Methodology for Important Sensor Screening for Fault Detection and Classification in Semiconductor Manufacturing

Feng Zhu
City University of Hong Kong
15 PUBLICATIONS **96** CITATIONS
SEE PROFILE

Xiaodong Jia
University of Cincinnati
71 PUBLICATIONS **2,052** CITATIONS
SEE PROFILE

Marcella Miller
University of Cincinnati
9 PUBLICATIONS **69** CITATIONS
SEE PROFILE

Xiang Li
Xi'an Jiaotong University
107 PUBLICATIONS **5,969** CITATIONS
SEE PROFILE

Methodology for Important Sensor Screening for Fault Detection and Classification in Semiconductor Manufacturing

Feng Zhu, Xiaodong Jia, Marcella Miller, Xiang Li, Fei Li, Yinglu Wang, and Jay Lee

***Abstract*—Feature design and selection is challenging because of huge data volume and high-mix production systems. Most engineers still rely on human experts to suggest the specific sensor channel and specific time frames of data from which to design the features. This study proposes a novel approach for important sensor screening to prioritize the useful sensor channels for FDC model development in semiconductor manufacturing. The proposed method can be used as a preprocessing step prior to feature extraction, and the selected sensor channels can be leveraged by process engineers for finer feature design. In this research, firstly, time series alignment kernels (TSAKs) are proposed to handle multivariate trace data. Then, the proposed method combines 5 different time series alignment kernels (TSAKs) with a feature selection algorithm, minimum Redundancy Maximum Relevance (mRMR), to identify the important sensor channels. Furthermore, a TSAK+Kernel Principal Component Analysis (KPCA) algorithm is proposed for a visualization tool. Lastly, the TSAK+Support Vector Machine (SVM) is employed for results validation. In this study, validation of the proposed method is based on both open-source datasets and the proprietary datasets from a real production line.**

***Index Terms*—Semiconductor, time series alignment kernel, sensor selection, fault detection and classification.**

I. Introduction

FAULT Detection and Classification (FDC) is an integral part of Advanced Process Control (APC) in semiconductor manufacturing [1]. The main objective of FDC is to monitor and analyze the variations in the process data to detect anomalies and to identify the potential root causes. Recently, driven by the increasing miniaturization of semiconductor products, tighter process control at a wafer-to-wafer level is highly desired by the practitioners. Therefore, FDC based quality control is also becoming popular. In FDC, the quality of all wafers is controlled based on the process measurements, so that every wafer can be covered without additional metrology devices, and the potential metrology delay can be avoided.

In the literature, various data-driven techniques are explored and proposed to build FDC models in semiconductor manufacturing. These methods include Principal Component Analysis (PCA) [2], k-Nearest Neighbor (kNN) [3], Mahalanobis distance [4], and one-class SVM (OS-SVM) [5], etc. By reviewing these works, it is found that the modeling techniques and the data-driven models are extensively discussed in the literature. However, feature extraction and selection, as a critical step to convert raw data into modellable features, is still less attended. Typical feature extraction techniques include 1) feature transformation, such Discrete Fourier transform (DFT), Discrete Wavelet Transform (DWT). 2) dimension reduction, such as PCA, Kernel PCA, etc., and 3) descriptive summary statistics, such as mean, standard deviation, max, min, etc. After feature extraction, feature selection techniques develop a selection criterion to highlight the important features for modeling. To the author's experience, the quality of features is one of the most important factors that can significantly influence the performance of an FDC model. The importance of feature extraction and selection is justified in one of our recent publications in [6]. In this study, the statistical features based on 1) whole trace; 2) recipe steps; and 3) signal patterns are benchmarked in the application of in-situ particle monitoring for etching chambers. It is found that the pattern-based features demonstrate significant improvements in the results compared with the other two sets of features. Therefore, the importance of feature extraction must be highlighted in FDC model development.

However, designing useful and reliable features from process measurements is not an easy task due to the following challenges. 1) The data volume can be extremely large, especially when hundreds of sensors are monitored in the manufacturing processes; 2) The diversity of products in the high-mix production systems makes the feature design even more difficult. Even worse, most engineers still rely on human experts to suggest the specific sensor channel and specific time frames of data from which to design the features. This will significantly lower the efficiency of model development and might inadvertently eliminate crucial sensory information. Therefore, it is imperative to have a methodology to quantify, visualize, and validate the sensor importance before building the FDC models. Also, it is better to rate the sensor importance based on the sensory readings directly than on the extracted features, because the human intervention in the

Manuscript received August 4, 2020; revised September 14, 2020 and October 24, 2020; accepted November 6, 2020. Date of publication November 10, 2020; date of current version February 3, 2021. *(Corresponding author: Xiaodong Jia.)*

The authors are with the Department of Mechanical and Materials Engineering, University of Cincinnati, Cincinnati, OH 45220 USA (e-mail: jiaxg@ucmail.uc.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TSM.2020.3037085.

Digital Object Identifier 10.1109/TSM.2020.3037085

feature extraction process will lead to subjective evaluation results. Therefore, different with the well-formulated research tasks in feature transformation, dimension reduction, statistical feature extraction and feature selection, this research focuses on the identifying the important sensor channels that are relevant to a product failure.

This article explores a novel approach for important sensor screening to prioritize the useful sensor channels for FDC applications. The proposed method can quickly scan hundreds of sensor channels to rate the usefulness of each sensor channel and to highlight the important ones for finer feature design. To verify the sensor selection results, a visualization tool is designed in this study to directly visualize the usefulness of the individual sensor channel or the selected sensor group. As a summary, the contributions and novelties of this research can be summarized as follows. 1) Time Series Alignment Kernels (TSAKs), including Dynamic Time Warping (DTW), Dynamic Time Alignment Kernel (DTAK), Global Alignment (GA) kernel and Triangular Global Alignment (TGA) kernel, are investigated and benchmarked. The characteristics of each TSAK are summarized and reviewed. 2) A novel algorithm that combines TSAK and minimum Redundancy Maximum Relevance (mRMR) is proposed to select important sensor channels. In addition, a TSAK+Support Vector Machine(SVM) approach is employed to cross-validate the results and to build the classifier; 3) Data visualization is explored based on TSAK and Kernel Principal Component Analysis (KPCA) to visualize the selected sensors on the manifold.

The rest of this article is organized as follows. Section II reviews relevant techniques for feature extraction and selection and FDC in semiconductor manufacturing; Section III elaborates the proposed methodology. In Section IV, two case studies are demonstrated. The concluding remarks are stated in Section V.

II. Literature Review

FDC has been widely studied in the semiconductor industry. He and Wang [3] proposed a k-Nearest Neighbor (kNN) based FD method which can detect anomalies by evaluating the distance of observation compared to the normal operating region. Yu [8] investigated a Principal Components (PCs)-based Gaussian Mixture Model (GMM) (PCGMM) to cope with the non-linearity in most batch processes and processes steps with variable duration. Verdier and Ferreira [4] proposed an adaptive Mahalanobis distance to detect anomalies in non-Gaussian distribution. Compared with Euclidean distance, this kind of distance is more reliable based on the local covariance structure of the observations under monitoring. He and Jin [9] proposed a fast pattern recognition based FD method by combining PCA and kNN as an ensemble model, which is capable of processing a large number of variables and is easily implemented for online process monitoring. Mahadevan and Shah [5] proposed an approach for FD based on One-class SVM (OCSVM). The use of kernel functions makes the OS-SVM model more robust to capture and model the non-linearity in the system.

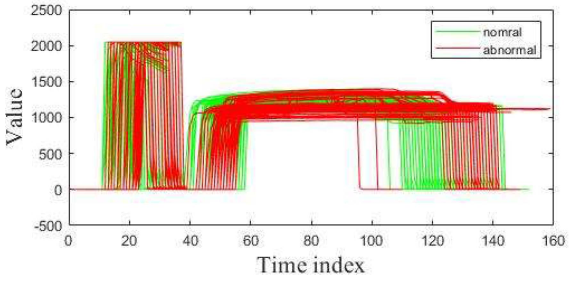

Fig. 1. Sensor signals of normal and abnormal data sets for the 405 nm parameter in the wafer database [17]: the time length of two wafer runs is different; moreover, they are not aligned in the time axis, such that tradition similarity functions and kernel functions, such as Euclidean distance, which assumes the ith point in one sequence is aligned with the ith point in the other, will produce a pessimistic dissimilarity measure.

Feature selection is also widely studied in the semiconductor industry. For flat features, where features are assumed to be independent, there are three groups of feature selection models: filter models, wrapper models, and embedded models [9]. Filter models rely on characteristics of data, which can evaluate features without utilizing any classification [10]. In filter models, firstly, feature evaluation ranks feature based on certain criteria. Then the features with the highest-ranking are chosen for the classification modeling. There are a number of criteria have been proposed for filter-based feature selection such as Fisher score [11], methods based on mutual information [12], and ReliefF and its variants [13], [14]. Because filter models evaluate features individually, they cannot handle feature redundancy, which means that features are selected independently from any specific classifiers. This kind of model ignores the effects of the selected feature subset on the performance of the induction algorithm [15]. To handle this issue, wrapper models utilize a specific classifier to evaluate the feature subset. Based on a predefined classifier, a typical wrapper model will perform as follows: 1) define a feature subset and evaluate its performance based on the classifier; 2) repeat until the desired quality is reached. However, the wrapper models need to run the classifier multiple times, which is very computationally expensive. Embedded models, which embed feature evaluation with classifier construction, have the advantages of both wrapper models and filter models. Recursive feature elimination using SVM [16] is a typical embedding model. However, for embedding models, in order to get the coefficient of each feature, only linear classifiers based on a linear combination of features, such as SVM with a linear kernel and logistic regression, can be used. This type of classifier is difficult to apply to temporal data. For filter models, the selected feature subset contains redundancy such that the subset may not have good performance like the subset selected by wrapper methods. Therefore, we propose the minimum Redundancy Maximum Relevance (mRMR) framework, which is a powerful feature selection method that selects the important features by trading-off between relevance and redundancy.

In addition, the time variations among different wafer runs must be properly handled when extracting features from the raw trace. Common similarity functions and kernel functions cannot properly handle the time variation. DTW has been identified as an effective method in [3] to consider the time variation in semiconductor manufacturing. However, DTW suffers several disadvantages: 1) it is very computationally expensive; 2) it does not satisfy the triangular inequality; 3) it is not a positive definite (p.d) kernel and thus cannot be plugged into SVM or other kernel-based machine learning algorithms for model development. To fix these disadvantages of DTW, DTW-SC [18] is proposed to speed up the DTW algorithm. DTAK is proposed in [19] to construct a new kernel that can be plugged into SVM. The study found that DTW, DTW-SC and DTAK are not ensured to be p.d kernels. To meet the admissible condition of SVM and other kernel-based machine learning algorithms, GA and TGA kernel are further proposed in [20] and [21] as a p.d kernel.

As a summary, the current research limitations that are addressed in this study include:

1) Lack of methodology for sensor selection, sensor importance quantification, visualization, and validation before developing FDC models for semiconductor manufacturing process.

2) Commonly used FDC algorithms, such as OCSVM, kNN, and GMM, can only take vectorized input rather than multivariate trace data. Therefore, feature extraction, which requires tremendous trial-and-error efforts, is required to pre-process the trace data into feature vectors.

3) Though DTW+SVM provides a seemingly practical solution to build detectors with time-series input directly, an investigation in [19] indicates that the kernel matrix (or gram matrix) constructed by DTW is non-p.d and it fails to meet the admissible condition of SVM and most kernelized machine learning algorithms.

III. Methodology

Based on the current limitations, the objectives of this study are to propose a methodology for sensor selection, sensor importance quantification, visualization, and validation. The proposed method serves as a preliminary sensor screening tool before building FDC models.

The core novelty in the proposed method is an innovative way to construct the kernel matrix in most kernelized machine learning algorithms by using Time Series Alignment Kernel (TSAK). TSAK is different from traditional kernels since it takes multivariate trace data as algorithm inputs, and it can measure that discrepancy between two multivariate time series with different lengths. Traditional kernels, such as linear, polynomial, and Gaussian kernel, can only take vectorized input with equal lengths. Also, they cannot take multivariate trace data as algorithm inputs directly. Based on the kernel matrix constructed by TSAK, three combined algorithms are further devised to attain the objectives of this study. 1) the TSAK+mRMR is exploited for sensor selection and sensor importance quantification; 2) TSAK+KPCA is employed for sensor importance visualization; 3) TSAK+SVM is utilized to validate the sensor importance ratings.

A. Time Series Alignment Kernel

Let π be the alignment between two discrete time series x= $(x_1, \ldots, x_n)$ and y= $(y_1, \ldots, y_m)$ of lengths n and m respectively. Then, π is a pair of increasing vectors $(\pi^{\mathrm{x}}, \pi^{\mathrm{y}})$ of length $p \leq n+m-1$, where $1 = \pi_1^{\mathrm{x}} \leq \cdots \leq \pi_p^{\mathrm{x}} = n$ and $1 = \pi_1^{\mathrm{y}} \leq \cdots \leq \pi_p^{\mathrm{y}} = m$ are the warping path for x and y respectively. Based on the alignment π, the DTW distance is defined as:

$$DTW(\mathrm{x}, \mathrm{y}) \overset{\mathrm{def}}{=} \min_{\pi \in \mathcal{A}(\mathrm{x},\mathrm{y})} \left(\sum\nolimits_{i=1}^{p} \left\| \mathrm{x}\left(\pi_i^{\mathrm{x}}\right) - \mathrm{y}\left(\pi_i^{\mathrm{y}}\right) \right\|_{l_2} \right) \tag{1}$$

where $\mathcal{A}(\mathrm{x}, \mathrm{y})$ is the set of all possible alignment paths and $\|\cdot\|_{l_2}$ is the Euclidean distance. According to [22], [23], calculation of $k_{DTW}(\mathrm{x}, \mathrm{y})$ in Eq.(1) amounts to performing the following computation between x and y:

$$M_{i,j} = \min\left(M_{i-1,j},\ M_{i-1,j-1}, M_{i,j-1}\right) + \left\| x_i - y_j \right\|_{l_2} \tag{2}$$

The resultant $M_{m,n}$ is the DTW distance between x and y. It is noted that the Euclidean distance term in Eq.(1) and Eq.(2) can be replaced by any other distance measure $\varphi\left(\mathrm{x}\left(\pi_i^{\mathrm{x}}\right), \mathrm{y}\left(\pi_i^{\mathrm{y}}\right)\right)$.

DTW has been extensively studied since was first proposed in [23], and many variants are suggested to enhance its performance. In [19], Sakeo and Chiba proposed a constrained alignments method to speed up the algorithm by restricting $\mathcal{A}(\mathrm{x}, \mathrm{y})$ to a small subset of all alignments that are close to the diagonal. This is achieved by introducing a band constraint:

$$d(\mathrm{x}, \mathrm{y}) \overset{\mathrm{def}}{=} \min_{\pi \in \mathcal{A}(\mathrm{x},\mathrm{y})} \left(\sum\nolimits_{i=1}^{p} \varphi\left(\mathrm{x}\left(\pi_i^{\mathrm{x}}\right), \mathrm{y}\left(\pi_i^{\mathrm{y}}\right)\right) \cdot r(\pi_i^{\mathrm{x}},\ \pi_i^{\mathrm{y}}) \right)$$
$$r\left(\pi_i^{\mathbf{x}},\ \pi_i^{\mathrm{y}}\right) = \begin{cases} 1 \text{ if } \left|\pi_i^{\mathrm{x}} - \pi_i^{\mathrm{y}}\right| < T \\ \infty \text{ if } \left|\pi_i^{\mathrm{x}} - \pi_i^{\mathrm{y}}\right| \geq T \end{cases} \tag{3}$$

where T is the user-defined parameter that defines the largest deviation against the diagonal entries. In the following discussion, DTW in Eq.(3) will be mentioned as DTW-SC.

Although DTW is effective to measure the discrepancy between time series, it does not satisfy the triangle inequality. Therefore, the Gram matrix or Kernel matrix that is constructed by DTW is non-positive definite (n.p.d), and it fails to meet the admissible condition of an SVM kernel. This can be fixed by adding regularization terms to the Gram matrix [21]. An alternative solution is presented in [19], which is known as Dynamic Time Alignment Kernel (DTAK) in the literature. Comparing with DTW, DTAK replaces the distance measure in Eq.(1) and Eq.(2) with local kernels, and the minimization problem in DTW becomes maximization.

$$k_{DTAK}(\mathrm{x}, \mathrm{y}) \overset{\mathrm{def}}{=} \max_{\pi \in \mathcal{A}(\mathrm{x},\mathrm{y})} \left(\sum\nolimits_{i=1}^{p} k_L\left(\mathrm{x}\left(\pi_i^{\mathrm{x}}\right), \mathrm{y}\left(\pi_i^{\mathrm{y}}\right)\right) \right) \tag{4}$$

Eq.(2) should be modified as

$$M_{i,j} = \max\left(M_{i-1,j},\ M_{i-1,j-1}, M_{i,j-1}\right) + k_L\left(x_i, y_j\right) \tag{5}$$

where $k_L\left(x_i, y_j\right)$ denotes the local kernel, which can be a polynomial kernel, linear kernel, Gaussian kernel, etc. According to [19], the DTAK is not an admissible kernel of SVM. However, it is symmetric and satisfies the Cauchy-Schwartz like inequality. Moreover, it can be understood as the inner product of the two time series.

The author in [20] proposes a Global Alignment (GA) kernel for time series, which is guaranteed to be a p.d kernel for time series, and thus it is admissible to SVM. The GA kernel can be described as below:

$$\begin{aligned} k_{GA}(\mathrm{x}, \mathrm{y}) &\overset{\text{def}}{=} \sum_{\pi \in \mathcal{A}(\mathrm{x},\mathrm{y})} e^{-\sum_{i=1}^{p} \varphi\left(\mathrm{x}(\pi_i^{\mathrm{x}}), \mathrm{y}(\pi_i^{\mathrm{y}})\right)} \\ &= \sum_{\pi \in \mathcal{A}(\mathrm{x},\mathrm{y})} \prod_{i=1}^{p} e^{-\varphi\left(\mathrm{x}(\pi_i^{\mathrm{x}}), \mathrm{y}(\pi_i^{\mathrm{y}})\right)} \\ &= \sum_{\pi \in \mathcal{A}(\mathrm{x},\mathrm{y})} \prod_{i=1}^{p} k_L\left(\mathrm{x}(\pi_i^{\mathrm{x}}), \mathrm{y}(\pi_i^{\mathrm{y}})\right) \end{aligned} \tag{6}$$

Unlike k_{DTW} and k_{DTAK}, k_{GA} is defined as the summation of exponentiated distance over all possible alignment paths, which will not exist the cost and risk of the bad wrapper path. The theoretical discussion in [20] indicates the p.d property of k_{GA} is ensured in theory. The computation of GA kernel can be stated as:

$$M_{i,j} = \left(M_{i-1,j} + M_{i-1,j-1} + M_{i,j-1}\right) \cdot k_L\left(x_i, y_j\right) \tag{7}$$

where $M_{m,n}$ is the GA distance between x and y.

Later investigation in [24] reported that the k_{GA} may produce a diagonal dominant Gram matrix where the trace of the matrix is much larger than the sum of off-diagonal entries. Discussions and numerical experiments in [21] demonstrates that the issue of diagonal dominance is not prominent if the input time sequences have similar lengths. In the semiconductor application, this is valid since the length of trace data over different wafer cycles only varies in a small range. Therefore, GA can be a good option for constructing a supervised classifier by using SVM. To better address the diagonal dominance phenomenon in the Gram matrix, [21] further proposes a Triangular GA (TGA) kernel. Interested readers can refer to [21] for more details about TGA.

As a summary, this subsection section revisited five time series alignment kernels – regularized DTW, regularized DTW-SC, regularized DTAK, GA, and TGA. Before moving to the next subsection, it is important to note the following details for implementation. 1) According to [21], the Gram matrix of the training fold produced by DTW, DTW-SC, and DTAK maybe not positive definite. We can regularize it with a ridge to ensure it becomes positive definite, which can be described by $\widetilde{\mathrm{K}} = \mathrm{K} + \lambda \mathrm{I}$; 2) GA is useful to compare time series with similar lengths, and it is ensured to be a p.d kernel; 3) if one sequence is more than two times longer than the other sequence, GA may produce a diagonal dominant Gram matrix; 4) DTW-SC and TGA improve the efficiency of DTW and GA respectively by adding additional constraints to the alignment; 5) Regarding the parameter tuning strategy for TSAK+SVM, readers can follow the instructions given in [21, Table 2]; 6) For all five of these kernels, the normalized counterparts $\tilde{k}(\mathrm{x}, \mathrm{y})$ can be described by $k(\mathrm{x}, \mathrm{y})/\sqrt{k(\mathrm{x}, \mathrm{x}) \cdot k(\mathrm{y}, \mathrm{y})}$. In the following discussion, we will employ $\tilde{k}_{GA}(\mathrm{x}, \mathrm{y})$ for algorithm development. 7) Since TSAK measure the similarity between the different wafer runs directly, each considered sensor channel is treated equally for classification modeling.

Algorithm 1 TSAK+mRMR for Sensor Selection

```
Input: 1) Dataset DS = $\{(\mathrm{X}_i, c_i)\}_{i=1}^{N}$;
       2) User-defined parameters $m$, $\alpha$
Output: Sensor set ss
$\mathrm{ss}_{All} \leftarrow \{1, 2, \ldots, G\}$
ss←Ø // selected sensors set by mRMR
$\mathrm{ss}_{\alpha}$ ← Ø // A set for candidate sensors
      for each $\mathrm{S}_j = \{\mathrm{t}_i^j\}_{i=1,\ldots,N}$ do
        $\mathrm{d}^j \leftarrow \emptyset$
        for $i = 1$ to $N$ do
(I)       $d_i^j = k_{TSAK}(\mathrm{t}_{baseline}^j, \mathrm{t}_i^j)$
        end for
        $F(\mathrm{S}_j, \mathrm{c}) = Relevance(\mathrm{d}^j, \mathrm{c})//\mathrm{Eq.}(8)$
      end for
      for $n = 1$ to $round(\alpha G)$ do
(II)    $\mathrm{ss}_{\alpha} \leftarrow \mathrm{ss}_{\alpha} \cup \arg\max_{j \in \mathrm{ss}_{All} \setminus \mathrm{ss}_{\alpha}} F(\mathrm{S}_j, \mathrm{c})$
      end for
      $\mathrm{ss} \leftarrow \arg\max_{j \in \mathrm{ss}_{ALL}} F(\mathrm{S}_j, \mathrm{c})$
      While $length(\mathrm{ss}) < m$ do
        for each $b \in \mathrm{ss}_{\alpha} \setminus \mathrm{ss}$ do
          $\mathrm{ss}' \leftarrow \mathrm{ss} \cup b$
(III)     $W_{TSAK}^{(b)} = \frac{1}{length(\mathrm{ss}')^2} \sum_{p,q \in \mathrm{ss}'} R(\mathrm{S}_p, \mathrm{S}_q)//\mathrm{Eq.}(9)$
          $V_F^{(b)} = \frac{1}{length(\mathrm{ss}')} \sum_{j \in \mathrm{ss}'} F(\mathrm{S}_j, \mathrm{c})$
        end for
        ss ← $\mathrm{ss} \cup \arg\max_b(\frac{V_F^{(b)}}{W_{TSAK}^{(b)}})$
      end while
return ss
```

B. TSAK+mRMR for Direct Sensor Selection

The minimum Redundancy Maximum Relevance (mRMR) is a powerful feature selection method that selects the important features by trading-off between relevance and redundancy for the flat feature. Generally, in mRMR, an F-statistic is used to describe the correlation of selected features with target labels. The feature redundancy is calibrated by evaluating the relevance between features. The optimization is done by performing a greedy search through the feature space. In this study, we preserve the idea of the mRMR algorithm and adapt it to handle multivariate sensor signals.

Let us denote by $\mathrm{DS} = \{\mathrm{X}_i, c_i\}_{i=1,\ldots,N}$ the given dataset for important sensor selection. $\mathrm{X}_i \in \mathbb{R}^{G \times T_i}$ denotes the trace matrix for the i-th wafer run with G sensor channels of length T_i. N is the total number of wafer runs. $c_i \in \{1, .., H\}$ represents the class label for sample i, where H is the total number of classes. $\mathrm{c} = \{c_i\}_{i=1,\ldots,N}$ represents the class label for all samples. Each row in X_i is a time series of length T_i, and we use t_i^j to denote the j-th sensor reading for the i-th wafer run. Then $\mathrm{S}_j = \{\mathrm{t}_i^j\}_{i=1,\ldots,N}$is a set of j-th sensor readings for all the wafers in **DS**. Based on the notation, the proposed algorithm for important sensor selection can be described as:

In Algorithm 1, the pseudo-code in Section (I) ranks the sensor importance individually based on the F-statistic in the below equation:

$$\mathrm{F}(\mathrm{S}_j,\mathrm{c})=\frac{\sum\limits_{h=1}^{H} n_h\left(\bar{\mathrm{d}}^j_{(h)}-\bar{\mathrm{d}}^j\right)^2/(H-1)}{\sum\limits_{h=1}^{H}\sum\limits_{l=1}^{n_h}\left(\mathrm{d}^j_{l,(h)}-\bar{\mathrm{d}}^j_{(h)}\right)^2/(N-H)} \tag{8}$$

where $\mathrm{d}^j=\{d_i^j\}_{\ i=1,\ldots,N}$ is a set of TSAK distances between trace series t_i^j and $\mathrm{t}^j_{baseline}$, as shown in Algorithm 1. It is important to note that the $\mathrm{t}^j_{baseline}$ can be one trace series or set of multiple trace series. In the latter case, d_i^j in Algorithm 1 will be the averaged TSAK distance.

d^j in Eq.(8) represents the kernel distance between current trace data and the baseline data, as defined in Section (I) in Algorithm 1. The superscript j represents the j-th sensor channel in the trace data. The $k_{TSAK}(\cdot,\ \cdot)$ in algorithms 1 is a similarity measure that quantifies the similarity between the two input time series. When the two input time series are similar, a large output value is expected, and vice versa. For different TSAKs, the definition of $k_{TSAK}(\cdot,\ \cdot)$ is different, as shown in Table I. It is also noted that the $k_{TSAK}(\cdot,\ \cdot)$ for DTAK, GA and TGA can be written as inner product of two input time series.

The F-statistic in Eq.(8) is a variance ratio, where the numerator is inter-class variance, and the denominator is the intra-class variance. If the clustering tendency in the kernel distance d^j indicates a stronger correlation with the class labels, a larger F-statistic value is observed. Therefore, the F-statistic in Eq.(8) measures the relevance of the sensor clustering tendency with the class labels.

Section (II) in Algorithm 1 establishes a candidate feature set ss_α for subsequent selection. The user-defined parameter α controls the sizes of the candidate feature set ss_α. Section (III) in Algorithm 1 selects the important features one at a time by maximizing relevance while minimizing the redundancy. The iteration starts with the most important feature that is identified in code Section (I). $W^{(b)}_{TSAK}$ evaluates the redundancy score at b-th for a group of sensors in ss′. $R(\mathrm{S}_p, \mathrm{S}_q)$ calculates the redundancy score between any pair of sensors in the set ss′.

$$R\left(\mathrm{S}_p\ ,\mathrm{S}_q\right)=\frac{1}{N}\sum_{i=1}^{N}k_{TSAK}(\mathrm{t}_i^p,\mathrm{t}_i^q) \tag{9}$$

where $V_F^{(b)}$ evaluates the relevance score of the sensor group ss' toward the target class labels.

In addition, it is important to note that sensor normalization is a required pre-processing step to make a fair judgment of different sensor channels. The sensor normalization step is written as:

$$\mathrm{t}_i^p=(\bar{\mathbf{t}}_i^p-m_i^p)/\sigma_i^p \tag{10}$$

where $\bar{\mathbf{t}}_i^p$ is the original time series for sensor p and sample i, and m_i^p and σ_i^p are the mean and standard deviation of the sensor readings respectively.

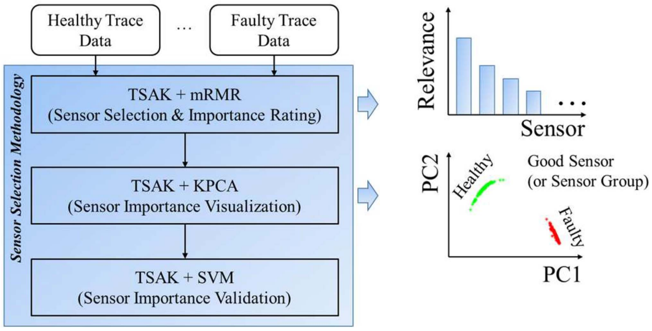

Fig. 2. Flow chart for the proposed methodology.

C. TSAK+KPCA for Sensor Importance Visualization

In this subsection, TSAK+KPCA is explored for sensor (group) importance visualization, which can provide a visual summary of the samples makes it easier to identify trends and patterns with your data. It is not directly related to the classification modeling. This algorithm seeks to obtain a mapping ϕ:$\mathrm{X}_i \in \mathbb{R}^{s\times T_i} \rightarrow \mathrm{V}_i \in \mathbb{R}^d$, which maps the input trace X_i onto the embedded manifold. X_i is the trace matrix for wafer i with $G \geq s \geq 1$ different sensor channels of length T_i. $\mathrm{V}_i \in \mathbb{R}^d$ is the projection of X_i onto the manifold and with dimensionality d. The usefulness of the sensor (group) can be visualized by plotting the $\{\mathrm{V}_i\}_{i=1,\ldots,N}$ against class labels. If there is a clear separation between different classes, the selected sensor (group) is important. Otherwise, it is not important. It is important to note that the KPCA is selected for its simplicity and popularity. The proposed algorithm for sensor importance visualization can be easily extended to other kernel-based dimension reduction methods, such as a diffusion map, Laplacian eigenmaps, etc.

Given set $\mathrm{X}=\{\mathrm{X}_i\}_{i=1,\ldots,N}$, key steps in TSAK + KPCA can be stated as follows.

1) Construct an $N\times N$ Gram matrix with elements $K_{ij}=k_{TSAK}\left(x_i, x_j\right)$;

2) Center the Gram matrix $\overline{\mathrm{K}}=\mathrm{K}-\mathrm{KE}-\mathrm{EK}+\mathrm{EKE}$, where $\mathbf{E}_{ij}$=$1/N$;

3) Eigenvalue decomposition $\lambda\alpha\ =\ \left(\frac{1}{N}\right)\overline{\mathrm{K}}\alpha$, where $\alpha_{\ 1},\ \alpha_{\ 2},\ \ldots,\ \alpha_{\ N}\in\mathbb{R}^N$ are the eigenvectors.

4) Given new input X_*, compute $v_*^k=\sum\limits_{i=1}^{N}\alpha_i^k k_{TSAK}(\mathrm{X}_i,\mathrm{X}_*)$ for $k=1,\ldots,d$. Then $\mathrm{V}_*=\left[v_*^1,\ \ldots,v_*^d\right]^{\mathrm{T}}$ is the principal component of the input X_*.

D. Proposed Methodology

An overview of the proposed methodology for sensor selection is presented in Fig. 2. The proposed methodology consists of three parts. 1) TSAK+mRMR for sensor selection. The expected output from this algorithm is a subset of selected sensor channels, and the relevancy in mRMR can be used as sensor importance ratings; 2) TSAK+KPCA is employed for visual confirmation of the sensor importance. The selected sensor (group) is expected to give clear separation among different classes in the manifold; 3) TSAK+SVM is used for cross-validation to further verify the sensor important. We compare

TABLE I
TSAKs IN THIS ARTICLE AND THE TUNING PARAMETERS

Kernel	Property	Math Description	Params.
DTW	pseudo p.d	$k_{DTW} \overset{\text{def}}{=} e^{-\frac{1}{t}DTW}$	t
DTW-SC	pseudo p.d	$k_{DTW_{SC}} \overset{\text{def}}{=} e^{-\frac{1}{t}DTW_{SC}}$	t, T
DTAK	pseudo p.d	$k_{DTAK}(\boldsymbol{x}, \boldsymbol{y}) \overset{\text{def}}{=}$ $\underset{\pi \in \mathcal{A}(\mathbf{x},\mathbf{y})}{max} \left(\sum_{i=1}^{p} k_\sigma \left(\boldsymbol{x}(\pi_i^{\mathbf{x}}), \boldsymbol{y}(\pi_i^{\mathbf{y}})\right)\right),$ Where $k_\sigma(x, y) = \exp\left(-\frac{\lVert x-y \rVert_{l_2}}{\sigma}\right)$	t, σ
GA	p.d	$k_{GA}(\mathbf{x}, \mathbf{y}) \overset{\text{def}}{=}$ $\sum_{\pi \in \mathcal{A}(\mathbf{x},\mathbf{y})} \prod_{i=1}^{p} k_L \left(\mathbf{x}(\pi_i^{\mathbf{x}}), \mathbf{y}(\pi_i^{\mathbf{y}})\right),$	σ
		Where $k_L = \frac{k_\sigma}{2-k_\sigma}$ is employed to fix diagonal dominance	
TGA	p.d	$k_{TGA}(\mathbf{x}, \mathbf{y}) \overset{\text{def}}{=}$ $\sum_{\pi \in \mathcal{A}(\mathbf{x},\mathbf{y})} \prod_{i=1}^{p} k_L \left(\boldsymbol{x}(\pi_i^{\mathbf{x}}), \boldsymbol{y}(\pi_i^{\mathbf{y}})\right),$ Where $k_L = \frac{w \cdot k_\sigma}{2-w \cdot k_\sigma}$ and $w\left(\pi_i^{\mathbf{x}}, \pi_i^{\mathbf{y}}\right) = \left(1 - \frac{\lvert \pi_i^{\mathbf{x}} - \pi_i^{\mathbf{y}} \rvert}{T}\right)_+$ limits the possible alignment paths close to the diagonal.	σ, T

the classification accuracy before and after sensor selection to validates the effectiveness of the proposed TSAK+mRMR sensor selection algorithm. It is noted that the whole methodology is a supervised learning, which requires the labelled dataset. With satisfactory accuracy given by TSAK+SVM in the cross-validation, the user can always improve the detection and classification accuracy by designing detailed features from the prioritized trace channels. A systematic strategy for parameter tuning for different TSAKs is given in [21, Table 2].

Table I tabulates the TSAKs that were employed in the experiments and corresponding tuning parameter for a summary. It is important to note that these three algorithms in the proposed methodology share the same kernel parameter. Users can find the optimal kernel parameters based on the cross-validation results given by TSAK+SVM. Regarding the parameter range, the author strictly follows the range that is suggested in [21, Table 2].

In semiconductor applications, the proposed method is an effective sensor screening approach that should be used before applying finer statistical analysis on the trace signal. The proposed methodology can give an objective sensor importance rating without extracting features from the raw trace. More generally, the GA kernel and TGA kernel can be potentially useful for other whole trace analysis applications, which will be explored in future investigations. It is also noted that the KPCA and SVM in the proposed method in Fig. 2 can be replaced by any other kernel-based learning algorithms for dimension reduction and classification.

To show the superiority of the proposed methodology, we employ the different feature extraction and selection methods in Fig. 3 to for the benchmark. Three feature sets are generated using various methods, including the frequency-domain features given by DFT (FeaSet #1), statistical features (FeaSet #2), and Principal components (PCs) given by PCA (FeaSet #3). For FeaSet #3, the data resampling technique is employed to ensure that all samples' time length is consistent, then PCs of each sensor are extracted as features.

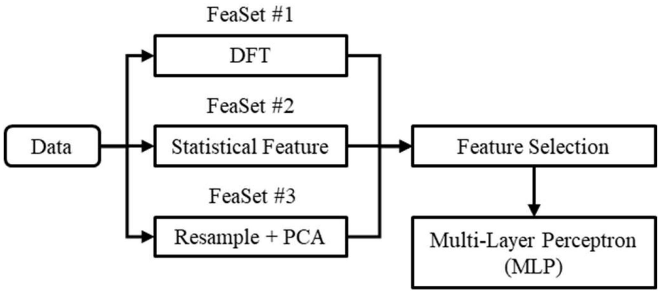

Fig. 3. Flow chart for the traditional feature extraction and selection.

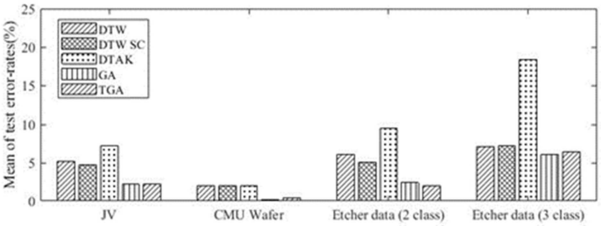

Fig. 4. Mean classification error rates on test folds, using a 3 folds 4 repeats cross-validation procedure for each kernel/data pair. Parameters were selected independently for each test iteration by applying an adaptive grid search within each training fold, using 3 folds 3 repeats cross-validation.

The traditional mRMR framework is employed for feature selection to find the critical features and their corresponding sensor channels. We call this framework as feature extraction based sensor selection. Multi-Layer Perceptron (MLP), which is the most commonly used feedforward neural network, is used for cross-validation to verify the feature quality and classification model performance.

IV. Results and Discussion

A. Case Study 1: Validation of TSAK+SVM

Three different data sets are employed for algorithm validation. The first one is the Japanese Vowel data set, which can be downloaded from the University of California at Irvine (UCI) data repository. In this dataset, nine male speakers uttered two Japanese Vowels /ae/ successively. For a detailed data description, interested readers can refer to [25]. The second dataset is a collection of in-line process control measurements recorded from an etching process that is donated by Carnegie Mellon University (CMU). Details about this dataset can be found in [17]. This data is widely studied in the literature for algorithm validation and benchmarking [26], [27]. The third dataset in this study is collected from a HITACHI Conductor Etch System 9000 Series, which is the same dataset that is described in our previous work in [6]. In this dataset, over 300 trace signals are collected from roughly 1000 to 1500 wafer runs, and the objective is to monitor the chamber particles based on process measurements. This dataset contains the run to fail trace records collected from three different etching chambers. To create a manageable dataset for algorithm validation, we randomly took 100 wafer runs right after the chamber cleaning activities, which are labeled as healthy, and another 100 wafer runs prior to the chamber cleaning that is labeled as faulty. The trace data from another 100 wafer runs in the middle of the

TABLE II
MEAN TYPE I AND TYPE II ERROR IN 3-FOLD CROSS-VALIDATION

	Mean type I error of TSAK+SVM				
	DTW	DTW SC	DTAK	GA	TGA
CMU Wafer	0%	0%	0%	0%	0%
Etcher 2-class	2.51%	2.01%	6.55%	0%	0%
	Mean type II error of TSAK+SVM				
CMU Wafer	2.01%	2.01%	2.01%	0.17%	0.41%
Etcher 2-class	3.51%	3.02%	3.02%	2.50%	1.99%

particle accumulation is also included as the 3*rd* class and is labeled as medium. As a summary, 4 different datasets are considered separately in this validation experiment: 1) JV dataset from UCI data repository; 2) CMU data for semiconductor etching; 3) Etcher dataset with healthy and faulty; 4) Etcher dataset with healthy, faulty and medium. It is important to note that the first two datasets are public, and the latter two datasets are proprietary.

To evaluate the performance of each kernel, mean classification errors from 3-fold cross-validation are employed, as shown in Fig. 4. Since CMU dataset and etcher dataset only contain healthy and faulty data, false alarm rate (type I error) and miss alarm rate (type II error) have also been provided in Table II for the dichotomous problem. These results in Fig. 4 and Table II suggest the following findings: 1) GA and TGA kernels indicate the lowest classification error rates on all the datasets compared with the other three kernels. 2) DTAK has the largest error in general, which is possible because it is not regularized. 3) DTW and DTW-SC are regularized in this study, and the weighting coefficient for the regularization term is manually tuned to the best results. 4) The parameter selection strategy in this study strictly follows the suggested parameter range in [20, Table 2]. Based on these benchmarking results, the GA kernel is recommended for use in semiconductor FDC considering its good accuracy and the p.d property.

B. Case Study 2: Validation of the Proposed Methods

This sub-section validates the effectiveness of TSAK+mRMR and TSAK+KPCA by comparing their performance with the traditional feature extraction based sensor selection in Fig. 3. The data employed in this sub-section are the Etcher data (2 class) and Etcher data (3 class) in Fig. 4. In the raw dataset, there are over 300 sensor channels. 71 of these channels are useful for monitoring, and the rest of them are control inputs. Therefore, the sensor selection task in this study will focus on identifying the useful sensor subset from 71 sensor channels. The mean classifier error from 3-fold cross-validation based on TSAK+SVM and MLP is utilized to justify the effectiveness of the proposed sensor selection approach.

Table III compares the classification error before and after sensor selection given by TSAK+SVM and MLP. The results demonstrate that both sensor selection and feature selection can significantly increase the classification accuracy. Comparing with the proposed methodology, the accuracy of MLP is relatively lower, because the performance of MLP is depends on the feature quality and feature extraction might inadvertently eliminate crucial sensory information, such that it is difficult to produce high-quality feature set. In Table IV, the selected sensors given by different methods are compared. Comparing with the proposed method, traditional feature extraction based sensor selection cannot find all the important sensors.

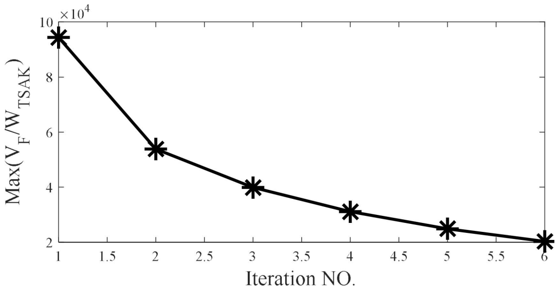

Fig. 5. Objective criterion of mRMR over iterations.

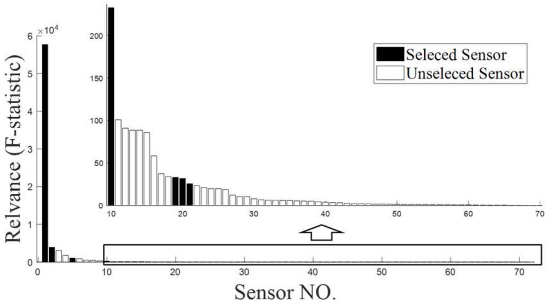

Fig. 6. Relevance score of each sensor.

In this study, 7 important sensors are selected by the proposed TSAK+mRMR. Fig. 5 shows the convergence of the objective criterion $V_F^{(b)}/W_{TSAK}^{(b)}$ in mRMR over the 6 iterations. It is observed that the sensors that are identified at a later stage of the iteration tend to have smaller rating scores. This is either because either these sensors have smaller relevance to the class labels, or they have larger redundancy with the previously selected sensors.

Fig. 6 visualizes the relevance scores of individual features after finishing step (I) in algorithm 1. In Fig. 6, the sensors are ranked by the relevance score, and the shadowed bars represent the selected sensors. From the results, one can easily see that the sensors with high relevance scores are not only selected as important sensors. Sensors with high relevance but duplicative information will be effectively excluded by the proposed algorithms. Therefore, it is safe to conclude that the proposed algorithm TSAK+mRMR can help identify the important sensor groups with the least redundant information among sensors.

Fig. 7 visualizes the usefulness of three sensor channels by using TSAK+KPCA, including sensor #5, sensor #16 and sensor #25. Sensor #25 has been selected by all methods and sensor #16 only has been selected by the proposed method. Sensor #5 has not been selected, so it should be a useless sensor. Since the data is proprietary, the author is not allowed the disclose the variable names. Fig. 7(a) and Fig. 7(d) show the raw trace and scatter plot on the manifold of sensor #25, which ranks 2*nd* among all evaluated sensors in Fig. 6. One can clearly see the difference in the trace data based on three

TABLE III
GENERALIZATION ERROR IN 3-FOLD CROSS-VALIDATION

	TSAK+SVM **before** sensor selection					MLP **before** feature selection		
	DTW	DTW SC	DTAK	GA	TGA	FeaSet #1	FeaSet #2	FeaSet #3
2-class	6.03%	5.02%	9.51%	2.50%	1.99%	6.99%	5.50%	4.00%
3-class	7.11%	7.23%	18.40%	6.10%	6.47%	11.33%	8.33%	12.33%
	TSAK+SVM **after** sensor selection					MLP **after** feature selection		
2-class	0%	0%	0%	0%	0%	4.50%	3.50%	4.00%
3-class	2.38%	2.10%	5.12%	2.23%	2.40%	8.00%	5.67%	5.33%

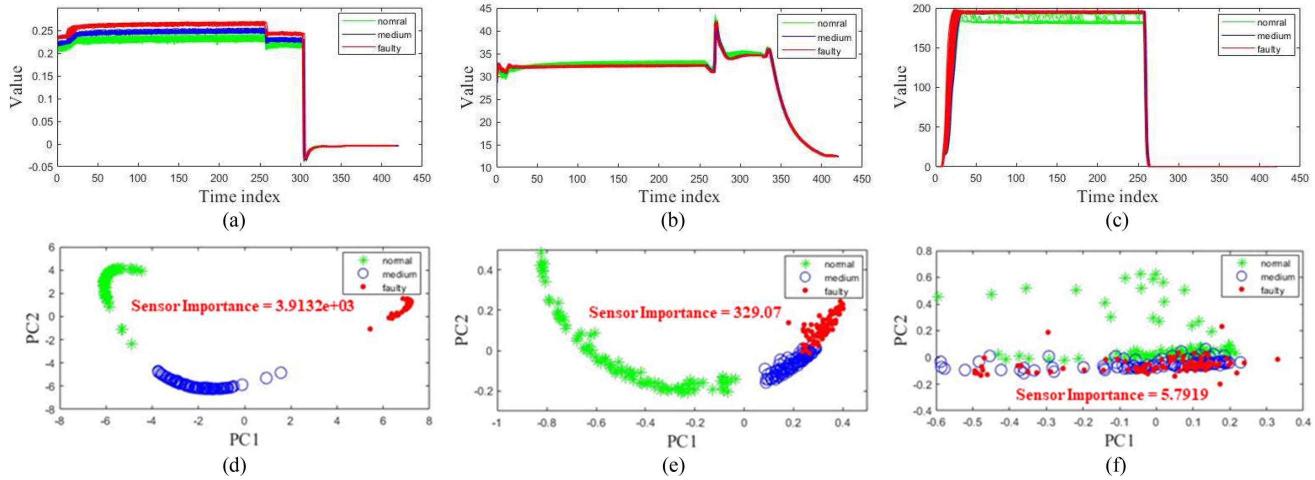

Fig. 7. KPCA visualization. (a) Raw trace signal of sensor #25. (b) Raw trace signal of sensor #16. (c) Raw trace signal of sensor #5. (d) Visualization of sensor #25 on the manifold. (e) Visualization of sensor #16 on the manifold. (f) Visualization of sensor #5 on the manifold.

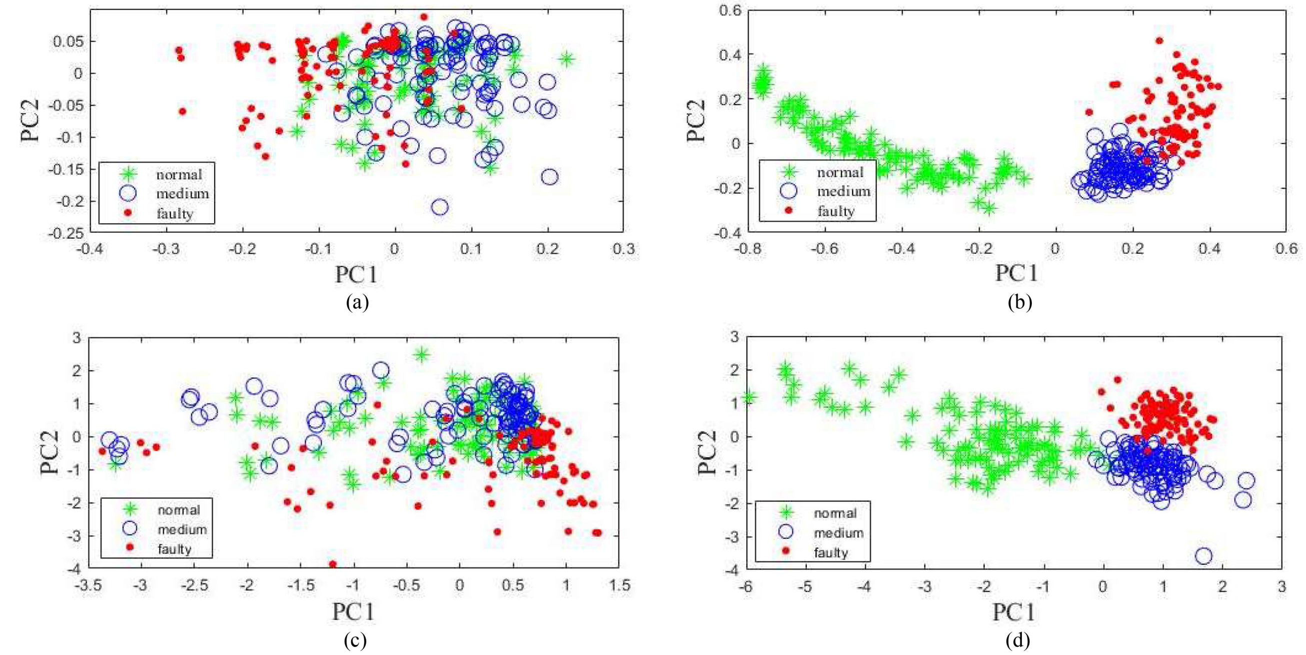

Fig. 8. Visualization for sensor group used in classification model. (a) KPCA before TSAK-mRMR sensor selection. (b) KPCA after TSAK-mRMR sensor selection. (c) PCA before mRMR feature selection. (d) PCA after mRMR feature selection.

classes. Unsurprisingly, Fig. 7(d) demonstrates three distinctive data clusters on the manifold. Likewise, Fig. 7(c) and Fig. 7(f) visualize sensor #5, which ranks 34^{th} in Fig. 6. Visualization of the raw trace data suggests no distinctive difference among different classes and the data points on the scatter plot in the PC space fail to demonstrate distinctive class. Fig. 7(b) and Fig. 7(e) visualize sensor #16, which is only selected by the proposed method. Fig. 7(e) also demonstrates three distinctive data clusters on the manifold. However, unlike sensor #25, the raw trace data of sensor #16 has not shown a clear difference, which explains why the traditional feature techniques miss it.

Fig. 8 visualizes the usefulness of a group of sensors by using TSAK+KPCA and PCA. For TSAK+KPCA, Fig. 8(a) shows the scatter plot in PC space based on all 71 sensors. One can clearly see that the data from different

TABLE IV
SELECTED SENSOR GIVEN BY DIFFERENT METHODOLOGY

	Selected Sensors Index
GA+mRMR (Proposed)	#53 **#16** #12 #7 #25 #47 #11
FeaSet #1	#25 #12
FeaSet #2	#53 #47 #25 #11
FeaSet #3	#53 #47 #25 #7

classes are not separated. That explains why TSAK+SVM cannot reach very high accuracy before the sensor selection, see Table III. Fig. 8(b) shows the scatter plot given by the 7 selected sensors. One can clearly see separation among different classes. That verifies how the classification accuracy after sensor selection can be very close to 100%, see Table III. In Fig. 8(c) and (d), the result is given by FeaSet #2. PCA has a similar performance with the TSAK+KPCA. However, traditional feature extraction is difficult to reach high accuracy for classification modeling and none of the feature extraction methods can guarantee that all the important sensor channels are founded.

V. CONCLUSION

This study proposes a systematic methodology for quickly important sensor screening, which will highlight the important sensor channels for finer feature design. In this research, we apply Time Series Alignment Kernels (TSAKs) to handle the multivariate sensor signals. Different TSAKs, including DTW, DTW-SC, DTAK, GA, and TGA, are investigated and benchmarked. Then, combining TSAKs and mRMR framework, the proposed TSAK+mRMR algorithm can effectively prioritize the important sensors while excluding the sensors that have smaller importance ratings or contain redundant information. To confirm the selection results, a visualization algorithm TSAK+KPCA is also proposed to visualize the usefulness of an individual sensor or a group of sensors. TSAK+SVM is used for cross-validation to further verify the sensor's importance. The effectiveness of the proposed methodology is demonstrated in Case 1 and 2 by using public data and the proprietary data. Based on these case studies, the following conclusions are reached:

1) Case study 1 evaluates the performance of each kernel as shown in Fig. 4 and Table II, which suggests GA and TGA have better performance and are recommended for use in semiconductor FDC considering its good accuracy and the p.d property.

2) Case study 2 validates the effectiveness of the proposed TSAK+mRMR sensor selection algorithm and demonstrates that sensor selection can significantly increase the classification accuracy and find the critical sensor channels missed by the traditional feature extraction based sensor selection.

In future investigations, the sensor screening method for virtual metrology will be explored. The utilization of TSAK for profile monitoring will also be investigated.

REFERENCES

[1] X. Jia, Y. Di, J. Feng, Q. Yang, H. Dai, and J. Lee, "Adaptive virtual metrology for semiconductor chemical mechanical planarization process using GMDH-type polynomial neural networks," *J. Process Control*, vol. 62, pp. 44–54, Feb. 2018.

[2] B. M. Wise and N. B. Gallagher, "The process chemometrics approach to process monitoring and fault detection," *J. Process Control*, vol. 6, no. 6, pp. 329–348, 1996.

[3] Q. P. He and J. Wang, "Fault detection using the *k*-nearest neighbor rule for semiconductor manufacturing processes," *IEEE Trans. Semicond. Manuf.*, vol. 20, no. 4, pp. 345–354, Nov. 2007.

[4] G. Verdier and A. Ferreira, "Adaptive mahalanobis distance and *k*-nearest neighbor rule for fault detection in semiconductor manufacturing," *IEEE Trans. Semicond. Manuf.*, vol. 24, no. 1, pp. 59–68, Feb. 2011.

[5] S. Mahadevan and S. L. Shah, "Fault detection and diagnosis in process data using one-class support vector machines," *J. Process Control*, vol. 19, no. 10, pp. 1627–1639, 2009.

[6] A. Ng, *Machine Learning and AI Via Brain Simulations*, Stanford Univ., Stanford, CA, USA, 2013.

[7] P. Li *et al.*, "A novel method for deposit accumulation assessment in dry etching chamber," *IEEE Trans. Semicond. Manuf.*, vol. 32, no. 2, pp. 183–189, May 2019.

[8] J. Yu, "Fault detection using principal components-based Gaussian mixture model for semiconductor manufacturing processes," *IEEE Trans. Semicond. Manuf.*, vol. 24, no. 3, pp. 432–444, Aug. 2011.

[9] Q. P. He and W. Jin, "Large-scale semiconductor process fault detection using a fast pattern recognition-based method," *IEEE Trans. Semicond. Manuf.*, vol. 23, no. 2, pp. 194–200, May 2010.

[10] J. Tang, S. Alelyani, and H. Liu, "Feature selection for classification: A review," in *Data Classification: Algorithms and Applications*, Cham, Switzerland: Springer, 2014, p. 37.

[11] H. Liu and H. Motoda, *Computational Methods of Feature Selection*. Boca Raton, FL, USA: CRC Press, 2007.

[12] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. London, U.K.: Wiley, 2012.

[13] J. R. Vergara and P. A. Estévez, "A review of feature selection methods based on mutual information," *Neural Comput. Appl.*, vol. 24, no. 1, pp. 175–186, 2014.

[14] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.

[15] K. Kira and L. A. Rendell, "A practical approach to feature selection," in *Machine Learning Proceedings*. Amsterdam, The Netherlands: Elsevier, 1992, pp. 249–256.

[16] M. A. Hall and L. A. Smith, "Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper," in *Proc. FLAIRS Conf.*, 1999, pp. 235–239.

[17] K.-B. Duan, J. C. Rajapakse, H. Wang, and F. Azuaje, "Multiple SVM-RFE for gene selection in cancer classification with expression data," *IEEE Trans. Nanobiosci.*, vol. 4, no. 3, pp. 228–234, 2005.

[18] R. T. Olszewski, *Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data*, School Comput. Sci., Carnegie-Mellon Univ., Pittsburgh, PA, USA, 2001.

[19] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust. Speech Signal Process.*, vol. ASSP-26, no. 1, pp. 43–49, Feb. 1978.

[20] H. Shimodaira, K.-I. Noma, M. Nakai, and S. Sagayama, "Dynamic time-alignment kernel in support vector machine," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 921–928.

[21] M. Cuturi, J.-P. Vert, O. Birkenes, and T. Matsui, "A kernel for time series based on global alignments," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, vol. 2, 2007, pp. 413–416.

[22] M. Cuturi, "Fast global alignment kernels," in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*, 2011, pp. 929–936.

[23] E. J. Keogh and M. J. Pazzani, "Derivative dynamic time warping," in *Proc. SIAM Int. Conf. Data Min.*, 2001, pp. 1–11.

[24] H. Sakoe and S. Chiba, "A similarity evaluation of speech patterns by dynamic programming," in *Proc. Nat. Meeting Inst. Electron. Commun. Eng. Japan*, 1970, p. 136.

[25] S. Gudmundsson, T. P. Runarsson, and S. Sigurdsson, "Support vector machines and dynamic time warping for time series," in *Proc. IEEE Int. Joint Conf. Neural Netw. IEEE World Congr. Comput. Intell.*, 2008, pp. 2772–2776.

[26] A. Frank and A. Asuncion. (2010). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml

[27] K. Ø. Mikalsen, F. M. Bianchi, C. Soguero-Ruiz, and R. Jenssen, "Time series cluster kernel for learning similarities between multivariate time series with missing data," *Pattern Recognit.*, vol. 76, pp. 569–581, Apr. 2018.

[28] L. Wang, Z. Wang, and S. Liu, "An effective multivariate time series classification approach using echo state network and adaptive differential evolution algorithm," *Exp. Syst. Appl.*, vol. 43, pp. 237–249, Jan. 2016.