Adaptive Process Monitoring for Multimode Industrial Processes through Machine Learning

Liang Cao, Xiaolu Ji, Yankai Cao, Bhushan Gopaluni

***Abstract*—In complex industrial processes, real-time monitoring of critical variables is essential for ensuring operational safety and efficiency. Traditional process monitoring models often struggle with processes characterized by multiple operating modes, leading to decreased prediction accuracy and reliability. Existing methods typically require prior knowledge of the number of operating modes and cannot adapt to new modes that emerge over time, limiting their applicability in dynamic industrial environments. To address these challenges, we propose an adaptive process monitoring framework that automatically identifies operating modes using change point detection and classifies data using Gaussian mixture models. Specialized sub-soft sensor models are then constructed for each identified mode. This approach eliminates the need for prior knowledge of operating modes and enables the system to adapt to new operating conditions in real time. The effectiveness of the proposed methodology is demonstrated through a case study on the fluid catalytic cracking unit at the Parkland Refinery. The results show that our adaptive segmented model achieves an RMSE of 2.645 and an R^2 of 0.819, significantly outperforming the non-segmented model with an RMSE of 5.037 and a negative R^2 of -0.597. This adaptive framework enhances operational safety and efficiency by providing a robust and flexible monitoring solution for dynamically changing industrial processes.**

***Index Terms*—Change Point Detection, Gaussian Mixture Models, Process Monitoring, Multimode Industrial Processes, Machine Learning Modeling**

I. Introduction

In modern industrial automation and process control, particularly in sectors such as chemicals, petroleum, and semiconductors, real-time monitoring and prediction of key variables are crucial for ensuring safe and efficient production [1], [2]. Soft sensors, also known as inferential sensors, utilize existing process data and advanced algorithms to estimate critical variables that are either difficult or expensive to measure directly [3]. With real-time estimation of critical variables, soft sensors enable enhanced process monitoring and control, significantly improving safety and reliability by providing redundant measurements and early detection of potential equipment failures.

Recent advances in data-driven algorithms have significantly enhanced industrial process monitoring capabilities. Federated learning architectures enable robust monitoring across distributed edge devices while preserving data privacy [4], while graph neural networks have demonstrated effectiveness in analyzing complex time-series data from battery management systems [5].

Complementing these monitoring advances, significant progress has been made in fault diagnosis and process variable analysis. Research has established new methodologies for sensor fault detection in discrete-time systems [6] and introduced novel statistical process monitoring algorithms for handling high-dimensional variables [7]. These developments have been further strengthened by implementations leveraging both deep architectures [8] and advanced symbolic representations [9]. The innovations in matrix factorization techniques [10] and edge-intelligent architectures [11] have also improved the computational efficiency of process monitoring systems, particularly in resource-constrained environments. Additionally, attention-based deep learning approaches [12] have enhanced the robustness of monitoring systems when dealing with complex or noisy process conditions.

However, despite the potential benefits of soft sensors, their widespread implementation in industrial settings has been hindered by several significant challenges. Real-world industrial processes often experience multiple operating modes. These modes may be influenced by a variety of factors, including changes in raw material properties, equipment conditions, environmental variables, and operational set points [13], [14]. As illustrated in Figure 1, multiple operating modes can occur under such circumstances, and ignoring these modes can cause soft sensors to fail catastrophically.

To address the challenge of multiple operating modes, researchers have devoted considerable effort to develop multimode methods [15]–[20]. Numerous methods for phase segmentation have been introduced to separate the entire process into distinct operating modes, and then independent soft sensor models are trained for each mode. Although these approaches have shown potential, they still often rely on prior knowledge of the number of modes and cannot adapt if new modes emerge during system operation.

One key technique for autonomous mode identification is change point detection (CPD), a fundamental problem in statistical analysis and machine learning [21]–[23]. CPD does

This work was supported in part by Mitacs. (Corresponding author: R. Bhushan Gopaluni.)

Liang Cao is with Department of Chemical Engineering, Massachusetts Institute of Technology, Boston, 02139, United States(e-mail:liangcao@mit.edu).

Xiaolu Ji is with the Department of Statistics, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada(e-mail: xiaoluji@student.ubc.ca).

Yankai Cao, R. Bhushan Gopaluni are with the Department of Chemical and Biological Engineering, University of British Columbia, Vancouver, BC, V6T 1Z3, Canada(e-mail: yankai.cao@ubc.ca, bhushan.gopaluni@ubc.ca).

not require prior knowledge of the number of operating modes; it detects points in time where the dataâs underlying statistical properties shift significantly. This segmentation paves the way for building accurate sub-soft sensor models per operating mode.

Gaussian Mixture Models (GMMs), widely employed in data classification, can further refine the segmentation by probabilistically clustering samples into distinct modes. As new data arrives, GMMs classify the incoming samples into one of the existing modes, triggering the corresponding sub-soft sensor model [24]–[26].

The proposed framework integrates CPD and GMMs for robust pattern segmentation and classification in industrial processes. The key contributions of this research are:

1) Development of an autonomous mode detection framework based on change point detection that eliminates the need for prior knowledge of operating modes.
2) Design of an adaptive modeling approach that combines GMM classification with specialized sub-soft sensor models for each identified mode.
3) Implementation of a real-time monitoring system capable of detecting and modeling new operating conditions as they emerge.

The remainder of this paper is organized as follows. Section II introduces the methodology for detecting change points. Section III discusses Gaussian mixture models and their use in modeling and classifying segmented data patterns. Section IV presents the proposed adaptive process monitoring framework for multi-mode industrial processes. Section V demonstrates the effectiveness of the proposed method through a case study on a fluid catalytic cracking unit. Finally, Section VI concludes the work.

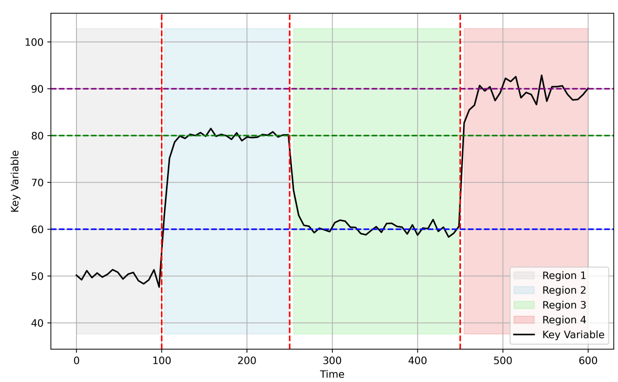

Fig. 1. An Illustration of Multiple Operating Modes

II. Change Point Detection and Soft Sensor Modeling

Let $\{X_1, X_2, \ldots, X_T\}$ denote a sequence of observations, where each X_t represents a random variable at time t. Assume that the data sequence can be partitioned into $k+1$ segments separated by k change points $\tau_1, \tau_2, \ldots, \tau_k$, where $1 \leq \tau_1 < \tau_2 < \cdots < \tau_k < T$. Each segment i (for $i = 1, 2, \ldots, k+1$) is characterized by a distinct probability distribution P_i with

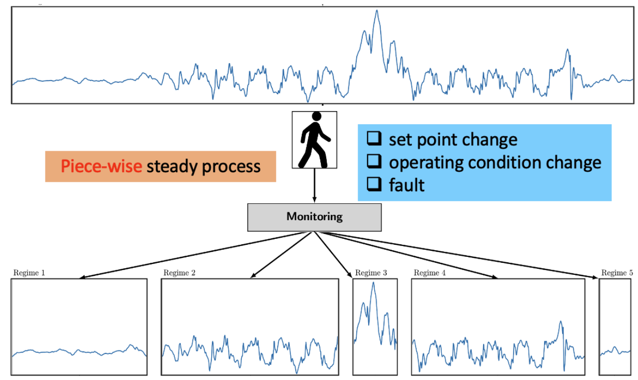

Fig. 2. An Example of Change Point Detection

parameters θ_i. An example of the change point detection is shown in Figure 2. Formally, this can be expressed as:

$$X_t \sim P_i(\theta_i) \quad \text{for} \quad \tau_{i-1} < t \leq \tau_i \tag{1}$$

where $\tau_0 = 0$ and $\tau_{k+1} = T$. To ensure the segmentation process captures meaningful data patterns, we also introduce a minimum segment length, $L_{\min}$, which prevents the creation of segments that are too short for reliable statistical modeling. In this work, we chose $L_{\min}$ by balancing two considerations: (1) providing sufficient samples within each segment to reliably train a sub-soft sensor model, and (2) avoiding over-segmentation where too many short segments could lead to spurious operating modes. We performed cross-validation using different candidate values of $L_{\min}$ to evaluate segmentation stability and model performance. In general, practitioners may adjust $L_{\min}$ based on the sampling rate and process characteristics to achieve a suitable trade-off between detection granularity and false-alarm avoidance. Each identified segment $\mathcal{D}_i = \{X_t, y_t\}_{t=\tau_{i-1}+1}^{\tau_i}$ must meet this minimum length criterion to avoid overfitting.

The primary objectives in change point detection are to determine the number k of change points, identify their locations $\{\tau_i\}$, and estimate the parameters $\{\theta_i\}$ for each segment. The joint likelihood of the entire data sequence, given the change points and their corresponding parameters, is the product of the likelihoods of each segment:

$$\mathcal{L}\left(\tau_{1:k}; \theta_{1:k+1} \mid X_1, X_2, \ldots, X_T\right) = \prod_{i=1}^{k+1} \prod_{t=\tau_{i-1}+1}^{\tau_i} f(X_t \mid \theta_i) \tag{2}$$

where $f(X_t \mid \theta_i)$ is the probability density (or mass) function of P_i. The goal is to identify the set of change points $\{\tau_1, \tau_2, \ldots, \tau_k\}$ and the corresponding parameters $\{\theta_1, \theta_2, \ldots, \theta_{k+1}\}$ that maximize the joint likelihood:

$$\{\hat{\tau}_{1:k}, \hat{\theta}_{1:k+1}\} = \arg \max_{\{\tau_i\},\{\theta_i\}} \mathcal{L}\left(\{\tau_i\}, \{\theta_i\} \mid X_{1:T}\right) \tag{3}$$

where $X_{1:T}$ denotes the entire data sequence. Directly maximizing the likelihood over all possible combinations of change points and parameters is computationally infeasible due to the combinatorial explosion as T increases. Therefore, various optimization strategies are employed to efficiently identify the optimal set of change points and corresponding parameters [21].

A. Online Change Point Detection

In dynamic industrial environments, it is crucial to detect changes in the process in real time as new data arrives. Online change point detection methods aim to identify points in time where the statistical properties of a data sequence change, without processing the entire dataset retrospectively. This capability allows the system to adapt promptly to new operating conditions, ensuring accurate predictions and reliable performance even when faced with unfamiliar process dynamics. An effective method for online change point detection is the Bayesian Online Change Point Detection (BOCPD) algorithm, introduced by Adams and MacKay [23].

The BOCPD algorithm maintains a run length variable r_t, which denotes the number of time steps since the last change point. At each time t, the algorithm updates the posterior distribution $P(r_t \mid X_{1:t})$ over the possible run lengths given the data observed so far. The probability of a change point occurring can be determined from $P(r_t \mid X_{1:t})$. It is updated using the following recursion:

$$P(r_t \mid X_{1:t}) = \frac{\sum_{r_{t-1}} P(r_t, r_{t-1}, X_t \mid X_{1:t-1})}{P(X_t \mid X_{1:t-1})} \tag{4}$$

where the numerator can be decomposed into terms involving the hazard function and the predictive probabilities:

$$\begin{aligned} P(r_t, r_{t-1}, X_t \mid X_{1:t-1}) = P(r_t \mid r_{t-1}) P(X_t \mid r_t, X_{t-r_t:t-1}) \\ P(r_{t-1} \mid X_{1:t-1}) \end{aligned} \tag{5}$$

The transition from r_{t-1} to r_t is governed by the hazard function $H(r_{t-1})$:

$$P(r_t \mid r_{t-1}) = \begin{cases} H(r_{t-1}) & \text{if } r_t = 0, \\ 1 - H(r_{t-1}) & \text{if } r_t = r_{t-1} + 1 \end{cases} \tag{6}$$

The hazard function $H(r_t)$ controls the expected duration of segments in the process. A common choice is to assume a constant hazard, where $H(r_t) = h$ and h is a small probability, representing the likelihood of a change point occurring at each time step. The predictive probability $P(X_t \mid r_t, X_{t-r_t:t-1})$ depends on the data since the last change point and is calculated using a parametric model for the data X_t. For each run length r_t, we maintain sufficient statistics ψ_{r_t} that summarize the data in the current segment, such as the mean and variance for Gaussian models. The predictive probability is then computed by integrating over all possible parameter values:

$$P(X_t \mid r_t, X_{t-r_t:t-1}) = \int P(X_t \mid \psi) P(\psi \mid X_{t-r_t:t-1}) d\psi \tag{7}$$

where ψ is the parameter of the assumed data model. If conjugate priors are used, this integral can often be computed analytically. This allows for efficient updates of the posterior run length distribution $P(r_t \mid X_{1:t})$ and the evidence $P(X_t \mid X_{1:t-1})$, ensuring that the algorithm remains computationally efficient in real-time scenarios. At each time t, we update the posterior run length distribution $P(r_t \mid X_{1:t})$ and compute the evidence $P(X_t \mid X_{1:t-1})$ as:

$$P(X_t \mid X_{1:t-1}) = \sum_{r_{t-1}} P(r_t, r_{t-1}, X_t \mid X_{1:t-1}) \tag{8}$$

We can then compute the probability of a change point occurring at time t as the probability that the run length resets to zero:

$$P(r_t = 0 \mid X_{1:t}) = \sum_{r_{t-1}} H(r_{t-1}) P(r_{t-1} \mid X_{1:t-1}) \tag{9}$$

If this probability exceeds a predefined threshold δ, we declare a change point at time t. In this work, we select the threshold δ to balance the trade-off between detection sensitivity and false alarm rate. Specifically, a higher δ reduces false positives but may miss true change points, whereas a lower δ captures more changes but risks over-segmentation. We first leveraged domain knowledge of typical process variability to narrow down a plausible range for δ. We then performed empirical tuning on a validation set to minimize both false positives and missed detections. In real industrial applications, δ can also be periodically revisited or dynamically adjusted according to factors like seasonal shifts or scheduled maintenance intervals.

B. Soft Sensor Modeling per Segment

Building upon the change point detection results, we can develop a framework for multi sub-soft sensors that adapts to different operating modes identified in the process. This approach allows us to capture the unique characteristics of each segment, potentially improving overall prediction accuracy.

Let's consider the time series data $\{X_t\}_{t=1}^T$ partitioned into $k+1$ segments by k change points $\{\tau_1, \tau_2, \ldots, \tau_k\}$. For each segment i, we construct a specialized soft sensor model using the input-output pairs within segment.

$$\mathcal{D}_i = \{(\mathbf{x}_t, y_t) \mid \tau_{i-1} < t \leq \tau_i\} \tag{10}$$

where $\mathbf{x}_t \in R^m$ is the vector of input features (different with X_t) and $y_t \in R$ is the target variable. The specialized soft sensor model define a function $f_i : R^m \to R$ that maps the inputs to the output:

$$\hat{y}_t = f_i(\mathbf{x}_t; \boldsymbol{\theta}_i), \quad \tau_{i-1} < t \leq \tau_i \tag{11}$$

where $\hat{y}_t$ is the estimated output and $\boldsymbol{\theta}_i$ are the parameters specific to segment i. For each segment i, find the optimal parameters $\boldsymbol{\theta}_i^*$ by solving the optimization problem:

$$\boldsymbol{\theta}_i^* = \arg\min_{\boldsymbol{\theta}_i} \frac{1}{n_i} \sum_{t=\tau_{i-1}+1}^{\tau_i} \ell(y_t, f_i(\mathbf{x}_t; \boldsymbol{\theta}_i)) + \gamma_i \mathcal{R}(\boldsymbol{\theta}_i) \tag{12}$$

where $n_i = \tau_i - \tau_{i-1}$ is the number of observations in segment i, $\ell(\cdot)$ is the loss function, $\mathcal{R}(\cdot)$ is the regularization term, and γ_i is the regularization strength for segment i.

III. Gaussian Mixture Models

In this section, GMMs are used to model the probability distributions of distinct data segments identified by change point detection and to classify new data points based on the fitted distributions.

A. Gaussian Mixture Model Framework

Within each segment i, we assume that the data $\{X_t\}_{t=\tau_{i-1}+1}^{\tau_i}$ are generated from a mixture of multiple Gaussian distributions. The probability density function for a data point X_t in the segment i is expressed as:

$$p(X_t \mid \lambda_i) = \sum_{m=1}^{M_i} \pi_{i,m} \mathcal{N}(X_t \mid \boldsymbol{\mu}_{i,m}, \boldsymbol{\Sigma}_{i,m}) \tag{13}$$

where $\mathcal{N}(X_t \mid \boldsymbol{\mu}_{i,m}, \boldsymbol{\Sigma}_{i,m})$ denotes the Gaussian distribution with mean $\boldsymbol{\mu}_{i,m}$ and covariance $\boldsymbol{\Sigma}_{i,m}$, $m = 1, 2, \ldots, M_i$. The weights $\pi_{i,m}$ denote the proportion of each component within segment i, satisfying $\sum_{m=1}^{M_i} \pi_{i,m} = 1$. $\lambda_i = \{\pi_{i,m}, \boldsymbol{\mu}_{i,m}, \boldsymbol{\Sigma}_{i,m}\}_{m=1}^{M_i}$ encapsulates all parameters of the GMM for segment i.

Estimating the parameters λ_i involves determining the mixture weights $\pi_{i,m}$, means $\boldsymbol{\mu}_{i,m}$, and covariances $\boldsymbol{\Sigma}_{i,m}$ that best fit the data within the segment i. The expectation maximization (EM) algorithm is commonly used for this purpose due to its effectiveness in handling latent variables associated with mixture models [25]. We assign random values to the means, covariances, and mixture weights, repeating the EM algorithm multiple times with different random seeds to select the best solution.

Selecting the appropriate number of mixture components M_i for each segment i is essential to balance model complexity and goodness of fit. Techniques such as the Bayesian Information Criterion (BIC) can be used to compare models with different numbers of components [27]. The model that minimizes the chosen criterion is typically selected:

$$\text{BIC} = -2 \ln \mathcal{L} + p_i \ln n_i \tag{14}$$

where $\mathcal{L}$ is the maximized likelihood of the model, p_i is the number of estimated parameters in segment i, and n_i is the number of observations in segment i.

In the expectation step (E-Step), we compute the posterior probabilities (responsibilities) $\gamma_{i,m}(X_t)$ that each data point X_t (for $t = \tau_{i-1} + 1, \ldots, \tau_i$) belongs to each Gaussian component m within segment i:

$$\gamma_{i,m}(X_t) = \frac{\pi_{i,m} \mathcal{N}(X_t \mid \boldsymbol{\mu}_{i,m}, \boldsymbol{\Sigma}_{i,m})}{\sum_{m'=1}^{M_i} \pi_{i,m'} \mathcal{N}(X_t \mid \boldsymbol{\mu}_{i,m'}, \boldsymbol{\Sigma}_{i,m'})} \tag{15}$$

In the maximization step (M-Step), using the responsibilities computed in the E-Step, we update the parameters:

$$\pi_{i,m}^{\text{new}} = \frac{1}{n_i} \sum_{t=\tau_{i-1}+1}^{\tau_i} \gamma_{i,m}(X_t), \tag{16}$$

$$\boldsymbol{\mu}_{i,m}^{\text{new}} = \frac{\sum_{t=\tau_{i-1}+1}^{\tau_i} \gamma_{i,m}(X_t) X_t}{\sum_{t=\tau_{i-1}+1}^{\tau_i} \gamma_{i,m}(X_t)}, \tag{17}$$

$$\boldsymbol{\Sigma}_{i,m}^{\text{new}} = \frac{\sum_{t=\tau_{i-1}+1}^{\tau_i} \gamma_{i,m}(X_t)(X_t - \boldsymbol{\mu}_{i,m}^{\text{new}})(X_t - \boldsymbol{\mu}_{i,m}^{\text{new}})^\top}{\sum_{t=\tau_{i-1}+1}^{\tau_i} \gamma_{i,m}(X_t)} \tag{18}$$

where $n_i = \tau_i - \tau_{i-1}$ is the number of observations in segment i. These steps are performed iteratively until convergence. After convergence, GMM can partition the data points into different Gaussian components. We can calculate the posterior probability of each data point X_t (i.e., the probability that it belongs to a particular Gaussian component) to determine its most likely component. Based on these probabilities, we can further classify and model new data points.

B. Classification of New Data Points

GMM can be applied to classification when we have some pre-labeled training data. When new data arrive, we can determine to which segment a new observation most likely belongs by calculating the probability density of that data point under each mode.

For a new data point X_{new}, we first compute the log probability density $\ln p\left(X_{\text{new}} \mid \lambda_i\right)$ under each segment's GMM. This log probability is then exponentiated to obtain the actual probability density:

$$p\left(X_{\text{new}} \mid \lambda_i\right) = \exp\left(\ln p\left(X_{\text{new}} \mid \lambda_i\right)\right) \tag{19}$$

After computing the probability densities for each GMM, the new data point is assigned to the segment i^* whose GMM yields the highest probability density:

$$i^* = \arg\max_i \, p\left(X_{\text{new}} \mid \lambda_i\right) \tag{20}$$

Then the corresponding segment-specific soft sensor model can be used for prediction:

$$\hat{y}_{\text{new}} = f_{i^*}\left(X_{\text{new}}; \boldsymbol{\theta}_{i^*}^*\right) \tag{21}$$

IV. Adaptive Process Monitoring for Multi-Model Industrial Processes

This section introduces an adaptive process monitoring framework for multi-model industrial processes. It provides a systematic approach to handling multi-model industrial processes. By integrating change point detection, GMM-based classification, and segment-specific soft sensor models, the framework adapts to dynamic changes in the process, maintains high prediction accuracy, and improves operational safety and efficiency. Figure 3 illustrates the framework of multiple operating modes in industrial processes.

Algorithm 1 details the offline phase, which processes historical data to establish initial models for different operating modes. This phase begins by initializing empty lists for change points and segment models. It then iterates through the historical data, accumulating points in a window until the minimum segment length is reached. For each window, it computes a change point score and, if this score exceeds a predefined threshold, it marks a new change point. When a change point is detected, the algorithm extracts the segment data, builds a soft sensor model for that segment, fits a GMM to the segment data, and stores these models. This process continues until all historical data have been processed, resulting in a set of initial models that capture the different operating modes present in the historical data.

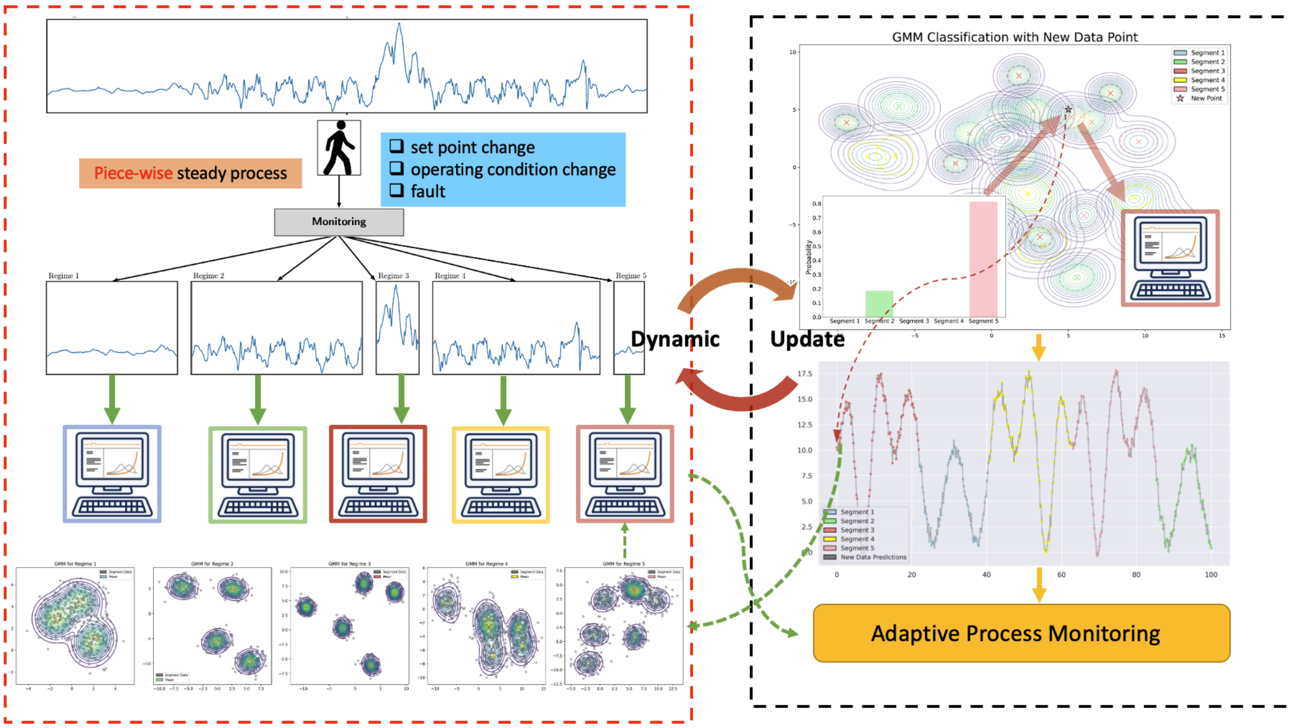

Fig. 3. Flowchart of Adaptive Process Monitoring for Muliphase Industrial Processes

The online phase, described in Algorithm 2, processes new data points in real-time, enabling the framework to adapt to new operating conditions and maintain prediction accuracy over time. As new data points arrive, the algorithm computes a change point score. If a new change point is detected, indicating a potential new operating mode, the algorithm creates a new soft sensor model and GMM for this segment and adds them to the model list. If no change point is detected, the algorithm classifies the new data point using existing GMMs and uses the corresponding soft sensor model to make a prediction. This adaptive approach allows the framework to continuously update its understanding of the process, incorporating new operating modes as they emerge, and ensuring that predictions remain accurate even as process conditions evolve.

Over extended operating periods, industrial processes may exhibit gradual drifts due to equipment wear, catalyst deactivation, or shifts in feedstock quality. Our framework inherently addresses such drift by continually monitoring for changes in the data distribution via change point detection. However, even within a detected mode, the model could degrade slowly over time. To mitigate this, we periodically evaluate the prediction performance of each sub-soft sensor model; if the error exceeds a specified threshold for a prolonged duration, partial retraining is triggered using a rolling window of recent data. This strategy ensures that sub-soft sensor models remain up-to-date with evolving process characteristics while maintaining computational feasibility. Such periodic or performance-based retraining offers a practical means to manage long-term model drift in real industrial deployments.

Although the proposed framework prioritizes accuracy and adaptability, interpretability of the sub-soft sensor models also plays a pivotal role in industrial decision-making. By examining feature importance metrics or techniques such as SHAP (SHapley Additive exPlanations) [28] within each mode-specific model, stakeholders can identify the most influential variables under different operating conditions. This insight empowers process engineers to perform targeted interventions, such as adjusting feed composition or modifying reactor settings, thereby facilitating root-cause analysis and process optimization. Furthermore, mode-specific sub-soft sensor outputs can be combined with operational context (e.g., known feed characteristics or temperature ranges), enabling more transparent and explainable monitoring results.

Moreover, integrating our adaptive monitoring framework with industrial control systems opens up opportunities for proactive interventions. For instance, if the monitoring system detects a shift to a high-variability mode, advanced process controllers (such as model predictive control) can automatically adjust control parameters to maintain product quality and minimize deviations. Similarly, predictions from the sub-soft sensor models can serve as feedforward signals to controllers, enabling timely tuning or setpoint changes that better accommodate transient disturbances. This synergy between real-time monitoring and closed-loop control strategies can ultimately enhance both process stability and operational efficiency in multimode industrial environments.

V. Case Study

In this case study, we demonstrate the application of adaptive process monitoring in the Fluid Catalytic Cracking (FCC) Unit at the Parkland Refinery in British Columbia, Canada [29].

Algorithm 1 Adaptive Process Monitoring: Offline Phase

Input: Training data $\{(\mathbf{x}_t, y_t)\}_{t=1}^{T}$
Hyperparameters: Regularization coefficient γ, Minimum segment length L_{min}, Change point detection threshold δ
Output: Change point list CP; Segment models list M
begin
  Initialize change point list CP $\leftarrow [0]$,
  Initialize segment models list M $\leftarrow [\ ]$,
  Initialize window $W \leftarrow [\ ]$, Set segment index $i \leftarrow 1$
  for *data point* $(\mathbf{x}_t, y_t)$ **do**
    Append $(\mathbf{x}_t, y_t)$ to W
    if *len*$(W) \geq L_{min}$ **then**
      score $\leftarrow$ Compute Change Point Score(W)
      if *score* $> \delta$ **then**
        new_CP $\leftarrow$ len(W)
        Append new_CP to CP
        Extract segment data $X_{\text{segment}} \leftarrow W$
        Build soft sensor model $f_i \leftarrow f(X_{\text{segment}})$
        Fit GMM $\lambda_i \leftarrow \text{GMM}(X_{\text{segment}})$
        Append (f_i, λ_i) to M
        Increment segment index $i \leftarrow i+1$
        Reset $W \leftarrow [\ ]$
      end
    end
  end
  return CP M
end

Algorithm 2 Adaptive Process Monitoring: Online Phase

Input: New data points $\mathbf{x}_{\text{new}}$
Hyperparameters: Regularization coefficient γ, Minimum segment length L_{min}, Change point detection threshold δ
Change point list CP; Segment models list M
Output: Predictions $\hat{y}$
begin
  Initialize window $W \leftarrow [\ ]$
  Initialize predictions $\hat{y} \leftarrow [\ ]$
  for *new data* $\mathbf{x}_{new}$ **do**
    Append $\mathbf{x}_{\text{new}}$ to W
    if *len*$(W) \geq L_{min}$ **then**
      score $\leftarrow$ Compute Change Point Score(W)
      if *score* $> \delta$ **then**
        Append new_CP to CP
        Extract segment data $X_{\text{new\_segment}} \leftarrow W$
        Build model $f_{\text{new}} \leftarrow f(X_{\text{new\_segment}})$
        Fit GMM $\lambda_{\text{new}} \leftarrow \text{GMM}(X_{\text{new\_segment}})$
        Append $(f_{\text{new}}, \lambda_{\text{new}})$ to M
        Reset $W \leftarrow [\ ]$
      end
      else
        Classify segment $i^* \leftarrow \text{Classify}(\mathbf{x}_{\text{new}}, \text{M})$
        Predict target $\hat{y}_{\text{new}} \leftarrow \text{M}[i^*].\text{model}(\mathbf{x}_{\text{new}})$
        Append $\hat{y}_{\text{new}}$ to $\hat{y}$
      end
    end
  end
  return $\hat{y}$ CP M
end

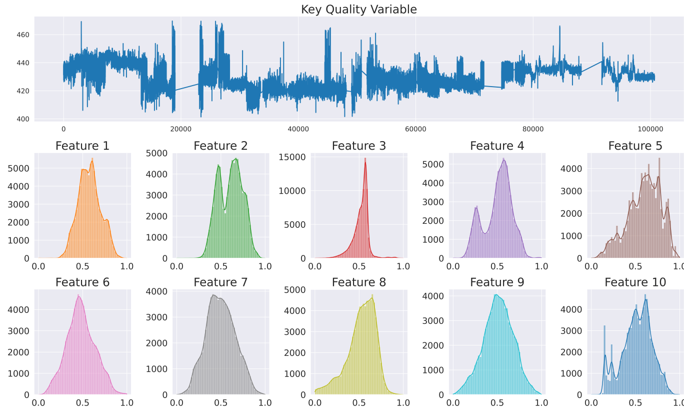

Fig. 4. Visualization of Selected Process Variables at the FCC Unit

The FCC unit is a crucial component in refining operations, converting heavy hydrocarbons into lighter compounds that serve as the basis for various petroleum products. The complex interactions within the FCC process are characterized by multi-phase systems, making it an ideal candidate for adaptive process monitoring. In the FCC process, the distillation temperature serves as a crucial quality indicator. The input variables are chosen based on process expertise and consist of easily measurable factors that affect the distillation temperature. A dataset containing 83,833 samples, gathered from January 2019 to October 2024, is utilized. This dataset is divided, with 80% used for training and the remaining 20% for testing purposes. Before applying the proposed framework, we performed essential data preprocessing steps to ensure data consistency and improve model training. Specifically, all input variables were normalized to have zero mean and unit variance. We also employed the $3-\sigma$ rule to detect and remove outliers. Figure 4 illustrates the distillation temperature values along with the distribution of the normalized selected input variables.

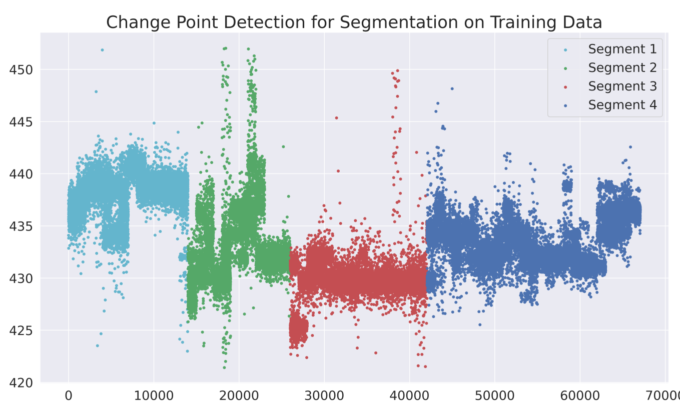

Fig. 5. Change Point Detection and Data Segmentation Result

We enhanced the soft sensor to recognize different modes of industrial processes by applying change point detection techniques. Figure 5 shows the results on the training data, where 4 different segments are identified, each corresponding to a different operating mode. Using these segments identified through change point detection, we employ a GMM to classify the training data and online change point detection to find new modes. Figures 5 and 6 show how the use of GMM to classify the data into different segments.

As we can see in Figure 6, it detected a new segment (Segment 5) based on the online change point detection mentioned in Algorithm 2. Although five modes are detected in this case study, the proposed framework is not inherently restricted to five modes. The change point detection algorithm can adaptively identify any number of operating modes as they emerge. However, as the number of modes increases, a larger dataset is generally required to train each sub-soft sensor model reliably. Practitioners should consider data availability, model complexity, and computational resources when applying this framework to processes with many operating modes.

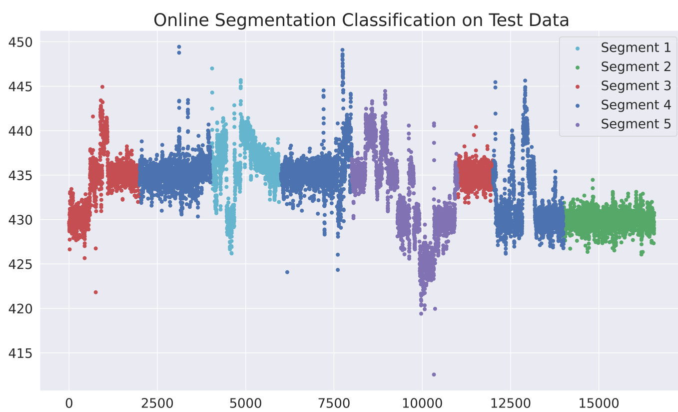

Fig. 6. GMM Classification and Online Change Point Detection for Segmentation

Beyond the clear improvements in predictive accuracy, segment-specific modeling also provides valuable operational insights. By analyzing key statistical properties within each detected mode (e.g., mean, variance, and dominant input features), process engineers can discern how varying feed compositions or catalyst conditions translate into distinct operating regimes. This added interpretability underscores the practical benefits of adaptive segmentation for both diagnosis and optimization in industrial contexts.

During the testing phase, the test data is classified using these trained GMM models. By calculating the likelihood of each test data point in different segments, the data point can be assigned to the segment corresponding to the maximum likelihood value. Figure 7 shows the classification probability heatmap of test data across the 5 different segments.

For each identified segment, we developed a sub-soft sensor model. The sub-soft sensor model can better capture the unique dynamics of each operating mode according to each segment. To further improve the effectiveness of the segmented soft sensor model, we compared the performance of different types of machine learning models under adaptive segmented and non-segmented data conditions. These models include ElasticNet [30]; Support Vector Machine (SVM) [31]; tree-based models: Decision Tree [32], Random Forest [33], Extra Trees [34], XGBoost [35], CatBoost [35], LightGBM [35]; neighbor-based models: K-Neighbors [36]; deep learning models: Long Short-Term Memory (LSTM) [37] and Transformer [38].

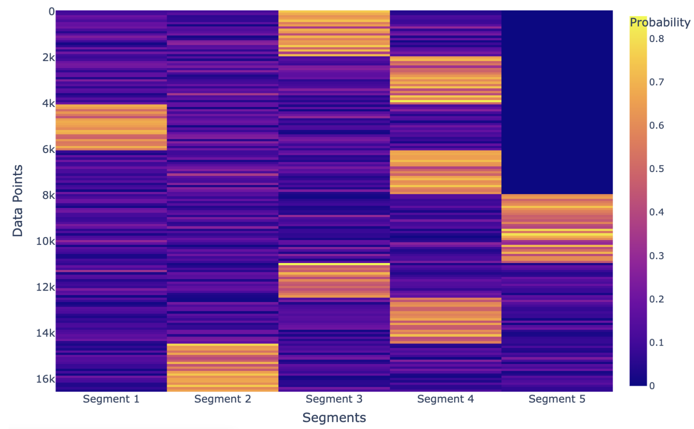

Fig. 7. Classification Probability of Test Data on Each Segment

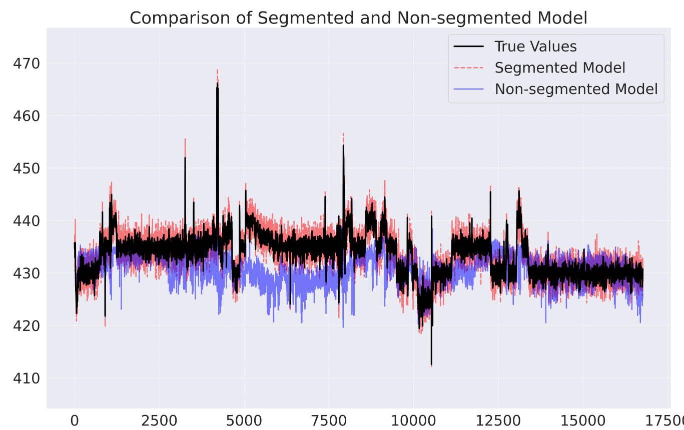

Fig. 8. Extra Trees Soft Sensor on Segmented and Non-segmented Data

We use Root Mean Square Error (RMSE) and R^2 (coefficient of determination) on the test dataset to evaluate the performance. The results summarized in Table I clearly show that the performance of the adaptive segmented model is significantly better than that of the non-segmented model across all metrics and models tested.

It is worth noting that the Extra Trees regressor has the lowest RMSE of 2.645 and the highest R^2 of 0.819 in the adaptive segmented case, while its RMSE is 5.037 with a negative R^2 of -0.597 in the non-segmented case. This represents a substantial improvement in predictive performance. Figure 8 shows the results for the Extra Trees soft sensor model. All models show negative or very low R^2 values in the non-segmented case, indicating poor fit, while the adaptive segmented approach consistently yields positive and much higher R^2 values, demonstrating a marked improvement in model explanatory power. This pattern holds for all 11 models tested, underscoring the effectiveness of the adaptive segmentation strategy.

To assess the real-time feasibility of the proposed framework, we also evaluated its computational performance. Specifically, we measured the average processing time per

TABLE I
COMPARISON OF DIFFERENT SOFT SENSORS ON TEST DATA

	Non-Segmented Model		Adaptive Segmented Model	
	RMSE	**R^2**	**RMSE**	**R^2**
Extra Trees	5.037	-0.597	**2.645**	**0.819**
Transformer	5.863	-0.842	3.176	0.781
LightGBM	5.786	-0.549	3.253	0.734
LSTM	5.892	-0.242	3.471	0.729
SVM	6.014	-0.583	3.589	0.704
CatBoost	6.176	0.093	3.697	0.691
XGBoost	6.594	0.017	3.736	0.688
K-Neighbors	6.872	-0.437	3.874	0.675
Random Forest	7.146	0.007	3.908	0.672
ElasticNet	7.388	0.121	3.954	0.668
Decision Tree	8.129	-0.762	3.985	0.651

incoming data point, accounting for change point detection, GMM classification, and sub-soft sensor prediction. On our experimental platform (an Intel Core i7 CPU with 16 GB RAM), the framework processes each data point in approximately 0.1 seconds, which is suitable for many industrial scenarios where sampling periods often range from seconds to minutes. We also observed that the most time-consuming step is the GMM classification (0.08 seconds), suggesting that parallelization or incremental GMM updates may further improve real-time responsiveness.

VI. CONCLUSION

We presented an adaptive process monitoring framework for multimode industrial processes that addresses the limitations of traditional methods in dynamic operating environments. By integrating change point detection for automatic identification of operating modes and Gaussian mixture models for data classification, the framework constructs specialized sub-soft sensor models tailored to each mode. The case study on the fluid catalytic cracking unit demonstrates the effectiveness of the proposed approach, showing significant improvements in prediction performance across various machine learning models when compared to non-segmented models. The adaptive framework not only improves prediction accuracy but also improves operational safety and efficiency by providing a robust and flexible monitoring system capable of adapting to the dynamic nature of industrial processes. Future research could explore integrating this framework with advanced process control systems, extending the methodology to handle high-dimensional data, and investigating its robustness to noisy and variable industrial conditions.

ACKNOWLEDGMENT

We gratefully acknowledge the financial support from Mitacs. We also thank the colleagues at Parkland Refinery for their valuable collaboration and support.

REFERENCES

[1] R. B. Gopaluni, A. Tulsyan, B. Chachuat *et al.*, “Modern machine learning tools for monitoring and control of industrial processes: A survey,” *IFAC-PapersOnLine*, vol. 53, no. 2, p. 218â229, 2020.

[2] N. P. Lawrence, S. K. Damarla, J. W. Kim, A. Tulsyan, F. Amjad, K. Wang, B. Chachuat, J. M. Lee, B. Huang, and R. Bhushan Gopaluni, “Machine learning for industrial sensing and control: A survey and practical perspective,” *Control Engineering Practice*, vol. 145, p. 105841, 2024.

[3] P. Kadlec, B. Gabrys, and S. Strandt, “Data-driven soft sensors in the process industry,” *Computers & Chemical Engineering*, vol. 33, no. 4, pp. 795–814, 2009.

[4] A. Arunan, Y. Qin, X. Li, and C. Yuen, “A federated learning-based industrial health prognostics for heterogeneous edge devices using matched feature extraction,” *IEEE Transactions on Automation Science and Engineering*, 2023.

[5] K. Q. Zhou, Y. Qin, and C. Yuen, “Graph neural network-based lithium-ion battery state of health estimation using partial discharging curve,” *Journal of Energy Storage*, vol. 100, p. 113502, 2024.

[6] D. Zhao, D. Shen, and Y. Wang, “Fault diagnosis and compensation for two-dimensional discrete time systems with sensor faults and time-varying delays,” *International Journal of Robust and Nonlinear Control*, vol. 27, no. 16, pp. 3296–3320, 2017.

[7] Z. Lou, Y. Wang, Y. Si, and S. Lu, “A novel multivariate statistical process monitoring algorithm: Orthonormal subspace analysis,” *Automatica*, vol. 138, p. 110148, 2022.

[8] C. Shang, F. Yang, D. Huang, and W. Lyu, “Data-driven soft sensor development based on deep learning technique,” *Journal of Process Control*, vol. 24, no. 3, pp. 223–233, 2014.

[9] B. Yang, S. Hong, H. Wang, and D. Zhao, “Complex dynamic slow independent symbol aggregation approximation for multimodal temporal fuzzy cognitive map,” *IEEE Transactions on Industrial Informatics*, 2024.

[10] J. Shang and T. Chen, “Unilateral orthogonal nonnegative matrix factorization,” *SIAM Journal on Control and Optimization*, vol. 61, no. 4, pp. 2497–2519, 2023.

[11] P. Kumar and S. Kumar, “Edge of transfer learning-based long short-term memory neural networks in the application of battery surface temperature prediction for electric vehicles,” *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, vol. 5, no. 4, pp. 1529–1536, 2024.

[12] S. Dash and N. C. Sahoo, “Attention-based deep learning approach for nonintrusive and simultaneous state detection of multiple appliances in smart buildings,” *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, vol. 5, no. 3, pp. 1248–1258, 2023.

[13] K. Kuang, P. Cui, S. Athey, R. Xiong, and B. Li, “Stable prediction across unknown environments,” in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD ’18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1617â1626. [Online]. Available: https://doi.org/10.1145/3219819.3220082

[14] L. Cao, F. Yu, F. Yang, Y. Cao, and R. B. Gopaluni, “Data-driven dynamic inferential sensors based on causality analysis,” *Control Engineering Practice*, vol. 104, p. 104626, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967066120301969

[15] Y. He, L. Yao, X. Zhang, X. Kong, and Z. Song, “Causality-driven

sequence segmentation for enhancing multiphase industrial process data analysis and soft sensing," *arXiv preprint arXiv:2407.05954*, 2024.

[16] H. Wu and J. Zhao, "Self-adaptive deep learning for multimode process monitoring," *Computers & chemical engineering*, vol. 141, p. 107024, 2020.

[17] L. Zhou, J. Zheng, Z. Ge, Z. Song, and S. Shan, "Multimode process monitoring based on switching autoregressive dynamic latent variable model," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 10, pp. 8184–8194, 2018.

[18] S. J. Zhao, J. Zhang, and Y. M. Xu, "Performance monitoring of processes with multiple operating modes through multiple pls models," *Journal of process Control*, vol. 16, no. 7, pp. 763–772, 2006.

[19] J. Liu, T. Liu, and J. Chen, "Sequential local-based gaussian mixture model for monitoring multiphase batch processes," *Chemical engineering science*, vol. 181, pp. 101–113, 2018.

[20] K. Wang, L. Rippon, J. Chen, Z. Song, and R. B. Gopaluni, "Data-driven dynamic modeling and online monitoring for multiphase and multimode batch processes with uneven batch durations," *Industrial & Engineering Chemistry Research*, vol. 58, no. 30, pp. 13 628–13 641, 2019.

[21] C. Truong, L. Oudre, and N. Vayatis, "Selective review of offline change point detection methods," *Signal Processing*, vol. 167, p. 107299, 2020.

[22] S. Aminikhanghahi and D. J. Cook, "A survey of methods for time series change point detection," *Knowledge and information systems*, vol. 51, no. 2, pp. 339–367, 2017.

[23] R. P. Adams and D. J. MacKay, "Bayesian online changepoint detection," *arXiv preprint arXiv:0710.3742*, 2007.

[24] J. Yang, X. Liao, X. Yuan, P. Llull, D. J. Brady, G. Sapiro, and L. Carin, "Compressive sensing by learning a gaussian mixture model from measurements," *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 106–119, 2014.

[25] G. Xuan, W. Zhang, and P. Chai, "Em algorithms of gaussian mixture model and hidden markov model," in *Proceedings 2001 international conference on image processing (Cat. No. 01CH37205)*, vol. 1. IEEE, 2001, pp. 145–148.

[26] H. Permuter, J. Francos, and I. Jermyn, "A study of gaussian mixture models of color and texture features for image classification and segmentation," *Pattern recognition*, vol. 39, no. 4, pp. 695–706, 2006.

[27] J. Kuha, "Aic and bic: Comparisons of assumptions and performance," *Sociological Methods & Research*, vol. 33, no. 2, pp. 188–229, 2004. [Online]. Available: https://doi.org/10.1177/0049124103262065

[28] L. Cao, J. Su, J. Saddler, Y. Cao, Y. Wang, G. Lee, L. C. Siang, R. Pinchuk, J. Li, and R. B. Gopaluni, "Real-time tracking of renewable carbon content with ai-aided approaches during co-processing of biofeedstocks," *Applied Energy*, vol. 360, p. 122815, 2024.

[29] J. Su, L. Cao, G. Lee, B. Gopaluni, L. C. Siang, Y. Cao, S. van Dyk, R. Pinchuk, and J. Saddler, "Tracking the green coke production when co-processing lipids at a commercial fluid catalytic cracker (fcc): combining isotope 14c and causal discovery analysis," *Sustainable Energy & Fuels*, vol. 6, pp. 5600–5607, 2022.

[30] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 67, no. 2, pp. 301–320, 2005.

[31] C. Shang, X. Gao, F. Yang, and D. Huang, "Novel bayesian framework for dynamic soft sensor based on support vector machine with finite impulse response," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1550–1557, 2014.

[32] S. B. Kotsiantis, "Decision trees: a recent overview," *Artificial Intelligence Review*, vol. 39, no. 4, pp. 261–283, 2013.

[33] Q. Cheng, Z. Chunhong, and L. Qianglin, "Development and application of random forest regression soft sensor model for treating domestic wastewater in a sequencing batch reactor," *Scientific Reports*, vol. 13, no. 1, p. 9149, 2023.

[34] L. Cao, X. Ji, Y. Cao, Y. Li, L. C. Siang, J. Li, V. K. Pediredla, and R. B. Gopaluni, "Interpretable soft sensors using extremely randomized trees and shap," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 8000–8005, 2023.

[35] J. M. Ahn, J. Kim, and K. Kim, "Ensemble machine learning of gradient boosting (xgboost, lightgbm, catboost) and attention-based cnn-lstm for harmful algal blooms forecasting," *Toxins*, vol. 15, no. 10, p. 608, 2023.

[36] G. Lin, A. Lin, and D. Gu, "Using support vector regression and k-nearest neighbors for short-term traffic flow prediction based on maximal information coefficient," *Information Sciences*, vol. 608, pp. 517–531, 2022.

[37] B. Y. . H. LeCun, Y., "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[38] L. Cao, X. Ji, Y. Cao, Y. Luo, Y. Wang, L. C. Siang, J. Li, and R. B. Gopaluni, "Interpretable industrial soft sensor design based on informer and shap," *IFAC-PapersOnLine*, vol. 58, no. 14, pp. 73–78, 2024.