

Research Article

TS-PADM: Anomaly Detection Model of Wireless Sensors Based on Spatial-Temporal Feature Points

Fengjiao Wang ^{1,2}, Ruixing Li ^{1,2}, Hua Wang ^{1,2}, Hengliang Zhu ¹,
and Naixue Xiong ³

¹Department of Computer, Min Jiang Teachers College, Fuzhou Fujian, China

²Fujian University Applied Engineering Centre for Internet of Things, Fuzhou, Fujian, China

³Northeastern State University, Oklahoma, USA

Correspondence should be addressed to Fengjiao Wang; wangfengjiao@fzmjtc.cn

Received 20 October 2020; Revised 15 November 2020; Accepted 14 March 2021; Published 12 April 2021

Academic Editor: Alessandro Bazzi

Copyright © 2021 Fengjiao Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In the practical application, sensor data collection is an essential means for the system to perceive the intrinsic features of data. The anomaly detection of data points can improve data quality and explore the potential information of data. The anomaly detection can be classified as two basic types, that is, classification and clustering. Those methods usually depend on the spatial correlation of data and have high computation complexity, so they are not suitable for the smart home and another mini-Internet of Things (IoT) environment. To overcome these problems, we propose a novel method for anomaly detection. In this paper, we first define the temporal and spatial feature of data flows; then, a time series denoising autoencoder (TSDA) is proposed to extract the discriminative high-dimensional characteristics to represent the data points. Moreover, a probability statistics-based anomaly detection model (PADM) was proposed for identifying the abnormal data. Extensive experimental results demonstrated that our method has fewer parameters and is easy to adjust and optimize. More importantly, our approach has higher precision and recall rate than the gradient boosted decision tree and XGBoost.

1. Introduction

Anomaly points are also called outliers or inconsistent points in the data mining and statistical analysis, and the anomaly point detection is a process to find the abnormal points whose behaviours are different from others. The development of IoT and sensor technologies makes smart home richer and more prosperous in the monitoring field [1]. Due to the uncertainty of the sensor deployment area and the limitation of equipment resources, the sensor data is easily interfered with and broken by external factors, so there is a widespread problem of sensor data unreliability. Therefore, it is urgent to solve the problem of ensuring and improving the quality of sensor data. Anomaly detection is a hot research topic in academia and industry, such as disease recognition in the medical system, network intrusion detection in the security field, credit card antifraud in the financial field, and anomaly detection in urban traffic management [2]. Besides, under the age of big data, there is the abuse of false data in

monitoring data, so detecting these outliers has also become a new focus.

Traditional anomaly detection methods mainly include hierarchical clustering, partition-based clustering, density-based clustering, and fuzzy clustering. The clustering algorithms are widely applied, such as the Gaussian mixture model (GMM), *K*-means clustering, and fuzzy clustering analysis [3]. However, when these methods are applied to small IoT environments, such as smart homes or smart buildings, the detection based on spatial correlation can increase the algorithm's complexity. On the other hand, because the number of same-type sensor nodes deployed in the environment is small and the spatial correlation of data is insufficient, so the detection results of these algorithms are usually unsatisfactory. Also, the high dimension of data is a massive challenge for anomaly detection, and traditional methods usually utilize the nearest neighbour or neighbourhood information to search the nearest data points. However, the data in a high-dimensional space tends to be sparser than

in the lower dimensions, and the distance among the data points is no longer intuitive [4]. Therefore, by the temporal correlation and statistical features of data, in this paper, we creatively propose a novel model for anomaly point detection, which employs the spatial-temporal features and probability statistics (TS-PADM). We also propose a time series denoising autoencoder (TSDA) network to compress high-dimensional monitoring data, which can be used to represent the temporal and spatial feature of detection points.

Furthermore, a Gaussian model is introduced into the anomaly detection. In this model, we use the auxiliary target variables to gain the anomaly points by the objective function of region partitioning. Here, the EM algorithm is used to determine the target solution, and then, the anomaly points are detected. Experimental results show that the method proposed in this paper achieves better performance than random forest (RF) [5], gradient boosted decision tree algorithm (GBDT) [6], and XGBoost algorithm [7] in terms of accuracy.

This paper's structure is organized as follows: Section 2 discusses the relevant research work of the anomaly detection method. Section 3 introduces the representation of data spatial-temporal features based on time series denoising autoencoder in detail. Section 4 elaborates on the new anomaly detection algorithm of this paper. Section 5 shows the experimental results and verification. Finally, Section 6 summarizes the paper.

2. Relation Work

The anomaly detection methods in wireless sensor networks are divided into a statistics-based method, distance-based method, classification-based method, and cluster-based method. Literature [8] proposed the abnormal data detection through the variable-width histogram of computational data; i.e., the dynamic data in the technical network of data fusion were aggregated into a variable-width histogram to detect the anomaly data. Liu et al. [9] proposed an isolation forest algorithm which built anomaly indexes according to the path length from leaf node to root node, with good detection results for the global outliers but bad results for dealing with local sparser points. Wang et al. [10] proposed gradient boosted decision tree, which combined weaker learners into a stronger learner through iteration and reduced residual errors in successive iterations, thus generating trees deepened lengthwise. This method has the advantages of high prediction accuracy and strong robustness to outliers [11, 12].

Chen et al. [13] proposed the extreme gradient boosting tree algorithm, which took a step in the negative gradient of the loss function but conducted second-order Taylor series expansion of empirical errors and added some standard terms to make the loss function expandable. However, this algorithm had too many parameters, and the using effect relies too heavily on parameter tuning results. Literature [14] proposed an anomaly detection method based on presumptive mathematical statistics model and kernel density function but required the prior knowledge of sensor data distribution and relied on a definite mathematical model, so it had the features of limitations and weak universality. Literature

[15] proposed an anomaly detection algorithm based on support vector machines (SVM), which learned a classification model with a training dataset and classified data instance into the class learned. In this algorithm, when the class has less data or the data did not belong to any class, such data were regarded as anomaly data. This algorithm required larger data samples to be training sets, so the number of samples would become its bottleneck, and the effect of anomaly detection was unsatisfactory. Literature [16] proposed an anomaly detection algorithm based on the K -means algorithm, which classified similar data instances into the data families with similar behaviours to achieve anomaly detection. This method did not introduce and utilize data streams' statistical features based on the data's spatial correlation. Literature [17] proposed an anomaly detection algorithm based on combining the K -means algorithm and the FP growth algorithm, which first conducted data modelling and then detected abnormal data. The cluster centre location and number selection of this method greatly impacted detection results, so its detection effects were not stable, and its complexity was very high [18]. Literature [19, 20] proposed an anomaly detection method based on distributed computation, which had sound detection effects in the large distributed environment. However, it was complicated and not easy to be achieved, so it was not suitable for the wireless sensor networks deployed in the home environment. Also, its data anomaly detection was achieved by the temporal and spatial correlation among detection data [21, 22].

In the specific environment of IoT, the same type of sensor nodes is insufficient in the number, so using the features of the time series of the data stream will have better detection effects [23]. This paper adapts autoencoder to extract the temporal-spatial features of detection points specific to this environment and the defects of existing methods. Considering the correlation of the spatial position of detection points and the change law of time series, auxiliary distribution variables were introduced to optimize the objective function of deep clustering. According to space-time features, detection points were clustered, and a multivariate Gaussian model based on probability statistics is designed to detect the anomaly data. Compared with the literature [24], our method does not need additional sensor nodes to provide data and saves the cost of communication, storage, and computation. The detection in standard scenarios such as causing the equidistant change of data stream when the equipment was turned on reduced the false alarm rate. Data anomaly can be effectively and accurately detected under the environment of a single sensor data stream [25].

3. Spatial-Temporal Features

3.1. Feature Description. Suppose that the total number of the sensor detection points included in the smart equipment is n , and define all detection point sets as $X = \{x_i = i | 1, \dots, n\}$, where $x_i \in dx$ refers to the physical quantity features of detection points, including air quality, temperature, illumination, heart rate, and noise. d_x refers to the number of dimensions of spatial-temporal features. The original features are mapped to the latent feature space by feature mapping

$\varphi : x \rightarrow Z$ to gain the original features of time series and space, so it can be expressed as $Z = \{z_i | i = 1, \dots, n\}$. Based on latent feature data z_i , the probability p_{ij} of detection point x_i belonging to the region r_j is calculated by feature mapping $f_\theta : z \rightarrow P$, where θ refers to parameter set.

Most spatial features that are original data features of the sensor have dynamic properties, and time series data is highly dimensional and continuously variable. The code mapping of $\varphi : x \rightarrow Z$ can reduce time series data and feature representation. Latent feature data is $z_i \in Z$, and feature dimensions are $d_x * dz$, where z_i consists of the features of time series and space of detection points. The model realizes dimension reduction and compression of time series data and uses normalization method to process spatial feature data of detection points.

3.2. Representation of Features of Time Series. A time series denoising autoencoder (TSDA) is proposed in this paper for the time series data of the detection points with high dimensionality and much noise. Random noise data were added into the sample dataset in the training process in order to enhance the antinoise property of TSDA. Convolutional layers (Conv2D) and maximum pooling layers (MaxPooling2D) were used in the encoding phase to achieve the feature extraction representation of time series data; the convolutional layers opposed to the encoding process and upsampling layers (UpSampling2D) were used in the decoding phase to reconstruct extraction representation into primitive input. TSDA has the same input and output, and its objective function is reconstruction error to optimize encoder and decoder.

- (1) Series input: the time series (normalized) $T_{x_i} = [t_1, t_2, \dots, t_\gamma]$ with a length of γ in the detection point x_i is selected. In order to facilitate the subsequent operations such as convolution, pooling, and upsampling, $T(x_i)$ is converted into two-dimensional matrix $T'(x_i)$ by reshaping; meanwhile, Gaussian random noise was added into $T'(x_i)$ to gain TSDA input
- (2) Encoding phase: multiple convolutional layers and maximum pooling layers are alternately stacked to compress input data to gain the feature representation
- (3) Decoding phase: multiple convolutional layers and upsampling layers are alternately stacked to restore the data feature representation to reconstruction input
- (4) Objective function: the error between the primitive input and the reconstruction input is taken as the loss function

Weight was saved after TSDA training, and only the encoder is used to complete the feature extraction representation of time series.

Suppose feature of time series $YT(x_i)$ of x_i as $YT(x_i) = [yt_1, yt_2, \dots, yt_x]$, where t_x refers to the number of dimensions of the feature of time series.

3.3. Representation of Spatial Features. The spatial features of detection points included coordinate position information, degree of importance of detection points, and instrument type, which were processed with the normalization method. Suppose the spatial features (YS) of x_i (YS) as $YS(x_i) = [ys_1, ys_2, \dots, ys_x]$, where s_x refers to the feature dimension.

The coordinate values determined the primitive spatial position of the detection points in the measuring coordinate system. When the dimensional units were different and the numerical values are significantly differentiated, it is not easy to give the relative position relation an accurate description among the detection points with specific quantities. Therefore, removing the influence of dimension and large values could effectively retain the relative position information of detection points and improve the convergence rate of subsequent training. The detailed practice is as follows:

- (1) Unified dimension: the dimension units of coordinate values were unified through the unit conversion in the coordinate system (the unit is usually unified as meter)
- (2) Nonlinear normalization: the extensive space range occupied by the structural body causes broadly differentiated position coordinate data of each detection point [26]. The function

$$y = \text{sgn}(x) \frac{\ln(|x|)}{\ln(\max|x|)} \quad (1)$$

Function (1) is used for conversion, where $\text{sgn}(x)$ is a mathematical formula that avoids negative coordinate values caused by different origins of coordinates selected in some coordinate systems.

4. Anomaly Detection Algorithm Based on Spatial-Temporal Features

4.1. Data Partitioning. The data partitioning problem can be equivalent to the problem that n detection points are partitioned into m regions. Suppose the region set finally partitioned as $R = \{r_j | j = 1, \dots, m\}$. Formula (2) is used to calculate the possibility p_{ij} of the detection point z_i in the region $r_j, i = 1, \dots, n$.

$$p_{ij} = f(z_i, \theta) = \frac{e^{\theta_j^T z_i}}{\sum_{j=1}^m r_j e^{\theta_j^T z_i}} \quad (2)$$

The softmax activation function ($f_\Theta = e^{x_i} / \sum e^{x_i}$) is adopted to calculate the possibility, where $\Theta = [\theta_1, \theta_2, \dots, \theta_m] \in d_z \times m$, and then, the gradient descent method is used to optimize the parameters. When, i.e., $p_{ij'} = \max\{p_{ij} | j = 1, \dots, m\}$, when $p_{ij'}$ is the maximum possibility, the detection point x_i is partitioned into the region $r_{j'}$ [27].

4.2. Iterative Solution. Suppose the time series features as $YT(x_i) = [yt_1, yt_2, \dots, yt_x]$, where t_x refers to the dimensions

of features of time series. Suppose space as $YS(x_i) = [y_{s_1}, y_{s_2}, \dots, y_{s_x}]$, where s_x refers to the dimensions of features that is the normalized parameter X . It is a training set containing m data, in which each sample is an n -dimension datum:

$$X = [x^{(1)}, x^{(2)}, \dots, x^{(m)}],$$

$$x^{(i)} = \begin{bmatrix} x_1^{(1)} \\ x_1^{(2)} \\ \vdots \\ x_n^{(i)} \end{bmatrix}. \quad (3)$$

It can be judged whether a sample is abnormal or not through the following function:

$$p(x^{(j)}; \mu, \sigma^2) = \prod_{i=1}^n p(x_i^{(j)}, u_i, \sigma^2) < \varepsilon. \quad (4)$$

We aim to acquire μ and σ according to the training set to gain a definite multivariate regular distribution model. Specifically, the following conclusion can be obtained from the maximum likelihood estimators:

$$\mu = \frac{1}{m} \sum_{j=1}^m x^{(j)}, \quad (5)$$

$$\Sigma = \frac{1}{m} (X - \mu)(X - \mu)^T,$$

where Σ is a covariance diagonal matrix, and the multivariate regular distribution model finally acquired can be written as

$$p(x^{(j)}; \mu, \sigma^2) = p(x; \mu, \Sigma) = (2\pi)^{-(n/2)} |\Sigma|^{-1/2} \cdot \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right). \quad (6)$$

The expectation maximum (EM) algorithm is used to find the maximum likelihood estimation of parameters or the maximum posterior estimation in the probabilistic model. The probabilistic model relies on the unobservable latent variables.

The expectation maximum algorithm is used for alternate calculation in three steps:

- (1) Calculation of expectation (E): the existing estimated values of the probabilistic model parameters are used to calculate the expectation of latent variables
- (2) Maximization (M): the expectation of latent variables acquired in the above E step is used to perform the parameter model's maximum likelihood estimation
- (3) E step

The latent variable $y_{t,k}$ is introduced into the existing sample set $Y = (y_1, y_2, \dots, y_T)$, and the data can be expanded as perfect numbers, where $t = 1, 2, 3, \dots, T$.

The latent variable $y_{t,k}$ refers to the detection point of the sample coming from the k th smart device. The likelihood function of perfect numbers

$$P(y, y | \mu, \varepsilon, \pi) = \prod_{t=1}^T (y_t, y_{t,1}, y_{t,2}, y_{t,3}, \dots, y_{t,k})$$

$$= \prod_{t=1}^T \prod_{k=1}^K (\pi_k N(y_{t,k}, \mu_k * \varepsilon_k))^{y_{t,k}}$$

$$= \prod_{k=1}^K \pi_k \varepsilon_{t=1}^T y_{t,k} \prod_{t=1}^T (\pi_k N(y_{t,k}, \mu_k * \varepsilon_k))^{y_{t,k}}. \quad (7)$$

The log-likelihood function of perfect numbers is R :

$$\ln p(y, y | \mu, \varepsilon, \pi) = \sum_{k=1}^K \left\{ \left(\sum_{t=1}^T y_{t,k} \right) \ln \pi_k \right. \\ \left. + \sum_{t=1}^T y_{t,k} [-\ln(2\pi) - 0.5 \ln |\Sigma_k|] \right\}. \quad (8)$$

4.3. Description of TS-PADM Algorithm. A feature set of wireless sensors $X = \{x_i = i | 1, \dots, n\}$ is taken as the input parameters of the TS-PADM algorithm, and the probability distribution $\{p_{ij} | i = 1, \dots, n; j = 1, \dots, t\}$ of detection points in various regions is taken as the output. The hyperparameters requiring manual setting in the algorithm include training batches, maximum number of iterations, and threshold of iteration errors. During the classification of detection points, the space and time of detection points are used for unsupervised clustering, and the number of regions t need not be designated. See the details in Algorithm 1.

Feature mapping $\varphi: x \rightarrow Z$ represents the features of time series of detection points through the denoising compression of time series and represents the spatial features with the normalization method, that is, mapping the primitive features of detection points to the latent feature space. Generally, the results of a clustering algorithm (such as K -means and Gaussian mixture model) are selected to initialize the target distribution R , avoiding the uncertainty of random initialization and speeding up the convergence process. Because R is the actual possibility distribution of detection points in their regions, the initialization by different clustering algorithms does not affect it. The EM method is used for iterative solution. As the target distribution R has been initialized, the EM method is used in line 4 to update the parameter ε and the possibility distribution P and then to fix the parameter to update the target distribution R . The algorithm finally gets the abnormal detection points back.

Input: a feature set of detection points $X = \{x_i = i | 1 \cdots n\}$
Output: the probability of detection points in various regions $\{p_{ij} | i = 1, \dots, n; j = 1, \dots, t\}$

1. Map the primitive features of detection points to the latent feature space, $f_\theta : z \rightarrow P$,
2. Use a clustering algorithm to initialize the target distribution R
3. WHILE NOT converged:
4. Fix the target distribution $\min_y R$ and update the parameter ϵ
5. Calculate the possibility of the detection points in the region $p_{ij} = f(z_i, \Theta)$ and update P
6. Fix the parameters and calculate $\gamma_{ij} = \varphi(p_{ij}, \epsilon)$ to update target distribution R
7. END WHILE
8. RETURN $\{p_{ij} | i = 1, \dots, n; j = 1, \dots, t\}$

ALGORITHM 1: Anomaly detection algorithm based on spatial-temporal features.

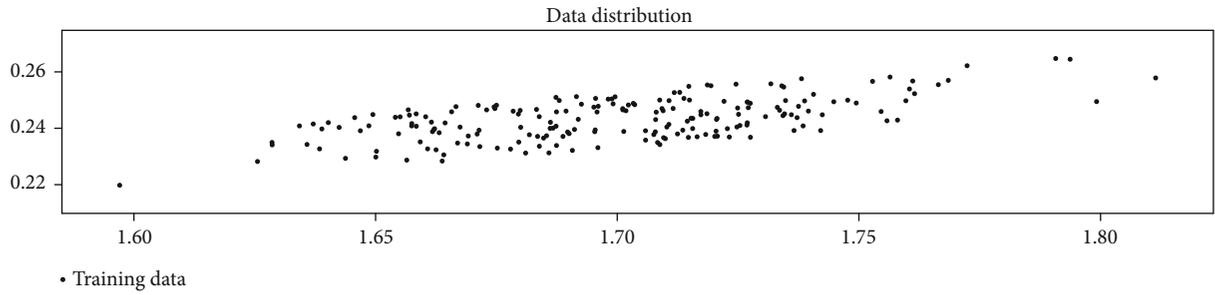


FIGURE 1: Test distribution of data samples.

5. Experimental Results and Verification

5.1. Data Sample Structure. Numerous sensors and smart devices in the smart home environment generate a large amount of data at all times, and conversely, those data provide a basis for altogether mining user behaviours and realizing smart control of home devices. Smart home environment data usually include monitoring data and smart device status. The monitoring data can be divided into two parts. The first part is environmental monitoring data of temperature, humidity, time, weather, brightness, etc., which are generally acquired through sensors and network information; the second part is user information involving user's position, status, and movement, which are acquired through indoor positioning technology and wearable devices. The status of a smart device refers to its current work status and parameter settings.

On the one hand, the status becomes some input dimension of decision-making by the Bayesian model as equipment information. On the other hand, it is also the decision-making objectives of the Bayesian model. The model is used for decision-making based on the current environmental data to achieve intelligent control of home devices by modifying the device status. There are distributed and multichannel methods to collect the smart home's environmental data, so the integration of scattered data is an essential prerequisite for the decision-making of device control. An expression specification of the home environment's data samples is proposed in this thesis: the current home environment data are sampled every 30 seconds and sorted into a data sample structured as in Figure 1. It is a modelling representation of

the current home environment's sensor data, forming a primitive dataset. For the detailed derivative processes of maximum likelihood estimator, covariance matrix, and the maximum likelihood estimation of the multivariate normal distribution, please refer to literature [28].

- (1) Data generation. Partial log data of sensor is input to test the algorithm model in this paper. The Gaussian model in the experiment had a mean value of $[0,1]$, a covariance of $[[0.3,0],[0,0.1]]$, and 30 data samples; the Gaussian model has a mean value of $[1, 2]$, a covariance of $[[0.2,0],[0,0.3]]$, and 30 data sample
- (2) Data preprocessing. The algorithm in this paper is used to preprocess the points of the sample set as a decomposition time series feature set $YT(x_i) = [yt_1, yt_2, \dots, yt_x]$ and an actual spatial feature set $YS(x_i) = [ys_1, ys_2, \dots, ys_x]$. The data need to be preprocessed before applying the EM algorithm of the Gaussian mixture model to the samples. That is, all the sample values are scaled to between 0 and 1. In literature [29], Mikalsen et al. proposed a method of calculating the distance between data objects for anomaly detection. It defines the data anomaly as a case that in the dataset X , the number of data objects existing in the circle in the range of R is less than ρ for a data object 0. On this basis, the features of time series and spatial-temporal features are creatively decomposed. Data partitioning of the feature set is applied to the comprehensive utilization of time correlation and the data stream's statistical features. The data

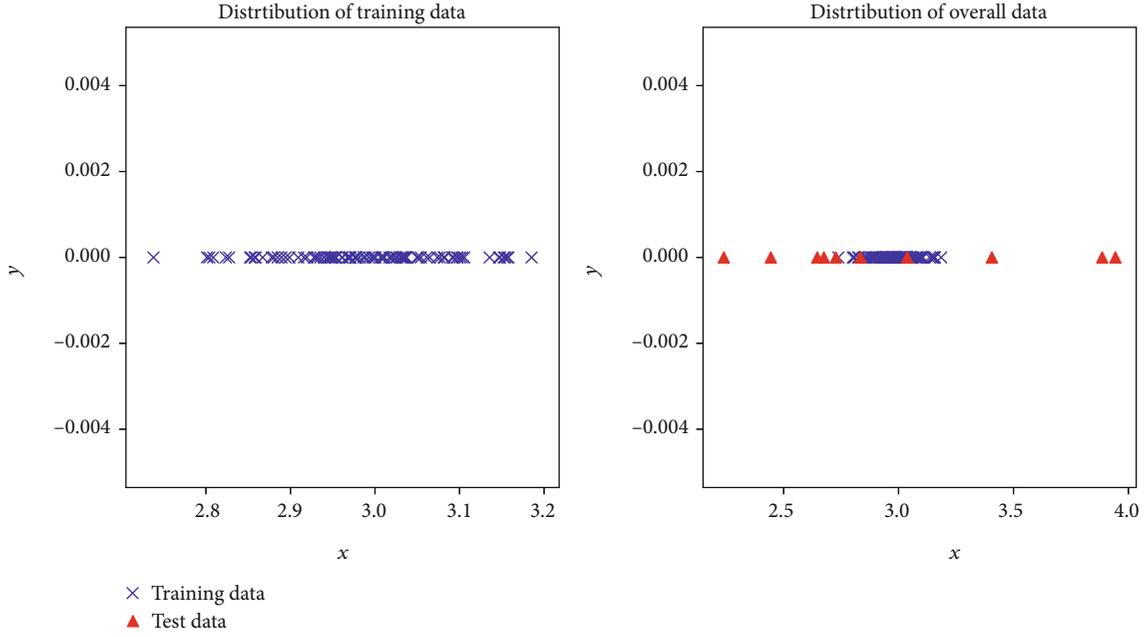


FIGURE 2: Test distribution of data samples.

samples from the dataset are trained for the model, and the results are obtained as shown in Figure 2

5.2. Experiment Preparation

5.2.1. Measured Dataset. The measured data are about 100,000 data records of detection points obtained from an environmental monitoring platform's processed monitoring data from April 1, 2019, to June 1, 2019. Nine hundred sixty-four measuring points are involved, and the observed physical quantity includes air quality, temperature, illumination, pollutants, and noise.

5.3. Benchmark Methods. The most commonly used K -means clustering and Gaussian mixture model are selected as benchmark methods based on a probability statistics-based anomaly detection model (PADM).

On the measured dataset, the model in this paper (TS-PADM) was used to extract features and modify K -means and GMM.

5.4. Evaluation Index. Experiments are performed with the above methods: K -means, GMM, and TS-PADM are used in the experiment on the dataset, and Silhouette Coefficient (SC) and Rand Index (RI) are adopted to evaluate the region partitioning performance.

The evaluation index SC is used to quantify the distribution of detection points. $SC \in [-1, 0)$ indicates that region partitioning performance is basically wrong, 0 indicates that it is involved into the locally optimal solution, and $SC \in [0, 1]$ indicates that the more the value, the more reasonable and uniform the distribution of detection points. The RI index is used to measure the cohesion degree of the region. The more the value, the higher the cohesion degree [30].

For the accuracy rate, the models are used to classify the test set; it refers to the proportion of the number of samples correctly classified in total samples shown as follows [31]:

$$\text{Accuracy} = \frac{N_{\text{correct}}}{N_{\text{total}}}. \quad (9)$$

For the recall rate, it refers to the proportion of the true-positive (TP) samples truly classified in all positive samples (TP + TN), where TN (true negative) refers to the data falsely classified as follows:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{TN}}. \quad (10)$$

5.5. Experimental Results of Measured Dataset. To make it simple, we suppose that all the data in X_{train} are normal data. In this way, fit (X_{train}) refers to the calculated model parameters of multivariate normal distribution, and Gaussian (X_{test}) refers to the density value of multivariate normal distribution calculated according to the objective function.

All the samples in X are detected by $\text{predict}(X)$ and returned to the list of detection results corresponding to X . The elements in the list are a two-tuple. The first element records whether x_i is normal data, and the second element records $p(x_i; \mu, \Sigma)$. Since we have assumed that all the data in X_{train} are normal data, the smallest density value in X_{train} is selected as ϵ here.

20 test data in X_{test} are possible abnormal samples. A large number of samples are needed by the intelligent detection program to train a function model. Perhaps, we firstly consider to label the samples as "normal" and "abnormal" like supervised learning and then train the model through classification algorithm. Suppose that x_{test} is a data sample, and $\text{predict}(x_{\text{test}})$ is used to judge whether x_{test} is a qualified

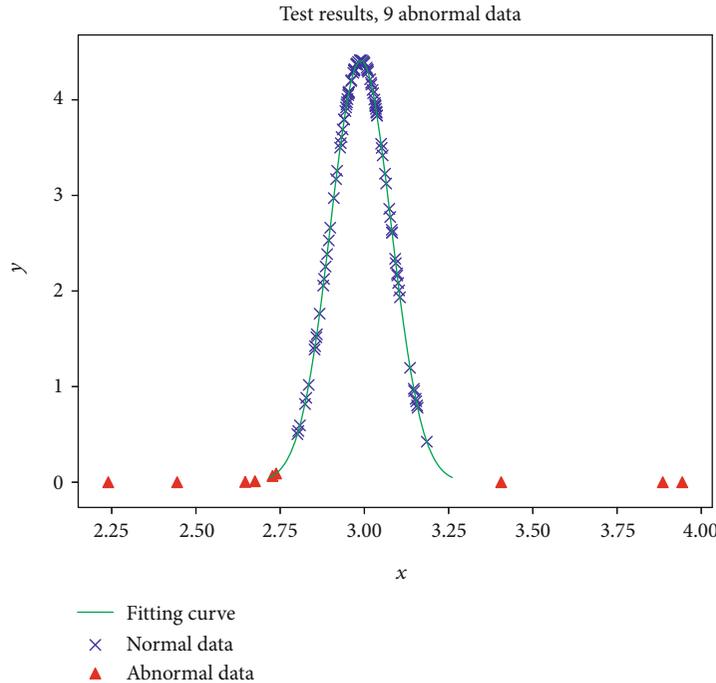


FIGURE 3: Detection results.

sample. From this, it can be seen that most of the training data are concentrated in the mean area of normal distribution, while the abnormal data are deviated towards both ends of the “reverse clock.” Next, the anomaly detection model is trained by fit method, and the results are $\mu = 2.98961537626059$ and $\sigma^2 = 0.008178221946938589$. After the model parameters are obtained, the objective function can be used to predict the data. Gussian (X) realizes the density function of normal distribution, while predict (X) detects all samples in X and returns to the list of detection results corresponding to X . Its visualization results are shown in Figure 3 as follows.

5.6. Experimental Control and Conclusions. The index factor SC was used to quantify the abnormal situation of measuring points, and RI index was used to measure the cohesion degree of region. The larger the values, the better the corresponding performance [32]. The average SC and RI indexes of K -means, GMM, and TS-PADM are calculated as shown in Table 1.

As shown in Table 1, SC and RI indexes of TS-PADM were significantly higher than those of K -means and GMM. The air quality coefficients of the three methods were within the interval, and the SC of TS-PADM was 44.3% and 37.6% higher than that of K -means and GMM, respectively, indicating that TS-PADM had better region partitioning results. TS-PADM preceded K -means and GMM in the RI index by 36.8% and 21.8%, respectively, indicating that its areas partitioned had a higher cohesion degree, higher correlation of detection point single domain, and lower correlation among single domains. TS-PADM was superior to K -means and

TABLE 1: K -means, GMM, and TSPADM test results.

Indicators	K -means	GMM	TS-PADM
SC	0.532	0.558	0.768
RI	590.7	731.1	935.3

GMM in regional cohesion and distribution of measuring points.

The accuracy rate and recall rate of the proposed algorithm were compared with those of the random forest (RF) algorithm [33], gradient boosted decision tree (GBDT) algorithm [5], and XGBoot algorithm [6]. The compared results of different algorithms on dataset are calculated as shown in Table 2.

From the analysis of the experimental results, it can be found that the detection effect of the proposed method was equivalent to that of XGBoot when anomaly classification was simulated on the dataset; TS-PADM has a higher accuracy rate and recall rate than the other methods and shows good anomaly classification performance. The experimental results show that the proposed method’s effect is better than that of the GBDT algorithm and the RF algorithm. In practical problems, XGBoot has many hyperparameters that are complicated to adjust and optimize. The method proposed in this thesis based on feature segmentation and cascaded random forest has fewer hyperparameters and better practicability, so it is of better research and application value in the field of future outlier detection.

Although the TS-PADM algorithm performs well in the measured dataset in this paper, it may not also show such

TABLE 2: Compared results of different algorithms on dataset.

Algorithms	Accuracy	Recall
TS-PADM	0.9812	0.9867
RF	0.9623	0.9423
GBDT	0.9451	0.9532
XGBoot	0.9634	0.9654

TABLE 3: Characteristics of experimental dataset.

Dataset	Total instances	Outliers	Features
Heart disease	303	120	75
WPBC	198	47	34
Parkinson	197	48	23
Internet ads	3279	454	1558

WPBC is Breast Cancer Wisconsin (Prognostic) Dataset.

TABLE 4: Compared results of different algorithms on Heart disease.

Algorithms	Accuracy	Recall
TS-PADM	0.9417	0.9502
RF	0.9515	0.9412
GBDT	0.8762	0.9121
XGBoot	0.9192	0.9234

The RF algorithm is comparable to TS-PADM.

TABLE 5: Compared results of different algorithms on WPBC.

Algorithms	Accuracy	Recall
TS-PADM	0.9452	0.9562
RF	0.9411	0.9318
GBDT	0.8642	0.8579
XGBoot	0.9327	0.9236

The TS-PADM algorithm performs best and is stable.

good accuracy and recall rate with the amount of data and feature data increasing.

5.7. Control Experiment and Analytical Results. In order to verify the effectiveness of the method proposed in this paper, tests were conducted on the four datasets shown in Table 3. These datasets were obtained from the UCI [34] open resource. The UCI currently maintains 559 datasets as a service to the machine learning community. The tenfold cross-validation method is used on all the experimental datasets, with 80% for training and 20% for validation.

Tables 4–7 present the comparison results obtained by different outlier detection algorithms.

Analysis of the experimental results shows that the algorithm in this paper is comparable to the RF algorithm when the amount of data is similar. As the number of data

TABLE 6: Compared results of different algorithms on Parkinson.

Algorithms	Accuracy	Recall
TS-PADM	0.9445	0.9552
RF	0.9423	0.9123
GBDT	0.8478	0.8636
XGBoot	0.9122	0.8356

The WPBC and Parkinson have similar data volume, and the TS-PADM algorithm performs best.

TABLE 7: Compared results of different algorithms on Internet ads.

Algorithms	Accuracy	Recall
TS-PADM	0.9745	0.9612
RF	0.9411	0.9378
GBDT	0.8778	0.8832
XGBoot	0.8423	0.8356

With the increasing number of Internet ad data instances, TS-PADM has outstanding advantages.

instances increased, the method outperformed the other three algorithms in accuracy and had a higher recall on the Internet ad datasets.

Combining the above data shows that the TS-PADM performs a slight disadvantage on heart disease. But it does not perform as consistently as TS-PADM on high-dimensional datasets with many features. TS-PADM achieves high recall on both high- and low-dimensional datasets while ensuring accuracy in the anomaly classification task, and the performance advantage is even more pronounced on high-dimensional datasets.

6. Conclusions

In this paper, the anomaly detection model of wireless sensors based on temporal-spatial feature points (TS-PADM) is proposed to compress high-dimensional monitoring data to represent the detection points' temporal-spatial features. On this basis, an anomaly point model based on probability statistics is adopted, and auxiliary target variables are introduced to optimize the objective function of region partitioning. The detection data obtained by sensors achieve better results in anomaly detection modelling. Extensive experiments show that the proposed TS-PADM algorithm can obtain relatively superior detection results and respond to the abnormal data changes of smart equipment in time. In future work, we will exploit deep learning methods to extract high-level semantic features of anomaly data to improve the prediction of anomaly detection.

Data Availability

The measured data which are about 100,000 data records used to support the findings of this study are available from the corresponding author upon request. And the four

datasets shown in Table 3 were obtained from the UCI (<http://archive.ics.uci.edu/ml/index.php>.) open resource.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

The financial support for this work provided by the Fundamental Research Funds for Department of Education, Fujian Province (No. JZ180635) is gratefully acknowledged.

References

- [1] C. Wang, Z. Liu, H. Gao, and Y. Fu, "Applying anomaly pattern score for outlier detection," *IEEE Access*, vol. 7, pp. 16008–16020, 2019.
- [2] M. Li, R. Kashef, and A. Ibrahim, "Multi-level clustering-based outlier's detection (MCOD) using self-organizing maps," *Big Data and Cognitive Computing*, vol. 4, no. 4, p. 24, 2020.
- [3] M. Ahmed, A. N. Mahmood, and M. R. Islam, "A survey of anomaly detection techniques in financial domain," *Future Generation Computer Systems*, vol. 55, no. 6, pp. 278–288, 2016.
- [4] Y. Djenouri and A. Zimek, "Outlier detection in urban traffic data," in *Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics*, pp. 1–12, Novi Sad, Serbia, 2018.
- [5] Z. H. Gong, J. N. Wang, and C. A. Su, "Weighted Deep Forest Algorithm," *Computer Applications and Software*, vol. 36, no. 2, pp. 274–278, 2019.
- [6] K. Guo, D. Liu, Y. Peng, and X. Peng, "Data-driven anomaly detection using OCSVM with boundary optimization," in *2018 Prongs tics and System Health Management Conference IEEE*, pp. 244–248, Chongqing, China, 2018.
- [7] T. Chen and C. Guestrin, "XGBboost: a scalable tree boosting system," in *Proceedings of the 22nd ACM Sigkdd International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, San Francisco, CA, USA, 2016.
- [8] H. Zhang and Z. Li, "Anomaly detection approach for urban sensing based on credibility and time-series analysis optimization model," *IEEE Access*, vol. 7, pp. 49102–49110, 2019.
- [9] F. T. Liu, K. M. Ting, and Z. H. Zhou, "Isolation- based anomaly detection," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 6, no. 1, pp. 1–39, 2012.
- [10] H. Wang, M. J. Bah, and M. Hammad, "Progress in outlier detection techniques: a survey," *IEEE Access*, vol. 7, pp. 107964–108000, 2019.
- [11] Y. Yang, N. Xiong, N. Y. Chong, and X. Défago, "A decentralized and adaptive flocking algorithm for autonomous mobile robots," in *2008 The 3rd International Conference on Grid and Pervasive Computing - Workshops*, Kunming, China, 2008.
- [12] A. Shahzad, M. Lee, Y. K. Lee et al., "Real time MODBUS transmissions and cryptography security designs and enhancements of protocol sensitive information," *Symmetry*, vol. 7, no. 3, pp. 1176–1210, 2015.
- [13] B. Chen, T. Hu, Z. Huang, and C. Fang, "A spatio-temporal clustering and diagnosis method for concrete arch dams using deformation monitoring data," *Structural Health Monitoring*, vol. 18, no. 5-6, pp. 1355–1371, 2018.
- [14] Y. Yu, X. Liu, E.-Z. Wang, K. Fang, and L. Huang, "Dam safety evaluation based on multiple linear regression and numerical simulation," *Rock Mechanics and Rock Engineering*, vol. 51, no. 8, pp. 2451–2467, 2018.
- [15] D. Dua and C. Graff, *UCI Machine Learning Repository*, University of California, School of Information and Computer Science, Irvine, CA, 2019, <http://archive.ics.uci.edu/ml>.
- [16] V. K. Samparathi and H. K. Verma, "Outlier detection of data in wireless sensor networks using kernel density estimation," *International Journal of Computer Applications*, vol. 5, no. 6, pp. 28–32, 2010.
- [17] V. Nalina, A. S. Sandesh, R. D. Sequiera, and P. Jayarekha, "Cloud computing based vehicle tracking information system," in *2015 IEEE International Advance Computing Conference (IACC)*, vol. 2, Bangalore, India, 2011no. 1.
- [18] S. Misra, A. Singh, S. Chatterjee, and M. S. Obaidat, "Mils-Cloud: a sensor-cloud-based architecture for the integration of military tri-services operations and decision making," *IEEE Systems Journal*, vol. 10, no. 2, pp. 628–636, 2016.
- [19] C. Srimathi, S. Park, and N. Rajesh, "Proposed framework for underwater sensor cloud for environmental monitoring," in *2013 Fifth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 104–109, Da Nang, Vietnam, 2013.
- [20] H. Paulheim and R. Meusel, "A decomposition of the outlier detection problem into a set of supervised learning problems," *Machine Learning*, vol. 100, no. 2-3, pp. 509–531, 2015.
- [21] Q. Zhang, C. Zhou, N. Xiong, Y. Qin, X. Li, and S. Huang, "Multimodel-based incident prediction and risk assessment in dynamic cybersecurity protection for industrial control systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 10, pp. 1429–1444, 2016.
- [22] K. Huang, Q. Zhang, C. Zhou, N. Xiong, and Y. Qin, "An efficient intrusion detection approach for visual sensor networks based on traffic pattern learning," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 10, pp. 2704–2713, 2017.
- [23] W. Wu, N. Xiong, and C. Wu, "Improved clustering algorithm based on energy consumption in wireless sensor networks," *IET Networks*, vol. 6, no. 3, pp. 47–53, 2017.
- [24] O. Ghorbel, W. Ayedi, H. Snoussi, and M. Abid, "Fast and efficient outlier detection method in wireless sensor networks," *IEEE Sensors Journal*, vol. 15, no. 6, pp. 3403–3411, 2015.
- [25] Y. Zhang, N. Meratnia, and P. J. M. Havinga, "Distributed online outlier detection in wireless sensor networks using ellipsoidal support vector machine," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1062–1074, 2013.
- [26] F. Eljorje, S. Kim, and J. Lee, "A wind turbine fault detection approach based on cluster analysis and frequent pattern mining," *KSII Transactions on Internet and Information Systems*, vol. 8, no. 2, pp. 664–677, 2014.
- [27] I. A. Klampanos, A. Davvetas, S. Andronopoulos, C. Pappas, A. Ikonomopoulos, and V. Karkaletsis, "Autoencoder-driven weather clustering for source estimation during nuclear events," *Environmental Modelling & Software*, vol. 102, pp. 84–93, 2018.
- [28] K. G. Dizaji, A. Herandi, C. Deng, W. Cai, and H. Huang, "Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization," in *2017 IEEE*

- International Conference on Computer Vision (ICCV)*, pp. 1–13, Venice, Italy, 2017.
- [29] K. Ø. Mikalsen, F. M. Bianchi, C. Soguero-Ruiz, and R. Jenssen, “Time series cluster kernel for learning similarities between multivariate time series with missing data,” *Pattern Recognition*, vol. 76, pp. 569–581, 2018.
- [30] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, “Deep clustering: discriminative embeddings for segmentation and separation,” in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 31–35, Shanghai, China, 2016.
- [31] J. W. Branch, C. Giannella, B. Szymanski, R. Wolff, and H. Kargupta, “In-network outlier detection in wireless sensor networks,” *Knowledge and Information Systems*, vol. 34, no. 1, pp. 23–54, 2013.
- [32] H. Su, J. Ren, and Z. Wen, “An approach using Dempster–Shafer evidence theory to fuse multi-source observations for dam safety estimation,” *Soft Computing*, vol. 23, no. 14, pp. 5633–5644, 2019.
- [33] B. Wei, D. Yuan, H. Li, and Z. Xu, “Combination forecast model for concrete dam displacement considering residual correction,” *Structural Health Monitoring*, vol. 18, no. 1, pp. 232–244, 2019.
- [34] D. Dua and C. Graff, *UCI Machine Learning Repository*, 2019, <http://archive.ics.uci.edu/ml/index.php>.