and the Ph.D. degree from the University of Hagen, Hagen, Germany, in 1996. He is the Chair Professor of Computational Intelligence and concurrently as an Associate Vice-President (Strategic Research) with Lingnan University, Hong Kong. Dr. Kwong was listed as one of the Top 1% of the World's Most Cited Scientists by Clarivate in 2022. He is the President of the IEEE Systems, Man, and Cybernetics Society from 2021 to 2023.

Shuang-Hua Yang is currently a professor and the Head of Department of Computer Science at the University of Reading, the UK and the Director of Shenzhen Key Laboratory of Safety and Security for Next Generation of Industrial Internet, China. He was selected as a member of European Academy of Sciences and Arts in 2024, and awarded DSc from Loughborough University in 2014 to recognize his academic contribution to wireless monitoring research. He is a Fellow of IET and a Fellow of InstMC, U.K. His current research interests include cyber–physical system safety and security, Internet of Things.