(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
08 October 2026 (08.10.2026)

WIPO | PCT

(10) International Publication Number
WO 2026/212573 A1

(51) International Patent Classification:
H04L 67/12 (2022.01) *G06N 20/00* (2019.01)

(21) International Application Number:
PCT/SG2026/050212

(22) International Filing Date:
01 April 2026 (01.04.2026)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
10202500862P 02 April 2025 (02.04.2025) SG

(71) Applicant: **NANYANG TECHNOLOGICAL UNIVERSITY** [SG/SG]; 50 Nanyang Avenue, Singapore 639798 (SG).

(72) Inventors: **FU, Yuguang**; c/o Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798 (SG). **YU, Xiao**; c/o Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798 (SG). **CUI, Shuaiwen**; c/o Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798 (SG).

(74) Agent: **LINDSAY, Jonas Daniel**; Marks & Clerk Singapore LLP, Privy Box No. 920258, Singapore 929292 (SG).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UY, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, CV, GH, GM, KE, LR, LS, MU, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO,

(54) **Title:** ONBOARD FAULT DIAGNOSIS AND RECOVERY IN WIRELESS SENSOR NODE DEVICE AND METHOD

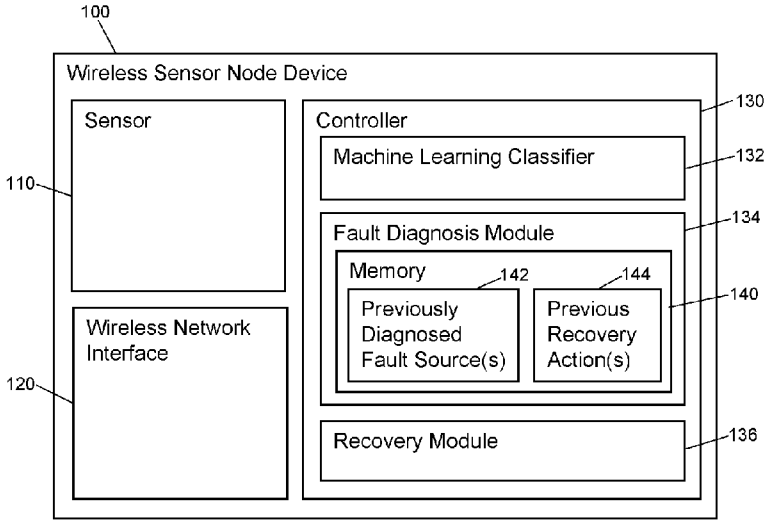


FIG.2

(57) **Abstract:** Wireless sensor node devices and methods in wireless sensor node devices are disclosed A wireless sensor node device comprises: a sensor configured to generate sensor data; a wireless network interface configured to transmit the sensor data from the sensor node device; and a controller comprising: a machine learning classifier configured to identify anomalous data in the sensor data and to classify the anomalous data; a fault diagnosis module configured to diagnose a fault source based on a classification of the anomalous data; and a recovery module configured to generate a recovery action indication based on the fault source.

[Continued on next page]

WO 2026/212573 A1

RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

ONBOARD FAULT DIAGNOSIS AND RECOVERY IN WIRELESS SENSOR NODE DEVICE AND METHOD

TECHNICAL FIELD

The present disclosure relates to Internet of Things (IoT) systems and wireless sensor nodes.

BACKGROUND

The new generation of wireless structural health monitoring (SHM) systems leverages the Internet of Things (IoT) paradigm, integrating smart sensors that not only acquire measurement data but also process and transmit it via the Internet. The data collected by edge devices, aka IoT nodes, can be analysed using machine learning (ML) algorithms to detect patterns or anomalies that may indicate structural weaknesses or the need for maintenance. This enables real-time monitoring of structural integrity by assessing performance, identifying damage or deterioration, and providing a comprehensive understanding of the structure's overall condition. It provides an alternative due to their compact node size, facilitating easy deployment, wireless communication, and their capability for flexible expansion and upgrade, serving as integral components of wireless SHM systems. However, edge devices often face various interferences in the harsh environment, which may result in the transmission of erroneous data. The data quality can be compromised by anomalies such as sensor faults, poor data transmission, harsh environments, unexpected events, and deliberate attacks. Due to harsh environment and vulnerability of low-cost hardware, sensors may produce extensive time-series data frequently interspersed with anomalies that represent a considerable obstacle to automated and reliable alerts for potential damage or accidents and can compromise their efficacy for long-term SHM applications.

Traditional methods for addressing sensor faults include physical and analytical redundancy. Physical redundancy uses dual sensors to compare sensor data with readings from "redundant" sensors placed alongside the primary sensors in SHM systems. While effective, this approach consumes additional resources and costs.

Analytical redundancy uses mathematical correlations among data streams from multiple sensors to generate virtual sensor outputs, reducing reliance on additional hardware. Recent advancements in machine learning (ML) and data-driven approaches have introduced techniques like anomaly detection, spatial-temporal correlation analysis, and predictive modelling to diagnose and recover faulty data. While ML-based strategies offer significantly higher accuracy, most of them are processed in a centralized manner, that is, collecting data from all the sensor nodes and processing it in a central station (e.g., workstations or cloud servers). Central processing not only increases the transmission load for wireless SHM systems, which may result in data loss issues but also extends the significant delay of subsequent data analytics for timely structural condition assessments.

SUMMARY

According to a first aspect of the present disclosure a wireless sensor node device is provided. The wireless sensor node device comprises: a sensor configured to generate sensor data; a wireless network interface configured to transmit the sensor data from the sensor node device; and a controller comprising: a machine learning classifier configured to identify anomalous data in the sensor data and to classify the anomalous data; a fault diagnosis module configured to diagnose a fault source based on a classification of the anomalous data; and a recovery module configured to generate a recovery action indication based on the fault source.

In an embodiment, the machine learning classifier is a lightweight deep multi-class classifier.

In an embodiment, the lightweight deep multi-class classifier is a student model trained using knowledge distillation.

In an embodiment, the fault diagnosis module implements a rule based diagnostic algorithm.

In an embodiment, the recovery module is configured to modify operation of the sensor based on the recovery action indication.

In an embodiment, the recovery module is configured to generate a message or alert based on the recovery action indication.

In an embodiment, the recovery module is configured to modify the sensor data based on the recovery action indication.

In an embodiment, the fault diagnosis module comprises a memory configured to store a previously diagnosed fault source and a corresponding previous recovery action indication and is further configured to determine if the anomalous data is related to the previously diagnosed fault source, and wherein the recovery module is configured to generate the recovery action indication taking into account the previous recovery action indication if the anomalous data is related to the previously diagnosed fault source.

In an embodiment, the memory is configured to store a plurality of previously diagnosed fault sources and a plurality of corresponding previous recovery action indications and the fault diagnosis module is further configured to iteratively determine if the anomalous data is related to a previously diagnosed fault source.

In an embodiment, the wireless network interface is further configured to receive second sensor data from a second sensor node device, the machine learning classifier is configured to identify anomalous data in the second sensor data and to classify the anomalous data, and wherein the recovery module is configured to modify operation of the wireless network interface based on the recovery action indication.

According to a second aspect of the present disclosure a method, in a wireless sensor node device is provided. The method comprises: generating sensor data from a sensor in the wireless sensor node device; transmitting the sensor data from the sensor node device; identifying anomalous data in the sensor data; classifying the anomalous data; diagnosing a fault source based on a classification of the anomalous data; and generating a recovery action indication based on the fault source.

In an embodiment, classifying the anomalous data comprises using a lightweight deep multi-class classifier.

In an embodiment, the lighiweight deep multi-class classifier is a student model trained using knowledge distillation.

In an embodiment, diagnosing a fault source comprises implementing a rule based diagnostic algorithm.

In an embodiment the method further comprises modifying operation of the sensor based on the recovery action indication.

In an embodiment the method further comprises generating a message or alert based on the recovery action indication.

In an embodiment the method further comprises modifying the sensor data based on the recovery action indication.

In an embodiment the method further comprises determining if the anomalous data is related to a previously diagnosed fault source, and generating the recovery action indication taking into account a previous recovery action indication if the anomalous data is related to the previously diagnosed fault source.

In an embodiment the method further comprises iteratively determining if the anomalous data is related to one of a plurality of previously diagnosed fault sources.

In an embodiment the method further comprises receiving second sensor data, via a wireless network interface, from a second sensor node device, identifying anomalous data in the second sensor data, classifying the anomalous data, and modifying operation of the wireless network interface based on the recovery action indication.

BRIEF DESCRIPTION OF THE DRAWINGS

In the following, embodiments of the present invention will be described as non-limiting examples with reference to the accompanying drawings in which:

FIG.1 is a block diagram showing a system comprising wireless sensor nodes according to an embodiment of the present invention;

FIG.2 is a block diagram showing a wireless sensor node device according to an embodiment of the present invention;

FIG.3 is a block diagram showing an overview of a framework for data anomaly classification, fault diagnosis, and sensor fault recovery used in embodiments of the present invention;

FIG.4 is a block diagram showing the structure of a classification model used in embodiments of the present invention;

FIG.5 is a block diagram showing the structure of an encoder-enhanced classification module used in embodiments of the present invention;

FIG.6 is a block diagram showing a knowledge distillation process used in training of classifiers in embodiments of the present invention;

FIG.7 shows an example deployment of a wireless sensor node device according to an embodiment of the present invention;

FIG.8 is a table showing sensor fault types and corresponding recovery strategies used in an embodiment of the present invention;

FIG.9 is a graph showing model size and accuracy for different classification models;

FIG.10 shows a confusion matrix for the classification model in an example implementation;

FIG.11 is a graph showing training loss and validation accuracy per epoch for the classification model in the example implementation;

FIG.12 shows a confusion matrix for a test set of samples onboard;

FIG.13 is a table showing results for a test set of samples on different platforms;

FIG.14a to FIG.14f show data from laboratory tests of a wireless sensor node device according to an embodiment of the present invention;

FIG.15a to 15c show data results of a power supply fluctuation test on a wireless sensor node device according to an embodiment of the present invention;

FIG.16a to 16c show data results of an environment influence test on a wireless sensor node device according to an embodiment of the present invention;

FIG.17 shows the frequencies of signal interference experiment on a wireless sensor node device according to an embodiment of the present invention;

FIG.18a and 18b show data results of a signal interference experiment on a wireless sensor node device according to an embodiment of the present invention;

FIG.19a and FIG.19b are flow diagrams showing autonomous recovery and data recovery of fault diagnosis in embodiments of the present invention; and

FIG.20A to FIG.20D show results of iterative data recovery.

DETAILED DESCRIPTION

The present disclosure introduces a new approach to sensor fault management in wireless IoT-based systems. Key novel features include:

Diagnosis of underlying sources of sensor faults. The approach goes beyond merely detecting anomalies by systematically diagnosing the underlying causes of sensor

faults. Through validation using real-world experiments, it correlates specific fault types (e.g., drift, missing data, or outliers) with their root cause types (e.g., power supply, signal interference). This capability enables accurate and actionable insights, addressing not just the symptoms but also the sources of sensor anomalies, which are often overlooked in traditional data-driven methods. The shift from symptom-based classification to a root-cause-driven recovery approach provides engineers with actionable insights for fault mitigation and reduces false alarms.

System recovery for sensor faults. The approach provides adaptive recovery strategies tailored to specific fault scenarios, preserving the diagnostic potential of raw sensor data and restoring system functionality. By implementing targeted recovery directly at the sensor level, the system restores normal operations without requiring manual intervention or additional hardware redundancies. This feature significantly enhances the reliability and longevity of SHM systems by addressing faults at their origin rather than compensating for their downstream effects on data analysis and decision-making processes. Existing techniques have additional cost-consuming or abandoned raw data for data recovery. Some require manual intervention or external decision-making after data anomaly classification. The proposed approach enables autonomous real-time fault diagnosis and recovery at the sensor level, significantly reducing maintenance costs and human intervention and enhancing the reliability of SHM systems in remote or inaccessible locations where immediate human intervention is not always feasible.

Decentralized sensor fault management. The mechanism is optimized for resource-constrained environments and deployed onboard, enabling real-time fault diagnosis directly on microcontroller-based sensor nodes without relying on external computational resources. By processing data locally, this approach eliminates the need for extensive data transmission to remote servers, reducing latency and computational overhead, enabling real-time system detection and recovery. By facilitating autonomous and decentralized fault management, the framework significantly enhances the resilience, autonomy, and operational efficiency of SHM systems. The proposed wireless sensor nodes implement an optimized lightweight classification model and a rule-based recovery algorithm for lower energy consumption and resource-constrained environments. They may be deployed on

MCU-based IoT nodes, eliminating the need for cloud-based AI processing and making it more suitable for large-scale, battery-powered IoT deployments and improving system efficiency. It introduces a decentralized approach with edge intelligence, allowing real-time, local fault diagnosis and recovery, reducing dependency on centralized data transmission, and minimizing latency.

Iterative diagnosis mechanism to address complex and multi-faults scenarios. The process begins with AI classification, which identifies the primary fault type. The classification result serves as an input to a rule-based diagnostic algorithm that establishes relationships between data anomalies and system faults. Once a fault is diagnosed, corresponding recovery actions are taken onboard immediately to avoid meaningless monitoring operations, and the mechanism returns to the detection part. If additional faults are detected, the data undergoes re-evaluation, and the cycle repeats until the data is classified as "normal," indicating stability. At this point, the process terminates automatically, conserving resources by bypassing further diagnosis and recovery steps. This can handle complex fault interactions and enable more effective real-world deployments where faults are often correlated, overcoming a limitation in existing studies, which assume independent single-fault classification. This iterative detection and correction mechanism allows for real-time adaptation to multiple faults rather than relying on brute-force deep learning models, reducing computational cost and model complexity.

Lightweight classification model development. The present disclosure provides a lightweight classification model that balances model compactness and accuracy. Firstly, an encoder-enhanced classification model (EECM) is proposed as a high-capacity teacher model capable of robust anomaly detection. The EECM incorporates a multi-head attention-based encoder to extract and enhance features from raw data using domain knowledge, excelling in classification tasks. Knowledge distillation is then applied to compress the teacher model into a distilled lightweight classification model (DLCM) optimized for edge deployment without significant loss in accuracy. Advanced data anomaly classification models process fault signals and classify them based on feature extraction and deep learning models. Many machine- learning and deep learning models for data classification often demonstrate a positive correlation between model complexity and task performance. Although such complex models

excel in classification tasks, they demand significant computational resources, rendering them unsuitable for deployment on edge devices. The model of the present disclosure not only achieves excellent classification performance but is also optimized to a small model size that can be deployed on low-cost MCU-based edge nodes, a comparison with some other deep learning models regarding accuracy and model size.

Deployment framework of the deep learning model. The present disclosure proposes a comprehensive deployment process of deep learning model. The deployment implementation framework involves model preparation and upload, converting the trained model from its original framework format (e.g., PyTorch) to the ONNX format. Once uploaded, the model undergoes benchmarking and online evaluation. Once the optimal microcontroller is selected, a complete project is generated including the necessary C code, libraries, and configuration files to run the model on the target hardware. Finally, the optimized code is flashed onto the physical microcontroller using a debugger/programmer interface. Deploying machine learning models on microcontrollers poses unique challenges due to limited computational resources and memory constraints. This ecosystem significantly enhances the efficiency of developing and deploying embedded applications. This framework simplifies tasks such as peripheral configuration, code writing and debugging, microcontroller programming, and firmware management. Their free availability and open-source nature facilitated the development process and ensured accessibility for replication.

Together, these features establish a novel paradigm for intelligent, fault-resilient SHM systems that dynamically adapt to real-world challenges. The present disclosure's integration of onboard fault diagnosis, real-world validation of fault sources, and tailored recovery mechanisms sets it apart from existing technologies, offering a scalable and efficient solution for IoT-enabled SHM applications.

The proposed wireless sensor node infrastructure introduces a novel system-centric approach to sensor fault diagnosis and recovery, departing from the traditional data-centric focus. Unlike prior methods, this mechanism targets the root causes of sensor faults by incorporating onboard diagnostic capabilities and retaining raw sensor data for further analysis. The key advantages are as follows:

Autonomous restoration of system operation. The present disclosure addresses the underlying causes of sensor faults, departing from traditional data-centric methods that only address the symptoms (e.g., data anomalies). Identifying sources of faults, such as environmental interference, sensor hardware issues, or calibration errors, and recovering them systematically, not only restores normal operations but also supports further data collection and analysis of structural health monitoring.

Preservation of raw data. Traditional methods rely on time-spatial correlations and AI-based reconstructions, replacing raw sensor data with approximations that often overlook valuable diagnostic insights. The proposed system preserves raw data by addressing faults at the sensor level and ensures minimal data loss through system-centric recovery.

Multi-fault detection and recovery. The iterative fault management mechanism enables the simultaneous detection and recovery of multiple faults, overcoming the limitations of traditional sensor fault diagnosis methods, which are typically restricted to addressing single faults. By addressing one primary fault type at a time and looping back to identify additional faults, the system overcomes the limitations of traditional methods, which often rely on complex models that can only handle two simultaneous fault types.

Efficient resource utilization. While recent advances improve detection accuracy, their reliance on centralized servers or cloud-based processing introduces significant data transmission overhead, delays, and risks associated with centralized failures. The present disclosure employs a lightweight classification model and a rule-based fault diagnosis and recovery algorithm, significantly reducing computational demands compared to complex machine learning-based approaches. The iteration of the sensor fault management process avoids complex data-driven models to identify data anomalies and reduces computational complexity. This makes the system particularly well-suited for low-cost, resource-constrained SHM applications.

Independence from time-spatial correlations. Unlike traditional methods that depend on neighboring sensors for fault detection and data reconstruction, the system-centric recovery strategies enable the independence of time-spatial correlations. This

independence enhances its adaptability and reliability, especially in scenarios where neighboring sensor data is unavailable or compromised.

FIG.1 is a block diagram showing a system comprising wireless sensor nodes according to an embodiment of the present invention. As shown in FIG.1, a plurality of wireless sensor node devices 100 are deployed in an environment 10. The environment 10 may be, for example, a structure such as a bridge which is undergoing structural health monitoring. The wireless sensor node devices 100 comprise sensors which generate sensor data. This sensor data is transmitted wirelessly to a gateway device 20 which sends the sensor data to a data analysis device 30. The wireless sensor node devices 100 may form an ad-hoc network in which the sensor data is sent between wireless sensor node devices 100 before reaching the gateway device 20. As described in more detail below, the wireless sensor node devices 100 implement on-device fault classification, identification and recovery.

FIG.2 is a block diagram showing a wireless sensor node device according to an embodiment of the present invention. As shown in FIG.2, the wireless sensor node device 100 comprises a sensor 110, a wireless network interface 120 and a controller 130. The sensor 110 may be an accelerometer which records vibration data, a gyroscope to measure angular velocity, and / or a thermometer to measure temperature; however, it will be appreciated embodiments of the present invention may implement various types of sensors. The wireless network interface 120 allows the wireless sensor node device 100 to send sensor data generated by the sensor 110 to other wireless sensor node devices or a gateway device. The wireless network interface 120 may also allow the wireless sensor node device 100 to receive sensor data generated by sensors on other wireless sensor node devices for forwarding as part of an ad-hoc network. The wireless network interface may be a Zigbee, Bluetooth, WiFi LoRa, NB-IoT, LTE-M, or other type of wireless network interface. It will be appreciated that embodiments of the present invention are not limited to these communication protocols. The controller 130 may be implemented as a microcontroller.

The controller 130 allows the wireless sensor node device 100 to perform on-device anomaly classification, fault diagnosis and recovery. As shown in FIG.2, the controller comprises a machine learning classifier 132, a fault diagnosis module 134 and a

recovery module 136. The machine learning classifier 132 may be implemented as a lightweight deep multi-class classifier, and is configured to identify anomalous data in sensor data and to classify the anomalous data. The fault diagnosis module 134 is configured to diagnose a fault source based on the classification of the anomalous data. The fault diagnosis module 134 comprises a memory 140 which stores previously diagnosed fault sources 142 and previous recovery actions 144. The previously diagnosed fault sources 142 and previous recovery actions 144 may be used in iteratively diagnosing faults. The recovery module 136 is configured to generate a recovery action indication based on the fault source.

FIG.3 is a block diagram showing an overview of a framework for data anomaly classification, fault diagnosis, and sensor fault recovery used in embodiments of the present invention. As shown in FIG.3 proposed system 300 operates in three main stages: data anomaly classification 310, sensor fault diagnosis 320, and system-centric sensor fault recovery 330. The workflow begins with the data anomaly classification 310 stage, where raw sensor data is pre-processed, and anomalies are identified by a machine learning model 312 as one of the predefined fault types. In the sensor fault diagnosis 320 stage, root causes of the anomalies are traced to specific factors such as environmental conditions 324, power depletion 322, signal interference 326 or sensor variables 328, leveraging onboard computational resources. Finally, the sensor fault recovery 330 stage applies tailored algorithms to correct identified faults, with seamless feedback loops to ensure continuous fault detection and correction. This may include autonomous recovery 332, in-situ recovery 334 and data recovery 336.

FIG.4 is block diagram showing the structure of a classification model used in embodiments of the present invention.

The machine learning classifier 132 of the wireless sensor node device 100 may be implemented by training an encoder-enhanced classification model (EECM) 420 to be capable of robust anomaly classification as a teacher model and applying knowledge distillation to produce a smaller, resource-efficient student model, DLCM 430. The distilled model maintains comparable accuracy while being optimized for deployment on MCU-based IoT nodes. Firstly, an encoder-enhanced classification model 420

(EECM) is developed for effective data anomaly classifications, which are designed to extract, compress, and classify features. Statistical (mean, range, and standard deviation) and histogram features 415 are extracted from the raw time series data 410, and an encoder 422 based on the attention mechanism is applied to quantify the correlation between these features 415, thereby highlighting information crucial for the subsequent classification task. A classifier 424 is used to classify different data types 426. It is effective to extract features from raw time series data 410 using domain knowledge and a transformer encoder.

The EECM 420 demonstrates excellent performance in accurately classifying data anomalies. However, its computationally intensive nature makes it not suitable for deployment on edge devices. To address this limitation, knowledge distillation 428 is employed to reduce model size by transferring knowledge from a high-capacity teacher model to a compact and lightweight student model, namely a distilled lightweight classification model (DLCM) 430.

The knowledge distillation process uses response-based knowledge and leverages both hard labels (ground truth) and soft labels (predictions from the teacher model) to guide the training of the student model DLCM 430. This dual-label approach ensures that the student model captures not only the ground truth but also the correlation among classes of the teacher model. The teacher model EECM 420 is pre-trained using a transformer encoder 422, which extracts and compresses features from raw timeseries data 410, yielding high-quality feature representations for superior data anomaly classification.

FIG.5 is a block diagram showing the structure of an encoder-enhanced classification module used in embodiments of the present invention.

Histogram and statistical features 515 are extracted from raw time series data 510. The extracted features are input into a transformer-encoder 520 which performs feature enhancement. As shown in FIG.5, the transformer-encoder 520 comprises a multi-head attention layer 530, a batch normalization layer 522, a feed forward layer 524 and an add & normalization layer 526. The multi-head attention layer 530 comprises a first linear layer 532, a scaled dot-product attention layer 534, a concat

layer 536 and a second linear layer 538. The transformer-encoder 520 processes input features through attention and feed-forward layers, dynamically assigning attention weights and applying nonlinear transformations to improve the discriminative power of the extracted features. This approach enables the model to learn and encode complex relationships within the data, leading to more effective anomaly classification. Additionally, residual connections and batch normalization are integrated to improve stability and performance.

Self-attention evaluates the significance of each feature relative to others within a sequence, while multi-head attention allows the model to focus on multiple parts of the input simultaneously. The keys, values, and queries are derived from the same output of the preceding feature extraction layer, allowing each position in the sequence to attend to all others. This mechanism captures feature dependencies, enhancing the effectiveness of feature extraction. Self-attention specifically enables the model to establish connections between different positions within a single sequence, facilitating the computation of a comprehensive representation of that sequence. The attention function is as follows:

$$Attention\,(Q,K,V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

where Q denotes the query of the signal, K denotes the key of the signal, V denotes the value of the signal, K^T denotes the transpose of the key, and $\sqrt{d_k}$ denotes the square root of the dimension of key. The attention function maps a query and a set of key-value pairs to an output which is a weighted sum of the values. The weights are calculated based on the compatibility between the query q, and each key k. The attention function processes a batch of queries as a matrix Q, with keys and values organized into matrices K and V, respectively. The output matrix is obtained by calculating the dot products between the queries and keys, normalizing by $\sqrt{d_k}$, and applying a Softmax function to derive the weights.

A single attention head, due to its averaging nature, may limit the model's ability to focus on multiple aspects of the data simultaneously, potentially constraining its

capacity to capture complex relationships and dependencies. Multi-head attention addresses this limitation by enabling the model to focus on different representation subspaces and positions concurrently, thereby enhancing information processing efficiency, as given above. The approach utilizes h-learned linear projections that transform queries, keys, and values to dimensions d_k, d_k, and d_v, respectively. For each head, the dimensions are set as $d_k = d_v = d_{model} = h$, allowing the attention function to be applied in parallel to the transformed queries, keys, and values. The resulting output values are then concatenated and passed through another linear projection to produce the final values. This method maintains a computational expense comparable to that of single-head attention with full dimensionality.

$$MultiHead(Q, K, V) = Concat\,(head_1, \ldots, head_h)$$

$$\text{Where } head_i = Attention\left(QW_i^Q, KW_i^K, VW_i^V\right)$$

Where the projections are parameter matrices $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}, W_i^K \in \mathbb{R}^{d_{model} \times d_k}, W_i^V \in \mathbb{R}^{d_{model} \times d_v}, W_i^0 \in \mathbb{R}^{hd_v \times d_{model}}$. The multi-head attention mechanism enhances the model's ability to capture a broader range of information and relationships within the data, thereby improving its understanding and representation of complex patterns. This approach not only accelerates data analysis but also efficiently allocates resources to extract the most relevant information. By assigning weights and computing inter-feature correlations, the attention mechanism boosts the representational power of the extracted features, making the model more effective in processing and interpreting intricate data.

The feedforward layer 524 applies a position-wise feedforward neural network to each position in the sequence independently, as given below. The feedforward layer processes the output of the attention layer by applying two linear transformations and a ReLU activation function in between, thereby generating a refined feature representation:

$$FNN(x) = max(0, xW_1 + b_1)W_2 + b_2$$

The first linear transformation increases the dimensionality of the features, allowing the network to capture more complex patterns and interactions within the data. The ReLU activation function introduces nonlinearity, enabling the model to learn and represent intricate relationships that cannot be captured by linear transformations alone. The second linear transformation reduces the dimensionality back to the original size, ensuring that the overall structure of the data is preserved. By incorporating these transformations, the feedforward layer 524 generates a more refined feature representation, enhancing the encoder's ability to model and understand the underlying structure of the input sequence.

Batch normalization is applied after each sublayer in the encoder, with residual connections around each sublayer to stabilize the training process. Unlike the conventional transformer encoder, which uses layer normalization, batch normalization is opted for because each feature extracted from the time series data in the study are independent. Batch normalization scales feature independently within each batch, whereas layer normalization normalizes the output of each layer. This modification has demonstrated improved performance, indicating that batch normalization is an effective optimization strategy for handling independent features in the specific application. Residual connections help maintain gradient flow and mitigate the vanishing gradient problem, thereby enhancing the model's stability and convergence. Since the sensor collected data are sequential and lack cross-data type dependencies, embedding or positional encoding in the transformer encoder is unnecessary.

The output from the transformer-encoder 520 is fed into a classifier 540. The classifier comprises a fully connected layer 542 and an output layer 544. After feature extraction and compression through the transformer encoder, the resulting sequence of feature vectors is considerably shorter yet richer in discriminative information compared to the original signal. Subsequently, this enhanced sequence of feature vectors will be input into a neural network for classification in the subsequent step. The data anomaly classifier includes a flattened layer, two fully connected layers of 64 neurons and 32 neurons, respectively, and an output layer of seven neurons same as the number of data classes. The feature vector ranges from 21 to 1 after normalization, the Tanh activation function is used in the fully connected layer establishing mappings between

features and output. The output layer applied the softmax activation function, with the element representing the probability of belonging to each class. The supervised learning approach is applied utilizing a well-suited classification algorithm. The classifier is trained on a labeled dataset comprising both normal and anomalous instances, allowing it to learn the distinguishing characteristics of anomalies and differentiate them from regular patterns. Mean-square error is used as the objective function to measure the difference in probability distribution between the true and predicted classes.

FIG.6 is a block diagram showing a knowledge distillation process used in training of classifiers in embodiments of the present invention.

Knowledge distillation is employed to reduce model size by transferring knowledge from a high capacity teacher model 610 which is an EECM to a compact and lightweight student model 620 which is a DLCM. The knowledge distillation process uses response-based knowledge and leverages both hard labels (ground truth) and soft labels (predictions from the teacher model 610) to guide the training of the student model 620. This dual-label approach ensures that the student model captures not only the ground truth but also the correlation among classes of the teacher model. The teacher model 610 (EECM) is pre-trained using a transformer encoder, which extracts and compresses features from raw time-series data 602, yielding high quality feature representations for superior data anomaly classification. During distillation, the teacher model outputs logits 612, which represent the probability distribution over classes. These logits 612 are then transformed into soft targets using a softmax function 614 with a temperature parameter T. The temperature T smooths the probability distribution, highlighting interclass relationships that guide the student model DLCM during training. Similarly, the student model generates logits 622 which are then transformed into soft targets using a softmax function 624 with a temperature parameter T.

A distillation loss 630 is calculated between the temperature from the teacher model 610 and the temperature from the student model 620. Additionally, a student loss 640 is calculated between the output from the student model 620 and a ground truth 642.

A total loss 650 is calculated by combining the student loss 640 and the distillation loss 630. The student model 620 is optimized to minimize this total loss 650.

This approach is simple yet effective for compressing large models and has been widely used in various applications. The response-based method captures "dark knowledge" embedded in the teacher model's soft targets, which contain critical insights about the relationships between different classes.

FIG.7 shows an example deployment of a wireless sensor node device according to an embodiment of the present invention.

The deployment workflow includes model conversion, performance evaluation, and on-board validation. First, in a model preparation stage 710 the trained model is converted to ONNX format to ensure hardware compatibility. In a process and evaluation stage 720 the model is then uploaded and benchmarked on various microcontrollers to assess memory usage, inference time, and power consumption. Based on benchmarking results, the optimal device is selected for deployment. Then in a project generation stage 730, a complete project is generated, including the necessary C code, libraries, and configuration files to run the model on the target hardware. The project is first simulated in a desktop environment to verify the accuracy of the model before deploying the optimized code onto the physical board for real-world testing. The model's accuracy is assessed by implementing it in C and running predictions against known datasets, which ensures the model's inference capabilities are tested thoroughly.

Feature extraction: The system derives statistical (mean, range, and standard deviation) and histogram features from raw time-series data to capture sensor anomalies. These features serve as the foundation for anomaly detection.

Feature compression: Using an attention-based encoder, the model highlights critical correlations among features, enhancing anomaly classification accuracy. Residual connections and batch normalization ensure computational stability.

Classification model: A neural network with fully connected layers is employed to categorize data into seven classes (e.g., normal, missing, drift). The model is trained using supervised learning on a labeled dataset.

Model compression: Knowledge distillation technique is applied to transfer the knowledge from a teacher model to a student model, generating a distilled lightweight classification model that can be deployed on resource-constrained edge devices.

Deployment: The classification model is deployed onto resource-constrained hardware (e.g., STM32 microcontrollers) following rigorous quantization and optimization. STM32Cube.AI facilitates model deployment by generating compatible C code, ensuring real-time fault detection onboard.

In summary, the proposed edge intelligence framework shows a systematic approach integrating advanced machine learning models into embedded systems, ensuring efficient performance and good accuracy while leveraging the robust development tools and ecosystem provided by STMicroelectronics.

FIG.8 is a table showing sensor fault types and corresponding recovery strategies used in an embodiment of the present invention.

The structural elements of wireless sensor networks, from individual sensor nodes to the network architecture, can contribute to fault generation. Major sources of faults include the power supply, environmental influences, transmission interference, or sensor variables, such as limitations in sensor range and resolution. Lab experiments are designed and conducted to investigate and evaluate the sources of sensor faults, providing insights for targeted recovery and mitigation strategies. Firstly, to simulate battery voltage variation and the influence of power supply, a source meter was employed. The source meter was programmed to apply a time-dependent voltage sequence to simulate voltage variation and record collected data. Secondly, a climate chamber is applied to simulate and assess the environmental effects, including healing and cooling experiments. Thirdly, signal interference from close equipment may have an effect on data quality. To evaluate the effects of signal interference on data collection, an experiment that positioned a TP-Link router near the sensor node and

tested various Zigbee channels for data collection. Finally, a high-level excitation test and an ambient vibration test are implemented to evaluate the effects of sensor range and resolution limitations. The results and relationship between data anomaly and sources of sensor faults are summarized in FIG.8.

A brief diagnosis algorithm is described below.

- Missing: Differentiates between hard faults (complete data loss) and soft faults (partial data loss). Soft faults are diagnosed based on the measured data count relative to the expected sampling rate.

- Minor: Diagnoses inadequate resolution by analyzing consecutive differences. If over 80% of the readings lie within the noise threshold, the system identifies insufficient resolution.

- Square: Detects sensor saturation by identifying clipped responses near the sensor's range limit (e.g., ±6 g) with a calibration tolerance (e.g., ±0.5 g).

- Outlier: Detects isolated spikes caused by extreme conditions like low temperatures (< -20°C) or loose electrical contacts by analyzing voltage and temperature thresholds.

- Drift: Identifies non-stationary vibration patterns caused by battery voltage variations using consecutive difference analysis (threshold: > 0.1 V).

- Trend: Diagnoses steady directional shifts attributed to temperature variations exceeding a rate-of-change threshold of 30%.

- Bias: Flags sudden measurement deviations caused by battery drops (e.g., 3.4 V) or sensor rotation using a derivative-based approach and gyroscope data.

Sensor data are recovered from identified faults using a system-centric approach to retain the raw data as much as possible and restore normal sensor function for future operations. In cases where environmental factors are responsible for sensor faults,

the sensor may need to remain idle until conditions improve and implement data recovery onboard to allow for accurate data processing in subsequent analyses.

Autonomous recovery: Includes dynamic channel adjustments for "missing" faults and range adjustments for "minor" and "square" faults.

"Missing" data often results from channel interference, with each channel operating at a different radio frequency and responding uniquely to the same signal source. Certain channels are free from interference for specific signal sources, and sensor nodes can switch channels as needed. So, the software initiates an alert and automatically switches to the next available channel to recover from "missing" faults. Channels are adjusted to maintain stable network performance, preserve data rate, and minimize latency. The warning signals to the engineer that data loss has occurred, prompting the automatic channel adjustment to ensure continued data collection.

The "square" and "minor" faults arise from limitations related to sensor range and resolution, respectively. The sensor node often has an accelerometer with different range settings, for example, ±2g, ±4g, ±8g, and ±16g. Sensor range and resolution have a negative correlation, and the trade-off is shown in Equation (1). The recovery strategy for "square" and "minor" faults involves modifying the sensor range automatically in software by one step. If a "square" fault is detected and diagnosed as a range issue, the sensor node checks whether it is already at the maximum range (16g). If so, it outputs a warning indicating that the maximum range has been reached and suggests replacing the sensor with a more suitable one. Otherwise, it increases the range by one step, for example, from 2g to 4g, noting both the original and updated range. For "minor" faults, identified as resolution issues, the recovery process works in reverse. When a "minor" fault is detected, the sensor checks if it is at the minimum range setting. If so, a notification is issued indicating that a more precise sensor is required. Otherwise, the sensor range is reduced by one step to improve resolution, for example, from 4g to 2g, thus addressing the fault.

$$resolution = R/2^{bit\ depth} \quad (1)$$

Where R represents the full-scale range.

In-situ recovery by engineers: Generates actionable alerts (e.g., "Charge battery" or "Fix loose contact") for faults requiring human intervention, such as "drift" and "outlier" caused by low battery or poor contact.

For “drift” caused by insufficient battery power and “outlier” resulting from poor contact, sensor maintenance is required on-site. To address these faults, the software code includes warning messages, such as “Please charge the battery! and “Please fix the loose contact!” to prompt on-site personnel to repair the sensors, which help prevent the collection of erroneous data, thereby supporting the accuracy of structural condition assessments and ensuring reliable structural health monitoring. Additionally, “drift” and “outlier” may occur with other anomaly data simultaneously due to other causes, prompting the need for data recovery. The recovery strategy for these anomalies is detailed in the following.

Data recovery: Faulty data segments are corrected using tailored algorithms. “Drift” and “trend” data are detrended using lightweight moving average filter. “Outlier” is replaced with mean-adjusted values based on the segment's standard deviation, and “bias” is recovered by restoring mean values to biased segments.

For some factors, such as temperature variation and sensor rotation, they need to wait for environmental recovery and sensor stabilization. The data recovery is implemented for these data. To account for temperature “trends”, an onboard detrend method is implemented to compensate for temperature effects. By applying a moving average filter, noise interference is effectively reduced, allowing clearer identification of primary data components and trends. The final trend-corrected data is obtained by subtracting the smoothed data from the raw data and adding the overall average of the raw data, as shown in equation (2). The moving average filter is computationally simple and requires minimal processing power, making it suitable for implementation on sensor devices. This method effectively handles both linear and nonlinear trends while being more resource-efficient than adaptive polynomial fitting for detrending.

To recover "bias" in data, an onboard derivative approach is used. The derivatives of raw accelerometer data within each window are first calculated during the detection stage. The window with a maximum derivative value that indicates the occurrence of bias is analyzed and recovered. Once the bias index is identified, data after the bias index is recovered by adding the mean of the preceding data, as shown in equation (3). This method is implemented directly on the sensor to correct bias when diagnosed as sudden rotation automatically. It's worth noting that "bias" usually happens with "drift", for this case, the recovery strategy for the "drift" after "bias" recovery implements data recovery as well. The "drift" recovery is implemented the same as the "trend" recovery.

$$y_a = y_r - y_s + x_r \quad (2)$$

$$y_a = y_{ab} - x_{bb} \quad (3)$$

Where y_a represents the data after recovering, y_r represents the raw data, y_s represents the data after smoothing, x_r represents the mean value of the raw data, y_{ab} represents the data after bias index, x_{bb} represents the mean value before bias index.

FIG.9 is a graph showing model size and accuracy for different classification models. The x-axis shows model size in megabytes and the y-axis shows accuracy as a percentage. As shown in FIG.9, models such as Googlenet, Pattern Recognition, and Base Classification Model all follow a general trend with accuracy maximum of between 95% and 96%. However, Encoder Enhanced Classification Model (EECM) and Distilled Lightweight Classification Model (DLCM) have accuracies of between 97% and 98% for relatively small model sizes. This makes these models suitable for implementation on low-cost MCU-based edge nodes.

An example implementation will now be described using the proposed lightweight deep learning (DL) classification method was validated using a real-world structural monitoring dataset obtained from a long-span cable-stayed bridge in China. The dataset consists of acceleration measurements collected over a continuous period of 31 days from 38 accelerometers distributed across the bridge. Each accelerometer

records vibration data at a sampling rate of 20 Hz, resulting in one-hour time-series segments containing 72,000 data points per channel. In total, the dataset comprises 28,272 labeled data instances.

FIG.10 shows a confusion matrix for the classification model in an example implementation. The data are categorized into seven classes, including one normal operating condition and six anomaly types: missing, minor, outlier, square, trend, and drift. Using this dataset, the proposed classification model achieved a final test accuracy of 96.36%, demonstrating its effectiveness in distinguishing normal signals from multiple anomaly types. The confusion matrix indicates a high proportion of correct classifications across all categories, with limited misclassification between fault types.

FIG.11 is a graph showing training loss and validation accuracy per epoch for the classification model in the example implementation. Training and validation results further show stable convergence behavior, with a rapid reduction in training loss and consistent validation accuracy over successive epochs. These results confirm the robustness and reliability of the proposed lightweight DL model for sensor data validation under diverse real-world conditions.

Further validation was performed to assess the deployability of the proposed classification method across heterogeneous computing platforms, including desktop systems and resource-constrained microcontroller-based IoT nodes.

FIG.12 shows a confusion matrix for a test set of samples on the microcontroller of STM32. The model was evaluated using a fixed test set of 550 samples and demonstrated consistent classification accuracy across all platforms, confirming hardware-independent performance.

FIG.13 is a table showing results for a test set of samples. When deployed on an STM32-based IoT node, inference was executed sequentially at the sensor node, with temporary buffering used solely for local feature extraction. The embedded implementation maintained the same classification accuracy as desktop implementations, while operating within the computational limitations of the

microcontroller. The total memory footprint of the deployed model, including feature extraction and inference modules, remained within the available on-chip flash and RAM resources. Although inference latency on the microcontroller increased relative to desktop environments due to reduced processing capability, the model remained stable and fully functional during continuous operation. These results demonstrate that the proposed lightweight classification method can be reliably embedded into sensor nodes, enabling on-device data validation and fault classification without reliance on centralized computing resources. This confirms the suitability of the wireless sensor node device for edge-intelligent, AIoT-based sensing systems operating under real-world resource constraints.

To validate the proposed fault diagnosis and recovery mechanism, laboratory experiments were designed to investigate primary sources of sensor faults, including sensor operating limits, power and hardware constraints, environmental influences, and signal interference. These controlled tests validate explicit causal links between observed data anomalies and their underlying physical sensor fault mechanisms, thereby enabling targeted recovery and mitigation strategies.

Two representative experimental scenarios were conducted: a high-level excitation test and an ambient vibration test to evaluate sensor limitations related to measurement range and resolution. In the high-level excitation test, a high-range reference accelerometer was co-located with a low-range wireless sensor node to enable simultaneous data acquisition at the same physical location. The sensors were mounted on a gantry-supported beam structure and subjected to repeated impact excitation using a calibrated modal hammer in a controlled laboratory environment. The excitation amplitude was intentionally increased beyond the operational range of the low-range sensor to assess the effects of signal saturation and clipping on data integrity. In the ambient vibration test, sensors were placed on a vibration-isolated optical table to evaluate performance under low-amplitude conditions. External disturbances were minimized to ensure stable measurement conditions. For comparative analysis, sensors with different resolution levels were deployed concurrently to assess the impact of quantization limits on signal fidelity. The results show that insufficient resolution leads to stepwise and distorted signal representations,

whereas higher-resolution sensing provides smooth and consistent responses under identical conditions.

FIG.14a to FIG.14f show data from laboratory tests of a wireless sensor node device according to an embodiment of the present invention. FIG.14a shows data results from the insufficient sensor range during the high-level excitation test. FIG.14b shows a zoomed-in view of the data results between indices 5000 and 8000 during the high-level excitation test. FIG.14c shows data results from insufficient resolution during the ambient vibration test. FIG.14d shows a zoomed-in view of the data results from the first 100 points during the ambient vibration test. FIG.14e shows data results of the ambient vibration comparison test from the node with sufficient resolution. FIG.14f shows zoomed-in view of the data results of the first 100 points in the experiment.

Controlled laboratory tests were conducted to simulate battery voltage fluctuations, with the recorded sensor outputs showing progressive drift and bias anomalies as the supply voltage decreased. A programmable source meter was used to supply a regulated and time-varying voltage to the wireless sensor node, replacing the nominal battery supply. The standard operating voltage of the sensor node is approximately 3.5 V. During testing, the supplied voltage was systematically varied to emulate realistic battery degradation scenarios while sensor data were continuously recorded. In the first test, the supply voltage was gradually reduced from 4.0 V to 3.5 V in small increments, and sensor responses were collected over a fixed duration under stable mechanical conditions. In a second test, the voltage was further reduced from 3.5 V to 3.3 V to examine the effects of under-voltage operation. The experimental results demonstrate a clear relationship between power supply degradation and sensor data anomalies.

FIG.15a to 15c show data results of a power supply fluctuation test on a wireless sensor node device according to an embodiment of the present invention. FIG.15a shows data results of the z-axis during the power supply experiment from voltage 4.0V to 3.5V. FIG.15b shows data results of the y-axis during the power supply experiment from voltage 3.5V to 3.3V. FIG.15c shows data results of the z-axis during the power supply experiment from voltage 3.5V to 3.3V. As the voltage decreased, the sensor output exhibited progressive drift and bias in the time-domain response. When the

voltage dropped below a critical threshold, data instability increased significantly, and intermittent signal interruption was observed. These behaviors confirm that power supply variation can directly induce characteristic data anomalies, including drift, bias, and data loss.

FIG.16a to 16c show data results of an environment influence test on a wireless sensor node device according to an embodiment of the present invention. FIG.16a shows data results from a heating experiment. FIG.16b shows data results from a cooling experiment. FIG.16c shows data results from a cooling experiment.

Controlled temperature variation tests were conducted using a climate chamber, and the resulting sensor data exhibited clear trend-type behavior in response to systematic temperature changes. The climate chamber enables precise and repeatable control of temperature, allowing systematic assessment of sensor behavior under thermal stress. Two representative scenarios were examined: a heating test and a cooling test. In the heating test, the ambient temperature was gradually increased from approximately 30°C to elevated temperature levels to assess high-temperature effects. In the cooling test, the temperature was reduced from room temperature to near-freezing conditions to evaluate low-temperature behavior. Sensor data were recorded continuously under both conditions using a fixed sampling rate and duration to ensure consistency. The experimental results demonstrate that temperature variation induces characteristic and repeatable data anomalies. Under increasing temperature, the sensor output exhibited an initial change followed by stabilization, indicating thermal sensitivity and subsequent equilibrium. Under decreasing temperature, the sensor response showed a sustained drift trend accompanied by increased noise. In addition, intermittent outlier events were observed during the cooling phase, indicating transient instability under low-temperature conditions. These observations confirm that temperature fluctuations can directly induce measurable drift, noise, and outlier behavior in sensor data.

Controlled tests were conducted on the Xnode wireless sensing system operating over a Zigbee communication link, and the resulting sensor data exhibited data loss anomalies under interference conditions. In the experiments, an external wireless source was positioned in close proximity to the sensor node to induce controlled interference, and sensor data were evaluated across multiple Zigbee channels.

FIG.17 shows the frequencies of signal interference experiment on a wireless sensor node device according to an embodiment of the present invention.

FIG.18a and 18b show data results of a signal interference experiment on a wireless sensor node device according to an embodiment of the present invention. FIG.18a shows data results of channel 22 during the signal interference experiment. FIG.18b shows a zoomed-in view of the data results from index 2000 to 3000 during the signal interference experiment.

The results demonstrate that wireless interference can directly lead to intermittent data loss, manifested as missing data points in the recorded sensor streams. The severity of data loss was observed to vary across communication channels, with certain channels exhibiting significantly higher loss rates. These results confirm that wireless interference constitutes a distinct and identifiable source of sensor data anomalies, characterized primarily by missing or incomplete data records. The findings validate the ability of the proposed system-centric framework to associate data loss patterns with underlying communication faults, thereby supporting automated fault diagnosis and recovery mechanisms for reliable operation in wireless IoT sensing environments.

FIG.19a and FIG.19b are flow diagrams showing autonomous recovery of fault diagnosis in embodiments of the present invention. FIG.19a shows autonomous recovery of a single fault diagnosis and FIG.19b shows autonomous recovery of a combined fault diagnosis.

The framework operates by continuously classifying incoming sensor data using an on-device neural network and triggering targeted recovery actions based on the identified anomaly type.

As shown in FIG.19a, for a single-fault scenario of the square, the system autonomously performs fault diagnosis and recovery without external intervention. When an input signal 1902 is classified by the neural network classification 1904 as a square-type anomaly 1906, the system performs a diagnosis 1908 in which it evaluates whether the measured signal exceeds the configured sensor range. A

determination is made whether the signal exceeds the sensor range for a threshold number of samples (in this example the threshold is 3 samples). If range saturation is not confirmed other reasons are checked 1910. Upon confirmation of range saturation 1912, autonomous recovery 1914 is carried out and the sensor range is automatically adjusted 1916, and data acquisition resumes 1918. The recovered data are re-evaluated by the classification model 1920 until normal operation 1922 is restored, at which point the recovery process terminates 1924.

As shown in FIG.19b, for combined fault scenarios, the framework applies an iterative classification–diagnosis–recovery cycle. The input signal 1932 is classified by the neural network classification 1934 so as to identify the dominant anomaly, such as a trend anomaly 1936 associated with environmental effects. The system performs a diagnosis 1938 in which it checks the temperature change rate. A determination is made whether the temperature change rate exceeds a threshold. If a trend anomaly is not confirmed other reasons are checked 1940. Upon confirmation of a trend anomaly 1942, targeted data recovery 1944 is then applied to compensate for the diagnosed cause.

The corrected data 1950 are subsequently reclassified 1952, allowing secondary anomalies, such as outliers 1954 induced by low temperature, to be detected and recovered in sequence. The iterative process continues until the data 1960 are classified as normal 1962. Experimental validation using laboratory data confirms that the proposed framework produces the expected recovery outcomes when executed on edge nodes. The results demonstrate that the system can reliably correlate data anomaly types with their underlying physical sensor faults and perform autonomous recovery accordingly. This capability enables distributed edge intelligence and reduces reliance on traditional physical redundancy approaches that require additional sensors, thereby validating the effectiveness and practicality of the proposed wireless sensor node device.

FIG.20A to FIG.20D show results of iterative data recovery. FIG.20A shows original data with a moving average window. FIG.20B shows data after trend recovery, FIG.20C shows data after trend recovery with outlier bound, and FIG.20D shows data after outlier recovery. Environmental factors, such as temperature variation, can

induce combined sensor faults manifested as multiple concurrent data anomalies. To address such conditions, the proposed framework applies an iterative diagnosis and recovery process. The AI model first identifies the dominant fault type and initiates a corresponding recovery operation before addressing secondary anomalies. For trend-related faults, a moving-average operation is applied to estimate the underlying trend component in the sensor data. The trend is then removed, and the signal mean is restored to obtain a corrected dataset. Following trend correction, the model evaluates the recovered data for residual anomalies. Data points that deviate beyond a predefined statistical threshold are identified as outliers. These outliers are corrected by replacing the abnormal values with representative values derived from the recovered data distribution. The validated results demonstrate that this sequential recovery process effectively removes trend- and outlier-type anomalies while preserving the overall characteristics of the original signal. The approach enables reliable correction of combined sensor faults and supports robust data recovery under environmentally induced disturbances, thereby validating the effectiveness of the proposed system-centric fault diagnosis and recovery mechanism.

Whilst the foregoing description has described exemplary embodiments, it will be understood by those skilled in the art that many variations of the embodiments can be made within the scope and spirit of the present invention.

CLAIMS

1. A wireless sensor node device comprising:

a sensor configured to generate sensor data;

a wireless network interface configured to transmit the sensor data from the sensor node device; and

a controller comprising:

a machine learning classifier configured to identify anomalous data in the sensor data and to classify the anomalous data;

a fault diagnosis module configured to diagnose a fault source based on a classification of the anomalous data; and

a recovery module configured to generate a recovery action indication based on the fault source.

2. The wireless sensor node device according to claim 1, wherein the machine learning classifier is a lightweight deep multi-class classifier.

3. The wireless sensor node device according to claim 2, wherein the lightweight deep multi-class classifier is a student model trained using knowledge distillation.

4. The wireless sensor node device according to any preceding claim, wherein the fault diagnosis module implements a rule based diagnostic algorithm.

5. The wireless sensor node device according to any preceding claim, wherein the recovery module is configured to modify operation of the sensor based on the recovery action indication.

6. The wireless sensor node device according to any preceding claim, wherein the recovery module is configured to generate a message or alert based on the recovery action indication.

7. The wireless sensor node device according to any preceding claim, wherein the recovery module is configured to modify the sensor data based on the recovery action indication.

8. The wireless sensor node device according to any preceding claim, wherein the fault diagnosis module comprises a memory configured to store a previously diagnosed fault source and a corresponding previous recovery action indication and is further configured to determine if the anomalous data is related to the previously diagnosed fault source, and wherein the recovery module is configured to generate the recovery action indication taking into account the previous recovery action indication if the anomalous data is related to the previously diagnosed fault source.

9. The wireless sensor node device according to claim 8, wherein the memory is configured to store a plurality of previously diagnosed fault sources and a plurality of corresponding previous recovery action indications and the fault diagnosis module is further configured to iteratively determine if the anomalous data is related to a previously diagnosed fault source.

10. The wireless sensor node device according to any preceding claim, wherein the wireless network interface is further configured to receive second sensor data from a second sensor node device, the machine learning classifier is configured to identify anomalous data in the second sensor data and to classify the anomalous data, and wherein the recovery module is configured to modify operation of the wireless network interface based on the recovery action indication.

11. A method, in a wireless sensor node device, the method comprising:

generating sensor data from a sensor in the wireless sensor node device;

transmitting the sensor data from the sensor node device;

identifying anomalous data in the sensor data;

classifying the anomalous data;

diagnosing a fault source based on a classification of the anomalous data; and

generating a recovery action indication based on the fault source.

12. The method according to claim 11, wherein classifying the anomalous data comprises using a lightweight deep multi-class classifier.

13. The method according to claim 12, wherein the lightweight deep multi-class classifier is a student model trained using knowledge distillation.

14. The method according to any one of claims 11 to 13, wherein diagnosing a fault source comprises implementing a rule based diagnostic algorithm.

15. The method according to any one of claims 11 to 14, further comprising modifying operation of the sensor based on the recovery action indication.

16. The method according to any one of claims 11 to 15, further comprising generating a message or alert based on the recovery action indication.

17. The method according to any one of claims 11 to 16, further comprising modifying the sensor data based on the recovery action indication.

18. The method according to any one of claims 11 to 17, further comprising determining if the anomalous data is related to a previously diagnosed fault source, and generating the recovery action indication taking into account a previous recovery action indication if the anomalous data is related to the previously diagnosed fault source.

19. The method according to claim 18, further comprising iteratively determining if the anomalous data is related to one of a plurality of previously diagnosed fault sources.

20. The method according to any one of claims 11 to 19, further comprising receiving second sensor data, via a wireless network interface, from a second sensor node device, identifying anomalous data in the second sensor data, classifying the anomalous data, and modifying operation of the wireless network interface based on the recovery action indication.

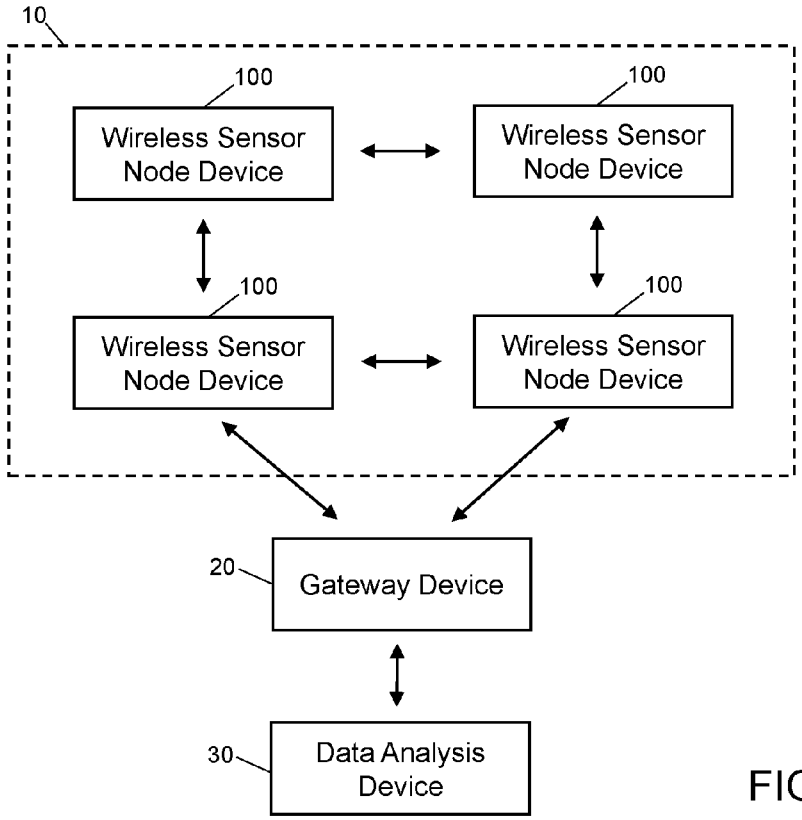

10
100
Wireless Sensor Node Device
100
Wireless Sensor Node Device
100
Wireless Sensor Node Device
100
Wireless Sensor Node Device
20
Gateway Device
30
Data Analysis Device

FIG.1

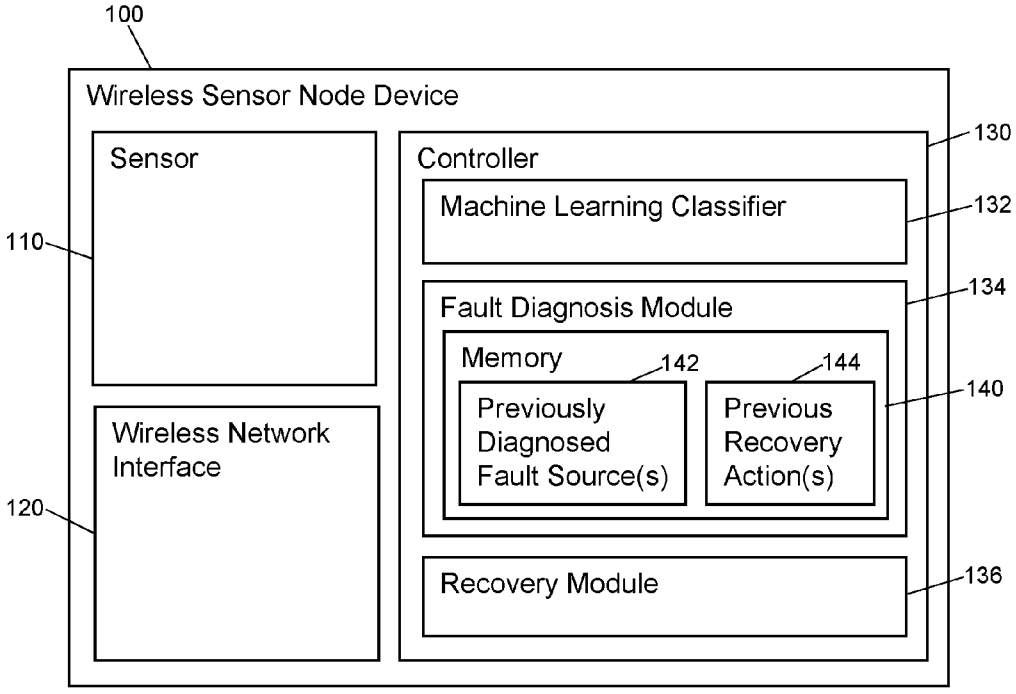

100
Wireless Sensor Node Device
Sensor
110
Wireless Network Interface
120
Controller
130
Machine Learning Classifier
132
Fault Diagnosis Module
134
Memory
142
144
Previously Diagnosed Fault Source(s)
Previous Recovery Action(s)
140
Recovery Module
136

FIG.2

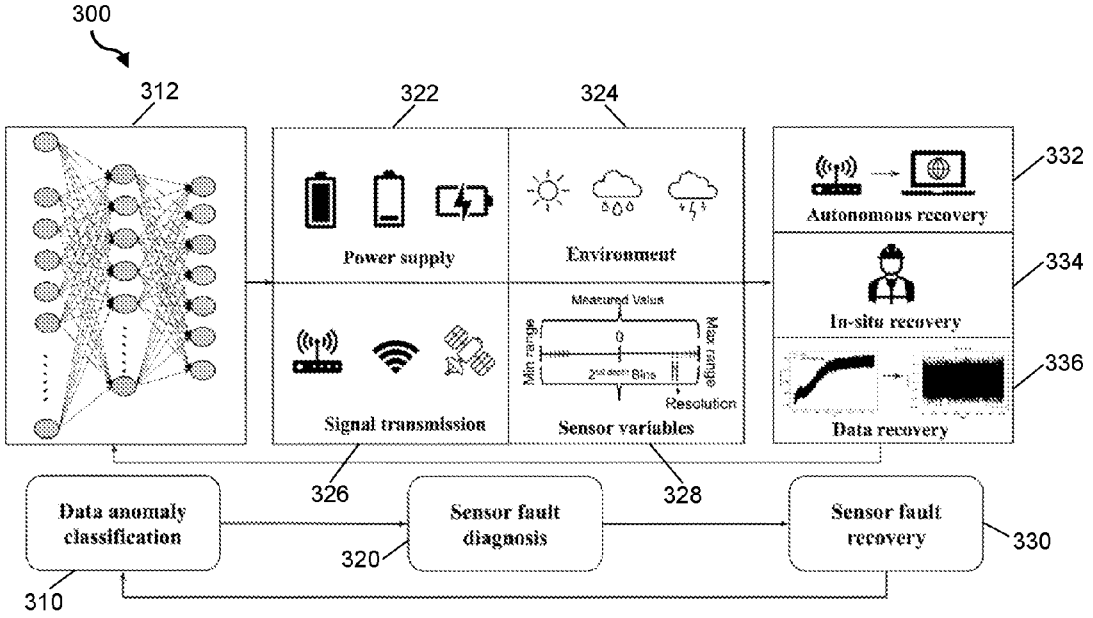

300
312
322
324
Power supply
Environment
Autonomous recovery
332
In-situ recovery
334
Measured Value
Min range
Max range
Bins
Resolution
Signal transmission
Sensor variables
Data recovery
336
326
328
Data anomaly classification
Sensor fault diagnosis
Sensor fault recovery
330
320
310

FIG.3

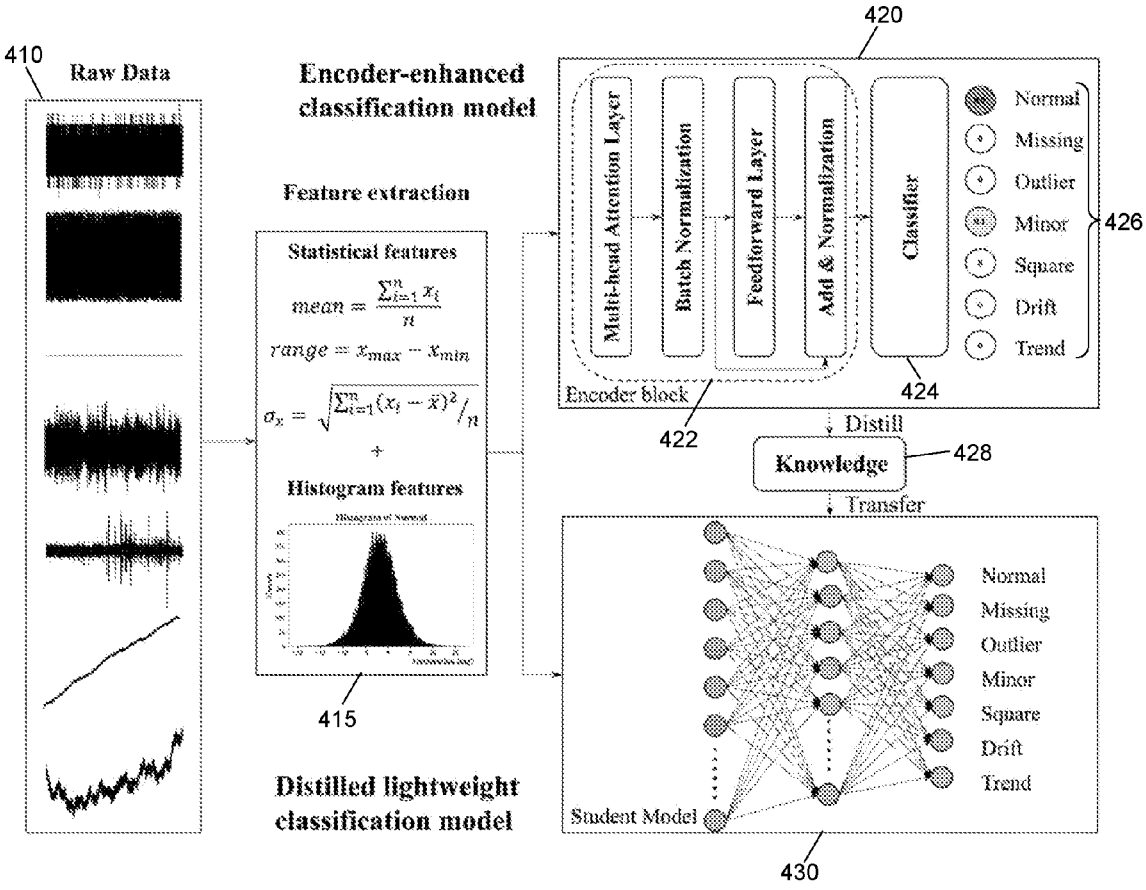

410
Raw Data
Encoder-enhanced classification model
420
Feature extraction
Statistical features
$mean = \frac{\sum_{i=1}^{n} x_i}{n}$
$range = x_{max} - x_{min}$
$\sigma_x = \sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 / n}$
+
Histogram features
Multi-head Attention Layer
Batch Normalization
Feedforward Layer
Add & Normalization
Classifier
Normal
Missing
Outlier
Minor
Square
Drift
Trend
426
Encoder block
424
422
Distill
Knowledge
428
Transfer
415
Distilled lightweight classification model
Student Model
Normal
Missing
Outlier
Minor
Square
Drift
Trend
430

FIG.4

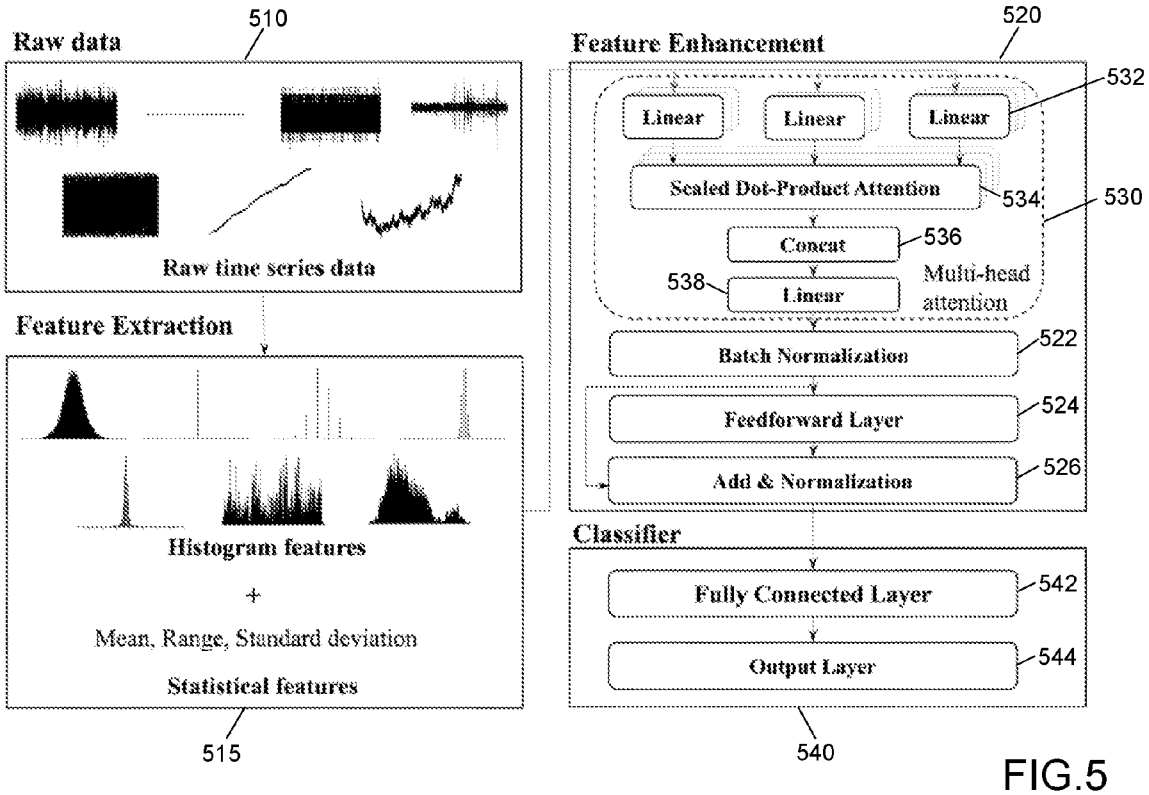

Raw data
510
Raw time series data
Feature Extraction
Histogram features
+
Mean, Range, Standard deviation
Statistical features
515
Feature Enhancement
520
Linear
Linear
Linear
532
Scaled Dot-Product Attention
534
530
Concat
536
538
Linear
Multi-head attention
Batch Normalization
522
Feedforward Layer
524
Add & Normalization
526
Classifier
Fully Connected Layer
542
Output Layer
544
540

FIG.5

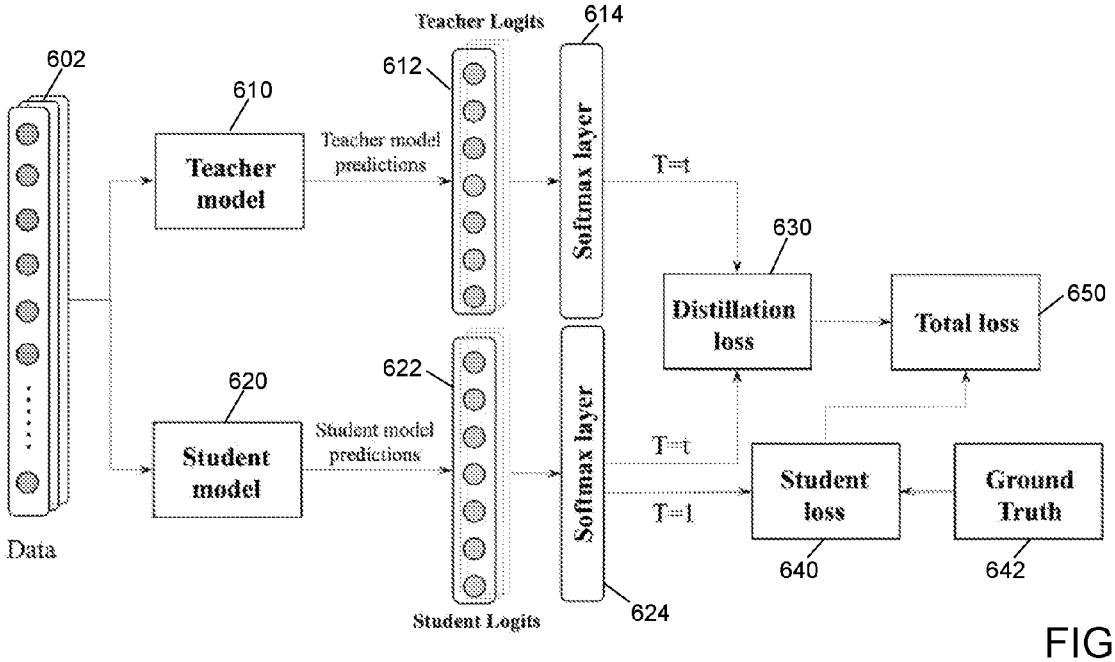

602
610
Teacher model
Teacher model predictions
Teacher Logits
612
614
Softmax layer
T=t
630
Distillation loss
Total loss
650
620
Student model
Student model predictions
622
Softmax layer
T=t
T=1
Student loss
Ground Truth
640
642
Data
Student Logits
624

FIG.6

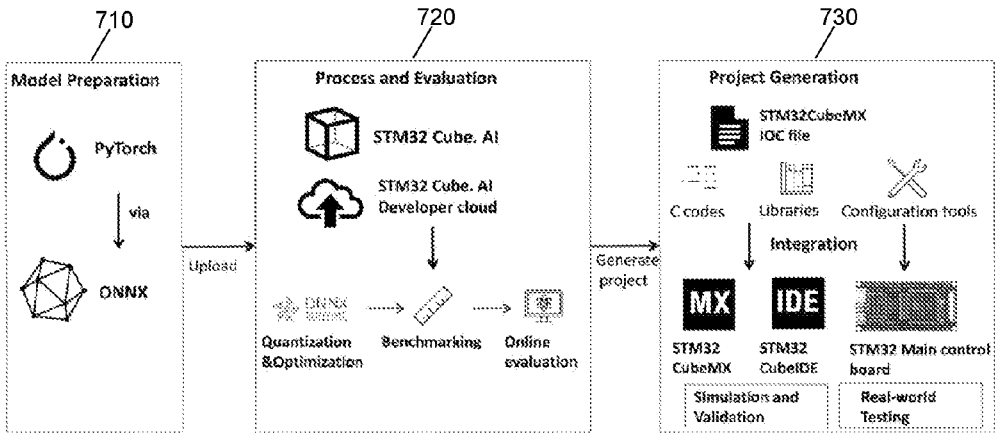


FIG.7

Data type	Sensor fault	Recovery strategy
Missing	Transmission inference	System recovery. Change the channel automatically.
Minor	Sensor resolution	System recovery. Modify the accelerometer to low range by one step automatically
Square	Sensor saturation	System recovery. Modify the accelerometer to high range by one step automatically
Outlier	Loose electrical contact	System recovery. Fix loose connections
	Cold temperature influence	Wait for temperature recovery and data recovery.
Drift	Battery voltage variation	System recovery. Charge the battery.
Trend	Temperature variation	Wait for temperature recovery and data recovery.
Bias	Battery voltage to 3.4V	System recovery. Charge the battery.
	Rotate suddenly	Wait for the location to be stable and data recovery.

FIG.8

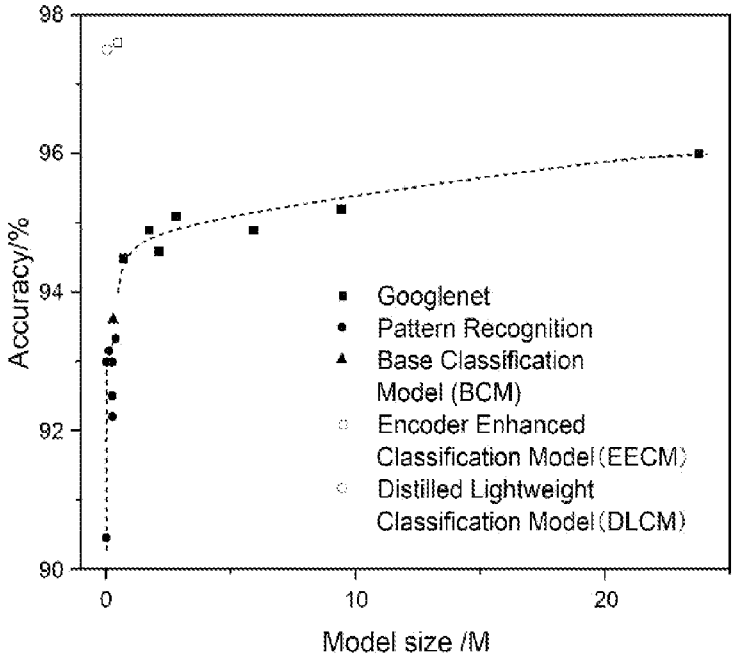

98
96
94
92
90
Accuracy/%
0
10
20
Model size /M
Googlenet
Pattern Recognition
Base Classification Model (BCM)
Encoder Enhanced Classification Model (EECM)
Distilled Lightweight Classification Model (DLCM)

FIG.9

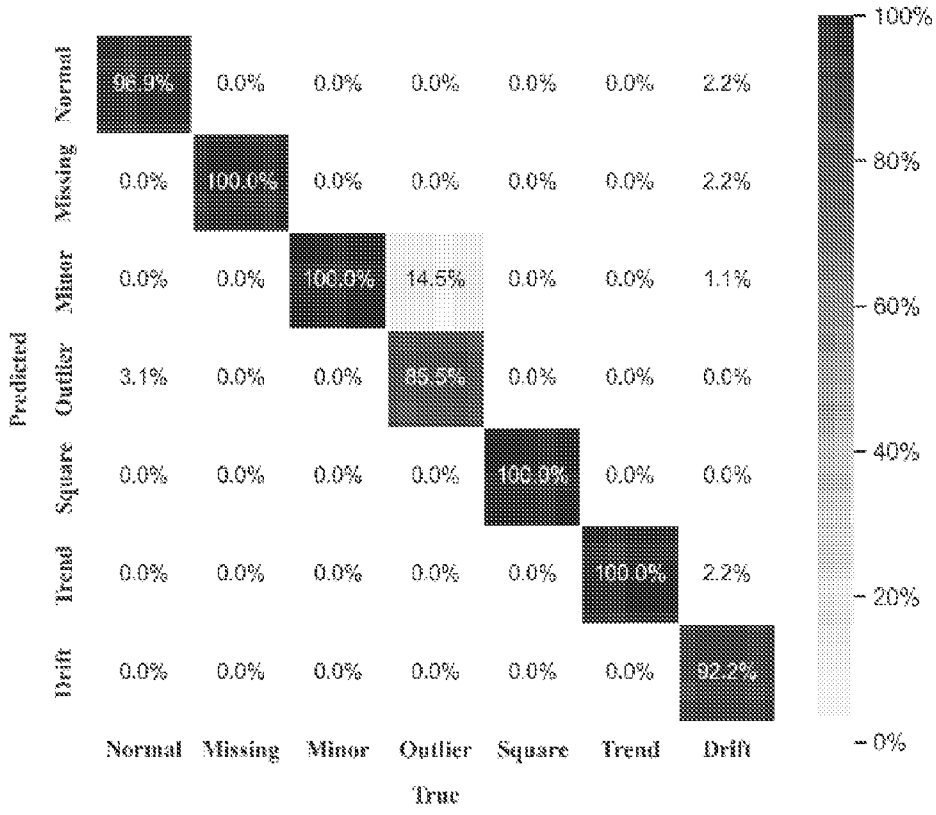

Predicted
Normal
Missing
Minor
Outlier
Square
Trend
Drift
96.9% 0.0% 0.0% 0.0% 0.0% 0.0% 2.2%
0.0% 100.0% 0.0% 0.0% 0.0% 0.0% 2.2%
0.0% 0.0% 100.0% 14.5% 0.0% 0.0% 1.1%
3.1% 0.0% 0.0% 85.5% 0.0% 0.0% 0.0%
0.0% 0.0% 0.0% 0.0% 100.0% 0.0% 0.0%
0.0% 0.0% 0.0% 0.0% 0.0% 100.0% 2.2%
0.0% 0.0% 0.0% 0.0% 0.0% 0.0% 92.2%
Normal
Missing
Minor
Outlier
Square
Trend
Drift
True
100%
80%
60%
40%
20%
0%

FIG.10

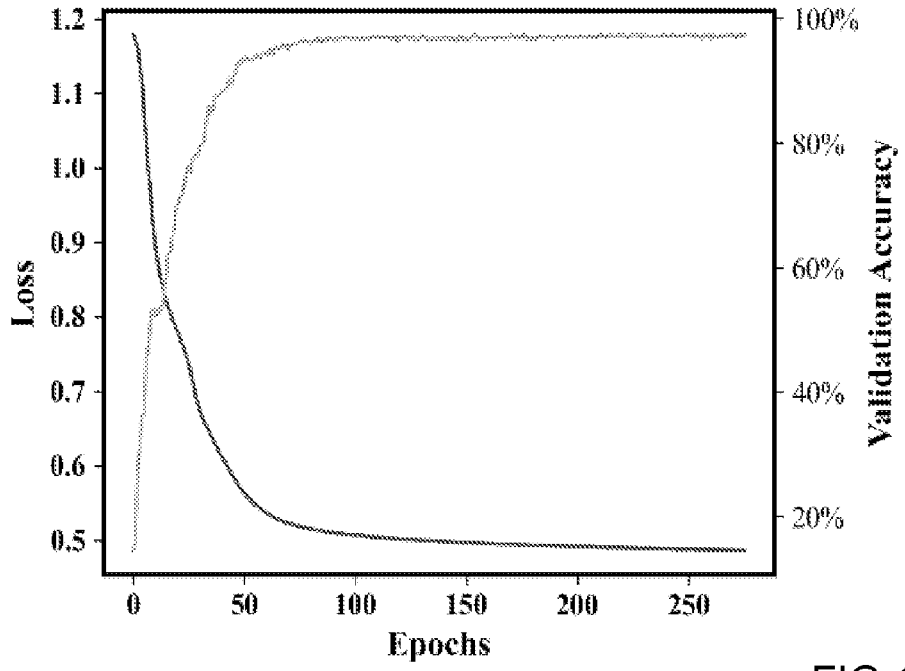

Loss
1.2
1.1
1.0
0.9
0.8
0.7
0.6
0.5
0
50
100
150
200
250
Epochs
100%
80%
60%
40%
20%
Validation Accuracy

FIG.11

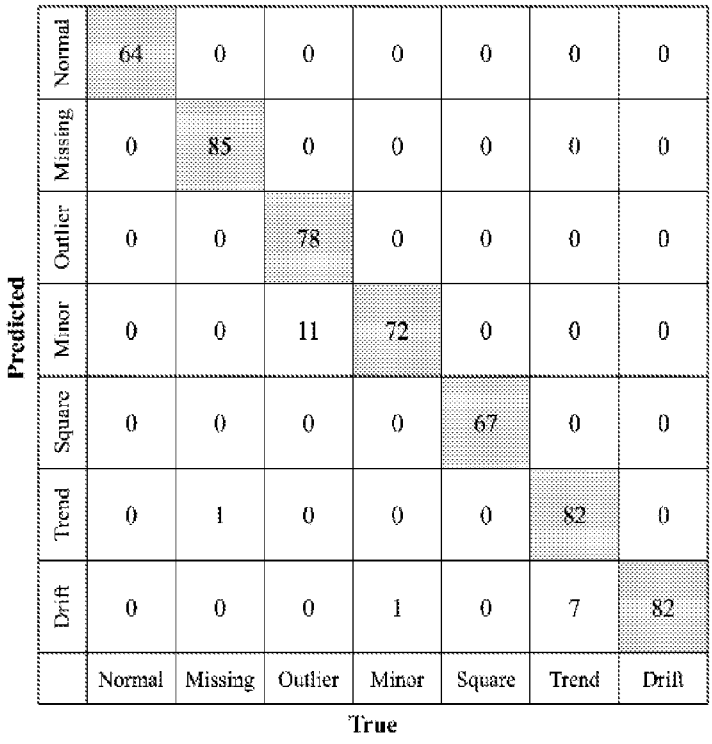

Predicted
Normal
Missing
Outlier
Minor
Square
Trend
Drift
64 0 0 0 0 0 0
0 85 0 0 0 0 0
0 0 78 0 0 0 0
0 0 11 72 0 0 0
0 0 0 0 67 0 0
0 1 0 0 0 82 0
0 0 0 1 0 7 82
Normal
Missing
Outlier
Minor
Square
Trend
Drift
True

FIG.12

Model type	Accuracy	Parameters	Inference time	Inference time/sample (ms)	Platform
Python model	96.37%	69,711	0.61s/550samples	0.011	Desktop
x86 c-model	96.36%	68,112	3.554s/550samples	0.065	Desktop
stm32 c-model	96.36%	68,112	339.35s/550samples	6.17	Stm32 board

FIG.13

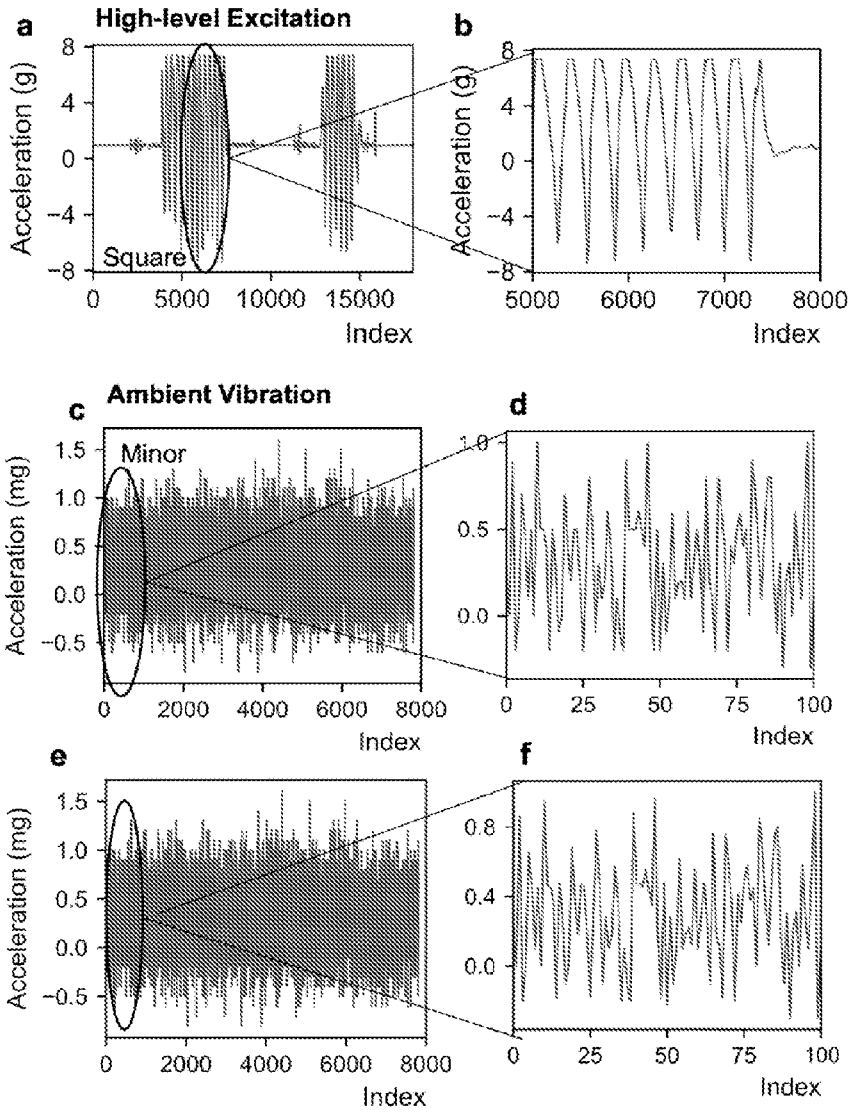


FIG.14

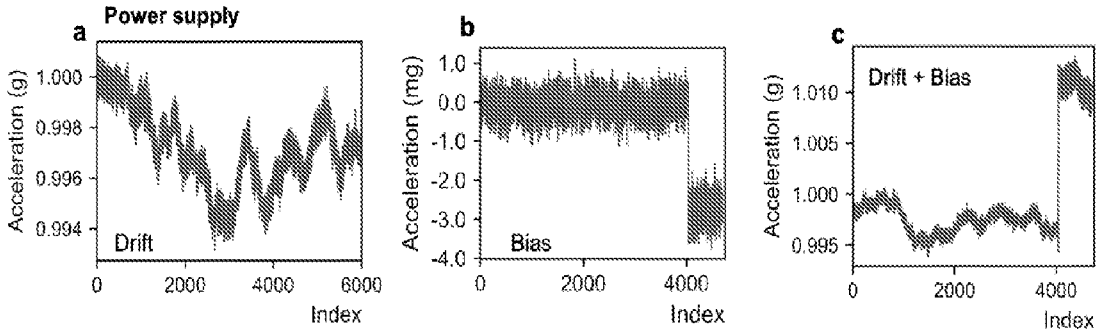


FIG.15

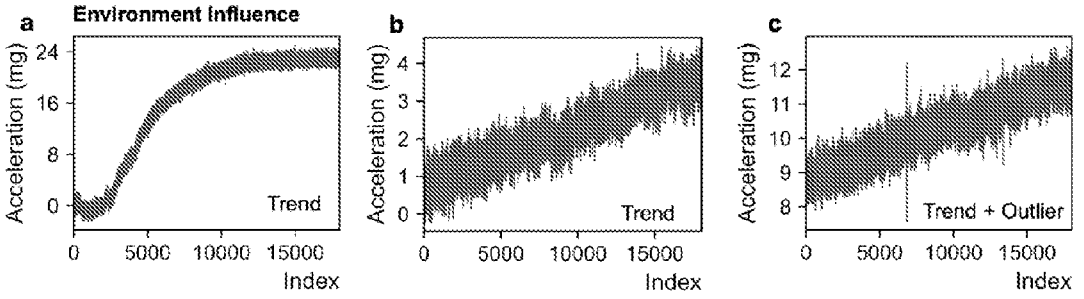


FIG.16

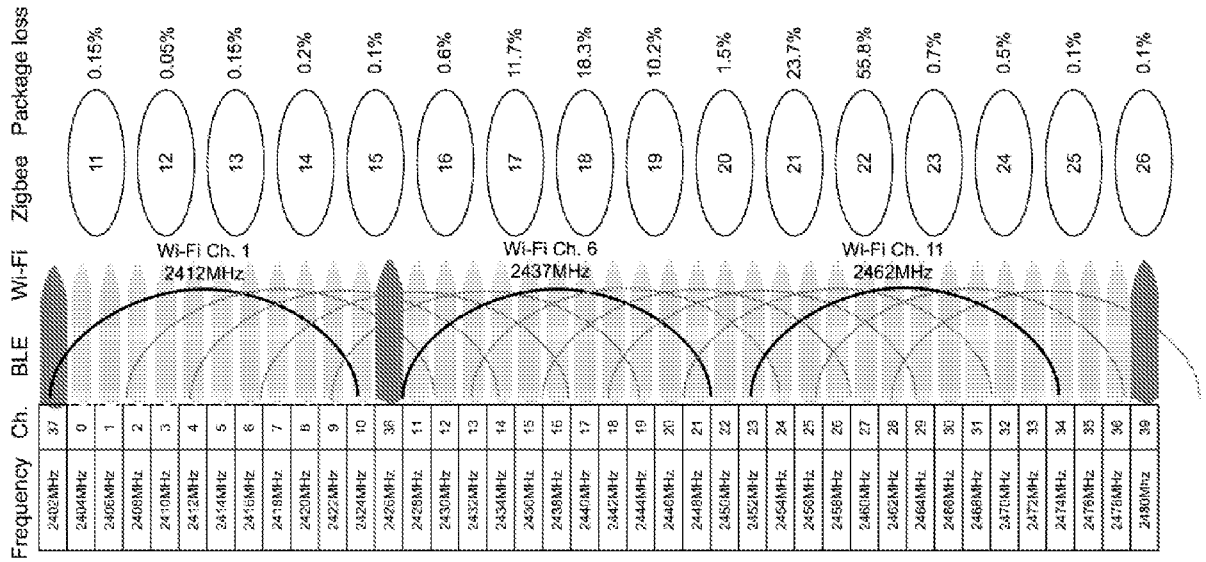


FIG.17

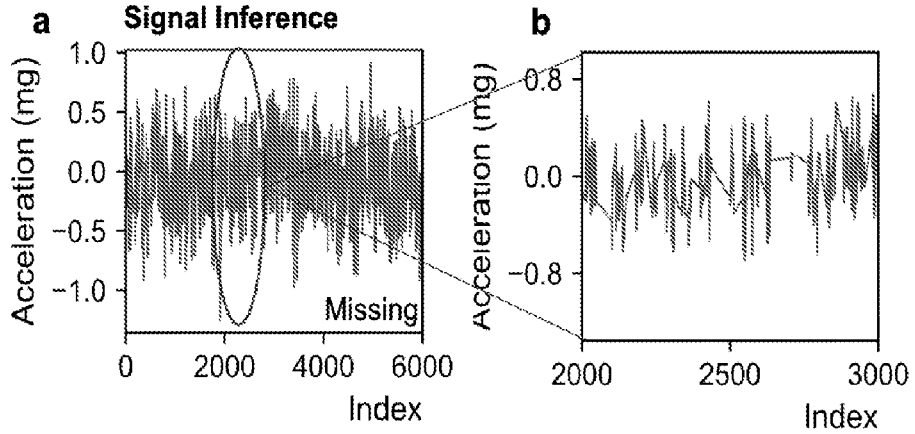

a
Signal Inference
Acceleration (mg)
1.0
0.5
0.0
−0.5
−1.0
Missing
0
2000
4000
6000
Index
b
Acceleration (mg)
0.8
0.0
−0.8
2000
2500
3000
Index

FIG.18

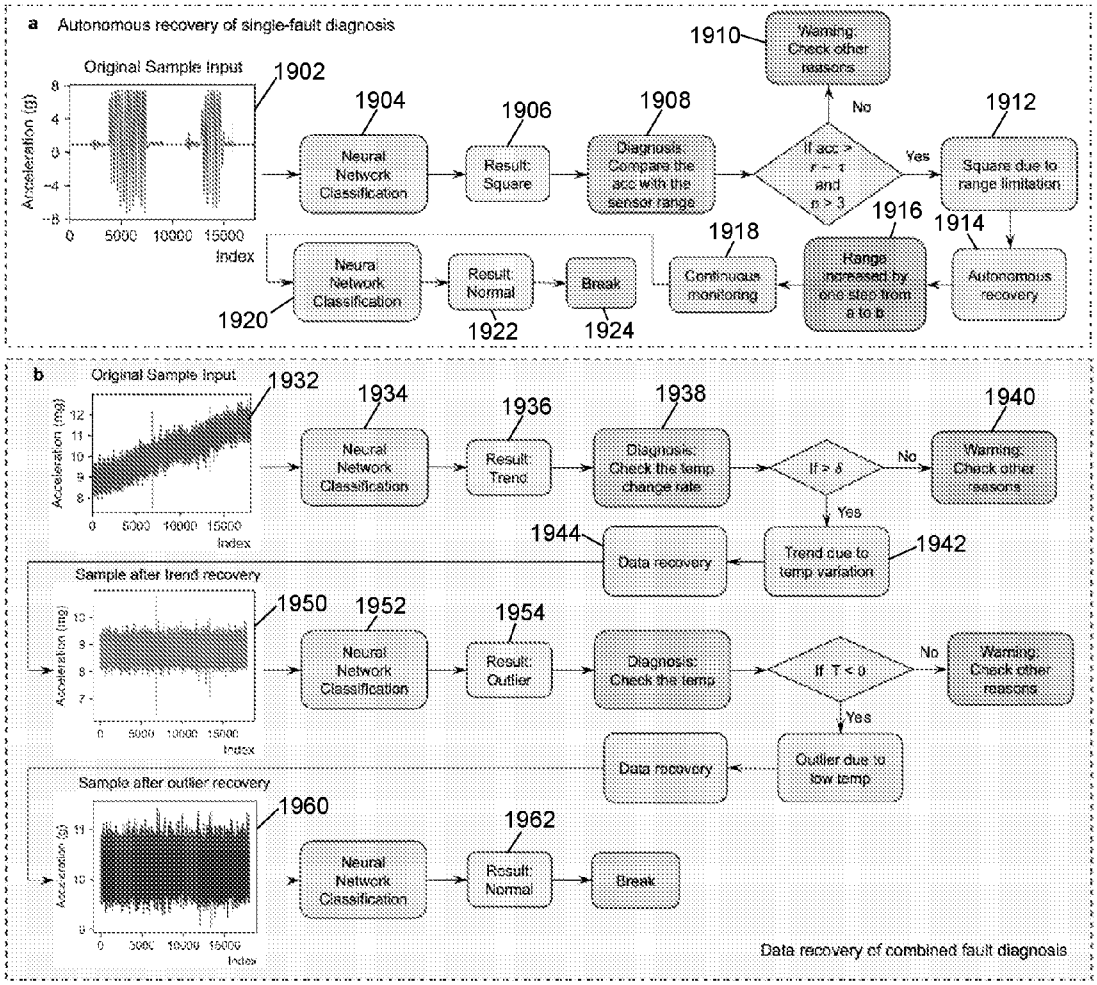

a Autonomous recovery of single-fault diagnosis
Original Sample Input
1902
1904
Neural Network Classification
1906
Result: Square
1908
Diagnosis: Compare the acc with the sensor range
1910
Warning: Check other reasons
No
Yes
1912
Square due to range limitation
1914
Autonomous recovery
1916
Range increased by one step from a to b
1918
Continuous monitoring
1920
Neural Network Classification
1922
Result: Normal
1924
Break
b
Original Sample Input
1932
1934
Neural Network Classification
1936
Result: Trend
1938
Diagnosis: Check the temp change rate
1940
Warning: Check other reasons
1942
Trend due to temp variation
1944
Data recovery
Sample after trend recovery
1950
1952
Neural Network Classification
1954
Result: Outlier
Diagnosis: Check the temp
If T < 0
Outlier due to low temp
Data recovery
Sample after outlier recovery
1960
1962
Neural Network Classification
Result: Normal
Break
Data recovery of combined fault diagnosis

FIG.19

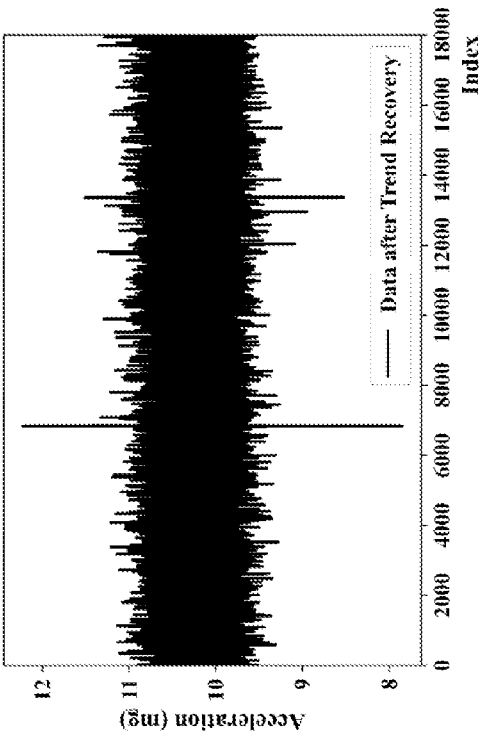


FIG.20B

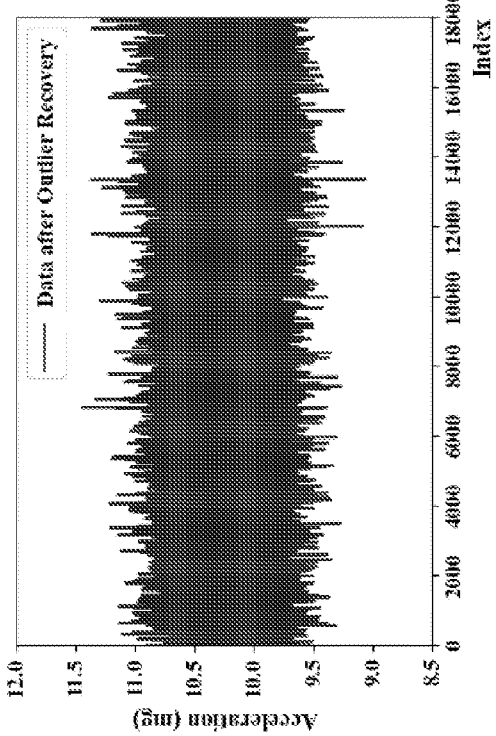


FIG.20D

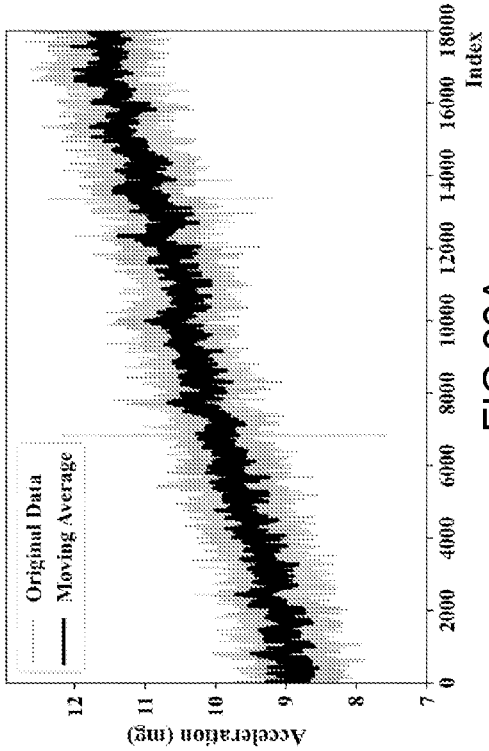


FIG.20A

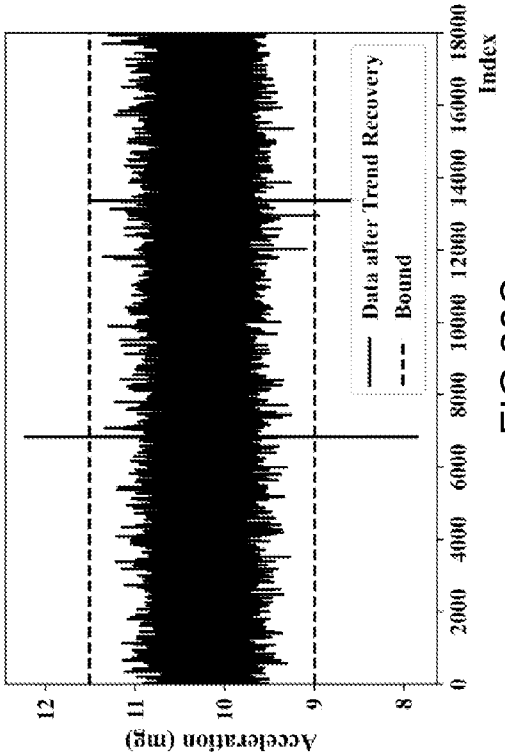


FIG.20C

INTERNATIONAL SEARCH REPORT

International application No.
PCT/SG2026/050212

A. CLASSIFICATION OF SUBJECT MATTER

H04L 67/12 (2022.01) G06N 20/00 (2019.01)

According to International Patent Classification (IPC)

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04L, G06N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic database consulted during the international search (name of database and, where practicable, search terms used)

Fampat, Internet: decentralized, local, sensor, machine learning, anomalous, classify, fault diagnosis, fault source, root cause, recovery and related terms

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 12216527 B1 (STAROSTA ET AL.) 4 February 2025 Paras. [0045], [0061], [0073], [0209], [0211], [0215], [0237]	1-20
X	US 2025/0080395 A1 (AGARWAL) 6 March 2025 Fig. 3, Paras. [0034], [0062]-[0063]	1-20
X	PENG C. ET AL., Sensor Fault Detection, Identification, and Recovery Techniques for Wireless Sensor Networks: A Full-scale Study. *The 13th International Workshop on Advanced Smart Materials and Smart Structures Technology*, 23 July 2017, pages 1-9 [Retrieved on 2026-04-22] Sections 1-2	1-20
A	CN 115630321 A (NORTH CHINA ELECTRIC POWER UNIV BAODING) 20 January 2023 Whole document particularly Para. [0021] of the original non-English language document (a machine translation is enclosed **only** for your reference)	-

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

*Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"D" document cited by the applicant in the international application

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
22/04/2026 *(day/month/year)*	24/04/2026 *(day/month/year)*

Name and mailing address of the ISA/SG

Intellectual Property Office of Singapore
1 Paya Lebar Link, #11-03
PLQ 1, Paya Lebar Quarter
Singapore 408533

Email: pct@ipos.gov.sg

Authorized officer

Mao Xiaohong (Dr)

IPOS Customer Service Tel. No.: (+65) 6339 8616

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/SG2026/050212

Note: This Annex lists known patent family members relating to the patent documents cited in this International Search Report. This Authority is in no way liable for these particulars which are merely given for the purpose of information.

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 12216527 B1	04/02/2025	NONE	
US 2025/0080395 A1	06/03/2025	NONE	
CN 115630321 A	20/01/2023	NONE	