

Incipient Fault Detection in Industrial Processes Using Sliding Window K-Means Shared Dictionary Learning

1st Shuchen Shen
School of Zhangjian
Nantong University
Nantong, China
0009-0009-2412-8360

2nd Shu Chen
School of Zhangjian
Nantong University
Nantong, China
0009-0008-2702-1158

3rd Liangliang Shang*
School of Electrical Engineering and Automation
Nantong University
Nantong, China
0000-0002-2969-2317

4th Chenxi Zhu
School of Zhangjian
Nantong University
Nantong, China
0009-0008-2467-1411

5th Gaosheng Hu
School of Zhangjian
Nantong University
Nantong, China
0009-0000-5317-8775

Abstract—Incipient faults characterized by low magnitude and slow evolution are often undetectable by conventional detection methods in their early stages. If left unaddressed, these subtle anomalies could escalate into serious safety incidents. Therefore, achieving early and sensitive detection of incipient faults is critical for ensuring safe industrial operations. This paper proposes an incipient fault detection method that integrates sliding window-based feature extraction with a shared dictionary constructed via K-means clustering. The core steps of the approach are as follows: First, multivariate time-series data are collected; Next, multiple statistical features are computed within overlapping sliding windows; Then, a shared dictionary is learned from all normal historical data using K-means clustering; Subsequently, the Euclidean distance between each new window and its nearest atom in the dictionary is used as the reconstruction error to measure the severity of anomalies; Finally, a control limit is derived to make fault detection decisions. The proposed approach is evaluated on the Tennessee Eastman (TE) process. Experimental results show that, under comparable experimental settings, the method achieves higher fault detection rates and lower false alarm rates than conventional approaches such as k-Nearest Neighbors (KNN) and Principal Component Analysis (PCA). Moreover, the method is computationally lightweight, easy to deploy online, and demonstrates strong robustness across varying operating conditions.

Index Terms—Incipient fault detection, K-means clustering, shared dictionary, reconstruction error, Principal Component Analysis

I. INTRODUCTION

Incipient faults are low-magnitude, slow-evolving anomalies typically obscured by process noise, leading to missed early detection. Unlike abrupt faults, their small magnitude makes them hard to identify and they can escalate into serious failures if ignored. With growing process complexity, reliable incipient fault detection is essential for ensuring safety and efficiency in industrial processes. The development of incipient fault detection methodologies has evolved significantly over the

past decades. Traditional univariate statistical process control techniques were initially applied but struggle with high dimensionality and complex correlations. Multivariate statistical process monitoring (MSPM) approaches emerged [1], [2], with Principal Component Analysis (PCA) and Partial Least Squares (PLS) becoming cornerstone techniques. Extensions include Dynamic PCA (DPCA) for temporal correlations [3], Kernel PCA (KPCA) for nonlinear relationships [4], and mixture models for multi-modal processes [5]. Recently, deep learning techniques have shown promise in capturing complex patterns [6].

Beyond traditional statistical methods, several advanced approaches have been explored for incipient fault detection. Independent Component Analysis (ICA) has been applied to capture non-Gaussian features in process data [7], while Slow Feature Analysis (SFA) extracts slowly varying features to detect gradual changes [8]. Shang *et al.* [9] proposed a Recursive Ensemble Canonical Variate Analysis (RECVA) method for incipient fault detection in dynamic processes. Meanwhile, Tang *et al.* [10] proposed a Hybrid Divergence (HD) method that effectively mitigates the adverse impact of abnormal data points on detection performance. However, these advanced methods often suffer from high computational complexity, require extensive hyperparameter tuning, or lack interpretability in practical industrial deployment scenarios.

Inspired by the success of dictionary learning in signal processing and pattern recognition [11], and particularly motivated by the work of Li *et al.* Building on the dictionary learning framework for fault diagnosis introduced in [12], a novel approach is developed that integrates sliding window feature extraction with a shared dictionary constructed through K-means clustering. The approach is built upon the vector quantization paradigm but tailored specifically for incipient fault detection in industrial processes subject to distribution

shifts. Unlike traditional dictionary learning methods that rely on complex optimization algorithms, K-means clustering [13] is employed for its simplicity, efficiency, and robustness in learning representative patterns from normal operation data.

The key components of the approach are: a sliding window feature extraction strategy that captures local temporal dynamics; a shared dictionary learning scheme that learns from all normal historical data across different operating phases, classifies the data, and forms a dictionary; a minimum distance-based reconstruction error metric that effectively quantifies deviations from normal patterns. Compared to existing approaches, the method offers a better balance between detection sensitivity, false alarm rate, computational efficiency, and robustness to distribution shifts.

The remainder of this paper is organized as follows: Section II presents the proposed fault detection methodology based on sliding window features and shared dictionary learning. Section III describes the experimental setup using the Tennessee Eastman process benchmark. Finally, Section IV concludes the paper and discusses potential future work.

II. FAULT DETECTION BASED ON SHARED DICTIONARY LEARNING

A. Sliding Window Feature Construction

Let $\{\mathbf{x}_t\}_{t=1}^N \in \mathbb{R}^m$ denote an m -dimensional time series. Given a window length w and a hop size s , the i -th sliding window starts at global time index $t_i = 1 + (i - 1)s$ and consists of w consecutive samples

$$\mathbf{X}_i = [\mathbf{x}_{t_i}, \mathbf{x}_{t_i+1}, \dots, \mathbf{x}_{t_i+w-1}] \in \mathbb{R}^{m \times w}, \quad (1)$$

where $i = 1, 2, \dots, Y$. The total number of windows is

$$Y = \frac{N - w}{s} + 1, \quad (2)$$

assuming $(N - w)$ is divisible by s . For the j -th variable in the i -th window, the sample mean and standard deviation are computed. Let $x_{t,j}$ denote the value of the j -th variable at global time t . The mean is

$$\mu_j^{(i)} = \frac{1}{w} \sum_{t=t_i}^{t_i+w-1} x_{t,j}, \quad (3)$$

and the standard deviation is

$$\sigma_j^{(i)} = \sqrt{\frac{1}{w} \sum_{t=t_i}^{t_i+w-1} (x_{t,j} - \mu_j^{(i)})^2}. \quad (4)$$

Stacking the means and standard deviations of all m variables yields the raw feature vector for the i -th window

$$\mathbf{f}_i = (\boldsymbol{\mu}^{(i)\top}, \boldsymbol{\sigma}^{(i)\top})^\top \in \mathbb{R}^{2m}, \quad (5)$$

where $\boldsymbol{\mu}^{(i)} = [\mu_1^{(i)}, \dots, \mu_m^{(i)}]^\top$ and $\boldsymbol{\sigma}^{(i)} = [\sigma_1^{(i)}, \dots, \sigma_m^{(i)}]^\top$.

To mitigate scale differences across features, \mathbf{f}_i is normalized using statistics computed from normal operating data. Let f_{ip} be the p -th component of \mathbf{f}_i , and let \bar{f}_p and s_p denote the

mean and standard deviation of the p -th feature over all normal training samples. The normalized feature vector is

$$\mathbf{z}_i = [z_{i1}, z_{i2}, \dots, z_{i2m}]^\top \in \mathbb{R}^{2m}, \quad (6)$$

with each element given by

$$z_{ip} = \frac{f_{ip} - \bar{f}_p}{s_p}, \quad p = 1, \dots, 2m. \quad (7)$$

The normalized features are then organized into dataset matrices by stacking them row-wise:

- Normal training set: $\mathbf{Z}_1 \in \mathbb{R}^{n_1 \times 2m}$
- Normal test set: $\mathbf{Z}_2 \in \mathbb{R}^{n_2 \times 2m}$
- Faulty test set: $\mathbf{Z}_3 \in \mathbb{R}^{n_3 \times 2m}$

where n_1 , n_2 , and n_3 are the numbers of windowed samples in each set. Specifically, the structure of \mathbf{Z}_1 is

$$\mathbf{Z}_1 = \begin{bmatrix} \mathbf{z}_1^\top \\ \mathbf{z}_2^\top \\ \vdots \\ \mathbf{z}_{n_1}^\top \end{bmatrix} \in \mathbb{R}^{n_1 \times 2m}, \quad (8)$$

To eliminate domain shift between training and testing phases, a shared dictionary learning dataset $\mathbf{Z}_{\text{normal}}$ is constructed by vertically concatenating the normal samples from both the training and test sets

$$\mathbf{Z}_{\text{normal}} = \begin{bmatrix} \mathbf{Z}_1 \\ \mathbf{Z}_2 \end{bmatrix} \in \mathbb{R}^{(n_1+n_2) \times 2m}. \quad (9)$$

This combined dataset is used to learn a shared dictionary that captures common patterns of normal operation across domains.

B. Shared Dictionary Learning via K-Means

In industrial incipient fault detection, distribution shifts commonly arise between different operational phases due to factors such as process condition variations, sensor drift, or environmental disturbances, which severely degrade the generalization capability of the model. To mitigate this issue, a shared dictionary is constructed by jointly leveraging all normal data from the initial phase, thereby achieving consistent alignment of normal behavior representations across stages.

Based on this, the K-means clustering is applied to the combined normal dataset $\mathbf{Z}_{\text{normal}}$ to learn a compact set of representative prototypes, which collectively constitute the shared dictionary \mathbf{D} .

Formally, K-means seeks to partition $\mathbf{Z}_{\text{normal}}$ into K disjoint clusters $\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_K\}$ by minimizing the within-cluster sum of squared Euclidean distances

$$\min_{\{\mathbf{d}_k\}_{k=1}^K} \sum_{k=1}^K \sum_{\mathbf{z}_i \in \mathcal{C}_k} \|\mathbf{z}_i - \mathbf{d}_k\|_2^2, \quad (10)$$

where $\mathbf{d}_k \in \mathbb{R}^{2m}$ denotes the centroid (also referred to as a dictionary atom) of cluster \mathcal{C}_k , and K is a pre-specified hyperparameter controlling the dictionary size.

The algorithm proceeds iteratively as follows:

- 1) **Initialization:** Randomly select K distinct samples from $\mathbf{Z}_{\text{normal}}$ as initial centroids $\{\mathbf{d}_k^{(0)}\}_{k=1}^K$. The random seed is fixed to ensure reproducibility.
- 2) **Assignment step:** For each sample \mathbf{z}_i , compute its Euclidean distance to all centroids and assign it to the cluster whose centroid yields the smallest distance. This process can be expressed as

$$k_i = \arg \min_{k \in \{1, \dots, K\}} \|\mathbf{z}_i - \mathbf{d}_k\|_2, \quad (11)$$

where k_i denotes the index of the assigned cluster, and $\arg \min$ returns the value of k that minimizes the distance.

- 3) **Update step:** Recompute each centroid as the mean of all samples assigned to its cluster

$$\mathbf{d}_k = \frac{1}{|C_k|} \sum_{\mathbf{z}_i \in C_k} \mathbf{z}_i, \quad (12)$$

where $|C_k|$ denotes the cardinality of cluster C_k .

- 4) **Convergence check:** Repeat steps 2 and 3 iteratively until one of the following termination conditions is met: the maximum number of iterations is reached; all cluster centroids remain unchanged; or the cluster assignments of all samples stabilize. Upon convergence, the resulting centroids are taken as the final dictionary atoms.

The K-means clustering process is illustrated in Fig. 1, which shows the iterative convergence from random initialization to the final stable dictionary atoms in a 2D feature space.

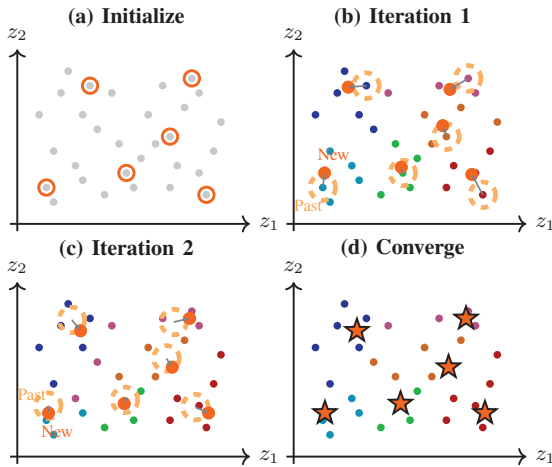


Fig. 1. Illustration of the Shared Dictionary Learning Algorithm in 2D Space.

To avoid suboptimal local minima, the K-means is executed with multiple random initializations, and the solution yielding the smallest final value of the objective function (10) is

retained. The resulting K centroids are arranged as rows to form the shared dictionary matrix:

$$\mathbf{D} = \begin{bmatrix} \mathbf{d}_1^T \\ \mathbf{d}_2^T \\ \vdots \\ \mathbf{d}_K^T \end{bmatrix} \in \mathbb{R}^{K \times 2m}. \quad (13)$$

In this formulation, each row of \mathbf{D} corresponds to a dictionary atom, capturing a typical pattern of normal system behavior. This shared dictionary strategy effectively integrates normal information across operational phases and significantly enhances the robustness and generalization capability of the detector under real-world distribution shifts.

C. Control Limit Determination

After the shared dictionary \mathbf{D} is constructed, a statistical threshold is required to distinguish between normal and anomalous conditions. Since the dictionary is learned solely from normal data, the reconstruction errors of normal samples are expected to concentrate at low values. To this end, the reconstruction errors are computed using all normal samples, and the control limit is determined based on their empirical distribution.

Specifically, for each normal sample \mathbf{z}_i , its reconstruction error is defined as the minimum Euclidean distance to all dictionary atoms in the shared dictionary

$$e_i = \min_{k=1, \dots, K} \|\mathbf{z}_i - \mathbf{d}_k\|_2, \quad (14)$$

where \mathbf{d}_k is the k -th dictionary atom, e_i is the reconstruction error.

Let $\{e_1, e_2, \dots, e_n\}$ denote the set of reconstruction errors from all normal samples, where n is the total number of normal samples. The control limit η is determined according to a pre-specified significance level α . During online detection, if the reconstruction error of a test sample exceeds η , it is classified as a faulty sample; otherwise, it is regarded as normal.

D. Offline Training and Online Detection

The proposed method consists of two phases: an offline training and an online detection. An offline shared dictionary model is constructed using normal operating data, and the proposed detection method is validated on a test dataset containing incipient faults. The detailed procedure is as follows:

Offline Training:

- Collect all normal data from the initial operating period of the system.
- Extract feature vectors from the raw multivariate time series using a sliding window, as described in (1)–(7).
- Apply uniform normalization to all feature vectors to obtain the standardized normal dataset $\mathbf{Z}_{\text{normal}}$, as per (8) and (9).
- Learn the shared dictionary $\mathbf{D} \in \mathbb{R}^{K \times 2m}$ by applying the K-means clustering algorithm on $\mathbf{Z}_{\text{normal}}$, as detailed in (10)–(13).

- Determine the control limit η based on the empirical distribution of reconstruction errors from normal samples at a pre-specified significance level α using (14).
- The model parameters (\mathbf{D} and η) are fixed and not updated thereafter.

Online Detection:

- Acquire real-time observations and generate the test feature vector \mathbf{z}_t using the same window length, feature type, and normalization parameters as in the offline phase in (1)–(7).
- Compute its reconstruction error e_t using (14).
- If $e_t > \eta$, classify the sample as faulty and trigger an alarm; otherwise, it is regarded as a normal sample.

III. APPLICATION TO THE TENNESSEE EASTMAN PROCESS

A. Tennessee Eastman Process Description

The Tennessee Eastman (TE) chemical process simulation platform is employed to generate datasets with various fault types for validating the effectiveness and performance of the proposed algorithm. The TE process is capable of simulating nonlinear, non-Gaussian, time-varying, and multi-modal characteristics, providing a standard benchmark for evaluating process modeling, control, fault detection, and diagnosis methods.

The TE process includes 41 measured variables and 12 manipulated variables, totaling 53 variables (often treated as 52 in practice due to one manipulated variable being fixed). Each measured variable is corrupted by additive Gaussian noise to simulate realistic industrial conditions. The sampling intervals vary: most variables are sampled every 3 minutes, 14 variables every 6 minutes, and 5 variables every 15 minutes. The process exhibits multiple fault types including step changes, random variations, slow drift, sticking, and constant position faults.

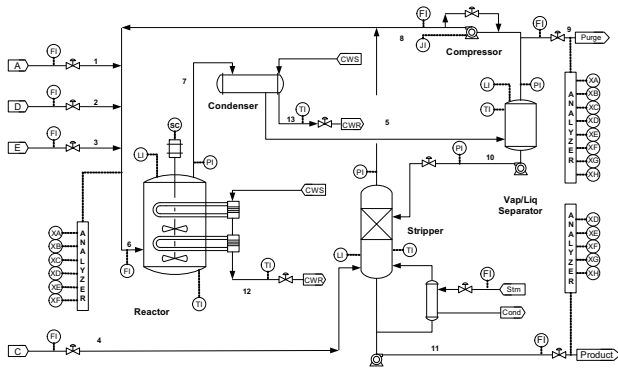


Fig. 2. Diagram of the Tennessee Eastman (TE) process.

B. Settings Parameters

The default parameters are: window length $w=20$, hop size $s=1$, shared dictionary size $K=60$. KNN anomaly detection method uses $k_{nn}=60$ with the average distance to the k_{nn} nearest neighbors as the anomaly score. PCA retains principal components accounting for 90% cumulative variance and

monitors the T^2 statistic. All control limits are set from normal validation data at the 99% quantile.

The fault detection rate (FDR) and false alarm rate (FAR) are defined as in [14].

The key hyperparameters, including the window length, hop size, and dictionary size, are selected based on an analysis of the dynamic characteristics of the TE process and empirical validation. Considering the sampling rate and fault evolution timescale of the TE process, the window length is set to 20, which effectively captures local steady-state statistical features while suppressing high-frequency noise. The hop size is set to 1 to preserve temporal resolution, thereby facilitating early fault detection. The dictionary size K is determined via grid search over the candidate set $K \in \{20, 40, 60, 80, 100\}$. Among these values, $K = 60$ achieves the best trade-off between FDR and FAR, adequately representing the underlying structure of normal operating conditions without introducing overfitting or degrading generalization performance.

C. Case Studies on TE Faults 3, 9, and 15

The proposed method effectively detects three types of incipient faults, significantly outperforming KNN and PCA in both detection rate and false alarm rate.

1) *Fault 3*: Fault 3 is an incipient step change in D feed temperature, presenting detection challenges due to its small magnitude, slow propagation, and control system compensation.

Fig. 3 shows that the proposed method achieves effective detection. First, sliding window feature extraction transforms point-wise analysis into temporal pattern analysis. This temporal aggregation amplifies weak fault signatures invisible to point-wise methods. Second, K-means clustering dictionary atoms provides multi-center representation of normal behavior, fundamentally different from PCA's single-center model. The reconstruction error is more sensitive than PCA's projection because the fault simply needs to differ from all normal patterns, regardless of deviation direction.

As shown in Fig. 4, PCA's catastrophic failure stems from two fundamental limitations. First, its single linear subspace cannot capture the nonlinear thermal dynamics. Second, PCA analyzes instantaneous samples independently, lacking sensitivity. Fig. 5 demonstrates that KNN's highly false alarm rate results from computing distances to individual training samples rather than learned prototypes.

2) *Fault 9*: Fault 9 introduces random variations in D feed temperature, making it difficult to distinguish from process noise.

As demonstrated in Fig. 6, while individual random samples are indistinguishable from noise, the statistical characteristics of fault-induced randomness differ from natural process noise over short time windows. The K-means dictionary atoms capture different patterns of normal variability. The fault-induced variance structure consistently differs from all learned normal patterns, causing systematic increases in reconstruction error.

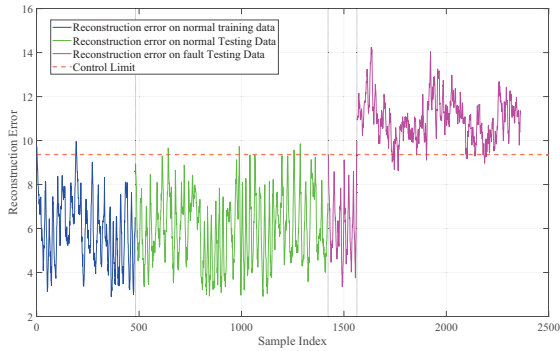


Fig. 3. Monitoring result of the proposed method for TE Fault 3.

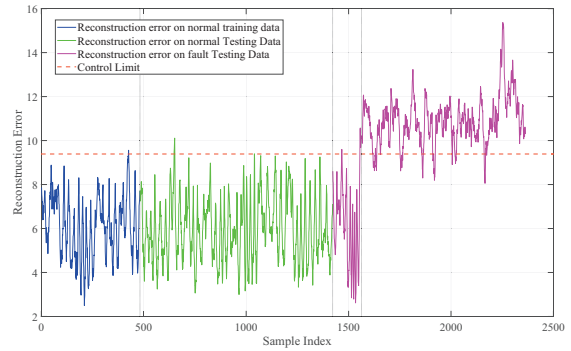


Fig. 6. Monitoring result of the proposed method for TE Fault 9.

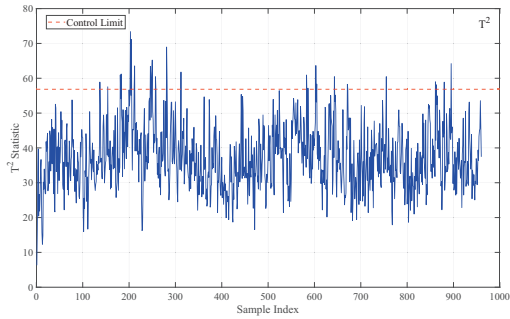


Fig. 4. PCA T^2 statistic monitoring chart for TE Fault 3.

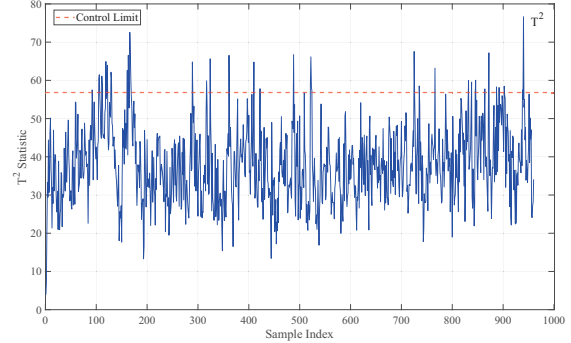


Fig. 7. PCA T^2 statistic monitoring chart for TE Fault 9.

As illustrated in Fig. 7, because the random disturbances propagate in different directions at each time point, they do not create consistent deviations in the PCA subspace—the random effects average out, making the fault statistically indistinguishable from normal noise. Fig. 8 shows that KNN cannot distinguish between normal process noise and fault-induced randomness. Its distance-based metric fluctuates wildly as random disturbances vary in magnitude.

3) *Fault 15*: Fault 15 involves condenser cooling water valve sticking, creating complex cascading dynamics with intermittent manifestation and control compensation.

Fig. 9 demonstrates superior performance because the dictionary atoms capture diverse normal operating patterns—each atom represents a distinct normal pattern. Normal mode transitions move from one atom to another with low reconstruction error, while fault-induced deviations move away from all atoms with high reconstruction error. When Fault 15 disrupts these patterns through control compensation, the abnormal variable combinations do not match any learned normal pattern, enabling detection even when individual deviations remain small.

As evident from Fig. 10, PCA is unable to differenti-

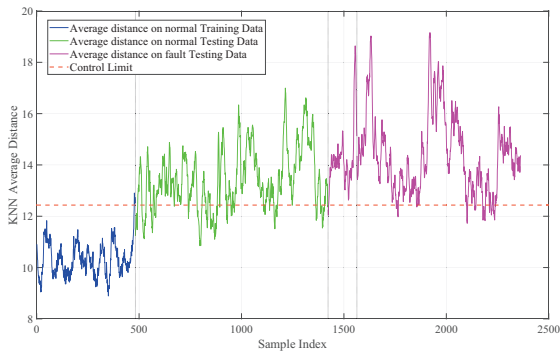


Fig. 5. Monitoring result of KNN anomaly detection for TE Fault 3.

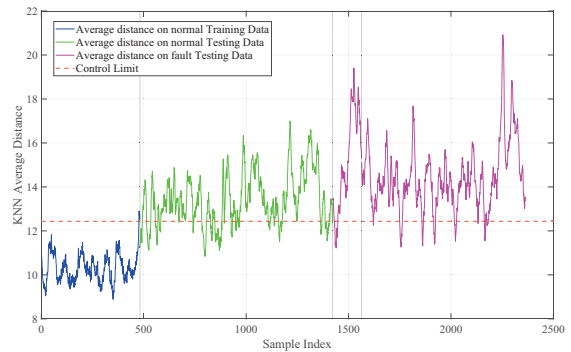


Fig. 8. Monitoring result of KNN anomaly detection for TE Fault 9.

ate among distinct normal operating modes. Fig. 11 reveals that KNN's low detection and high false alarm rates stem from instance-based distance computation in high-dimensional space.

The quantitative performance comparison is summarized in Table I.

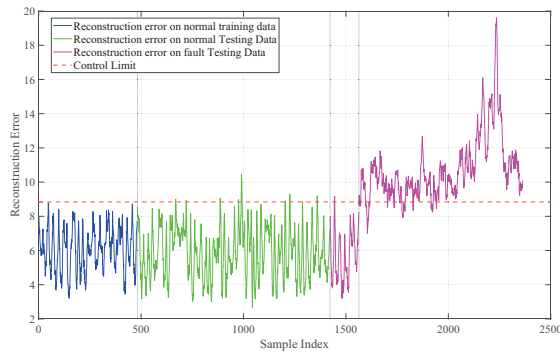


Fig. 9. Monitoring result of the proposed method for TE Fault 15.

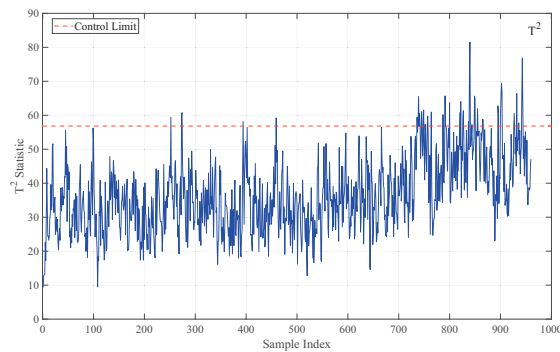


Fig. 10. PCA T^2 statistic monitoring chart for TE Fault 15.

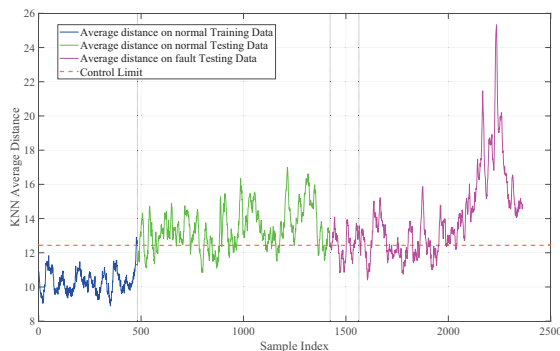


Fig. 11. Monitoring result of KNN anomaly detection for TE Fault 15.

IV. CONCLUSION

This paper proposes a lightweight effective incipient fault detection method for industrial processes by integrating sliding

TABLE I
FAULT DETECTION RATE (FDR, %) AND FALSE ALARM RATE (FAR, %) ON TE FAULTS.

Case Fault	Shared dict.		KNN		PCA	
	FDR	FAR	FDR	FAR	FDR	FAR
Fault 3	96.50	0.80	93.25	81.24	3.12	1.25
Fault 9	91.90	1.40	93.00	80.04	3.62	6.25
Fault 15	92.10	1.50	72.38	75.69	5.50	0.00

window feature extraction with a K-means based shared dictionary. The approach effectively mitigates domain shift and captures different normal operating modes. The reconstruction error to the nearest dictionary atom serves as a sensitive anomaly score, and the control limit derived from empirical quantiles ensures robust decision-making without distributional assumptions. Extensive experiments on the Tennessee Eastman process demonstrate that the proposed method significantly outperforms conventional KNN and PCA in detecting incipient faults, achieving high detection rates and low false alarm rates. The model is computationally efficient, does not need online retraining, and can be easily deployed in real-time monitoring systems.

REFERENCES

- [1] S. J. Qin, "Survey on data-driven industrial process monitoring and diagnosis," *Annu. Rev. Control*, vol. 36, no. 2, pp. 220–234, 2012.
- [2] S. Yin, X. Li, H. Gao, and O. Kaynak, "Data-based techniques focused on modern industry: an overview," *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 657–667, Jan. 2015.
- [3] W. Ku, R. H. Storer, and C. Georgakis, "Disturbance detection and isolation by dynamic principal component analysis," *Chemom. Intell. Lab. Syst.*, vol. 30, no. 1, pp. 179–196, 1995.
- [4] J.-M. Lee, C. Yoo, S. W. Choi, P. A. Vanrolleghem, and I.-B. Lee, "Non-linear process monitoring using kernel principal component analysis," *Chem. Eng. Sci.*, vol. 59, no. 1, pp. 223–234, 2004.
- [5] Q. Jiang and X. Yan, "Multimode process monitoring using variational Bayesian inference and canonical correlation analysis," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 4, pp. 1814–1824, 2019.
- [6] L. Shang, R. Fang, J. Liu, Y. Jing, C. Wan, and A. Qiu, "PSFCL: A probabilistic slow feature contrastive learning approach for incipient fault diagnosis in industrial processes," *Computers & Chemical Engineering*, p. 109584, 2026.
- [7] J.-M. Lee, C. Yoo, and I.-B. Lee, "Statistical process monitoring with independent component analysis," *Journal of Process Control*, vol. 14, no. 5, pp. 467–485, 2004.
- [8] Z. Zhang and D. Tao, "Slow feature analysis for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 436–450, Mar. 2012.
- [9] L. Shang, Y. Gu, Y. Tang, H. Fu, and L. Hua, "Recursive ensemble canonical variate analysis for online incipient fault detection in dynamic processes," *Measurement*, vol. 220, p. 113411, Oct. 2023.
- [10] Y. Tang, L. Shang, R. Zhang, J. Li, and H. Fu, "Hybrid divergence based on mean absolute scaled error for incipient fault detection," *Eng. Appl. Artif. Intell.*, vol. 129, p. 107662, Mar. 2024.
- [11] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
- [12] Y. Li, Y. Chai, and H. Yin, "Autoencoder embedded dictionary learning for nonlinear industrial process fault diagnosis," *J. Process Control*, vol. 101, pp. 24–34, 2021.
- [13] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probab.*, 1967, pp. 281–297.
- [14] L. Shang, A. Qiu, P. Xu, and F. Yu, "Canonical variate nonlinear principal component analysis for monitoring nonlinear dynamic processes," *J. Chem. Eng. Jpn.*, vol. 55, no. 1, pp. 29–37, 2022.