Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Dual-channel temporal convolutional autoencoder: Multi-scale representation with short-term fluctuation and long-term trend for dynamic process monitoring

Jian Huang
University of Science and Technology Beijing
Xin Pu
University of Science and Technology Beijing
Xu Yang
yangxu@ustb.edu.cn

University of Science and Technology Beijing
Kaixiang Peng
University of Science and Technology Beijing
Jingjing Gao
University of Science and Technology Beijing
Dajian Huang
University of Science and Technology Beijing

Research Article

Keywords: Dynamic process monitoring, multi-scale temporal representation, temporal convolutional autoencoder, short-term fluctuation, long-term trend

Posted Date: November 10th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11350376/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Dual-channel temporal convolutional autoencoder: Multi-scale representation with short-term fluctuation and long-term trend for dynamic process monitoring

Jian Huang[1,2], Xin Pu[1], Xu Yang[1,2*], Kaixiang Peng[1], Jingjing Gao[1,2], Dajian Huang[1]

[1]Key Laboratory of Knowledge Automation for Industrial Processes of Ministry of Education, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing, China.
[2]Shunde Innovation School, University of Science and Technology Beijing, Foshan, China.

*Corresponding author(s). E-mail(s): yangxu@ustb.edu.cn;

Abstract

Traditional autoencoder-based monitoring methods are constrained by the assumption of static data. This limitation prevents them from effectively capturing the inherent temporal autocorrelation and multi-scale dynamics, consequently leading to degraded monitoring performance. To solve this problem, a dual-channel temporal convolutional autoencoder (DTCAE) is proposed to effectively characterize multi-scale temporal dynamics. Specifically, two parallel temporal convolutional networks with distinct receptive fields are employed to separately extract short-term fluctuation and long-term trend features from the latent representation. This dual-channel representation reduces the interference between short-term fluctuations and long-term trends, enabling more discriminative temporal representations. Subsequently, the original data are reconstructed by combining both features, preserving complementary temporal information for accurate reconstruction. Finally, the monitoring statistics for extracted features are integrated by the Bayesian inference criterion, enabling a comprehensive assessment of process status. The proposed method is then validated on the penicillin fermentation process and the wastewater treatment process.

Keywords: Dynamic process monitoring, multi-scale temporal representation, temporal convolutional autoencoder, short-term fluctuation, long-term trend

1 Introduction

As modern chemical processes become increasingly complex, reliable fault detection is important for identifying abnormal operation (de Brito et al. 2026). From a process safety perspective, undetected faults and abnormal operating conditions can propagate through interconnected process units, potentially resulting in equipment failures, hazardous material releases, environmental emissions, and severe safety consequences (Santos et al. 2014). Therefore, reliable process monitoring plays a vital role in supporting the production activities of an enterprise (Wang et al. 2020; Khan et al. 2015). Classical statistical process monitoring established a data-driven foundation for fault detection and diagnosis (Qin 2003). To address more complex, time-varying behavior, newer data-driven monitoring models have been developed (Huang et al. 2025). Principal component analysis (PCA) (Amin et al. 2021) and slow feature analysis (SFA), as early data-driven methods, were developed based on linear assumptions and are inadequate for capturing the complex nonlinear dynamics prevalent in modern industrial processes (Li et al. 2025; Wang et al. 2020).

To address nonlinear problems, autoencoders (AEs) and their deep learning variants offer significant advantages in capturing complex nonlinear relationships within high-dimensional process data. The encoder maps raw inputs into a compressed latent space, while the decoder reconstructs the original data through inverse transformations. The inherent reconstruction mechanism of AEs enables them to capture the essential nonlinear features of normal samples. For instance, Lee et al. developed a variational autoencoder (VAE) for high-dimensional nonlinear processes (2019), while Li et al. proposed a distributed-ensemble stacked autoencoder (DE-SAE) to enhance nonlinear feature extraction (2021). These methods achieve effective nonlinear feature extraction through latent representation learning. However, in dynamic industrial processes where variables exhibit strong temporal coupling, conventional AE-based methods ignore these temporal dependencies during feature compression, significantly limiting their capability to capture dynamic features for effective process monitoring.

In dynamic industrial systems, process variables exhibit highly nonlinear interactions and temporal coupling (Pota et al. 2023). Traditional static monitoring methods often assume statistical independence between current and historical data, which neglects inherent temporal dependencies in time-series data. Recent efforts have sought to incorporate temporal dynamics (Jiang et al. 2018; Chen and Liu 2002). An attention mechanism was integrated into a long short-term memory (LSTM) architecture by Sun et al. to facilitate fuel ethanol fermentation monitoring (2023). Their model integrates static and dynamic variables to predict ethanol concentration. Tang et al. developed a dual-attention bidirectional generative adversarial network (BiGAN) to monitor time-varying process variables under dynamic uncertainty, using a self-attention mechanism to capture evolving dependencies (2023). Zeng et al. combined LSTMs with dual-attention modules in an autoencoder framework, achieving enhanced fault detection through dynamic nonlinear feature extraction (2024). Despite integrating temporal modeling and attention mechanisms, these methods still face difficulties in adequately characterizing multi-scale temporal dependencies among process variables. Time series may contain trend and fluctuating components at different temporal scales (Wu et al. 2021). In dynamic process monitoring, representing both types of

variation is therefore relevant to distinguishing normal evolution from fault-induced changes. Specifically, the normal operation of dynamic processes is characterized by multi-scale temporal coupling, where process variables interact across different time horizons and maintain a dynamic equilibrium. Fault occurrences may disturb this equilibrium by causing abnormal short-term fluctuations or long-term trend changes. Consequently, effective fault detection requires monitoring representations that can characterize both short-term fluctuations and long-term trends.

Time series decomposition methods have been widely used to extract distinct temporal patterns and reveal the underlying temporal dynamics of process data. Cleveland et al. decomposed the time series into seasonal, trend, and residual terms by locally weighted regression smoothing (Loess) (1990). On this basis, Wen et al. used least absolute deviations loss with sparse regularization to robustly extract the trend component (2019). In addition, different temporal components can also be extracted using variational mode decomposition (VMD) (Dragomiretskiy and Zosso 2014), discrete wavelet transform (DWT) (Mallat 1989), and singular spectrum analysis (SSA) (Azma et al. 2026). While these methods are effective in extracting multi-scale features, their capability to model nonlinear relationships in high-dimensional process data remains limited. To overcome these limitations, hybrid approaches that combine traditional decomposition methods with deep learning have been increasingly explored (Yang et al. 2021). Bedi et al. proposed a hybrid framework that integrates VMD with autoencoder architectures for electric load forecasting (2020). Yang et al. proposed a multi-step-ahead model (EMD-SAE) combining empirical mode decomposition with a stacked autoencoder, which reduces overfitting and improves temporal prediction accuracy (2019). These hybrid autoencoder-based methods both address nonlinear complexity and effectively capture the system's temporal dynamics. However, traditional decomposition methods depend on manually tuned parameters, which may compromise the effectiveness of hybrid models. Moreover, the computational burden of both decomposition techniques and deep autoencoder training makes these hybrid methods computationally expensive and difficult to optimize in an end-to-end manner. Therefore, it remains necessary to develop a unified deep learning framework that can capture nonlinear relationships while learning short-term fluctuation and long-term trend representations directly from process data.

Motivated by the above analysis, this paper seeks to develop an end-to-end deep learning framework that can simultaneously capture high-dimensional nonlinear relationships and learn multi-scale temporal features from process data. In the context of modeling temporal dependencies, temporal convolutional networks (TCNs) (Hewage et al. 2020; Abu Farha and Gall 2019) have received widespread attention for their capacity to process information across varying receptive fields. By stacking convolutional layers with different dilation rates, TCNs can model temporal features at multiple scales, ranging from local dynamic patterns to long-range evolving dependencies. Building on this capability, this paper proposes a dual-channel temporal convolutional autoencoder (DTCAE) with short-term fluctuation and long-term trend representation for dynamic process monitoring. First, an autoencoder is used to obtain latent features from the raw data, which reduces redundant information and noise. Then, two parallel temporal convolutional networks are employed to learn

complementary short-term fluctuation and long-term trend features from the latent representation. The fixed-dilation branch is designed to characterize local short-term fluctuations under a stable receptive field, whereas the progressively enlarged dilation branch is used to model long-term trends through hierarchical receptive-field expansion. This parallel structure provides complementary temporal representations for process dynamics at different time horizons, thereby enhancing the monitoring capability for dynamic operating conditions. Finally, the Bayesian Inference Criterion (BIC) is used to integrate the statistical information from both the extracted temporal features and the reconstructed features. The efficacy of DTCAE is validated on the penicillin fermentation process and the wastewater treatment process. The main contributions of this paper are as follows:

Short-term fluctuation and long-term trend features are obtained by using parallel TCN branches with different dilation strategies. Fixed and enlarged dilation rates of TCN are designed to describe the short-term fluctuations and long-term trends of normal samples, respectively, which explicitly capture multi-scale temporal dynamics.

By preserving the temporal dependencies, the learned multi-scale features uncover the intrinsic patterns of process variations. On this basis, a comprehensive monitoring strategy is developed by constructing separate statistics for short-term fluctuation and long-term trend features, which are then integrated through BIC, enabling more sensitive and reliable fault detection across different temporal scales.

2 Autoencoder

Autoencoders are neural networks trained to reconstruct their inputs and can learn compact, nonlinear representations for dimensionality reduction (Hinton and Salakhutdinov 2006). As depicted in Fig. 1, this architecture integrates two key modules: an encoder and a decoder. The encoding module transforms input data into a compact latent representation through hierarchical neural operations, while the decoding module performs the inverse transformation to reconstruct the input. Through this compression and reconstruction process, the model learns to extract salient structural features from the data.

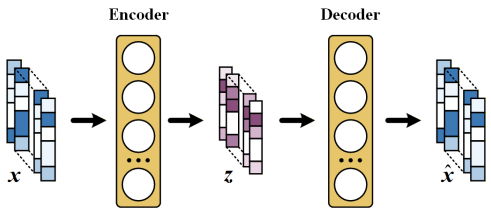


Fig. 1 Architecture of autoencoder network

Consider an input data matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{m \times n}$, where m denotes the data dimensionality and n indicates the sample size. The encoder transforms an input $\boldsymbol{x} \in \mathbb{R}^m$ into a latent representation $\boldsymbol{z} \in \mathbb{R}^k$, and the decoder maps $\boldsymbol{z}$ back to a reconstruction $\hat{\boldsymbol{x}} \in \mathbb{R}^m$, where $k < m$. The encoder and decoder have

independent parameters and need not use the same activation at their output layers. Their mathematical forms are:

$$\boldsymbol{z} = f_{\theta_{\text{enc}}}(\boldsymbol{x}) = \phi_{\text{enc}}\left(\boldsymbol{W}_{\text{enc}}\boldsymbol{x} + \boldsymbol{b}_{\text{enc}}\right) \tag{1}$$

$$\hat{\boldsymbol{x}} = g_{\theta_{\text{dec}}}(\boldsymbol{z}) = \phi_{\text{dec}}\left(\boldsymbol{W}_{\text{dec}}\boldsymbol{z} + \boldsymbol{b}_{\text{dec}}\right) \tag{2}$$

where $f_{\theta_{\text{enc}}}(\cdot)$ and $g_{\theta_{\text{dec}}}(\cdot)$ denote the encoder and decoder, respectively. Their parameters are $\theta_{\text{enc}} = \{\boldsymbol{W}_{\text{enc}}, \boldsymbol{b}_{\text{enc}}\}$, with $\boldsymbol{W}_{\text{enc}} \in \mathbb{R}^{k\times m}$ and $\boldsymbol{b}_{\text{enc}} \in \mathbb{R}^{k}$, and $\theta_{\text{dec}} = \{\boldsymbol{W}_{\text{dec}}, \boldsymbol{b}_{\text{dec}}\}$, with $\boldsymbol{W}_{\text{dec}} \in \mathbb{R}^{m\times k}$ and $\boldsymbol{b}_{\text{dec}} \in \mathbb{R}^{m}$. The symbols ϕ_{enc} and ϕ_{dec} distinguish the layer-wise encoder and decoder mappings. In the implemented network, ReLU is used in the hidden layers of both modules, whereas the final decoder layer is linear because the standardized process variables are continuous and may take either sign. The autoencoder is trained to reduce reconstruction error by backpropagation. The reconstruction loss is formulated using the mean squared error (MSE):

$$Loss\left(\boldsymbol{X}, \hat{\boldsymbol{X}}\right) = \frac{1}{n}\sum_{i=1}^{n}\left\|\boldsymbol{x}_i - \hat{\boldsymbol{x}}_i\right\|^2 \tag{3}$$

3 Process monitoring based on dual-channel temporal convolutional autoencoder (DTCAE)

3.1 Short-term fluctuation and long-term trend representation based on TCN

TCN is a specialized convolutional architecture tailored for time series analysis. Stacked dilated causal convolutions are employed in TCN to capture both local and global temporal dependencies while preserving causal relationships in sequential data (Lea et al. 2017). TCN is built upon three core components, namely causal convolution, dilated convolution, and residual blocks, as demonstrated in Fig. 2. These components are organized in a hierarchical structure where time series data flow through stacked residual blocks to generate predictions. By limiting the receptive field at each time step to current and past inputs, causal convolution prevents information leakage from future time steps and maintains temporal causality. Dilated convolution achieves exponential receptive field expansion through introducing spacing in the convolutional kernel, allowing the network to efficiently capture multi-scale temporal patterns without increasing computational cost (Thill et al. 2021).

Consider a one-dimensional temporal sequence $\boldsymbol{l} \in \mathbb{R}^n$ of length denoted by n and a filter of size represented by $\boldsymbol{f} : \{0, \ldots, k_{conv} - 1\} \to \mathbb{R}$. For each element s in the sequence, the dilated causal convolution $\boldsymbol{F}$ is defined as:

$$F(s) = \left(\boldsymbol{l} *_d \boldsymbol{f}\right)(s) = \sum_{r=0}^{k_{conv}-1} f(r) \cdot l_{s-d\cdot r} \tag{4}$$

In Eq. (4), d is the dilation factor, k_{conv} is the kernel size, and $s - dr$ indexes only the present or past, thereby preserving causality. For the multivariate latent sequence

used by DTCAE, let $\boldsymbol{H}^{(\ell-1)} = [\boldsymbol{h}_1^{(\ell-1)}, \ldots, \boldsymbol{h}_n^{(\ell-1)}]^{\mathrm{T}} \in \mathbb{R}^{n \times c_{\ell-1}}$ be the input to residual block ℓ. The residual transform $\mathcal{F}^{(\ell)}$ contains two causal dilated convolutional layers, each followed by weight normalization, ReLU activation, and dropout. A 1×1 convolution projects the shortcut whenever the channel dimensions differ. The block output is

$$\boldsymbol{H}^{(\ell)} = \mathrm{ReLU}\Big(\mathcal{F}^{(\ell)}\Big(\boldsymbol{H}^{(\ell-1)}\Big) + \boldsymbol{H}^{(\ell-1)}\boldsymbol{W}_{\mathrm{res}}^{(\ell)\mathrm{T}}\Big), \tag{5}$$

where $\boldsymbol{H}^{(\ell)} \in \mathbb{R}^{n \times c_\ell}$ and $\boldsymbol{W}_{\mathrm{res}}^{(\ell)} \in \mathbb{R}^{c_\ell \times c_{\ell-1}}$. Thus, both terms inside the addition have dimension $n \times c_\ell$; when $c_\ell = c_{\ell-1}$, the shortcut may reduce to the identity mapping.

Given an encoded latent sequence $\boldsymbol{H}^{(0)} \in \mathbb{R}^{n \times k}$, the proposed dual-channel TCN learns complementary short-term fluctuation and long-term trend representations. The short-term branch fixes the dilation at one in every block, maintaining a local receptive field. The long-term branch uses $d_\ell = 2^{\ell-1}$ in block ℓ, so its receptive field expands hierarchically. For output channel c at time s, the two branch convolutions are defined as follows:

$$q_{s,c}^{(\ell)} = b_{q,c}^{(\ell)} + \sum_{r=0}^{k_{conv}-1} \sum_{j=1}^{c_{\ell-1}} W_{q,cjr}^{(\ell)} h_{s-r,j}^{(\ell-1)}, \tag{6}$$

$$p_{s,c}^{(\ell)} = b_{p,c}^{(\ell)} + \sum_{r=0}^{k_{conv}-1} \sum_{j=1}^{c_{\ell-1}} W_{p,cjr}^{(\ell)} h_{s-d_\ell r,j}^{(\ell-1)}, \qquad d_\ell = 2^{\ell-1}. \tag{7}$$

Here ℓ is the residual-block index; j and c index the input and output channels; $\boldsymbol{W}_q^{(\ell)}, \boldsymbol{W}_p^{(\ell)} \in \mathbb{R}^{c_\ell \times c_{\ell-1} \times k_{conv}}$ and $\boldsymbol{b}_q^{(\ell)}, \boldsymbol{b}_p^{(\ell)} \in \mathbb{R}^{c_\ell}$ are learnable kernels and biases. Samples with negative time indices are zero-padded. The convolution outputs are subsequently processed by weight normalization, ReLU, and dropout within the residual block.

Figure 2 illustrates this causal sampling pattern for $k_{conv} = 3$. At time s, dilation factors $d = 1$, 2, and 4 use the index sets $\{s, s-1, s-2\}$, $\{s, s-2, s-4\}$, and $\{s, s-4, s-8\}$, respectively; no future sample is accessed. Accordingly, the fixed-dilation branch repeatedly emphasizes neighboring observations, whereas the progressively dilated branch aggregates information over increasingly long time horizons.

3.2 The structure of the DTCAE method

Although AE demonstrates strong capabilities in extracting features with complex nonlinear relationships, it lacks the ability to capture temporal dynamics in time-varying processes. To address this limitation, a dual-channel temporal feature learning strategy is introduced. Short-term fluctuation and long-term trend representations are learned using two temporal convolutional networks. These representations capture distinct temporal patterns across different process stages, thereby incorporating temporal dependencies into the latent features of the autoencoder. Based on the learned representations, the loss function is redesigned to enhance the consistency of short-term features while promoting the smooth evolution of long-term features.

First, feature extraction is performed on the industrial process using an encoder. Let $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{m \times n}$ be a dataset, wherein m corresponds to the data

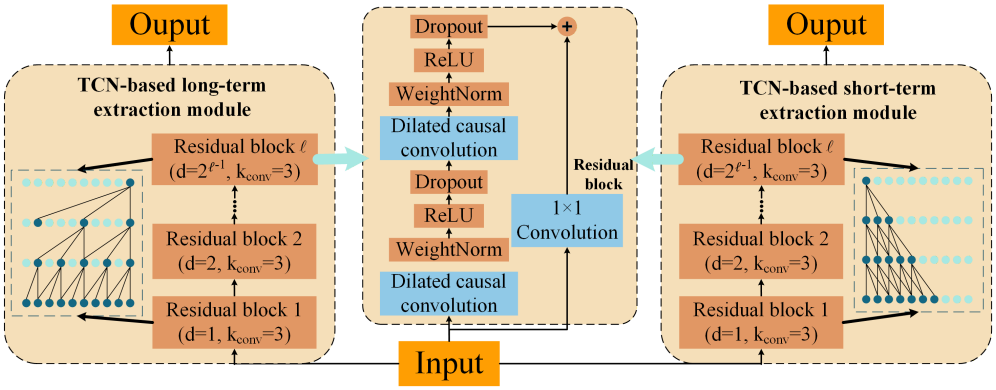


Fig. 2 Causal sampling with kernel size 3 and dilation factors $d = 1$, 2, and 4. The sampled past positions become progressively farther apart as d increases.

dimensionality and n corresponds to the time series length. The encoding operation is expressed as follows:

$$\hat{\boldsymbol{z}}_t = f_{\theta_{\text{enc}}}(\boldsymbol{x}_t) = \phi_{\text{enc}}(\boldsymbol{W}_{\text{enc}}\boldsymbol{x}_t + \boldsymbol{b}_{\text{enc}}) \tag{8}$$

where $f_{\theta_{\text{enc}}}(\cdot)$ uses the encoder parameters defined in Eq. (1), and $\hat{\boldsymbol{z}}_t \in \mathbb{R}^k$ denotes the latent feature at time t. The latent features are then fed into a dual-channel TCN to learn complementary short-term fluctuation and long-term trend features from the input sequence. As outlined in equations (5), (6) and (7), the temporal feature learning process is implemented using two parallel TCNs. Subsequently, the sum of short-term features $\boldsymbol{Q} = [\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \cdots, \boldsymbol{\xi}_n] \in \mathbb{R}^{k\times n}$ and long-term features $\boldsymbol{P} = [\boldsymbol{\tau}_1, \boldsymbol{\tau}_2, \cdots, \boldsymbol{\tau}_n] \in \mathbb{R}^{k\times n}$ is fed into the decoder. The reconstructed features $\boldsymbol{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_n] \in \mathbb{R}^{k\times n}$ are obtained through the decoding process, incorporating information from both temporal representations. The reconstruction process is defined as follows:

$$\boldsymbol{z}_t = \boldsymbol{\xi}_t + \boldsymbol{\tau}_t \tag{9}$$

$$\hat{\boldsymbol{x}}_t = g_{\theta_{\text{dec}}}(\boldsymbol{z}_t) = \phi_{\text{dec}}(\boldsymbol{W}_{\text{dec}}\boldsymbol{z}_t + \boldsymbol{b}_{\text{dec}}) \tag{10}$$

where $g_{\theta_{\text{dec}}}(\cdot)$ uses the independent decoder parameters $\theta_{\text{dec}} = \{\boldsymbol{W}_{\text{dec}}, \boldsymbol{b}_{\text{dec}}\}$ to generate $\hat{\boldsymbol{x}}_t \in \mathbb{R}^m$. As in Eq. (2), ReLU is used in the decoder hidden layers and the output layer is linear. Differences between the original and reconstructed data define the reconstruction loss function. To ensure the consistency of short-term fluctuation features and the smooth evolution of long-term trend features, explicit constraints are imposed on the learned temporal representations. Based on this, three distinct loss functions are constructed and mathematically formulated as:

$$Loss_{rec} = \frac{1}{n}\sum_{t=1}^{n} \|\boldsymbol{x}_t - \hat{\boldsymbol{x}}_t\|^2 \tag{11}$$

$$Loss_{short} = \frac{1}{(C-1)\,p} \sum_{c=1}^{C-1} \sum_{j=1}^{p} \left\| \boldsymbol{\xi}_{(c-1)p+j} - \boldsymbol{\xi}_{cp+j} \right\|^2 \tag{12}$$

$$Loss_{long} = \frac{1}{n-2} \sum_{t=1}^{n-2} \left\| \boldsymbol{\tau}_{t+2} - 2\boldsymbol{\tau}_{t+1} + \boldsymbol{\tau}_t \right\|^2 \tag{13}$$

where p and $C = \left\lfloor \frac{n}{p} \right\rfloor$ denote segment length and total number of segments, $c \in \{1, 2, \cdots, C-1\}$ is the segment index, and $j \in \{1, 2, \cdots, p\}$ is the intra-segment position index. The short-term consistency loss enforces consistency across adjacent temporal segments by penalizing deviations between temporally aligned feature positions. This constraint encourages the short-term feature to capture local temporal patterns while suppressing irregular disturbances. The long-term smoothness loss penalizes abrupt changes in the long-term trend feature by using the second-order temporal difference, thereby promoting gradual temporal evolution and reducing sensitivity to high-frequency noise. Together, these losses guide the dual-channel representation toward complementary short-term fluctuation and long-term trend features. The total loss function is formulated as a weighted sum of the three individual loss components:

$$Loss = Loss_{rec} + \lambda_1 Loss_{short} + \lambda_2 Loss_{long} \tag{14}$$

where λ_1 and λ_2 are the weight coefficients. The overall structure of the DTCAE is shown in Fig. 3.

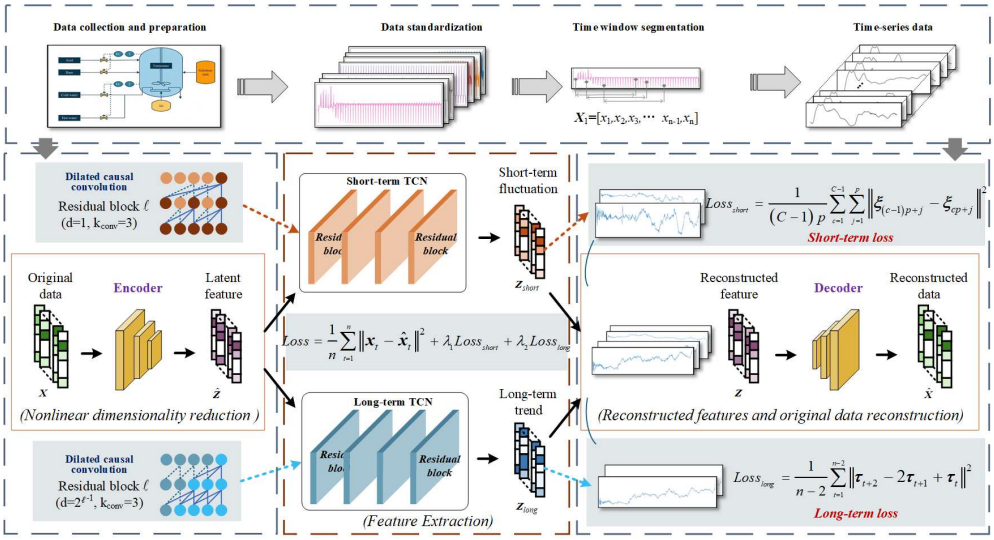


Fig. 3 Structure of the proposed DTCAE network

3.3 DTCAE-based monitoring method

The DTCAE is designed to capture temporal patterns in dynamic processes using a hierarchical structure. Given a standardized dataset $\boldsymbol{X}_{\text{test}} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n] \in$

$\mathbb{R}^{m\times n}$, the data are first encoded into latent features $\hat{\boldsymbol{z}} \in \mathbb{R}^k$. To better characterize process dynamics, $\hat{\boldsymbol{z}}$ is represented by two complementary temporal representations: short-term fluctuation features $\boldsymbol{\xi}$ and long-term trend features $\boldsymbol{\tau}$, learned through two separate temporal convolutional networks. The reconstructed features $\boldsymbol{z}$ are then passed to the decoder. This enables the model to learn a compact latent space that retains both short-term fluctuations and long-term trends. Monitoring statistics are constructed separately from the short-term fluctuation and long-term trend features. The mathematical expressions are shown in Eqs. (15), (16) and (17):

$$T^2 = (\boldsymbol{z} - \boldsymbol{\mu}_z)^{\mathrm{T}} \boldsymbol{\Sigma}_z^{-1} (\boldsymbol{z} - \boldsymbol{\mu}_z) \tag{15}$$

$$T^2_{short} = \left(\boldsymbol{\xi} - \boldsymbol{\mu}_\xi\right)^{\mathrm{T}} \boldsymbol{\Sigma}_\xi^{-1} \left(\boldsymbol{\xi} - \boldsymbol{\mu}_\xi\right) \tag{16}$$

$$T^2_{long} = (\boldsymbol{\tau} - \boldsymbol{\mu}_\tau)^{\mathrm{T}} \boldsymbol{\Sigma}_\tau^{-1} (\boldsymbol{\tau} - \boldsymbol{\mu}_\tau) \tag{17}$$

where $\boldsymbol{\mu}_z$, $\boldsymbol{\mu}_\xi$, and $\boldsymbol{\mu}_\tau$ are the mean vectors, and $\boldsymbol{\Sigma}_z$, $\boldsymbol{\Sigma}_\xi$, and $\boldsymbol{\Sigma}_\tau$ are the covariance matrices, estimated separately from the corresponding fused, short-term, and long-term features of the normal training samples. The T^2, T^2_{short} and T^2_{long} statistics are used to monitor variations in the learned feature space. In addition, the *SPE* statistic is computed from the residual space to detect abnormal variations not captured by the learned representations. Mathematically, the *SPE* statistic can be expressed as (18):

$$SPE = (\boldsymbol{x} - \hat{\boldsymbol{x}})^{\mathrm{T}} (\boldsymbol{x} - \hat{\boldsymbol{x}}) \tag{18}$$

To improve fault detection performance, short-term fluctuation and long-term trend features are integrated through Bayesian inference to construct a fused monitoring statistic. Based on Eqs. (19) and (20), the T^2 fault probability is calculated as:

$$p_{T^2}(\mathrm{F}|\boldsymbol{x}_{test}) = \frac{p_{T^2}(\boldsymbol{x}_{test}|\mathrm{F})\, p_{T^2}(\mathrm{F})}{p_{T^2}(\boldsymbol{x}_{test})} \tag{19}$$

$$p_{T^2}(\boldsymbol{x}_{test}) = p_{T^2}(\boldsymbol{x}_{test}|\mathrm{N})\, p_{T^2}(\mathrm{N}) + p_{T^2}(\boldsymbol{x}_{test}|\mathrm{F})\, p_{T^2}(\mathrm{F}) \tag{20}$$

in which $p_{T^2}(\boldsymbol{x}_{test} \mid \mathrm{N}) = e^{-T^2/T^2_{\lim}}$ and $p_{T^2}(\boldsymbol{x}_{test} \mid \mathrm{F}) = e^{-T^2_{\lim}/T^2}$ represent the conditional fault probabilities for the respective statistics. The control limit $T^2_{\lim}$ of the constructed statistic is determined by kernel density estimation (KDE), which is based on the statistic values of normal training samples at the confidence level α. Normal and faulty conditions are denoted by "N" and "F", respectively. Given $p_{T^2}(\mathrm{N}) = \alpha$, $p_{T^2}(\mathrm{F}) = 1 - \alpha$, and a confidence level α, the fault probabilities for T^2_{long} and T^2_{short} are computed using an identical approach. Therefore, the new statistic is expressed in Eq. (21):

$$BIC_{T^2} = \frac{p^2_{T^2}(\mathrm{F}|\boldsymbol{x}_{test}) + p^2_{T^2_{short}}(\mathrm{F}|\boldsymbol{x}_{test}) + p^2_{T^2_{long}}(\mathrm{F}|\boldsymbol{x}_{test})}{p_{T^2}(\mathrm{F}|\boldsymbol{x}_{test}) + p_{T^2_{short}}(\mathrm{F}|\boldsymbol{x}_{test}) + p_{T^2_{long}}(\mathrm{F}|\boldsymbol{x}_{test})} \tag{21}$$

Based on the integrated statistic and the extracted deep features, the operational roadmap of the proposed method is presented in Fig. 4. The specific algorithmic procedures for the offline and online phases are outlined as follows.

Offline modeling:

Step 1: Collect the normal data from industrial processes and standardize the data as training data $\boldsymbol{X}_{train}$.

Step 2: Feed the training data $\boldsymbol{X}_{\text{train}}$ into the encoder to obtain latent features, which are then processed by the dual-channel TCN to extract short-term fluctuation features and long-term trend features using fixed and progressively enlarged dilation rates, respectively.

Step 3: Combine the short-term fluctuation features and long-term trend features to obtain the fused feature representation $\boldsymbol{Z}$, and send $\boldsymbol{Z}$ to reconstruct the original data.

Step 4: Obtain the T^2, T^2_{short}, T^2_{long} and SPE statistics from the learned representations and reconstruction residuals.

Step 5: Construct the control limits T^2_{lim}, $T^2_{short,\text{lim}}$, $T^2_{long,\text{lim}}$, $BIC_{T^2,\text{lim}}$ and SPE_{lim} using KDE based on the corresponding statistic values of normal training samples.

Online monitoring:

Step 1: Acquire online data and normalize them to form test data $\boldsymbol{X}_{test}$.

Step 2: Process the test data $\boldsymbol{X}_{test}$ through the DTCAE model to derive the latent features, temporal representations, and reconstructed data.

Step 3: Obtain the T^2, T^2_{short}, T^2_{long} and SPE statistics based on the latent features and reconstructed data, and calculate the fused statistic BIC_{T^2}.

Step 4: Determine the process status by comparing the monitoring statistics with their corresponding control limits.

4 Case study

4.1 Penicillin fermentation process

Penicillin fermentation is a complex multi-stage nonlinear system that serves as a benchmark process in biopharmaceutical production (Bajpai and Reuss 1980). Key operational parameters such as pH, temperature, and dissolved oxygen levels directly influence penicillin quality. The time-varying and nonlinear dynamics of fermentation complicate real-time process monitoring. Moreover, the coexistence of short-term fluctuations and long-term trends in fermentation data poses significant challenges for conventional fault detection methods, often resulting in degraded monitoring performance and increased false alarm rates (FARs).

Table 1 Training hyperparameters of the proposed DTCAE

Hyperparameter	Value
Optimizer	Adam
Learning rate	1×10^{-4}
Batch size	32
Epochs	60
Dropout rate	0.1

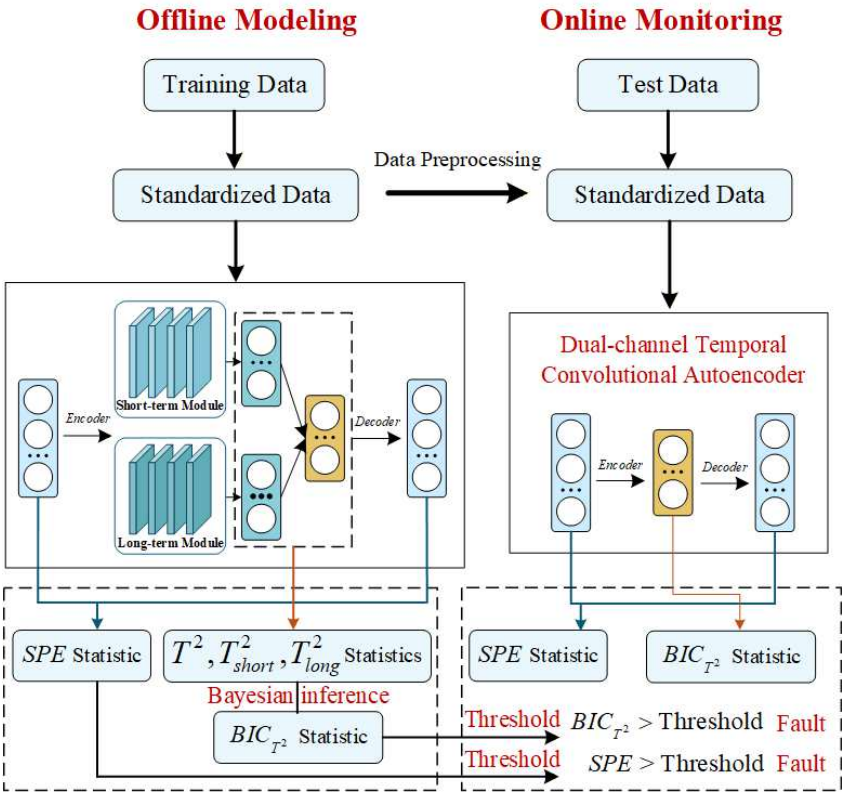


Fig. 4 Process monitoring flow chart of proposed DTCAE

The DTCAE method is validated using a penicillin fermentation dataset (Yang et al. 2024). The dataset comprises 20 batches of normal operational data for training and 6 batches of faulty data for testing. Each batch covers 400 hours with a 0.5 h sampling interval, yielding 800 samples. In each faulty batch, the first 200 samples represent normal conditions, while faults are introduced from sample 201 onward.

To validate the effectiveness of the DTCAE algorithm, a comparative study involving seven fault detection methods is conducted on the dataset: AE, DAE, GAN-AE, ConvAE (Mao et al. 2018), LSTM-AE (Chen et al. 2021), TCN-AE (Mo et al. 2023) and DTCAE. All algorithms used a 0.96 confidence threshold for fault detection. The AE and DAE models use a symmetric encoder-decoder architecture with layer dimensions of 16-12-16. The GAN-AE framework incorporates a generator and discriminator with dimensions 12-64 and 64-1, respectively. The AE, DAE, and GAN-AE models adopt a fully connected autoencoder structure. LSTM-AE integrates LSTM layers in the encoder-decoder, while ConvAE and TCN-AE employ convolutional layers. The DTCAE employs a four-layer TCN with a kernel size of three. The short-term channel uses a fixed dilation factor of one, while the long-term channel employs the dilation sequence 1, 2, 4, and 8 across the four residual blocks. Through this dual-dilation temporal modeling design, the model can capture both short-term fluctuations

and long-term trends. The core training hyperparameters shared across all compared models are summarized in Table 1.

Comparative fault detection rates (FDRs) and FARs for seven algorithms across different fault types on the penicillin fermentation dataset are summarized in Table 2.

Table 2 FDRs/FARs for penicillin fermentation process

Fault No.		1	2	3	4	5	6	Average
AE	T^2	0.39/0.01	0.74/0.01	0.12/0.01	0.54/0.00	0.37/0.02	0.53/0.00	0.45/0.01
	SPE	1.00/0.06	0.88/0.03	0.81/0.05	0.87/0.08	1.00/0.07	0.80/0.03	0.89/0.05
DAE	T^2	0.34/0.01	0.78/0.00	0.38/0.01	0.74/0.00	0.54/0.02	0.60/0.00	0.56/0.01
	SPE	1.00/0.06	0.90/0.03	0.64/0.04	0.82/0.08	1.00/0.06	0.79/0.02	0.86/0.05
GAN-AE	T^2	1.00/0.01	0.88/0.00	1.00/0.01	0.89/0.00	1.00/0.02	0.80/0.00	0.93/0.01
	SPE	0.94/0.06	0.83/0.02	0.32/0.04	0.77/0.06	0.90/0.06	0.77/0.02	0.76/0.04
ConvAE	T^2	0.53/0.00	0.73/0.00	0.06/0.01	0.54/0.00	0.68/0.02	0.75/0.00	0.55/0.01
	SPE	1.00/0.01	0.86/0.00	0.37/0.01	0.77/0.00	1.00/0.02	0.80/0.00	0.80/0.01
LSTM-AE	T^2	0.23/0.01	0.21/0.02	0.99/0.01	0.89/0.00	0.02/0.05	0.30/0.00	0.44/0.02
	SPE	1.00/0.01	0.90/0.00	0.59/0.01	0.85/0.00	1.00/0.02	0.80/0.00	0.86/0.01
TCN-AE	T^2	1.00/0.01	0.94/0.00	0.21/0.00	0.69/0.00	0.60/0.05	0.68/0.00	0.69/0.01
	SPE	1.00/0.01	0.88/0.01	0.70/0.01	0.87/0.01	1.00/0.05	0.81/0.00	0.88/0.01
DTCAE	T^2	1.00/0.01	0.89/0.01	1.00/0.01	0.89/0.01	1.00/0.03	0.78/0.00	0.93/0.01
	T^2_{short}	1.00/0.01	0.86/0.00	0.55/0.01	0.82/0.01	1.00/0.05	0.78/0.01	0.84/0.02
	T^2_{long}	1.00/0.01	0.86/0.02	0.40/0.01	0.86/0.05	1.00/0.09	0.90/0.01	0.84/0.03
	BIC_{T^2}	**1.00/0.01**	**0.86/0.00**	**1.00/0.01**	**0.89/0.02**	**1.00/0.09**	**0.83/0.00**	**0.93/0.02**
	SPE	**1.00/0.01**	**0.92/0.01**	**0.94/0.01**	**0.88/0.00**	**1.00/0.05**	**0.82/0.00**	**0.93/0.01**

The DTCAE outcomes and their mean values are bolded for visibility. Although the AE and DAE algorithms are capable of effective data reconstruction, the FDRs of their T^2 statistics remain relatively low. GAN-AE attains an average T^2 FDR of 0.93, equal to that of DTCAE, but has a lower average *SPE* FDR. Similarly, while ConvAE, LSTM-AE, and TCN-AE demonstrate acceptable performance on certain faults, none of these six algorithms achieve consistently good results across all fault types. In contrast, the DTCAE model achieves high FDRs across all fault categories. Due to differences in fault types and the characteristics of statistical indicators, detection rates vary across different statistics; for instance, Fault 3 and Fault 4 exhibit distinct behaviors. By combining the strengths of multiple statistical indicators and applying Bayesian inference criteria, improved detection performance can be achieved. Both BIC_{T^2} and *SPE* achieve an average FDR of 0.93. As shown in Table 2, DTCAE consistently achieves the highest or among the highest detection rates across the six fault types. This improvement demonstrates the superior ability of DTCAE to capture short-term fluctuations and long-term trends in the data.

Table 3 reports the penicillin fermentation ablation results. The fixed-dilation branch uses the sequence $1, 1, 1, 1$, whereas the progressive-dilation branch uses $1, 2, 4, 8$. The single-branch variants retain their respective auxiliary losses; the single-channel multi-dilation TCN has no auxiliary loss. The dual-channel variant without auxiliary losses retains one fixed and one progressive branch. Two additional controls use either two fixed-dilation branches or two progressive-dilation branches. Each table entry lists the fault detection rate followed by the false alarm rate.

Table 2 shows that DTCAE attains a mean FDR of 0.93 and a mean FAR of 0.01 under both the T^2 and *SPE* statistics. In the ablation study reported in Table 3, the

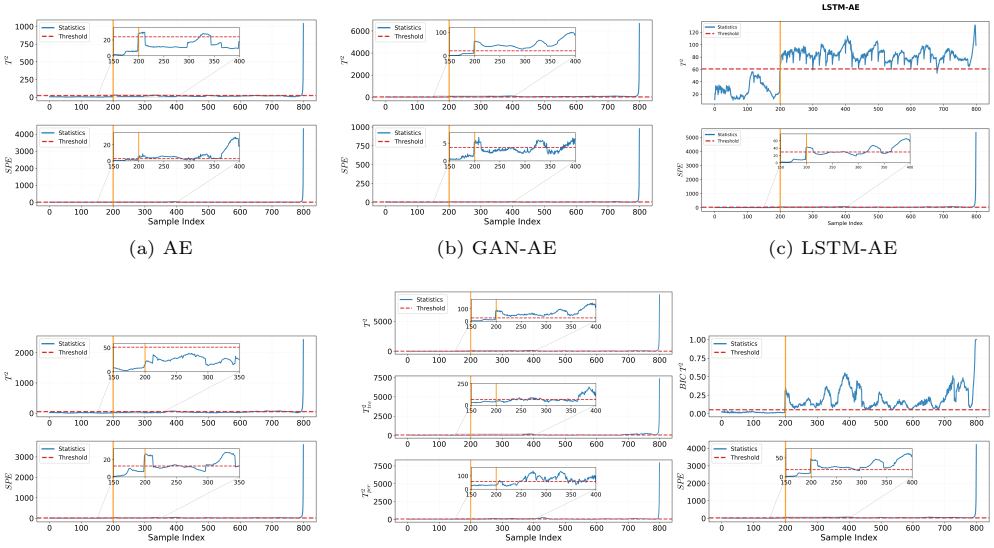
Table 3 FDRs/FARs for penicillin fermentation process in the ablation study

Fault No.		1	2	3	4	5	6	Average
AE	T^2	0.39/0.01	0.74/0.01	0.12/0.01	0.54/0.00	0.37/0.02	0.53/0.00	0.45/0.01
	SPE	1.00/0.06	0.88/0.03	0.81/0.05	0.87/0.08	1.00/0.07	0.80/0.03	0.89/0.05
Fixed dilation branch only	T^2	0.01/0.00	0.30/0.00	0.01/0.06	0.48/0.00	0.92/0.06	0.75/0.00	0.41/0.02
	SPE	1.00/0.01	0.94/0.01	1.00/0.06	0.89/0.00	1.00/0.06	0.82/0.00	0.94/0.02
Progressive dilation branch only	T^2	0.23/0.00	0.67/0.00	0.66/0.17	0.82/0.00	0.06/0.17	0.62/0.00	0.51/0.06
	SPE	1.00/0.01	0.94/0.01	1.00/0.12	0.89/0.01	1.00/0.12	0.85/0.00	0.95/0.05
Single channel multi dilation TCN	T^2	1.00/0.05	0.94/0.12	0.33/0.38	0.92/0.09	1.00/0.38	0.97/0.14	0.86/0.19
	SPE	1.00/0.01	0.84/0.00	0.15/0.06	0.74/0.00	0.99/0.06	0.79/0.00	0.75/0.02
Dual channel TCN without auxiliary losses	T^2	1.00/0.01	0.87/0.00	0.88/0.10	0.89/0.04	1.00/0.10	0.83/0.00	0.91/0.04
	SPE	1.00/0.01	0.92/0.01	0.94/0.05	0.88/0.00	1.00/0.04	0.83/0.00	0.93/0.02
Dual channel TCN with fixed dilations	T^2	1.00/0.03	0.88/0.01	0.82/0.06	0.92/0.01	1.00/0.06	0.94/0.00	0.93/0.03
	SPE	1.00/0.01	0.90/0.00	0.55/0.05	0.83/0.00	1.00/0.04	0.80/0.00	0.85/0.02
Dual channel TCN with progressive dilations	T^2	1.00/0.01	0.89/0.01	0.74/0.17	0.89/0.02	1.00/0.16	0.89/0.01	0.90/0.06
	SPE	1.00/0.01	0.90/0.00	0.58/0.06	0.86/0.00	1.00/0.06	0.80/0.00	0.86/0.02
DTCAE	BIC_{T^2}	**1.00/0.01**	**0.86/0.00**	**1.00/0.01**	**0.89/0.02**	**1.00/0.09**	**0.83/0.00**	**0.93/0.02**
	SPE	**1.00/0.01**	**0.92/0.01**	**0.94/0.01**	**0.88/0.00**	**1.00/0.05**	**0.82/0.00**	**0.93/0.01**

dual-channel controls with two fixed-dilation branches and two progressive-dilation branches attain mean T^2 FDRs of 0.93 and 0.90, respectively, with mean FARs of 0.03 and 0.06. Under *SPE*, the same controls attain mean FDRs of 0.85 and 0.86, each with a mean FAR of 0.02. Removing the auxiliary losses leaves the mean *SPE* FDR at 0.93 but increases the mean FAR to 0.02. The single fixed-dilation and progressive-dilation branches attain mean *SPE* FDRs of 0.94 and 0.95, respectively, with mean FARs of 0.02 and 0.05.

The monitoring trajectories for Fault 3 are visualized in Fig. 5. Most of the compared methods exhibit an alarm response around the initial fault onset, although their alarm persistence and overall detection coverage differ substantially as the fault evolves. However, as the process evolves, the detection performance of AE deteriorates, leading to missed alarms. This instability indicates that conventional shallow autoencoder architectures are less effective in characterizing the complex temporal dynamics inherent in the fermentation process data. Although GAN-AE imposes distribution constraints on features, its detection performance gradually declines as the fault progresses. Similarly, LSTM-AE and TCN-AE have limited capability in simultaneously modeling short-term fluctuations and long-term trends. Consequently, these algorithms fail to achieve a good balance between the T^2 and *SPE* statistics. In contrast, DTCAE detects the fault immediately upon occurrence and sustains consistent detection. The dual-channel temporal feature representation enables DTCAE to distinguish fault-induced deviations from normal process variations. DTCAE achieves detection rates of 1 and 0.94 for the BIC_{T^2} and *SPE* statistics, respectively, demonstrating its effectiveness in capturing multi-scale temporal characteristics in industrial fermentation processes.

Penicillin fermentation is a typical batch bioprocess characterized by significant temporal dynamics, including short-term fluctuations caused by operational interventions and long-term trends related to microbial metabolism. However, process faults can significantly disrupt these temporal patterns. To validate the feature extraction capability of DTCAE, Fig. 6 visualizes the short-term and long-term temporal features of Fault 1, alongside the final reconstructed feature for a selected dimension. Under normal operating conditions, the short-term feature exhibits relatively stable local variations, while the long-term feature maintains a smooth trajectory. Following



(a) AE (b) GAN-AE (c) LSTM-AE

(d) TCN-AE (e) DTCAE

Fig. 5 The monitoring charts of Fault 3

fault introduction, the short-term feature shows amplified local fluctuations that deviate markedly from the normal operating range. Simultaneously, the long-term feature demonstrates a gradual drift away from its baseline. The abnormal short-term fluctuations indicate disturbed local process dynamics, while the long-term drift reflects the accumulation of process deviations. By learning these two complementary temporal features, DTCAE enables more precise fault characterization. The combined evolution of short-term fluctuation and long-term trend features provides a comprehensive representation of fault progression, which enhances fault detection performance.

In addition, the two weight coefficients λ_1 and λ_2 in the loss function influence the performance of the DTCAE model. To determine appropriate values, a systematic exploration of different parameter combinations was conducted. Specifically, the short-term consistency weight coefficient λ_1 and the long-term smoothness weight coefficient λ_2 were varied over a range of values to evaluate their influence on fault detection performance. Fig. 7 presents the parameter sensitivity surfaces of BIC_{T^2} and SPE for Fault 3 under different combinations of λ_1 and λ_2. As can be observed, the detection results vary considerably with different parameter settings. Based on the overall detection performance, λ_1 and λ_2 were selected as 0.1 and 0.05, respectively, for all subsequent experiments.

The segment length p in the short-term consistency loss was also examined using Fault 3. Fig. 8 compares the FDRs obtained from the T^2 and SPE statistics for $p = 5$, 10, 15, 20, 25, and 30. Both statistics attain their highest FDRs among the examined values at $p = 20$. Accordingly, p was set to 20 for the subsequent experiments.

Since the DTCAE model relies on sequential observations to learn temporal dependencies, the selection of the sliding window length is another important factor. Fig. 9 shows the fault detection results under different window lengths for Fault 3. As shown

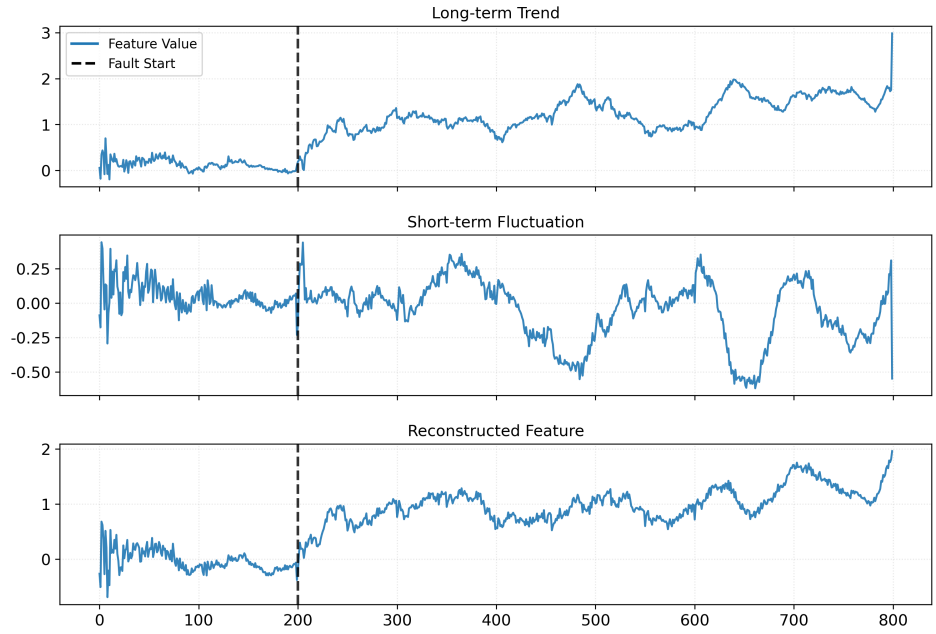
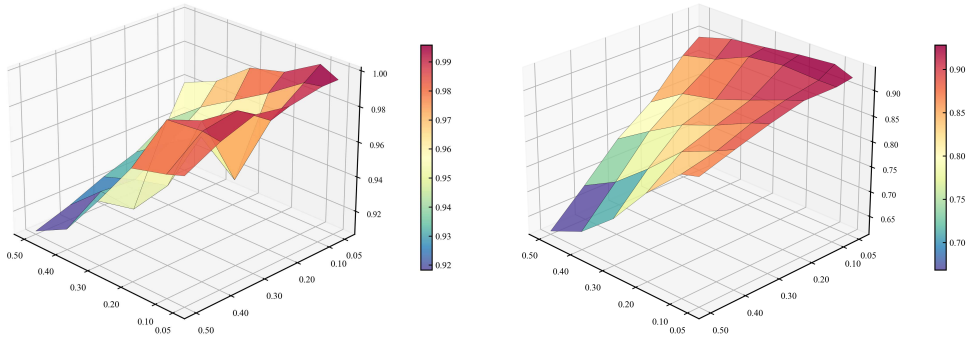


Fig. 6 Short-term, long-term, and final reconstructed features of Fault 1



(a) (b)

Fig. 7 Parameter sensitivity surfaces of different weight coefficients for Fault 3: (a) BIC_{T^2} and (b) SPE

in the figure, increasing the window length generally provides more temporal information, while excessively large windows may introduce redundant historical patterns. Considering both fault detection performance and model complexity, the window length was set to 80 in the remaining experiments.

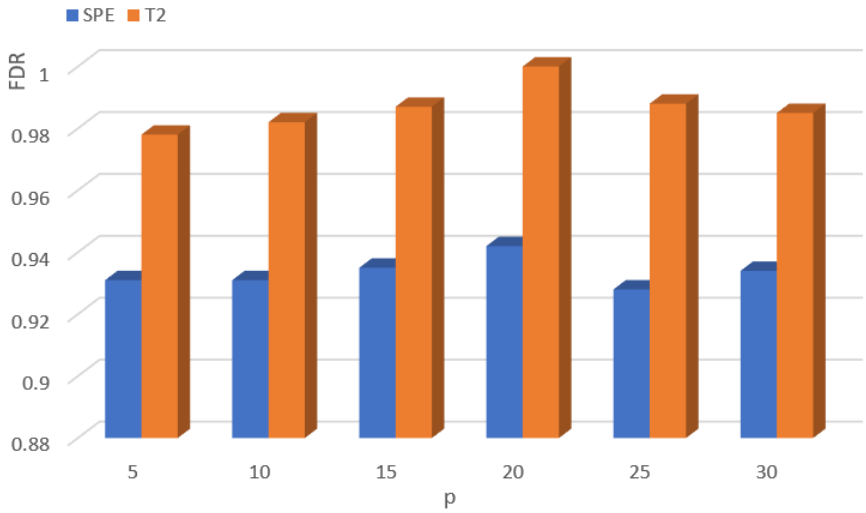
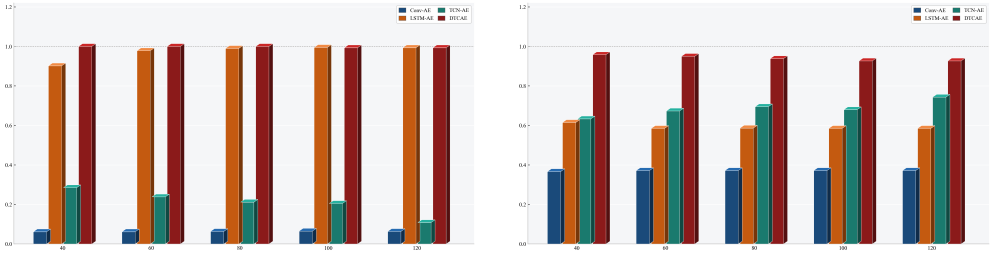


Fig. 8 Effect of segment length p on the FDRs of T^2 and SPE for Fault 3



(a) (b)

Fig. 9 FDR comparison of different methods under different sliding window lengths for Fault 3: (a) T^2/BIC_{T^2} and (b) SPE

4.2 Wastewater treatment process (WWTP)

The WWTP integrates physical, chemical, and biological technologies to remove and transform pollutants through complex process flows (Gernaey et al. 2004). Wastewater sequentially flows through various functional units, such as anaerobic, anoxic, and aerobic tanks. Due to microbial activity and interdependencies among process variables,

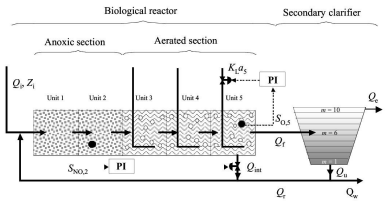


Fig. 10 General overview of the BSM1 plant

variable relationships are highly nonlinear, making the wastewater treatment process difficult to describe using simple linear models. Moreover, environmental factors often cause process data to deviate from Gaussian distributions. As the WWTP progresses through different operational stages, the process data exhibit short-term fluctuations and long-term trends, posing significant challenges for process monitoring. To address these challenges, this study employs the Benchmark Simulation Model 1 (BSM1) to simulate WWTP operations in a controlled environment (Peng et al. 2022; Alex et al. 2008).

Table 4 Seven fault descriptions

NO.	Fault type	Value
1	Sludge expansion fault	$\mu H = 2.5$
2	Sludge expansion fault	$\mu H = 3.5$
3	Toxic impact fault	$\mu H = 3.5, bH = 5$
4	Toxic impact fault	$\mu H = 3.75, bH = 4.5$
5	Ramp fault	X_{BA}, amplitude = -3%
6	Step fault	S_O, amplitude = +20%
7	Step fault	X_S, amplitude = +5%

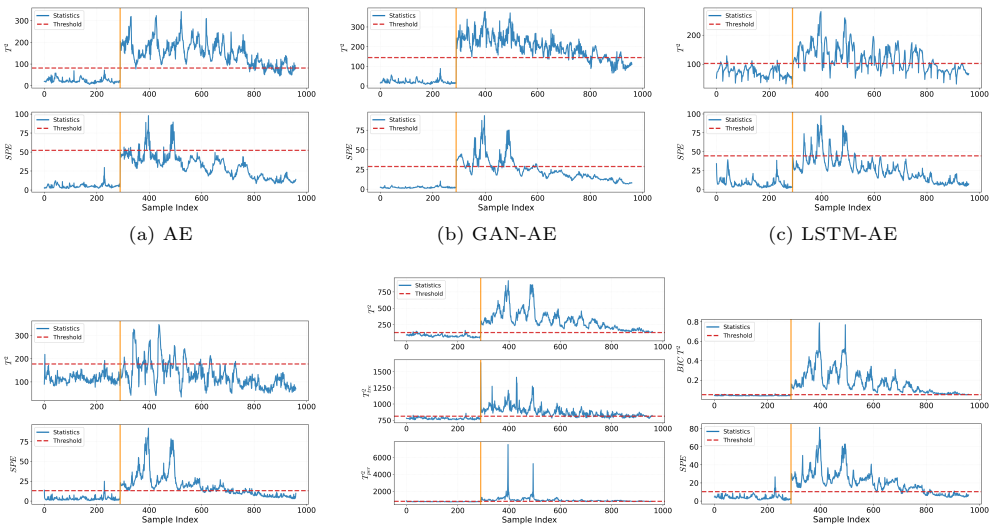
Table 5 FDRs/FARs for wastewater treatment process

Fault No.		1	2	3	4	5	6	7	Average
AE	T^2	0.89/0.00	0.99/0.00	0.99/0.00	0.89/0.00	0.99/0.00	0.99/0.00	0.99/0.00	0.96/0.00
	SPE	0.07/0.00	0.96/0.00	0.59/0.00	0.99/0.00	0.79/0.00	0.47/0.00	0.58/0.00	0.64/0.00
DAE	T^2	0.67/0.00	0.99/0.00	0.99/0.00	0.99/0.00	0.99/0.00	0.89/0.04	0.99/0.04	0.93/0.01
	SPE	0.27/0.00	0.99/0.00	0.77/0.00	0.99/0.00	0.99/0.00	0.63/0.04	0.75/0.04	0.77/0.01
GAN-AE	T^2	0.81/0.01	0.99/0.00	0.99/0.01	0.99/0.01	0.99/0.01	0.97/0.01	0.99/0.01	0.96/0.01
	SPE	0.28/0.00	0.99/0.00	0.84/0.00	0.99/0.00	0.98/0.00	0.64/0.00	0.83/0.00	0.79/0.00
ConvAE	T^2	0.70/0.05	0.96/0.05	0.83/0.05	0.99/0.05	0.99/0.00	0.83/0.00	0.88/0.00	0.88/0.03
	SPE	0.19/0.00	0.81/0.00	0.58/0.00	0.99/0.00	0.86/0.00	0.47/0.00	0.59/0.00	0.64/0.00
LSTM-AE	T^2	0.57/0.05	0.92/0.05	0.70/0.05	0.99/0.05	0.87/0.05	0.69/0.05	0.72/0.05	0.78/0.05
	SPE	0.14/0.03	0.85/0.03	0.60/0.03	0.99/0.03	0.65/0.03	0.47/0.03	0.58/0.03	0.61/0.03
TCN-AE	T^2	0.17/0.01	0.64/0.01	0.45/0.01	0.92/0.01	0.74/0.01	0.36/0.01	0.45/0.01	0.53/0.01
	SPE	0.59/0.01	1.00/0.01	0.84/0.01	1.00/0.01	1.00/0.01	0.78/0.01	0.83/0.01	0.86/0.01
DTCAE	T^2	0.99/0.03	0.99/0.01	0.99/0.01	0.99/0.01	0.99/0.01	0.99/0.03	0.99/0.01	0.99/0.02
	T^2_{short}	0.89/0.02	0.99/0.01	0.99/0.01	0.99/0.01	0.99/0.01	0.98/0.01	0.99/0.01	0.97/0.01
	T^2_{long}	0.82/0.02	0.99/0.01	0.99/0.01	0.99/0.01	0.99/0.01	0.99/0.02	0.99/0.01	0.97/0.01
	BIC_{T^2}	**0.98/0.02**	**0.99/0.01**	**0.99/0.01**	**0.99/0.01**	**0.99/0.01**	**0.99/0.01**	**0.99/0.01**	**0.99/0.01**
	SPE	**0.73/0.01**	**0.99/0.01**	**0.96/0.01**	**0.99/0.01**	**0.99/0.01**	**0.89/0.01**	**0.96/0.01**	**0.93/0.01**

The BSM1 model incorporates two primary biochemical processes: nitrification and denitrification, with detailed specifications in (Alex et al. 2008). Fig. 10 presents a general overview of the BSM1 plant. The study monitors 56 process variables. Under normal operating conditions, the BSM1 simulation runs for ten consecutive days with a sampling interval of 15 min, yielding 960 normal samples. Three categories of faults are introduced during the simulation: sludge bulking, toxic shock, and sensor faults. These are further classified into seven specific fault types, as detailed in Table 4. All faults are initiated on Day 4 of the simulation.

Seven algorithms are compared: AE, DAE, GAN-AE, ConvAE, LSTM-AE, TCN-AE, and DTCAE. All methods employ a confidence level of 0.96. The structures of AE,

DAE, and the autoencoder component of GAN-AE are configured as 56–80–24–80–56. The GAN structure uses dimensions of 24–80–1. ConvAE incorporates convolutional layers into its encoder-decoder structure, whereas LSTM-AE and TCN-AE employ LSTM units and TCN, respectively, following a similar reconstruction-based architecture. The DTCAE structure remains consistent with that used in the penicillin fermentation case, with a hidden layer dimension of 24. Table 5 summarizes the fault detection performance of all methods across seven fault types, with DTCAE results indicated in bold.



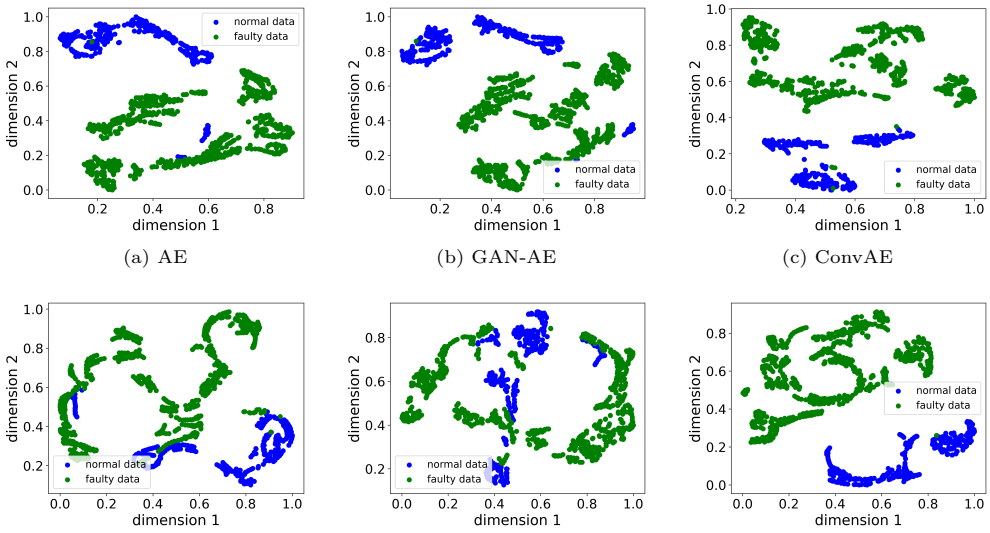
(a) AE (b) GAN-AE (c) LSTM-AE

(d) TCN-AE (e) DTCAE

Fig. 11 The monitoring charts of Fault 1: (a) AE, (b) GAN-AE, (c) LSTM-AE, (d) TCN-AE, and (e) DTCAE

For Faults 1, 3, and 6, AE, DAE, and GAN-AE achieve high FDRs based on the T^2 statistic. However, these models perform poorly on the *SPE* statistic, with none achieving average FDRs above 0.9. Additionally, ConvAE achieves a moderate T^2 average of 0.88 but exhibits poor *SPE* performance with an average of only 0.64. LSTM-AE shows suboptimal results in both statistics, with average FDRs of 0.78 and 0.61 for the T^2 and *SPE* statistics, respectively. Although TCN-AE attains a relatively high average *SPE* FDR of 0.86, its T^2 average drops to 0.53, indicating detection failures in several fault scenarios. In contrast, DTCAE achieves the highest overall detection performance across most fault scenarios, with average FDRs of 0.99 and 0.93 for the BIC_{T^2} and *SPE* statistics, respectively.

As shown in Fig. 11 for Fault 1 monitoring, most algorithms achieve successful fault detection via the T^2 statistic at sample 289. By contrast, their detection effectiveness using the *SPE* statistic is noticeably weaker. The WWTP dataset exhibits complex multi-scale temporal characteristics. Although existing methods can detect the initial



(a) AE (b) GAN-AE (c) ConvAE

(d) LSTM-AE (e) TCN-AE (f) DTCAE

Fig. 12 Visualization of Fault 1 with 2D t-SNE: (a) AE, (b) GAN-AE, (c) ConvAE, (d) LSTM-AE, (e) TCN-AE, and (f) DTCAE

fault occurrence, they fail to adequately capture both short-term fluctuations and long-term trends in the data, which leads to missed detections during the fault propagation phase. In contrast, DTCAE achieves higher detection performance in monitoring both the SPE and BIC_{T^2} statistics. The results indicate that the proposed dual-channel representation can effectively characterize multi-scale temporal patterns in dynamic process data.

In order to better demonstrate and analyze the fault detection performance, the t-distributed stochastic neighbor embedding (t-SNE) algorithm (van der Maaten and Hinton 2008), a nonlinear dimensionality reduction method, is used to visualize the latent feature representations for Fault 1. The visualization of Fault 1 is shown in Fig. 12 for the six comparative algorithms: AE, GAN-AE, ConvAE, LSTM-AE, TCN-AE, and DTCAE. Blue points represent normal samples, and green points represent faulty samples. The comparison shows that traditional methods cannot effectively distinguish faulty samples from normal ones, while DTCAE demonstrates better separation capability. These results indicate that DTCAE effectively identifies fault occurrences by using short-term fluctuation and long-term trend information among process variables, achieving improved process monitoring performance.

5 Conclusion

This paper proposes a dual-channel temporal convolutional autoencoder with short-term fluctuation and long-term trend representation for dynamic process monitoring. Two parallel temporal convolutional networks are employed to learn complementary temporal representations from the latent space. These representations enable a

comprehensive characterization of the underlying process dynamics, improving reconstruction quality and monitoring performance. The DTCAE was evaluated on the penicillin fermentation process and the WWTP. Results demonstrate its effectiveness in dynamic process monitoring compared with existing methods. However, several limitations remain. The extracted temporal features may still contain irrelevant fluctuations and noise, which can affect reconstruction accuracy and monitoring reliability. Future work will focus on refining the multi-scale temporal feature learning mechanism and incorporating noise reduction strategies.

Statements and Declarations

Author Contributions

Jian Huang: Conceptualization, Methodology, Software, Investigation, Data curation, Visualization, Writing – original draft. **Xin Pu:** Formal analysis, Validation, Visualization, Investigation, Writing – review & editing. **Xu Yang:** Supervision, Validation, Visualization, Investigation, Writing – review & editing. **Kaixiang Peng:** Conceptualization, Resources, Supervision, Project administration, Funding acquisition, Writing – review & editing. **Jingjing Gao:** Data curation, Validation. **Dajian Huang:** Supervision, Resources.

Funding

This work was supported in part by the National Natural Science Foundation of China under Grants 62573048, 62503045, and U25B2053; the Guangdong Basic and Applied Basic Research Foundation under Grants 2025A1515012016, 2024A1515110256, 2025A1515011375, and 2023A1515140121; and the Shagang Group Scholarship for Teachers.

Competing Interests

The authors have no relevant financial or non-financial interests to disclose.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Abu Farha, Y., Gall, J.: MS-TCN: Multi-stage temporal convolutional network for action segmentation. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., pp. 3575–3584 (2019). https://doi.org/10.1109/CVPR.2019.00369

Alex, J., Benedetti, L., Copp, J., Gernaey, K.V., Jeppsson, U., Nopens, I., Pons, M.-N., Steyer, J.-P., Vanrolleghem, P.A.: Benchmark simulation model no. 1 (BSM1). Technical Report TEIE-7229, Lund University, Lund, Sweden (2008). https://www.iea.lth.se/publications/Reports/LTH-IEA-7229.pdf

Amin, M.T., Khan, F., Ahmed, S., Imtiaz, S.: A data-driven bayesian network learning method for process fault diagnosis. Process Saf. Environ. Prot. **150**, 110–122 (2021) https://doi.org/10.1016/j.psep.2021.04.004

Azma, A., Liu, Y., Rezania, S.: A hybrid WPD–SSA machine learning transformer for modeling nonlinear dynamics of flow systems. J. Water Process Eng. **82**, 109448 (2026) https://doi.org/10.1016/j.jwpe.2026.109448

Bajpai, R.K., Reuss, M.: A mechanistic model for penicillin production. J. Chem. Technol. Biotechnol. **30**(1), 332–344 (1980) https://doi.org/10.1002/jctb.503300140

Bedi, J., Toshniwal, D.: Energy load time-series forecast using decomposition and autoencoder integrated memory network. Appl. Soft Comput. **93**, 106390 (2020) https://doi.org/10.1016/j.asoc.2020.106390

Chen, H., Liu, H., Chu, X., Liu, Q., Xue, D.: Anomaly detection and critical SCADA parameters identification for wind turbines based on LSTM-AE neural network. Renew. Energy **172**, 829–840 (2021) https://doi.org/10.1016/j.renene.2021.03.078

Chen, J., Liu, K.C.: On-line batch process monitoring using dynamic PCA and dynamic PLS models. Chem. Eng. Sci. **57**(1), 63–75 (2002) https://doi.org/10.1016/S0009-2509(01)00366-9

Cleveland, R.B., Cleveland, W.S., McRae, J.E., Terpenning, I.: STL: A seasonal-trend decomposition procedure based on loess. J. Off. Stat. **6**(1), 3–73 (1990)

de Brito, H.G., de Paula Maitelli, C.W.S., Chiavone-Filho, O.: Detection of obstructions in oil and gas pipelines: machine learning techniques for hydrate classification. Braz. J. Chem. Eng. (2026) https://doi.org/10.1007/s43153-026-00643-x

Dragomiretskiy, K., Zosso, D.: Variational mode decomposition. IEEE Trans. Signal Process. **62**(3), 531–544 (2014) https://doi.org/10.1109/TSP.2013.2288675

Gernaey, K.V., Van Loosdrecht, M.C.M., Henze, M., Lind, M., Jørgensen, S.B.: Activated sludge wastewater treatment plant modelling and simulation: State of the art. Environ. Model. Softw. **19**(9), 763–783 (2004) https://doi.org/10.1016/j.envsoft.2003.03.005

Hewage, P., Behera, A., Trovati, M., Pereira, E., Ghahremani, M., Palmieri, F., Liu, Y.: Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station. Soft Comput. **24**, 16453–16482 (2020) https://doi.org/10.1007/s00500-020-04954-0

Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006) https://doi.org/10.1126/science.1127647

Huang, J., Liu, Y., Yang, X., Lv, Z., Peng, K.: Cointegration stacked autoencoder model based on stationary features reconstruction for non-stationary process monitoring. Process Saf. Environ. Prot. **193**, 1287–1299 (2025) https://doi.org/10.1016/j.psep.2024.11.086

Jiang, G., Xie, P., He, H., Yan, J.: Wind turbine fault detection using a denoising autoencoder with temporal information. IEEE/ASME Trans. Mechatron. **23**(1), 89–100 (2018) https://doi.org/10.1109/TMECH.2017.2759301

Khan, F., Rathnayaka, S., Ahmed, S.: Methods and models in process safety and risk management: Past, present and future. Process Saf. Environ. Prot. **98**, 116–147 (2015) https://doi.org/10.1016/j.psep.2015.07.005

Lea, C., Flynn, M.D., Vidal, R., Reiter, A., Hager, G.D.: Temporal convolutional networks for action segmentation and detection. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 1003–1012 (2017). https://doi.org/10.1109/CVPR.2017.113

Lee, S., Kwak, M., Tsui, K.L., Kim, S.B.: Process monitoring using variational autoencoder for high-dimensional nonlinear processes. Eng. Appl. Artif. Intell. **83**, 13–27 (2019) https://doi.org/10.1016/j.engappai.2019.04.013

Li, Q., Wan, J., Yang, X., Huang, J., Cui, J., Yan, Q.: Two-stage stacked autoencoder monitoring model based on deep slow feature representation for dynamic processes. J. Process Control **147**, 103389 (2025) https://doi.org/10.1016/j.jprocont.2025.103389

Li, Z., Tian, L., Jiang, Q., Yan, X.: Distributed-ensemble stacked autoencoder model for non-linear process monitoring. Inf. Sci. **542**, 302–316 (2021) https://doi.org/10.1016/j.ins.2020.06.062

Mallat, S.G.: A theory for multiresolution signal decomposition: The wavelet representation. IEEE Trans. Pattern Anal. Mach. Intell. **11**(7), 674–693 (1989) https://doi.org/10.1109/34.192463

Mao, T., Zhang, Y., Ruan, Y., Gao, H., Zhou, H., Li, D.: Feature learning and process monitoring of injection molding using convolution-deconvolution auto encoders. Comput. Chem. Eng. **118**, 77–90 (2018) https://doi.org/10.1016/j.compchemeng.2018.07.009

Mo, R., Pei, Y., Venkatarayalu, N.V., Pereira Nathaniel Joseph, Premkumar, A.B., Sun, S., Foo, S.K.K.: Unsupervised TCN-AE-based outlier detection for time series with seasonality and trend for cellular networks. IEEE Trans. Wireless Commun. **22**(5), 3114–3127 (2023) https://doi.org/10.1109/TWC.2022.3216004

Peng, C., Wang, K., Zheng, K., Meng, F.: Monitoring of wastewater treatment process based on multi-stage variational autoencoder. Expert Syst. Appl. **207**, 117919 (2022) https://doi.org/10.1016/j.eswa.2022.117919

Pota, M., De Pietro, G., Esposito, M.: Real-time anomaly detection on time series of industrial furnaces: A comparison of autoencoder architectures. Eng. Appl. Artif. Intell. **124**, 106597 (2023) https://doi.org/10.1016/j.engappai.2023.106597

Qin, S.J.: Statistical process monitoring: basics and beyond. J. Chemom. **17**(8–9), 480–502 (2003) https://doi.org/10.1002/cem.800

Santos, R.B., de Sousa, E.O., da Silva, F.V., da Cruz, S.L., Fileti, A.M.F.: Detection and on-line prediction of leak magnitude in a gas pipeline using an acoustic method and neural network data processing. Braz. J. Chem. Eng. **31**(1), 145–153 (2014) https://doi.org/10.1590/S0104-66322014000100014

Sun, Y., Dong, Y., Yan, X.: Attention-based LSTM block model framework based on static and dynamic variables for modeling fuel ethanol fermentation process. Biochem. Eng. J. **199**, 109049 (2023) https://doi.org/10.1016/j.bej.2023.109049

Tang, X., Lu, W., Yan, X.: Dual attention bidirectional generative adversarial network for dynamic uncertainty process monitoring and diagnosis. Process Saf. Environ. Prot. **171**, 214–224 (2023) https://doi.org/10.1016/j.psep.2022.12.081

Thill, M., Konen, W., Wang, H., Bäck, T.: Temporal convolutional autoencoder for unsupervised anomaly detection in time series. Appl. Soft Comput. **112**, 107751 (2021) https://doi.org/10.1016/j.asoc.2021.107751

van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**, 2579–2605 (2008)

Wang, J., Zhao, Z., Liu, F.: Robust slow feature analysis for statistical process monitoring. Ind. Eng. Chem. Res. **59**(27), 12504–12513 (2020) https://doi.org/10.1021/acs.iecr.0c01512

Wang, Y., Pan, Z., Yuan, X., Yang, C., Gui, W.: A novel deep learning based fault diagnosis approach for chemical process with extended deep belief network. ISA Trans. **96**, 457–467 (2020) https://doi.org/10.1016/j.isatra.2019.07.001

Wen, Q., Gao, J., Song, X., Sun, L., Xu, H., Zhu, S.: RobustSTL: A robust seasonal-trend decomposition algorithm for long time series. In: Proc. AAAI Conf. Artif. Intell., vol. 33, pp. 5409–5416 (2019). https://doi.org/10.1609/aaai.v33i01.33015409

Wu, H., Xu, J., Wang, J., Long, M.: Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In: Adv. Neural Inf. Process. Syst., vol. 34, pp. 22419–22430 (2021)

Yang, H.F., Chen, Y.P.P.: Hybrid deep learning and empirical mode decomposition model for time series applications. Expert Syst. Appl. **120**, 128–138 (2019) https://doi.org/10.1016/j.eswa.2018.11.019

Yang, S., Deng, Z., Li, X., Zheng, C., Xi, L., Zhuang, J., Zhang, Z., Zhang, Z.: A novel hybrid model based on STL decomposition and one-dimensional convolutional neural networks with positional encoding for significant wave height forecast. Renew. Energy **173**, 531–543 (2021) https://doi.org/10.1016/j.renene.2021.04.010

Yang, X., Xiao, J., Huang, J., Peng, K.: Deep feature representation with online convolutional adversarial autoencoder for nonlinear process monitoring. J. Taiwan Inst. Chem. Eng. **155**, 105236 (2024) https://doi.org/10.1016/j.jtice.2023.105236

Zeng, L., Jin, Q., Lin, Z., Zheng, C., Wu, Y., Wu, X., Gao, X.: Dual-attention LSTM autoencoder for fault detection in industrial complex dynamic processes. Process Saf. Environ. Prot. **185**, 1145–1159 (2024) https://doi.org/10.1016/j.psep.2024.02.042