IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 25 April 2023, accepted 26 May 2023, date of publication 30 May 2023, date of current version 7 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3281407

RESEARCH ARTICLE

Failure Detection and Primary Cause Identification of Multivariate Time Series Data in Semiconductor Equipment

MINJAE BAEK AND SEOUNG BUM KIM
School of Industrial and Management Engineering, Korea University, Seoul 02841, South Korea

Corresponding author: Seoung Bum Kim (sbkim1@korea.ac.kr)

This work was supported in part by the Brain Korea 21 FOUR of the Ministry of Science and ICT (MSIT), South Korea, under the Information Technology Research Center (ITRC) Support Program supervised by the Institute for Information Communication Technology Planning and Evaluation under Grant IITP-2020-0-01749; and in part by the National Research Foundation of Korea's funded by the Korean Government under Grant RS-2022-00144190.

ABSTRACT Downtime caused by equipment failure is the biggest productivity problem in the 24-hour a day operations of the semiconductor industry. Although some equipment failures are inevitable, increases in productivity can be gained if the causes of failures can be detected quickly and repaired, thus reducing downtime. Univariate control charts are commonly used to detect failures. However, because of the complexity of the process and the structural characteristics of the equipment, detection and identification of the causes of failures may be difficult. The purpose of this study is to use correlations of variables to detect failures in semiconductor equipment, to predict the parts to be replaced and to identify the primary causes of failures. The proposed method consists of four steps: (1) conversion of the multivariate time series data of the equipment into signature matrixes, (2) detection of anomalies through a convolutional autoencoder, (3) learning classification models with supervised learning methods that use the residual matrixes of fault sections, and (4) application of an explainable algorithm to interpret the classification model. The effectiveness and applicability of the proposed method are demonstrated by the actual multivariate time series data obtained from 8-inch ashing process equipment that produces semiconductors on 8-inch silicon wafers.

INDEX TERMS Anomaly detection, explainable artificial intelligence, multivariate time series, semiconductor process, signature matrix.

I. INTRODUCTION

Semiconductors are essential components of the Internet of Things, smart TVs, smartphones, and autonomous vehicles. Equipment and process downtime should be minimized to improve productivity in the manufacturing of semiconductors. The major causes of downtime are preventive maintenance (PM) and breakdown maintenance (BM) [1]. PM is the replacement of parts before the expiration of their expected lifespan. The necessary replacement parts are typically readily at hand because of their predicted lifespans and replacement dates. Therefore, PM completion time is predictable. In other words, the idle time of the equipment can be predicted, and semiconductor production plans can be established accordingly. In contrast, BM is performed when unexpected failures occur. BM requires engineers to determine why equipment failed and take appropriate actions. However, these appropriate actions of identifying the equipment involved, procuring it, and implementing the repairs as quickly as possible hinge on promptly identifying the cause of a failure.

In the semiconductor industry, fault detection and classification (FDC) systems detect equipment failures. These systems rely on collecting sensor data (e.g., temperature, pressure, and power) in real time and detecting and signaling equipment abnormalities [2]. Various processes occur in

The associate editor coordinating the review of this manuscript and approving it for publication was Yongming Li.

a semiconductor factory, and each process has a different equipment structure. Consequently, the processes are complicated, and the engineers for each process set a sensor threshold for the FDC systems. When a specific sensor trend falls below the lower threshold or exceeds the upper threshold, the corresponding sensor is interlocked, preventing the equipment from operating under certain conditions. Engineers use FDC systems to determine sensor data that exceed the threshold and identify the parts that require inspection. However, semiconductor equipment functions by combining various process factors, such as pressure, temperature, and power. Notably, physical or structural correlations exist between the sensors in the equipment. Consequently, sensors alone may fail to detect failures [3], [4]. For example, plasma is used during etching. When a temperature of several thousand degrees is applied to an object, the object passes through solid-liquid-gas states, and the molecules collide, resulting in ionization. The ionized state is termed plasma, which is known as the fourth state. However, because it is challenging to supply temperatures of several thousand degrees, plasma in the semiconductor industry is generated by lowering the pressure and applying electric power [5]. If the prescribed pressure cannot be maintained consistently, normal plasma cannot be generated. Therefore, the standing and reflected waves of the power become unstable, and the data generated by the sensor that detects power deviate from the threshold value.

Currently, engineers monitor the FDC systems and subjectively determine the cause of a failure and the remedial action necessary. However, such a methodology based on work experience results in differences in BM that affect semiconductor production and productivity. Therefore, to reduce the time required for action and increase manufacturing productivity, it is crucial to use artificial intelligence (AI) that can use multivariate sensor data from the equipment to detect failures and determine their exact cause.

Many studies have been conducted on how to detect equipment anomalies in the semiconductor industry. Approaches that have been investigated include univariate control charts, including Shewhart, cumulative sum, and exponentially weighted moving average charts. However, univariate approaches can not take into account the correlations that can occur between sensors [6].

Therefore, the semiconductor industry needs monitoring and diagnostic methods that can handle multivariate processes. Data collected in the semiconductor manufacturing processes are unbalanced in that abnormal data are scarce [7], [8]. In situations in which a few abnormal data are available, studies capable of predicting anomalies based solely on normal data are essential [9]. Reference [3] trained a k-nearest neighbor (k-NN) model using multivariate sensor data from equipment in a semiconductor etching process. The equipment abnormality was detected based on its similarity to a sample of the training set that consisted of only normal data. The data were considered normal if the distances between the input data points in the sample and the trajectories of the normal samples in the training data were less than a threshold value. Reference [10] performed anomaly detection by combining convolutional autoencoder (CAE) with k-NN. The CAE was trained using the normal dataset from the etching process to extract its features. The distance between the normal data features was calculated using k-NN. When the distance between the feature of new data and the feature of normal data exceeds the k-NN distance, it was identified as abnormal. Reference [11] proposed a stacked autoencoder (AE) that can use time-frequency analysis to detect anomalies in multivariate sensor data from equipment used in chemical vapor deposition. Reference [12] proposed a variational AE (VAE) neural network by combining a convolutional neural network (CNN) and a VAE network. They used the multivariate time series data of the etching process equipment and used only normal data to train the network. The data were considered abnormal if the reconstruction errors between the input and output variables exceeded a specific threshold. Previous studies showed significant success in detecting anomalies based exclusively on learning from normal data. However, k-NN and AE methods cannot provide clear interpretations of the results. Therefore, these proposed networks failed to explain the primary causes of equipment anomalies and lacked the capability to interpret the relationships between the multivariate data variables [13].

The present study proposes a method that can detect equipment anomalies and identify the key process variables responsible for anomalies in multivariate processes. The main contributions of this study can be summarized as follows:

1) Signature matrixes using inner products were applied to detect equipment abnormalities to reflect the existence of correlations between the sensor data of semiconductor equipment.
2) Parts needing replacement were predicted by using the residual matrixes of the abnormal section.
3) The primary factors of the replacement parts were identified using Shapley additive explanations (SHAP).

The remainder of this paper is organized as follows. Section II briefly reviews the multivariate time-series anomaly detection. Section III explains the proposed methodology. Section IV presents a comparison of the experimental results. Section V presents our concluding remarks and future research directions.

II. RELATED WORKS

Multivariate time-series anomaly detection is a crucial problem in many industries. Detecting anomalies in the manufacturing process is important to minimize the production of defective products and prevent system failures. Reference [14] used only the normal data from the etching process of semiconductor manufacturing and performed anomaly detection through machine learning techniques such as over-sampling principal component analysis (OsPCA) [15], angle-based outlier detection (ABOD) [16], and local outlier factor (LOF) [17]. Reference [18] used an isolation forest (iForest) [19] to detect defective wafers using data from

particle level measurement sensors inside a semiconductor clean room. Reference [20] detected defect intrusions or anomalies in network traffic using the k-means clustering algorithm, which groups objects into k number of clusters based on their similarity to the centroid of each cluster. In addition, [21] used the one-class support vector machine (OCSVM) [22] to detect network intrusions. OCSVM isolates a set of data points representing normal behavior from the rest of the data and identifies all data points outside its boundaries as anomalous behavior. Reference [23] detected false data injection threats using a deep auto-encoding Gaussian mixture model (DAGMM) [24], which combines a compression network through an autoencoder and an estimation network through a Gaussian mixture model. While these anomaly detection methods have been proven effective in many applications, they may not work well with multivariate time series because of their limitations in capturing temporal dependencies. To address this issue, researchers have studied anomaly detection models that take temporal dependencies into account. Reference [25] applied long short term memory AE (LSTM-AE) [26] to multivariate time-series data from a semiconductor plasma etch machine to detect defective wafers. Reference [27] analyzed network event sequences with LSTM and classified the output features of LSTM with linear discriminant analysis (LDA) to detect anomalies in industrial control networks. Reference [28] used discrete-wavelet-transformation (DWT) and LSTM-AE to accurately detect short-term and long-term anomalies in multiple pumps. Reference [29] extracted the features of each sensor with a CNN head to create a feature map, and then used a recurrent neural network (RNN) to identify temporal patterns to detect abnormalities in real industrial service elevators. However, RNN and LSTM are sensitive to noises and thus can be less accurate if the input data is noisy. For noisy data, anomaly detection methods using denoising autoencoder (DAE) exist. Reference [30] proposed a fault detection method using DAE on etching and chemical vapor deposition process data. The DAE, trained only on normal data, could not remove the noise of abnormal data, leaving residual noise, which was used for fault detection. Similarly, [31] used DAE to detect wafer defects in the thin film process at a semiconductor fabrication plant. In addition, [32] used stacked denoising autoencoder (SDAE) with data from a photolithography tool to detect defects in wafers. Furthermore, [33] proposed unsupervised anomaly detection (USAD), which combines the stable learning advantages of autoencoder and the precise anomaly detection advantages of adversarial learning. This method was used to conduct anomaly detection on four public multivariate time-series datasets. USAD is also very efficient and robust to noise in the data. These studies have demonstrated excellent performance in anomaly detection. However, they could not provide interpretations for the results, leading to the need for research that can detect the causes of anomalies. Reference [34] attempted to detect anomalies with AE and applied SHAP to explore influencing factor of anomalies detected in the chiller data. Reference [35] adopted LSTM model to detect anomalies in hot rolling process and combined SHAP to find causal variables for anomalies. Reference [36] used an isolation forest (iForest) to detect anomalies in multivariate time series data from magnetic resonance imaging (MRI) devices and applied SHAP to identify critical sensors associated with scan failures or abnormal MRI device behavior. These studies provided an interpretation of the causal variables after anomaly detection. However, they did not provide interpretations associated with the correlation between variables, which is crucial in multivariate data when detecting anomalies.

In this study, we performed anomaly detection considering the correlation between multivariate time series data of semiconductor process equipment. In addition, we used the residual matrix to predict equipment failure parts in abnormal regions and applied SHAP to identify pair sensors that contribute significantly to the prediction of failure parts. This allows us to provide contribution points for the action decisions of engineers.

III. PROPOSED METHOD

Figure 1 shows the overall structure of the proposed method, which consists of four main steps: First, the multivariate time series data of the semiconductor equipment are transformed into signature matrixes. Next, anomaly detection is performed using the signature matrixes as the input to a convolutional AE (CAE). Classification, based on the residual matrixes generated in the abnormal sections, is then performed to predict the correct replacement part. Finally, the primary causes of failure are identified using SHAP.

A. SIGNATURE MATRIX OF TIME SERIES DATA

Because the semiconductor manufacturing processes typically involve multivariate sensor data, the correlations between the variables should be considered. In addition, these multivariate sensor data contain noises. Reference [37] proposed a signature matrix calculated by the inner product of time series data to address multivariate data with noises.

Figure 2 shows an example of a multivariate time series data where X_i represents the ith time series observation and X_i^w represents the time window (between time $t-w$ and time t) in the ith time series. Considering the two time series $X_i^w = (x_i^{t-w}, x_i^{t-w-1}, \cdots, x_i^t)$ and $X_j^w = (x_j^{t-w}, x_j^{t-w-1}, \cdots, x_j^t)$, the inner product of the two time series data can be calculated as follows:

$$m_{ij}^t = \frac{\sum_{\delta=0}^{w} x_i^{t-\delta} x_j^{t-\delta}}{k}, \tag{1}$$

where k is a rescale factor ($k = w$).

An $n \times n$ signature matrix M^t at time t is generated through the inner product of (1). Figure 3 shows an example of the signature matrix.

Signature matrixes are generated for each time window. In addition, three-channel matrixes are generated based on long, medium, and short sizes of a time window to extract various features from a signature matrix. Figure 4 shows an

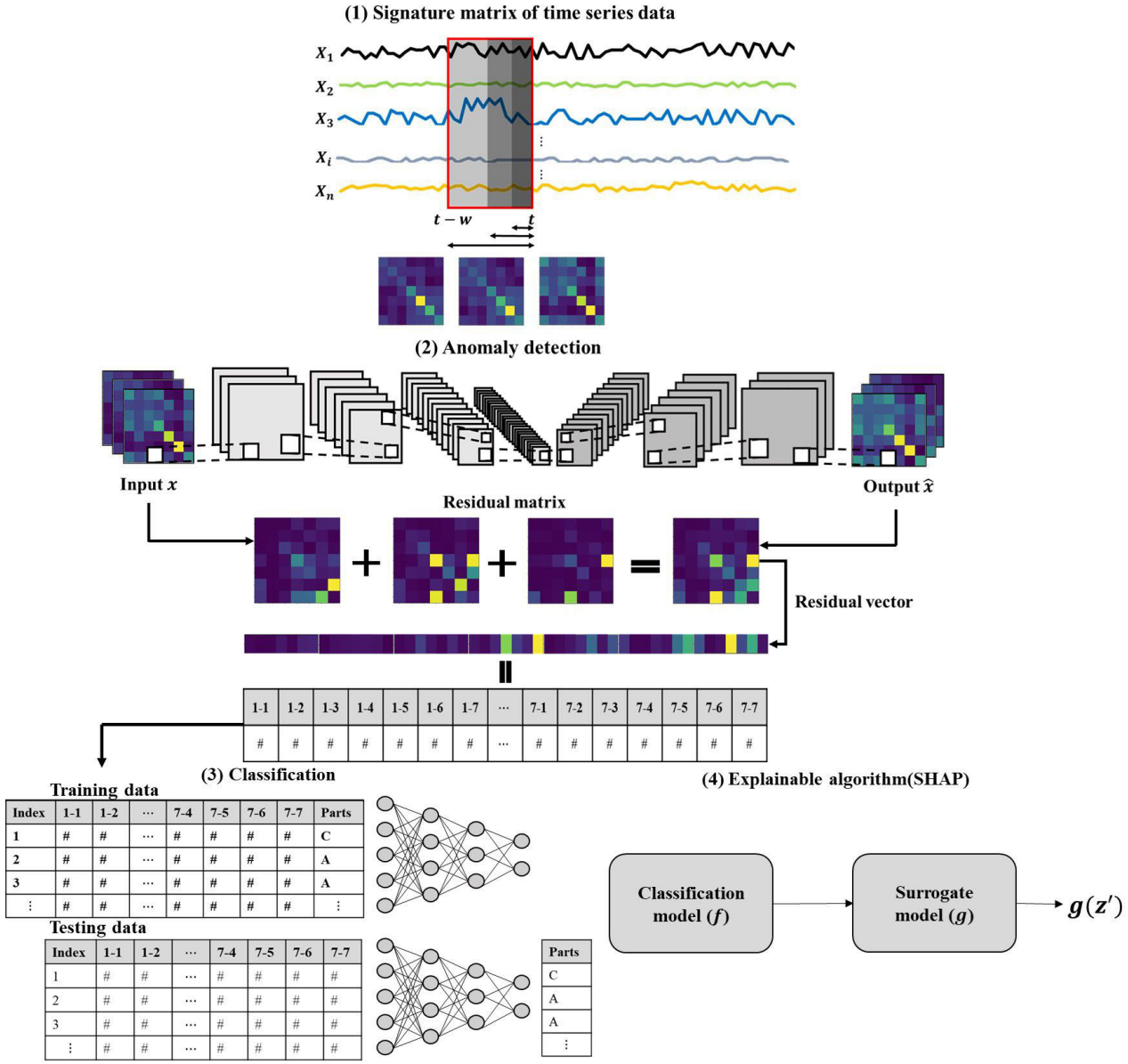

FIGURE 1. Framework of proposed methodology. The proposed method consists of four steps: (1) transformation of time-series data into signature matrix, (2) anomaly detection using signature matrix as input, (3) classification using residual matrix, and (4) interpretation of the classification model using SHAP.

example of three-channel matrixes and the process of how they are generated.

Subsequently, an $n \times n$ signature matrix is generated for each section, and a three-channel signature matrix is generated in a time window. In this study, the short, medium, and long sections are 25, 50, and 100, respectively.

B. ANOMALY DETECTION BASED ON SIGNATURE MATRIX

As the time window moves, the multivariate time series data are converted into a signature matrix. At this point, there are instances in which both the normal and abnormal data are included within the same time window. If any abnormal data are included within the time window, they are labeled as abnormal; however, this is not a definitively abnormal label. For instance, if 99% of normal data and 1% of abnormal data are included within the same time window, the corresponding signature matrix is labeled as abnormal; however, its abnormal status remains uncertain. Therefore, it is necessary to definitively detect abnormal labels. To this end, anomaly detection is performed to detect certain abnormalities.

AE has been primarily used in the field of unsupervised anomaly detection and has performed excellently [38]. The CAE is an AE structure that is converted into a convolutional structure. A CAE consists of an encoder and a decoder that contain convolution and transposed convolution layers, respectively. The encoder creates a feature representation by compressing and summarizing the characteristics of the input data. The decoder aids in the reconstruction of the feature

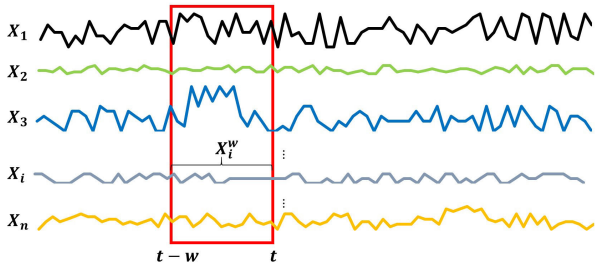

FIGURE 2. Example of multivariate time series data.

$$\begin{array}{|c|c|c|c|c|}\hline m_{11}^t & m_{12}^t & m_{13}^t & \cdots & m_{1n}^t \\ \hline m_{21}^t & m_{22}^t & m_{23}^t & \cdots & m_{2n}^t \\ \hline m_{31}^t & m_{32}^t & m_{33}^t & \cdots & m_{3n}^t \\ \hline \vdots & \vdots & \vdots & \ddots & \vdots \\ \hline m_{n1}^t & m_{n2}^t & m_{n3}^t & \cdots & m_{nn}^t \\ \hline \end{array}$$

FIGURE 3. Example of an $n \times n$ signature matrix M^t at time t.

representation compressed by the encoder [39]. Figure 5 shows the structure of a CAE.

As shown in Figure 5, the input is a three-channel signature matrix. "Conv" represents a convolution layer in which 3×3 means that the kernel size is 3. C and P denote the number of output channels and the padding size, respectively. The stride was set to 1. Both the encoder and decoder of the CAE are composed of four hidden layers, and each hidden layer consists of a batch norm and an activation function structure after convolution. Scaled exponential linear units (SELU) [40] are used as the activation function. In this study, we used SELU because it can activate a self-regulating neural network to increase the accuracy of the autoencoder and reduce the need for complex regularization techniques. SELU is also known to be superior to other activation functions in terms of convergence speed [41], [42]. Table 1 presents the CAE network structure.

The CAE is trained by three-channel signature matrixes consisting of only normal data as inputs. The following loss function is defined by minimizing the difference between the input and output data:

$$lossfunction = \sum_{t} \sum_{c=1}^{3} \left\| x_{:,:,c}^t - \widehat{x}_{:,:,c}^t \right\|_F^2, \qquad (2)$$

where x is the input matrix component, $\widehat{x}$ is the output matrix component. c is the channel, and F is the Frobenius norm that represents the Euclidean distance for each component of the matrix. If the signature matrixes, including abnormal data, are used as the input, the anomaly score between the input and output data is high because it cannot be reconstructed normally. The CAE can detect anomalies based on an anomaly score.

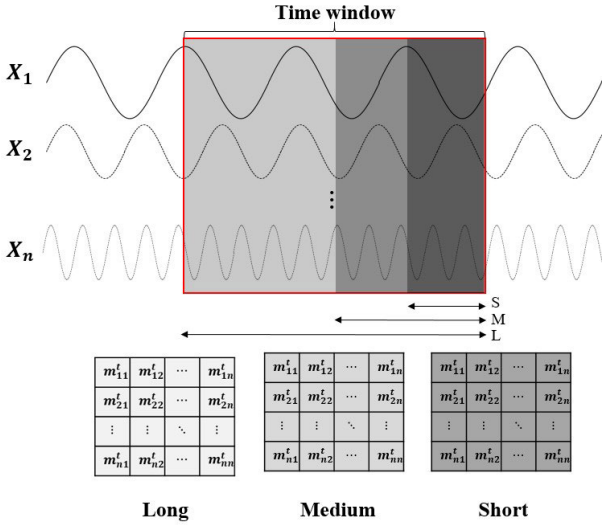

FIGURE 4. Example of a three-channel signature matrix of $\mathbf{n} \times \mathbf{n} \times \mathbf{3}$ in a time window. S, M, and L represent short, medium, and long sizes.

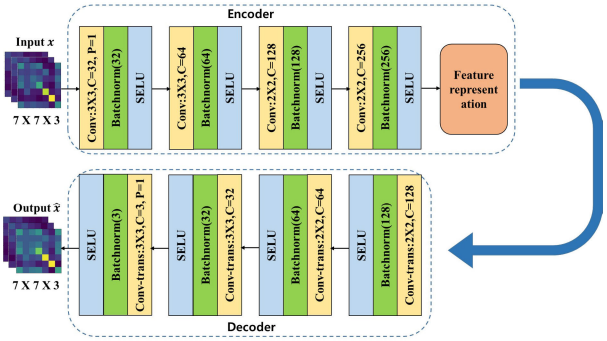

FIGURE 5. Structure of a convolutional autoencoder.

In this study, the anomaly score is the number of matrix components that exceed a specific threshold δ among the components of the residual matrixes developed using the difference between the input and output data. δ is determined empirically [37]. Because each residual matrix component represents a reconstruction error, a large residual matrix component indicates that the time series data constituting the component is abnormal. Moreover, if the anomaly score exceeds a threshold τ, it is considered an anomaly. The mean value of the anomaly score of the validation data of only normal observations can be set as τ, which is defined as

$$\tau = \alpha \cdot mean\{score(t)_{valid}\}, \qquad (3)$$

where $score(t)_{valid}$ are the anomaly scores of the validation data. Specifically, we set the threshold by multiplying the mean of anomaly scores in the validation data by the hyperparameter α. We have determined the appropriate value of α such that F1 score was maximized. Table 2 shows the performance of outcomes in terms of precision, recall, and F1 score by changing α. In our study, we set $\alpha = 1.7$.

TABLE 1. CAE network structure: The k, s, p, n, BN denote kernel size, stride, padding, the number of channels, and batch normalization, respectively. Output represents the output dimension.

Encoder	Output	Decoder	Output
Conv2D (k=3,s=1,p=1,n=32), BN(32), SELU	7 × 7 × 32	ConvTranspose2D (k=2,s=1,n=128), BN(128), SELU	4 × 4 × 128
Conv2D (k=3,s=1,n=64), BN(64), SELU	5 × 5 × 64	ConvTranspose2D (k=2,s=1,n=64), BN(64), SELU	5 × 5 × 64
Conv2D (k=2,s=1,n=128), BN(128), SELU	4 × 4 × 128	ConvTranspose2D (k=3,s=1,n=32), BN(32), SELU	7 × 7 × 32
Conv2D (k=2,s=1,n=256), BN(256), SELU	3 × 3 × 256	ConvTranspose2D (k=3,s=1,p=1,n=3), BN(3), SELU	7 × 7 × 3

TABLE 2. Performance of anomaly detection in terms of average values of precision, recall, and F1 scores as the α varies. Standard deviations are shown in parentheses.

α	Precision	Recall	F1 score
1.0	0.873 (0.0059)	0.987 (0.0028)	0.926 (0.0021)
1.1	0.879 (0.0139)	0.985 (0.0032)	0.929 (0.0063)
1.2	0.901 (0.0099)	0.983 (0.0044)	0.941 (0.0039)
1.3	0.904 (0.0074)	0.982 (0.0041)	0.942 (0.0023)
1.4	0.904 (0.0074)	0.982 (0.0041)	0.942 (0.0023)
1.5	0.922 (0.0084)	0.977 (0.0012)	0.948 (0.0045)
1.6	0.925 (0.0059)	0.977 (0.0012)	0.950 (0.0032)
1.7	0.927 (0.0054)	0.977 (0.0026)	0.952 (0.0023)
1.8	0.943 (0.0104)	0.956 (0.0181)	0.949 (0.0037)
1.9	0.955 (0.0017)	0.937 (0.0035)	0.946 (0.0010)
2.0	0.955 (0.0017)	0.937 (0.0035)	0.946 (0.0010)

C. CLASSIFICATION USING RESIDUAL MATRIX

A replacement part is predicted using the three-channel residual matrixes of the section that has been definitely identified as abnormal in the abnormal section. Three-channel residual matrixes have various characteristics because the three channels consist of short, medium, and long sections. The residual matrix of the three sections is added to create an $n \times n$ residual matrix that combines these characteristics. Subsequently, this matrix is transformed into a row of residual vectors. The residual vector is set as the input variable of classification model, and the part label replaced in the abnormal section is set as the target variable. The multilayer perceptron network (MLP) is trained by using a supervised learning method to predict the part to be replaced in the abnormal section.

D. INTERPRETATION OF CLASSIFICATION RESULTS

Classification and predictive performance have significantly improved because of advances in deep learning algorithms [43]. However, understanding how deep learning algorithms make decisions is challenging because of their complexity. Therefore, methods to explain the output of the deep learning model is necessary [44]. In this study we use the SHAP, one of the explainable methods that can identify the importance of variables; it does so by representing the absolute influence of each variable through calculating the absolute value of the variable importance score and averaging it [45]. In this study, we used the KernelSHAP algorithm, which is more efficient than the original SHAP [46]. Although DeepSHAP exists, which is specialized for deep neural networks, we decided to use the KernelSHAP because KernelSHAP tends to be more accurate because it computes Shapley values for all possible combinations in the input feature space and takes into account the importance of the input features [46]. In our experiment, we found that KernelSHAP and DeepSHAP produced similar results. Specifically, the KernelSHAP is applied to investigate the primary factors of replacement parts and the relationships among variables.

IV. EXPERIMENTS

A. DATA COLLECTION AND PREPROCESSING

The data used in this study were the multivariate time series data from the equipment of an 8-inch ashing process that produces semiconductors on 8-inch silicon wafers. A total of 242,780 time steps were collected, with 227,913(94%) and 14,867(6%), respectively, of normal data and abnormal data. Seven input variables were used; these include temperature, pressure in the equipment, flow rate by gas type, and power. As for the target variables, two labels were used. The first label indicates the status of the normal and abnormal sections of the equipment, and the second label indicates parts replaced for each abnormal section. Six abnormal sections and six classes of part labels were used. Six sections consist of two short-term abnormal sections (0.004%, 0.16%), three medium-term abnormal sections (0.84%, 1.29%, 1.39%), and one long-term abnormal section (2.4%). In addition, each abnormal section has a part label that indicates the primary cause of the failed equipment.

One-hot encoding to the label was applied in which the normal was 0 and the abnormal was 1. Because the part label had six classes, Part A was transformed into 100000, Part B was transformed into 010000, and in the same way, Parts C to F were transformed into 001000 to 000001. Subsequently, the time window was set to 100. Furthermore, while the time window was moving (i.e., stride = 5), the dataset in the time window was relabeled as abnormal (1) if an abnormal

label (1) was included in the time window. Moreover, when the dataset in the time window was relabeled as abnormal (1), the part label that was transformed into one-hot encoding was set by the second label. The preprocessed data were divided into three parts based on abnormal data in the test set. The test set consisted of both abnormal (30%) and normal data (70%), and the training and validation data consisted of only normal data. Namely, the data were divided into training (64%), validation (16%), and test set (20%). For further preprocessing, the training data were normalized by using the z-score normalization method with a mean and a standard deviation of 0 and 1, respectively, except for the label data. Note that normalization was performed on the validation and test set by using the mean and variance of the training data.

B. PERFORMANCE EVALUATION

In this study, precision, recall, and an F1 score were used to evaluate the performance of the anomaly detection algorithm. Furthermore, accuracy was used to evaluate the performance of the classification algorithm. Precision is the ratio of data that are correctly predicted as normal by the model to the actual normal data, and recall is the ratio of data that are correctly predicted as abnormal by the model to the actual abnormal data. The F1 score is the harmonic mean of the precision and recall. Accuracy is the proportion of a correct prediction made by a model. Precision, recall, F1 score, and accuracy are calculated by the following equations:

$$precision = \frac{TP}{TP + FP}, \tag{4}$$

$$recall = \frac{TP}{TP + FN}, \tag{5}$$

$$F1\,score = 2 \cdot \frac{precision \cdot recall}{precision + recall}, \tag{6}$$

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{7}$$

where TP (true positive) refers to the true cases that the model predicts as true. FP (false positive) is the false cases that the model predicts as true. FN (false negative) refers to the true cases that the model predicts as false. TN (true negative) is the false case that the model predicts as false.

C. ANOMALY DETECTION PERFORMANCE

The LOF [17], OCSVM [22], iForest [47], AE [48], VAE [49], DAGMM [24], LSTM-AE [26] and USAD [33] were used in comparative experiments. Comparisons of performance in anomaly detection were made before and after transforming the time series data into a signature matrix. The LOF, iForest, OCSVM, AE, VAE, DAGMM and USAD were tested before and after the signature matrix transformation. It is noted that because LSTM-AE performs anomaly detection on sequential data, we only tested LSTM-AE before reflecting the signature matrix. The CAE was only tested after the signature matrix transformation. Adam optimizer [50] was used to train the CAE for 30 epochs by using a minibatch gradient descent method. The experiments were repeated five times, and the average and standard deviation values were reported for comparison in Table 3.

TABLE 3. Comparison of anomaly detection methods in terms of average values of precision, recall, and F1 scores. Standard deviations are shown in parentheses.

Input	Type	Model	Precision	Recall	F1 score
Without signature matrix	Vector	LOF	0.782 (0.0033)	0.952 (0.0004)	0.858 (0.0018)
		OCSVM	0.772 (0.0010)	0.962 (0.0000)	0.857 (0.0005)
		iForest	0.803 (0.0043)	0.866 (0.0054)	0.833 (0.0030)
		AE	0.870 (0.0085)	0.941 (0.0077)	0.904 (0.0068)
		VAE	0.905 (0.0076)	0.928 (0.0101)	0.916 (0.0080)
		DAGMM	0.525 (0.0190)	0.687 (0.0275)	0.595 (0.0020)
		LSTM-AE	0.828 (0.0031)	0.914 (0.0016)	0.867 (0.0004)
		USAD	0.902 (0.0033)	0.904 (0.0086)	0.903 (0.0049)
With signature matrix	Vector	LOF	0.838 (0.0027)	0.974 (0.0002)	0.901 (0.0017)
		OCSVM	0.837 (0.0023)	0.961 (0.0000)	0.895 (0.0014)
		iForest	0.811 (0.0051)	0.874 (0.0028)	0.841 (0.0032)
		AE	0.916 (0.0048)	0.932 (0.0003)	0.924 (0.0025)
		VAE	0.916 (0.0012)	0.934 (0.0002)	0.917 (0.0007)
		DAGMM	0.669 (0.0011)	0.705 (0.0063)	0.687 (0.0030)
		USAD	0.914 (0.0059)	0.941 (0.0034)	0.927 (0.0047)
	Matrix	CAE	0.927 (0.0054)	0.977 (0.0026)	0.952 (0.0023)

The results revealed that the AE, VAE, LSTM-AE and USAD outperformed the LOF, OCSVM, and iForest in terms of F1 scores. In general, real-world multivariate time series data contains noise, which affects the performance of LSTM [37]. As a result, LSTM-AE performed worse than AE and VAE. Furthermore, when the correlations between variables were taken into account using the signature matrix, the performances of the LOF, OCSVM, iForest, AE, VAE and USAD models were improved. DAGMM performed somewhat worse than the other models. However, we could see that DAGMM also improved its precision, recall, and F1 score performance when using the signature matrix as input. Additionally, the CAE model with the signature matrix as the matrix input yielded the best results in terms of the F1 scores.

D. CLASSIFICATION PERFORMANCE

This study compared cases in which residual matrixes and residual vectors were used as inputs. A CNN model was used when residual matrixes were used as input. A MLP model was used when residual vectors were used as input. Table 4 presents the network structures of the classification models.

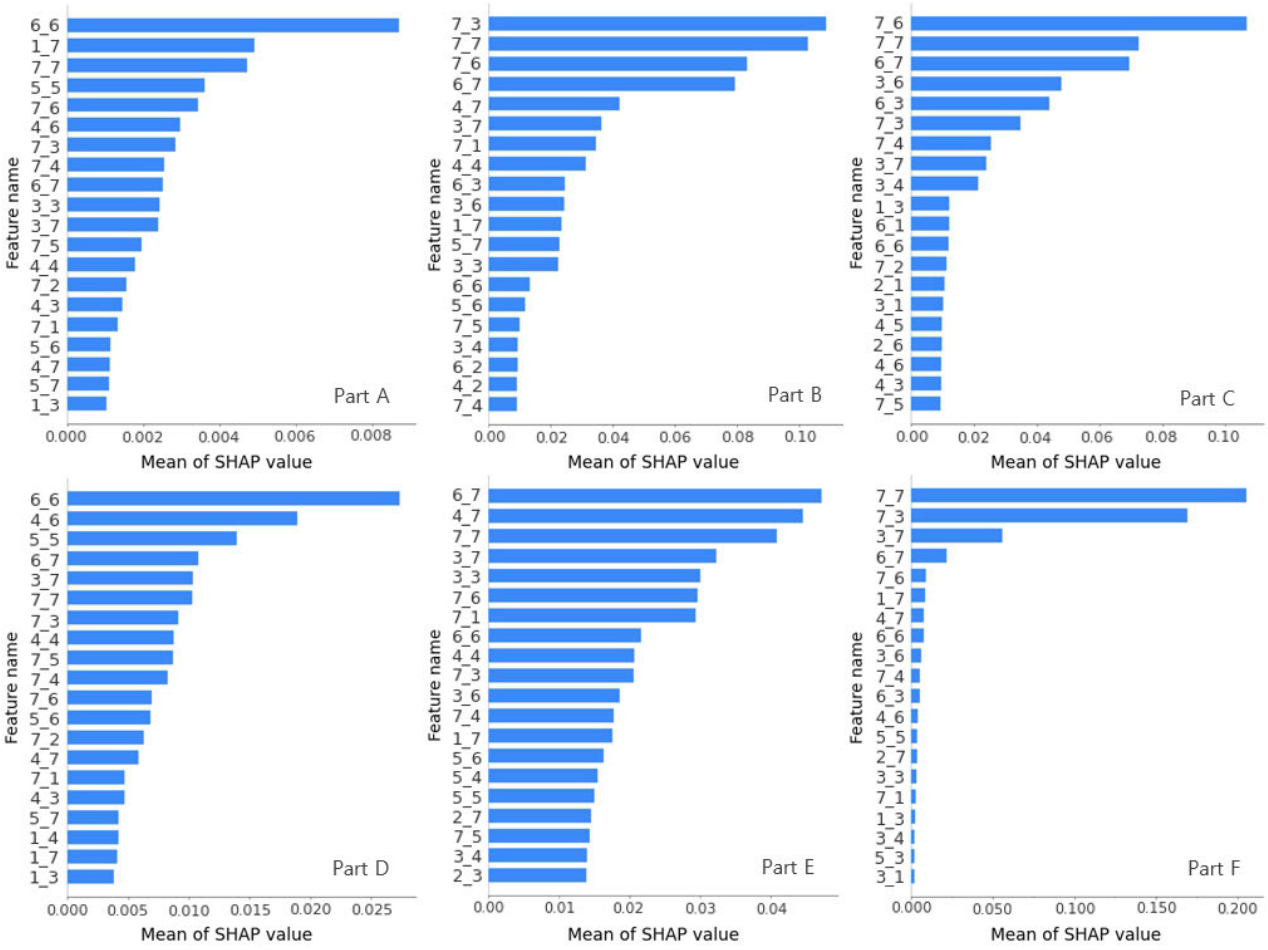

FIGURE 6. Importance of sensor pairsćorrelation of each parts, as ranked by the KernelSHAP method.

TABLE 4. Two classification model network structures: CNN feeding a signature matrix as input and MLP feeding a signature vector as input.

CNN		MLP	
Network	Output dimension	Network	Node
Input	$7 \times 7 \times 3$	Input	49
Conv1	$7 \times 7 \times 32$	Layer1	70
Conv2	$5 \times 5 \times 64$	Layer2	30
Conv3	$3 \times 3 \times 128$	Layer3	10
Fc1	625	Output layer	6
Fc2	256		
Output layer	6		

The CNN consisted of three convolution layers, two fully connected layers, and one output layer. Dropout [51] was applied to the fully connected layers. A rectified linear unit [52] was used as the activation function, and softmax was applied to the output layer. The MLP consisted of three hidden layers and one output layer, and the number of nodes in each hidden layer was 70, 30, and 10. Dropout was applied to the hidden layers. Tanh was used as the activation function, and softmax was applied to the output layer. For hyperparameter optimization, the minibatch gradient descent method was used in conjunction with Adam optimizer for 50 epochs for the CNN and MLP. The residual matrixes for the CNN, or the residual vectors for the MLP, were divided into three parts, namely, training (64%), validation (16%), and test set (20%). The experiment was repeated 10 times by changing the seeds each time. Table 5 and Table 6, respectively, summarize the classification accuracy for each iteration and part.

TABLE 5. Classification performance reported using the averaged accuracy score and standard deviation (in parentheses).

Count	1	2	3	4	5	6
CNN	0.972	0.957	0.970	0.974	0.957	0.959
MLP	0.974	0.978	0.981	0.974	0.991	0.979
Count	7	8	9	10	Avg(Std)	
CNN	0.967	0.967	0.969	0.967	0.966(0.005)	
MLP	0.981	0.978	0.978	0.979	0.979(0.004)	

Table 5 shows that the MLP with residual vectors as input outperformed the CNN with residual matrixes as input. Table 6 shows that, in the CNN, the classification accuracy of Part A and Part E was 0 and 0.6525, respectively. In the MLP, the classification accuracy of Part A and Part E was

TABLE 6. Classification performance reported using the averaged accuracy score and standard deviation (in parentheses).

Part	A	B	C	D	E	F
CNN	0.0000 (0.000)	0.9805 (0.014)	0.9778 (0.010)	0.9534 (0.028)	0.6525 (0.075)	0.9979 (0.003)
MLP	0.6750 (0.160)	0.9863 (0.005)	0.9400 (0.024)	0.9375 (0.048)	0.9744 (0.014)	0.9988 (0.002)

0.675 and 0.9744, respectively. In other words, because the MLP performs better than the CNN in terms of the classification accuracy of Parts A and E, the MLP is more accurate than CNN. It was believed that the failure of the CNN to classify Part A was caused by an insufficient number of training data. Table 7 presents the number of residual matrixes for each part.

TABLE 7. Number of residual matrices by part.

Part	A	B	C	D	E	F
Number of residual matrixes	21	645	698	427	97	1203

The number of residual matrixes for Part A was the smallest because the failure period of this part was shorter than those of the other parts. Therefore, it was determined that the CNN was not properly trained because of too few training data.

E. IMPORTANCE OF SENSOR PAIRS' CORRELATION BY USING KernelSHAP

Figure 6 shows the importance of sensor pairs' correlation that contributed to classifying Parts A, B, C, D, E, and F, as ranked by the SHAP method. The greater the absolute influence of pair in the graph, the more it can be interpreted as having a significant impact on prediction of the relevant part.

For example, it can be observed that 7_6 exists at the top in Part C. This implies the correlation between Sensors 7 and 6 had a significant impact on predicting replacement of Part C. For Part E, 6_7, 4_7, and 7_7 exist at the top. If Part E, which controls signal processing, encounters a challenge, the abnormality can be interpreted as occurring in the trends of Sensors 6, 4, and 7 that are related to Part E. Moreover, for both Part A and D, 6_6 is identified as the most important contributing factor. This suggests that the correlation of 6 and itself (i.e., Sensor 6 only) is an influential feature for both Part A and Part D classes. Application of SHAP to the MLP model permits explanation of the relationship between pair sensors (i.e., correlation between the sensors) in identifying the part at fault in an equipment failure.

V. CONCLUSION

This study presents a framework for the use of multivariate sensor data to detect abnormalities and identify their causes in semiconductor manufacturing processes. Signature matrixes are generated to reflect the correlation between the variables and are used as the input to the CAE for anomaly detection. Furthermore, residual matrixes are used to predict the parts to be replaced for each abnormal section detected by CAE. In this study, residual matrixes are transformed into residual vectors to be used as an input for the MLP model. Additionally, SHAP is applied to the MLP to identify the major factors related to the equipment failure. Several future research directions could enlarge upon this study. Because the data in this study had a small number of variables, MLP and SHAP could be applied as classification and explainable algorithms. When the number of variables is large, it is appropriate to apply CNN as a classification model. In addition, when CNN is used in the future as a classification model, a method to determine the relationship between the variables will be investigated and implemented. Nevertheless, even without this further research, the proposed method can contribute to timely and accurate decision making by engineers in trouble-shooting equipment failures and thus increase the productivity of the semiconductor industry.

REFERENCES

[1] S. Munirathinam and B. Ramadoss, "Big data predictive analtyics for proactive semiconductor equipment maintenance," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Oct. 2014, pp. 893–902.

[2] E. L. Park, J. Park, J. Yang, S. Cho, Y.-H. Lee, and H.-S. Park, "Data based segmentation and summarization for sensor data in semiconductor manufacturing," *Expert Syst. Appl.*, vol. 41, no. 6, pp. 2619–2629, May 2014.

[3] Q. Peter 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] T. Kourti and J. F. MacGregor, "Process analysis, monitoring and diagnosis, using multivariate projection methods," *Chemometric Intell. Lab. Syst.*, vol. 28, no. 1, pp. 3–21, Apr. 1995.

[5] H. Conrads and M. Schmidt, "Plasma generation and plasma sources," *Plasma Sources Sci. Technol.*, vol. 9, no. 4, pp. 441–454, Nov. 2000.

[6] A. Thieullen, M. Ouladsine, and J. Pinaton, "Application of principal components analysis to improve fault detection and diagnosis on semiconductor manufacturing equipment," in *Proc. Eur. Control Conf. (ECC)*, Jul. 2013, pp. 1445–1500.

[7] G. Michau, Y. Hu, T. Palmé, and O. Fink, "Feature learning for fault detection in high-dimensional condition monitoring signals," *Proc. Inst. Mech. Eng., O, J. Risk Rel.*, vol. 234, no. 1, pp. 104–115, Feb. 2020.

[8] W. Jiang, Y. Hong, B. Zhou, X. He, and C. Cheng, "A GAN-based anomaly detection approach for imbalanced industrial time series," *IEEE Access*, vol. 7, pp. 143608–143619, 2019.

[9] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection: A review," *ACM Comput. Surv.*, vol. 54, no. 2, pp. 1–38, Mar. 2021.

[10] H. Zhang, P. Wang, X. Gao, H. Gao, and Y. Qi, "Automated fault detection using convolutional auto encoder and k nearest neighbor rule for semiconductor manufacturing processes," in *Proc. 3rd Int. Conf. Intell. Auto. Syst. (ICoIAS)*, Feb. 2020, pp. 83–87.

[11] D. Liao, C. Chen, W. Tsai, H. Chen, Y. Wu, and S. Chang, "Anomaly detection for semiconductor tools using stacked autoencoder learning," in *Proc. Int. Symp. Semicond. Manuf. (ISSM)*, Dec. 2018, pp. 1–4.

[12] W. Yong, C. Xu, and W. Zhengying, "Fault detection of sensor data in semiconductor processing with variational autoencoder neural network," in *Proc. China Semiconductor Technol. Int. Conf. (CSTIC)*, Jun. 2020, pp. 1–3.

[13] L. Ruff, J. R. Kauffmann, R. A. Vandermeulen, G. Montavon, W. Samek, M. Kloft, T. G. Dietterich, and K. Müller, "A unifying review of deep and shallow anomaly detection," *Proc. IEEE*, vol. 109, no. 5, pp. 756–795, May 2021.

[14] G. A. Susto, M. Terzi, and A. Beghi, "Anomaly detection approaches for semiconductor manufacturing," *Proc+Manuf.*, vol. 11, pp. 2018–2024, Jan. 2017.

[15] Y.-R. Yeh, Z.-Y. Lee, and Y.-J. Lee, "Anomaly detection via over-sampling principal component analysis," in *New Advances in Intelligent Decision Technologies*. Cham, Switzerland: Springer, 2009, pp. 449–458.
[16] H.-P. Kriegel, M. Schubert, and A. Zimek, "Angle-based outlier detection in high-dimensional data," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2008, pp. 444–452.
[17] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2000, pp. 93–104.
[18] I. Jahan, Md. M. Alam, Md. F. Ahmed, and Y. M. Jang, "Anomaly detection in semiconductor cleanroom using isolation forest," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2021, pp. 795–797.
[19] F. T. Liu, K. M. Ting, and Z. Zhou, "Isolation-based anomaly detection," *ACM Trans. Knowl. Discovery Data*, vol. 6, no. 1, pp. 1–39, Mar. 2012.
[20] G. Münz, S. Li, and G. Carle, "Traffic anomaly detection using k-means clustering," in *Proc. GI/ITG Workshop MMBNET*, 2007, vol. 7, no. 9, pp. 1–10.
[21] M. Zhang, B. Xu, and J. Gong, "An anomaly detection model based on one-class SVM to detect network intrusions," in *Proc. 11th Int. Conf. Mobile Ad-hoc Sensor Netw. (MSN)*, Dec. 2015, pp. 102–107.
[22] L. M. Manevitz and M. Yousef, "One-class SVMs for document classification," *J. Mach. Learn. Res.*, vol. 2, pp. 139–154, Dec. 2001.
[23] C. Chen, Y. Wang, M. Cui, J. Zhao, W. Bi, Y. Chen, and X. Zhang, "Data-driven detection of stealthy false data injection attack against power system state estimation," *IEEE Trans. Ind. Informat.*, vol. 18, no. 12, pp. 8467–8476, Dec. 2022.
[24] B. Zong, Q. Song, M. R. Min, W. Cheng, C. Lumezanu, D. Cho, and H. Chen, "Deep autoencoding Gaussian mixture model for unsupervised anomaly detection," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–15.
[25] F. Hosseinpour, I. Ahmed, P. Baraldi, M. Behzad, E. Zio, and H. Lewitschnig, "An unsupervised method for anomaly detection in multiystage production systems based on LSTM autoencoders," in *Proc. 32nd Eur. Saf. Rel. Conf. (ESREL)*, M. C. Leva, E. Patelli, L. Podofillini, and S. Wilson, Ed. Singapore: ESREL, 2022.
[26] B. Lindemann, F. Fesenmayr, N. Jazdi, and M. Weyrich, "Anomaly detection in discrete manufacturing using self-learning approaches," *Proc. CIRP*, vol. 79, pp. 313–318, Jan. 2019.
[27] Z. Wang and Z. Jin, "Mining safety event in industrial control network based on the long-short term memory networks," in *Proc. 3rd IEEE Int. Conf. Comput. Commun. (ICCC)*, Dec. 2017, pp. 1255–1258.
[28] B. Lindemann, N. Jazdi, and M. Weyrich, "Anomaly detection and prediction in discrete manufacturing based on cooperative LSTM networks," in *Proc. IEEE 16th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2020, pp. 1003–1010.
[29] M. Canizo, I. Triguero, A. Conde, and E. Onieva, "Multi-head CNN–RNN for multi-time series anomaly detection: An industrial case study," *Neurocomputing*, vol. 363, pp. 246–260, Oct. 2019.
[30] J. Jang, B. W. Min, and C. O. Kim, "Denoised residual trace analysis for monitoring semiconductor process faults," *IEEE Trans. Semicond. Manuf.*, vol. 32, no. 3, pp. 293–301, Aug. 2019.
[31] S.-K.-S. Fan, C.-Y. Hsu, C.-H. Jen, K.-L. Chen, and L.-T. Juan, "Defective wafer detection using a denoising autoencoder for semiconductor manufacturing processes," *Adv. Eng. Informat.*, vol. 46, Oct. 2020, Art. no. 101166.
[32] H. Lee, Y. Kim, and C. O. Kim, "A deep learning model for robust wafer fault monitoring with sensor measurement noise," *IEEE Trans. Semicond. Manuf.*, vol. 30, no. 1, pp. 23–31, Feb. 2017.
[33] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, "USAD: UnSupervised anomaly detection on multivariate time series," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 3395–3404.
[34] S. Gupta, A. Venugopal, and M. J. Mohan, "Fault detection and diagnosis using AutoEncoders and interpretable AI–case study on an industrial chiller," in *Proc. IEEE Int. Symp. Adv. Control Ind. Processes (AdCONIP)*, Aug. 2022, pp. 198–203.
[35] J. Jakubowski, P. Stanisz, S. Bobek, and G. J. Nalepa, "Explainable anomaly detection for hot-rolling industrial process," in *Proc. IEEE 8th Int. Conf. Data Sci. Adv. Anal. (DSAA)*, Oct. 2021, pp. 1–10.
[36] E. Christoforou, K. Blom, Q. Gao, M. Börü, and T. Cataltepe, "MRI condition monitoring with explainable AI and feature selection," in *Proc. 30th Signal Process. Commun. Appl. Conf. (SIU)*, May 2022, pp. 1–4.
[37] C. Zhang, D. Song, Y. Chen, X. Feng, C. Lumezanu, W. Cheng, J. Ni, B. Zong, H. Chen, and N. V. Chawla, "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 1409–1416.
[38] K. Choi, J. Yi, C. Park, and S. Yoon, "Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines," *IEEE Access*, vol. 9, pp. 120043–120065, 2021.
[39] A. Azarang, H. E. Manoochehri, and N. Kehtarnavaz, "Convolutional autoencoder-based multispectral image fusion," *IEEE Access*, vol. 7, pp. 35673–35683, 2019.
[40] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, "Self-normalizing neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–12.
[41] O. Kuchaiev and B. Ginsburg, "Training deep AutoEncoders for collaborative filtering," 2017, *arXiv:1708.01715*.
[42] V. S. Tomar, "A critical evaluation of activation functions for autoencoder neural networks," Ph.D. dissertation, College Comput., Nat. College Ireland, Dublin, Ireland, 2022.
[43] J. Liu, X. Kong, F. Xia, X. Bai, L. Wang, Q. Qing, and I. Lee, "Artificial intelligence in the 21st century," *IEEE Access*, vol. 6, pp. 34403–34421, 2018.
[44] A. B. Arrieta, N. Díaz-Rodríguez, J. D. Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Inf. Fusion*, vol. 58, pp. 82–115, Jun. 2020.
[45] J. Mi, A. Li, and L. Zhou, "Review study of interpretation methods for future interpretable machine learning," *IEEE Access*, vol. 8, pp. 191969–191985, 2020.
[46] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–22.
[47] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. 8th IEEE Int. Conf. Data Mining*, Dec. 2008, pp. 413–422.
[48] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.
[49] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.
[50] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[51] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[52] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 807–814.

MINJAE BAEK received the B.S. degree, in 2016. He is currently pursuing the M.S. degree with the School of Industrial and Management Engineering, Korea University, Seoul, South Korea. His research interest includes anomaly detection and their industrial applications.

SEOUNG BUM KIM received the M.S. and Ph.D. degrees in industrial and systems engineering from the Georgia Institute of Technology, in 2001 and 2005, respectively. From 2005 to 2009, he was an Assistant Professor with the Department of Industrial and Manufacturing Systems Engineering, The University of Texas at Arlington. He is currently a Professor with the School of Industrial and Management Engineering, Korea University. He has published more than 150 internationally recognized journals and refereed conference proceedings. His research interests include utilize machine learning algorithms to create new methods for various problems appearing in engineering and science.