**Smart Structures and Systems**, *Vol. 29, No. 1 (2022) 77-91*
https://doi.org/10.12989/sss.2022.29.1.077


# SHM data anomaly classification using machine learning strategies: A comparative study

Jau-Yu Chou [1a], Yuguang Fu [2b], Shieh-Kung Huang[*3] and Chia-Ming Chang [1c]

[1] *Department of Civil Engineering, National Taiwan University, No.1 Sec. 4, Roosevelt Rd., Taipei 10617, Taiwan*
[2] *School of Civil and Environmental Engineering, Nanyang Technological University, 50 Nanyang Avenue, 639798 Singapore*
[3] *Department of Civil Engineering, National Chung Hsing University, 145, Xingda Road, Taichung 40227, Taiwan*

(*Received April 12, 2021, Revised July 28, 2021, Accepted August 3, 2021*)

**Abstract.** Various monitoring systems have been implemented in civil infrastructure to ensure structural safety and integrity. In long-term monitoring, these systems generate a large amount of data, where anomalies are not unusual and can pose unique challenges for structural health monitoring applications, such as system identification and damage detection. Therefore, developing efficient techniques is quite essential to recognize the anomalies in monitoring data. In this study, several machine learning techniques are explored and implemented to detect and classify various types of data anomalies. A field dataset, which consists of one month long acceleration data obtained from a long-span cable-stayed bridge in China, is employed to examine the machine learning techniques for automated data anomaly detection. These techniques include the statistic-based pattern recognition network, spectrogram-based convolutional neural network, image-based time history convolutional neural network, image-based time-frequency hybrid convolution neural network (GoogLeNet), and proposed ensemble neural network model. The ensemble model deliberately combines different machine learning models to enhance anomaly classification performance. The results show that all these techniques can successfully detect and classify six types of data anomalies (i.e., missing, minor, outlier, square, trend, drift). Moreover, both image-based time history convolutional neural network and GoogLeNet are further investigated for the capability of autonomous online anomaly classification and found to effectively classify anomalies with decent performance. As seen in comparison with accuracy, the proposed ensemble neural network model outperforms the other three machine learning techniques. This study also evaluates the proposed ensemble neural network model to a blind test dataset. As found in the results, this ensemble model is effective for data anomaly detection and applicable for the signal characteristics changing over time.

**Keywords:** data anomaly classification; ensemble neural network; GoogLeNet; machine learning

## 1. Introduction

In long-term structural health monitoring (SHM) applications, sensors are expected to autonomously measure the responses of civil infrastructure. However, data anomalies are likely to occur frequently due to issues with damaged components, harsh environments, or calibration errors (Sharma *et al*. 2010, Mahapatro and Khilar 2013). Sensor data anomalies (e.g., sensor faults) are generally examined and classified in accordance with the features of the time history measurement data. In general, data anomalies can be categorized as missing, minor, outlier (i.e., spikes), square, trend, drift, excessive noise, non-linearity, and others. For example, the drift is one of the most common anomalies, and the mean values of measured data change with time due to temperature effects. More detailed classification and discussion can be found in the papers (Mahapatro and Khilar 2013, Dragos and Smarsly 2016).

Existing data anomalies can significantly misjudge the structural conditions via SHM applications. Researchers found that the measurement data corrupted by anomalies can distort the signal processing, e.g., power spectral density calculation, inaccurate system identification, and false damage detection (Fu *et al*. 2019). For instance, the precision of mode shape estimation was dramatically decreased using the measurement data corrupted with anomalies, especially for higher-order modes (Peng *et al*. 2017a). Therefore, developing efficient techniques to autonomously detect and classify different types of anomalies are essential to yield reliable SHM outcomes.

Extensive research has been conducted to tackle data anomalies (Bao *et al*. 2019a). Conventional physics-based strategies for data anomaly detection and classification can be classified into two categories: model-based methods and model-free methods. The model-based methods are established from physics-based models to predict nominal sensor measurements and compare these predicted measurements with raw ones. Autoregressive modeling techniques are implemented to estimate the sensor outputs; the residuals between the estimated and measured responses are then used to identify sensor faults (Chang *et al*. 2017,

*Corresponding author, Assistant Professor,
E-mail: skhuang@nchu.edu.tw
[a] Ph.D. Student
[b] Assistant Professor
[c] Associate Professor



ISSN: 1738-1584 (Print), 1738-1991 (Online)

Lo *et al*. 2016). In addition, hypothesis tests have been proposed to examine raw measurement data and determined if measurement data followed the distribution of fault-free data (Peng *et al*. 2017b, Huang *et al*. 2017a, Li *et al*. 2019). Model-free methods mostly identified the anomalies by comparing sensor measurement in proximity and assumed that anomalies had similar characteristics (Huang *et al*. 2017b). In model-based strategies, each type of sensor faults needs a specific model, in which most applications are designed to diagnose limited types of anomalies. On the other hand, model-free methods usually do not have high-precision, especially for minor or moderate anomalies. Therefore, more powerful and scalable strategies are desired for signal anomaly detection.

Alternatively, data-driven methods leveraging machine learning (ML) techniques have been exploited to directly classify sensor faults (Zhao *et al*. 2011, Smarsly and Law 2014, Yu *et al*. 2014, Drago and Smarsly 2016). Recently, Tang *et al*. (2018) split time history data and built dual-channel image containing time and frequency information. A convolutional neural network is trained for data anomaly classification; both balanced and imbalanced training sets, as well as training ratios are investigated. Bao *et al*. (2019b) developed a deep learning-based strategy that imitated biological vision and logical thinking. The measurement data were first converted by data visualization, and then further classified by a deep neural network. The proposed strategy was eventually validated by evaluating the data measured from a long-span bridge in Chinasw. Fu *et al*. (2019) accessed the data anomalies in a context of wireless monitoring systems, in which outliers, drift, and bias were the three main types of anomalies. A three-stage strategy was developed to detect anomalies using distributed similarity tests, to classify anomalies using artificial neural networks, and to eliminate the anomalies through a correction function or replacing faulty data with estimated values. Munir *et al*. (2019) developed an anomaly detection technique, namely FuseAD, leveraging both statistical and deep-learning-based methods by fusing them in a residual fashion. The proposed strategy was successfully applied in streaming data. Mao *et al*. (2020) developed generative adversarial networks combined with unsupervised methods (e.g., autoencoders) to detect sensor anomalies, and these methods were able to reduce the amount of efforts required for labeling in conventional supervised learning strategies. The measurement data were transformed to Gramian Angular Field images, and two datasets from a full-scale bridge were utilized to validate the proposed methodology. In general, the data-driven methods can achieve high precision for many types of data anomalies. Nevertheless, typical challenges of data-driven methods include the requirement of extensive datasets for training and difficulties to achieve high accuracy for all anomalies. By comparing with physical-based strategies, a more effective and autonomous anomaly detection method can be established from ML-based models. In this paper, a comparative study is carried out to detect and classify data anomalies using a variety of ML techniques (i.e., statistic-based pattern recognition network, spectrogram-based convolutional neural network, image-based time history convolutional neural network, and image-based time-frequency hybrid convolution neural network). A field dataset consisting of one-month acceleration data obtained from a long-span cable-stayed bridge in China is employed to evaluate these four strategies. Model parameters including input dimension, epochs, and learning rate are studied and tuned to achieve high performance. The results obtained from these methods are effective with high accuracy, and the associated advantages and shortcomings are discussed in details. In addition, an autonomous online data anomaly classification is developed using 10 minutes segment data. An ensemble neural network model is proposed to leverage the advantage of different ML strategies and to achieve higher accuracy for data anomaly detection and classification.

## 2. Data anomaly classification

SHM systems require high-quality sensor measurements to guarantee accurate results. However, data anomaly sometimes occurs in multiple sensors, resulting in deteriorated measurements. Therefore, detecting data anomalies is crucial in SHM applications. Moreover, identifying the types of data anomalies helps engineers better understand the condition of the sensors, and fasten the maintenance process. In this study, both shallow and deep neural networks are considered for data anomaly classification. In particular, the pattern recognition network, as one of most popular shallow neural networks, is selected to detect a certain pattern in signals. As for the deep neural network, convolutional neural network is considered, following the successful studies reported in the literature, but various types of input are explored. To sum up, four types of ML models are employed for a comprehensive study. These models are 1) pattern recognition network, 2) spectrogram-based convolutional neural network (CNN), 3) image-based time history CNN, and 4) image-based time-frequency hybrid CNN. Comparisons on the complexity of network architecture and different types of data visualization are elaborated. The field dataset is provided by the 1st International Project Competition for Structural Health Monitoring (IPC-SHM) and the dataset is briefly introduced including classification, distribution, and dataset augmentation. Then, features of each model are presented including the architecture and input construction.

### *2.1 Field dataset from IPC-SHM*

The data are acceleration responses collected from a long-span cable-stayed bridge with 38 channels, as illustrated in Fig. 1 (Tang *et al*. 2018). The data records a full month measurement in January, 2012 with a sampling rate of 20 Hz.

The types of data anomalies each hour are provided by IPC-SHM (Bao *et al*. 2021). Fig. 2 presents the 7 data anomalies in terms of time history responses. These data anomalies are labeled as *normal*, *missing*, *minor*, *outlier*, *square*, *trend*, and *drift*. The distribution of labels (based on the 1-hour signal) can be found in Fig. 3. As shown in the

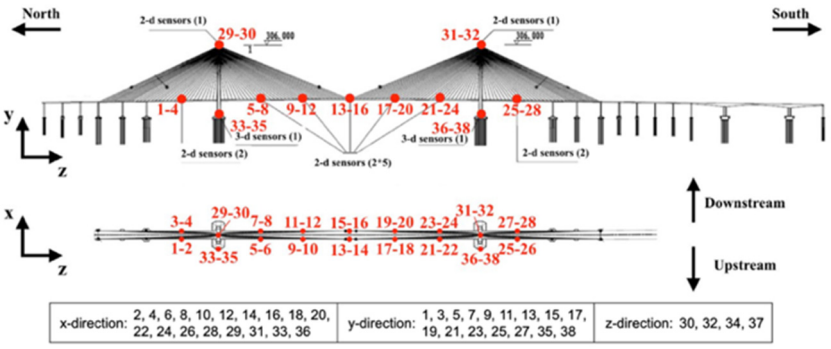
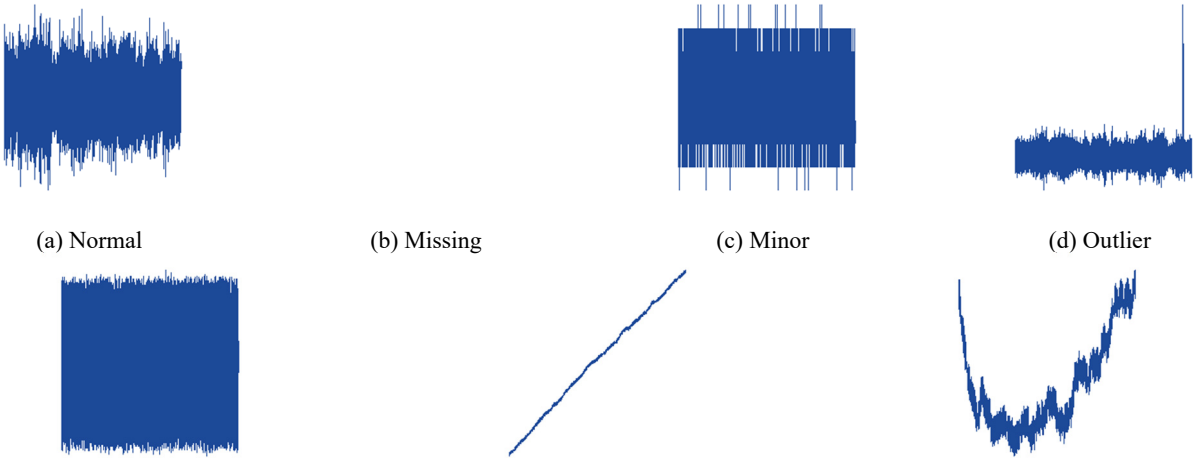


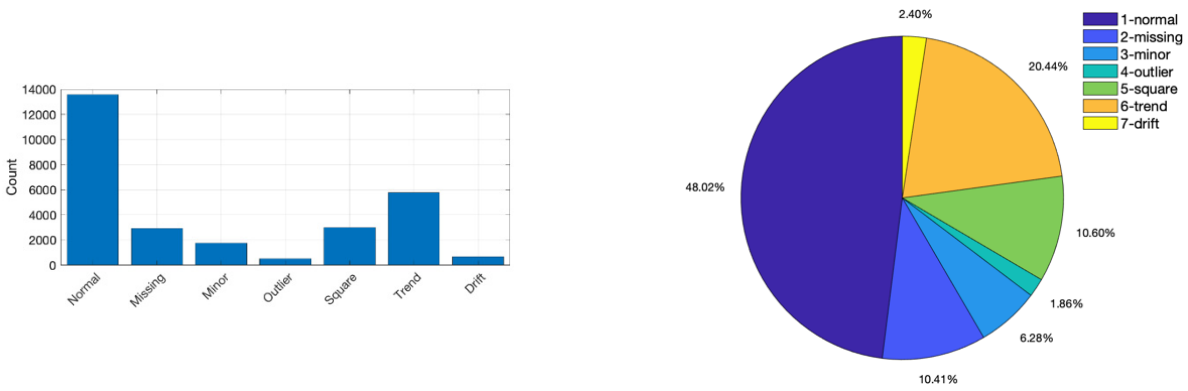

Fig. 1 Illustration of the long-span cable-stayed bridge and sensor locations



(a) Normal (b) Missing (c) Minor (d) Outlier

(e) Square (f) Trend (g) Drift

Fig. 2 Types of data anomalies




(a) Label counts (b) Percentage pie chart

Fig. 3 Types of data anomalies

figure, the number of the data anomalies is highly unbalanced, with the highest one up to around 12,000 samples (normal) and the lowest one down to 523 samples (outlier). To adjust the imbalance of the dataset (i.e., the normal signals are outweighed the outlier signals), the number of training datasets is fixed and equal to that of the outlier signals to prevent unpleasant training features. During training, some unclear labeling signals are manually removed to guarantee all features fully understood by the ML models. Thus, the size of the modified datasets is smaller than the provided one. In addition, depending on the selected input types, the dataset can be augmented twice by adding flipped responses over time to establish a high-quality training model. To produce a correctly trained model, the samples only with obvious features are utilized.

### 2.2 Pattern recognition network

#### 2.2.1 Network architecture

The first ML model used to classify the 7 data anomalies is pattern recognition network. This model is a multi-layer feedforward (MLF) network and can be trained to distinguish inputs in accordance with target labels. In this study, 64 derived parameters are exploited as an input including the histograms, arithmetic means, ranges, and standard deviations with respect to label. To be efficient, the network is comprised sequentially by an input layer, two hidden layers, and an output layer, as shown in Fig. 4. These four layers are simply assigned as the fully connected layers. The hyperbolic tangent is selected as the activation function for all neurons. The mean squared error (MSE) is chosen to measure the trained model performance in this study although the cross-entropy loss function is preferable in a multi-class classification problem in literature.

In the beginning, the weights and bias values of neurons are initialized using the Nguyen-Widrow algorithm (Pavelka and Proch 2004). Then, the MLF network updates the weight and bias values in accordance with the backpropagation learning algorithm. The Levenberg-Marquardt algorithm (LMA) (i.e., the damped least-squares (DLS) method) is employed for learning and optimizing this ML model. Training is terminated when the minimum gradient magnitude or maximum adaptive value of the learning algorithm is reached. Meanwhile, the training process should stop to avoid overfitting when the validation loss increases as the training loss continuously decreases. That is, if the accuracy over the training subset increases but the accuracy over the validation subset remains the same or decreases, the models are early terminated to hold on to the generalization capability. This method yields an inevitable trade-off similarly to the problem of curve fitting, where a higher number of degrees of freedom in the approximating function results in greater flexibility and better fitting performance for the input data. However, considering that the data may be affected by noise, a general model is preferred even with little inaccuracy.

#### 2.2.2 Input construction

The dataset contains 744 hours for 38 accelerometers recorded in 31 days. An hour of signal with sampling rate 20 Hz generates 72,000 points. The enormous data is not suitable for fast training and testing. Thus, extracting the specific features is better than direct use of the raw data in the model construction. To classify the 7 types of data anomalies with high efficiency, the statistic features are used to represent the signals, such as the histogram, arithmetic mean, range, and standard deviation, as seen in Fig. 5.

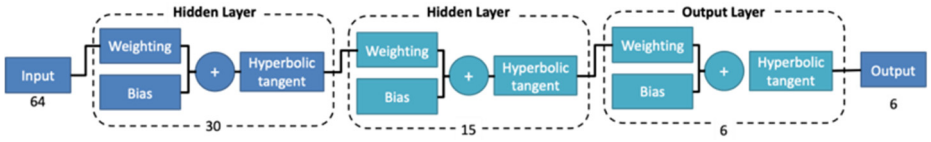
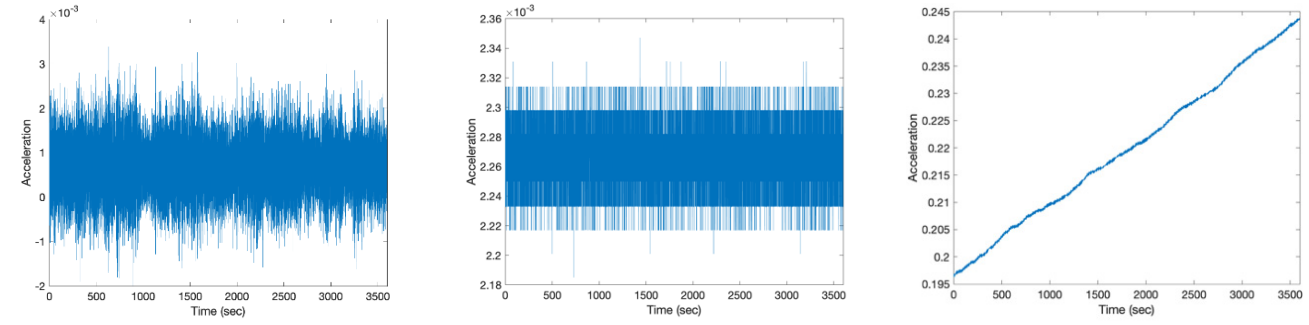
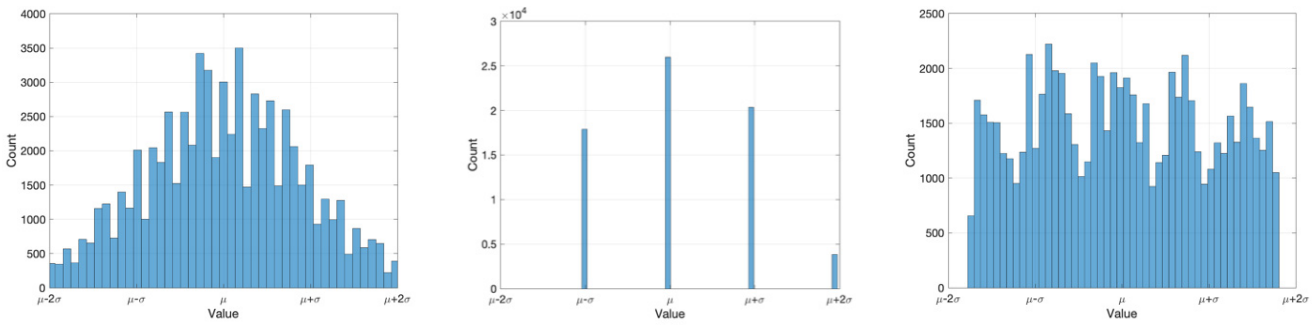



Fig. 4 Network architecture of pattern recognition network




(a-1) time history: normal (b-1) time history: minor (c-1) time history: trend




(a-2) histogram: normal (b-2) histogram: minor (c-2) histogram: trend

Fig. 5 Original signals and statistic features of different labels: time histories and histograms

The derived features are first introduced:

- The arithmetic means are the central value of a signal, which is also namely the expectation or average.
- The range is the difference between the largest and smallest value of a signal and can offer a rough idea of the signal distribution. For instance, the trend always has a much larger value of range than the range of minor data anomaly.
- The standard deviation measures the variation of a signal and shows the dispersion of a signal. A low standard deviation indicates that the values tend to be close to the mean (i.e., the minor and the outlier), while a high standard deviation indicates that the values are spread out over a wider range (i.e., the trend and the drift).

In addition to the three simple statistic features, histograms are also adopted in this study to distinguish different data anomalies. The histograms provide the occurrence of different values for a signal and interpret a random phenomenon in terms of signal sample space and event probability (i.e., the subsets of the sample space). In this study, the count of each bin in the histograms falling within two standard deviations is applied to distinguishing anomalies. In total, 61 sections are assigned to represent the signals, each section has a range as

$$r_s = {}^{\sigma_x}/_{30} \tag{1}$$

where $r_s$ is the range of a section and $\sigma_x$ is the standard deviation of the signal, $x$. For example, the signals label as the normal should have a normal distribution shown in Fig. 5(a-2), the signals label as the minor or the outlier should have a distribution with a lot of zero sections shown in Fig. 5(b-2), and the signals label as the trend should have a uniform distribution shown in Fig. 5(c-2). In accordance to the above statistic features, the labels can be easily classified, and are exploited as the inputs to establish the ML model.

Nevertheless, the missing can be identified just before the pattern recognition network because of the unique characteristics. In the database, two kinds of signals are labeled as missing. The first kind is the data with NAN (not a number), and these data cannot generate any statistic features but can be easily identified while preparing the input data. The other type of missing is that when the measurements read as a constant. These data have monotonous statistic features and, most importantly, the range of the data is always zero. Thus, these data can be also easily classified as missing while preparing the input data. A pre-screening step is facilitated to detect the missing data anomaly before implementing the pattern recognition network. The types of data anomalies classified by the model is reduced from 7 to 6.

The last step to derive the input is normalization. After processing the dataset with 744 hours for 38 accelerometers, the maximum absolute value in the dataset is used to normalize the statistic features. For example, the maximum values of the arithmetic mean, range, and standard deviation are 1.94, 1.36, and 5.42, respectively. Thus, these statistic values are individually normalized by 2, 2, and 6. For the histogram, each sample is normalized by the data length (72,000 points). Finally, the derived input is ranged between 0 and 1.

### 2.3 Spectrogram-based CNN (CNN-1)

#### 2.3.1 Network architecture

As introduced before, the spectrogram-based anomaly features are a visual representation of the energy level per signal. Thus, a convolutional neural network (CNN) is selected as the ML technique to process the image-like information and denoted as CNN-1. Fig. 6 shows the architecture of CNN used in this ML approach. The input layer is to process the spectrogram with a size of $103 \times 2400$. A total number of 7 feature learning layers are added to extract useful information from the input layer. Each feature learning layer has 16 filters at the size of $3 \times 3$, followed by an activation layer using RELU and a max pooling layer. Afterward, a fully connected layer is added that combines all the features learned by the previous layers to classify the anomalies. The number of nodes in the fully connected layer is 7, which is the same as the number of data types. Then, a softmax activation function is employed to normalize the output of the fully connected layer. Finally, a classification layer is adopted using the probabilities returned by the softmax activation function that eventually assigns the input to one of the mutually exclusive classes. The spectrogram-based CNN is implemented via the deep learning toolbox in MATLAB (MATLAB 2020b).

#### 2.3.2 Input construction

Each sensor reading spectrogram is considered as an alternative meaningful feature for anomaly detection and classification. Specifically, a spectrogram of measurement data is a visual way of representing its signal strength, i.e., energy distribution over the time-frequency domain. As can

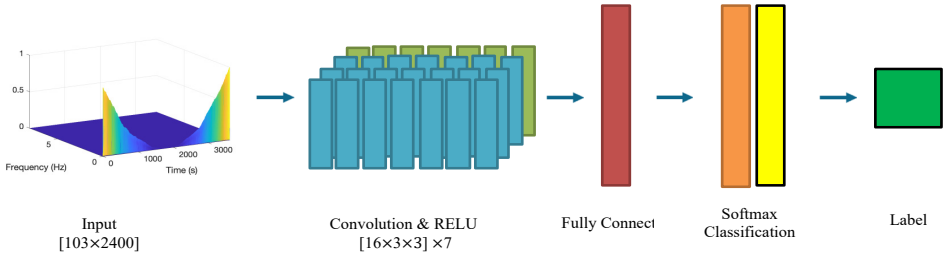
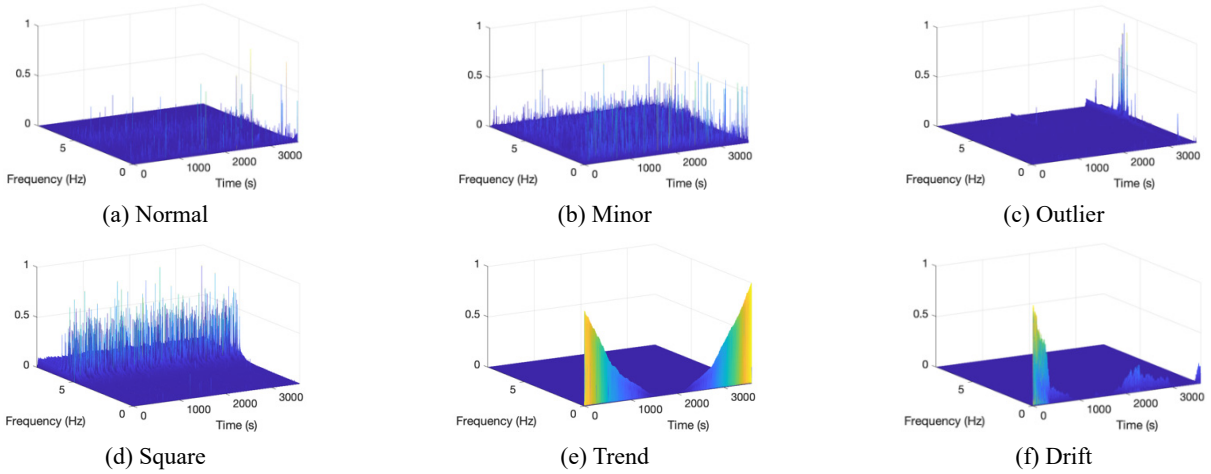



Fig. 6 Network architecture of spectrogram-based CNN model




(a) Normal (b) Minor (c) Outlier

(d) Square (e) Trend (f) Drift

Fig. 7 Spectrogram representation of different anomalies (Note that the missing anomaly are presented by a plain surface with all zeros)

be seen in Fig. 7, different data types have significant unique features in the spectrogram representation. The spectrograms are generated with 0.5 sec and 0.008 Hz in terms of time and frequency resolutions. The magnitudes of these spectrograms are converted into images. These images are then resized by a factor of 0.1 along both time and frequency axes. Eventually, the frequency domain resolution is still sufficient, while the input size is reduced.

In these spectrograms, each dataset is first pre-processed by removing offsets and linear trends. Then, the spectrograms are generated and normalized by the maximum value. The features of various data anomalies are described as below:

- The normal (Label 1) have relatively uniform distribution of energy.
- The miss (Label 2) cannot be processed to generate spectrograms, and these spectrograms are replaced by a "zero" plot (e.g., a plain surface with zeros).
- The minor (Label 3) has a small energy level with scattered peaks. Even though the minor dataset spectrograms are normalized, the magnitudes of the spectrograms are always below 0.5.
- The outlier (Label 4) has large peaks surrounding the spikes if using the projection of these peaks onto the time axis.
- The square (Label 5) has relatively small and high energy in low and high frequencies. Moreover, their distribution of energy is consistent over time.
- In both trend (Label 6) and drift (Label 7), the energy is concentrated at zero frequency, e.g., the DC component in a signal. The difference between these two anomalies is the energy distribution over the time axis. After linear trend removal, the trend datasets have zero spectrograms in the mid time span and a maximum value at both ends. The drift datasets have relatively random energy distribution at zero frequency.

The unique features of the spectrogram can be employed as an effective indicator to enable ML techniques in the detection and classification of data anomalies.

### 2.4 Image-based time history CNN (CNN-2)

#### 2.4.1 Network architecture

Because data anomalies are easy to classify manually but are tedious and time inefficient, a straightforward approach is to directly convert the time histories into an image and to exploit the converted images in training of a CNN model. The model used in this approach refers to the object recognition tool shared in the ImageNet Large Scale Visual Recognition Competition (ILSVRC) in 2012 (Krizhevesky *et al*. 2012, Russakovsky *et al*. 2015). In this section, a simple CNN model is developed and denoted as CNN-2. As compared with the CNN-1 model, this model uses images that are converted directly from time histories pixels as input, instead of taking using the spectrum magnitude. In this model, an image is first resized from 256 × 256 to 28 × 28 pixels to expedite the training process. Note that compressing image resolution would result in a certain extent of feature loss, which is a tradeoff between the performance and computational loading. Then, a series of convolution layers combined with batch normalization and max pooling layers are sequentially added to the image. In total, three series of the combined layers are employed with the dimensions shown in Fig. 8. RELU function is selected as the activation function for all layers. After extracting features from the previous layers, a fully connected layer, a softmax layer, and a classification layer are added at the end of the model to rank the scores of 7 labels. During the training process, the batch size is fixed to 128 images per epoch.

#### 2.4.2 Input construction

To directly transform the signals into images, each dataset is first pre-processed by removing offsets and linear trends. Next, 7 types of data anomalies and normal data are first normalized by

$$\boldsymbol{y}_{normalize} = \frac{\boldsymbol{y} - \boldsymbol{y}_{min}}{\boldsymbol{y}_{max} - \boldsymbol{y}_{min}} \tag{2}$$

where $\boldsymbol{y}_{normalize}$ is the normalized signal; $\boldsymbol{y}$ is the input response for a specific data anomaly type; $\boldsymbol{y}_{max}$, $\boldsymbol{y}_{min}$

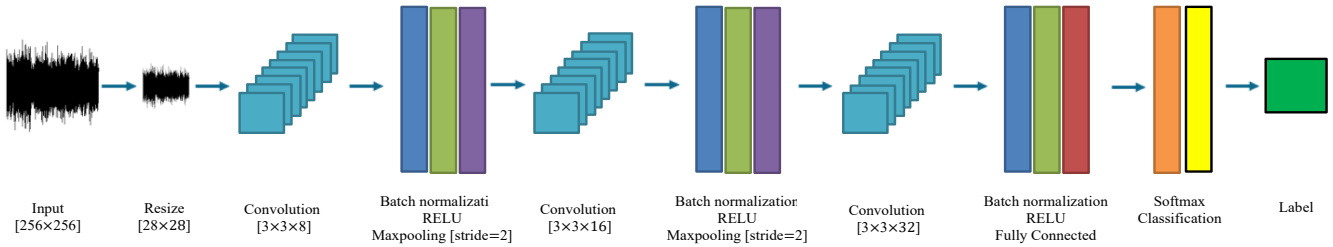



Fig. 8 Network architecture of image-based time history CNN model

are the maximum and minimum value in $\boldsymbol{y}$. The normalization eliminates the magnitude effect and can highlight only the features in the signal pattern. After normalization, $\boldsymbol{y}_{normalize}$ is drawn on an image plane corresponding to the time axis and acceleration axis. The axes are changed to invisible afterward because the axis should not be considered as training features. The completed image is then resized to an acceptable resolution which the features can be clearly classified. Note that the resolution has high impact on the model accuracy because some details would be discarded as the resolution decreases. However, the image resolution cannot be too high due to computational limits. In this study, images with resolution of $256 \times 256$ pixels are generated for the model input.

### 2.5 Image-based time-frequency hybrid CNN (GoogLeNet)

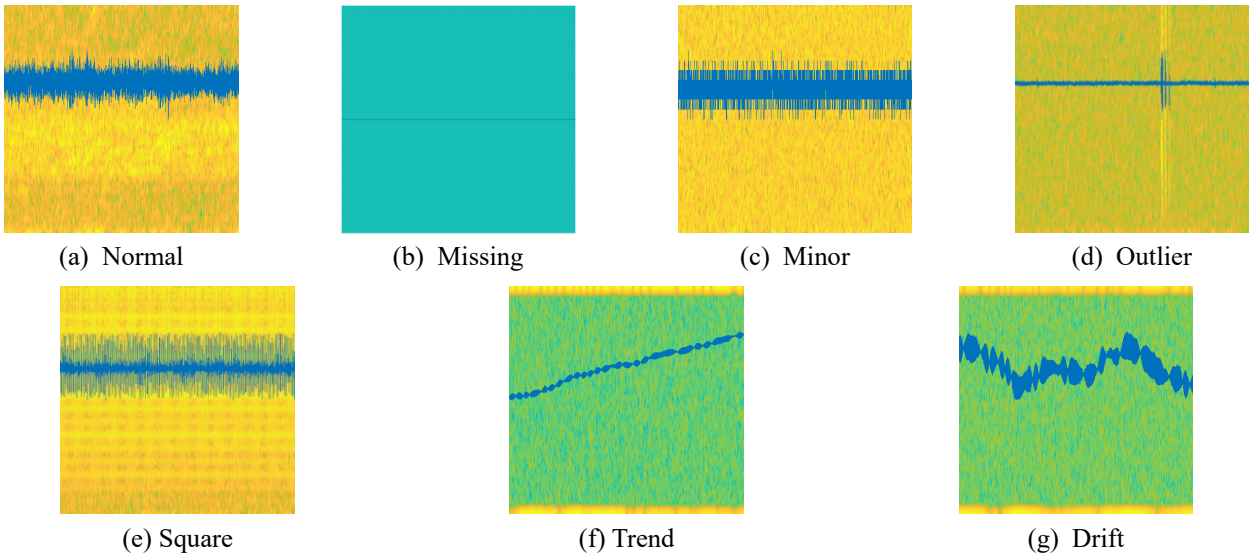
To further improve the detection accuracy, a more complicated, or deeper model architecture with hybrid input information is suggested (Baoshash and Ouelha 2016). Numerous studies exhibited the effectiveness of using hybrid inputs for model training, especially time-frequency responses (Krizhevesky *et al*. 2012, Christian *et al*. 2015). For a more complicated image, GoogLeNet is capable of extracting more detailed features because of the dense and complicated layers within the model. GoogLeNet was employed to classify 1000 types of objects and was rewarded as the 1st place in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2014 (Christian *et al*. 2015). Significant improvement in performance was established compared to ZFnet and AlexNet. Moreover, it has a lower accuracy error compared to the VGGNet. Consequently, both time history and time-frequency responses are combined and exploited to the GoogLeNet as the model inputs.

#### 2.5.1 Network architecture

In this study, GoogLeNet is utilized for data anomaly classification based on signal-based data visualization and combination space. First, the image is resized to $224 \times 224$ pixels to reduce computational loading. The architecture of the GoogLeNet consists of 22 layers, including convolution layers, max pooling layers, inception layers, average pooling layers, linear layer, softmax layer, and fully connected layer. More details of this architecture can be found in (Christian *et al*. 2015). In this study, the last classification layer and fully connected layer are replaced with layer dimensions of $1 \times 7$ to identify 7 types of data anomalies through the transfer learning (Pan and Yang 2010). The batch size is set to 10, with an initial learning rate set to 0.001.

#### 2.5.2 Input construction

To extract more features other than just considering the time histories such as in CNN-2 or only considering the spectrogram such as in CNN-1, the signals after removing offsets and linear trends are normalized by Eq. (2). The



(a) Normal (b) Missing (c) Minor (d) Outlier

(e) Square (f) Trend (g) Drift

Fig. 9 Input image using the hybrid CNN

time-frequency responses are generated by the same procedure presented in Section 2.3.2. The response is then plotted on a 2D surface as a heatmap, which the frequency magnitudes are represented in color gradients. To strengthen the features, the normalized signal is also overlaid on the time-frequency image. The two responses are combined into one image using a shared time-axis. A dual y-axis is assigned where one is the frequency and the other is the acceleration magnitude. Because the acceleration magnitude is not crucial to the classification, the response is shift to a value which will not hide important frequency content (i.e., low frequency portion). Sequentially, the constructed image is then resized to resolution of $512 \times 512$ pixels. Note that decreasing the resolution may have disadvantages to the trained model because the heatmap will be blurred due to lack of pixels. Moreover, when more complicated spectrum features are included in the background, a higher resolution is required to fully capture the details within an image. Fig. 9 represents the input images containing 7 different data anomalies. As shown in the figure, the features can be both observed in the time-domain history and the time-frequency response. Finally, the dataset is exploited to the GoogLeNet for model training.

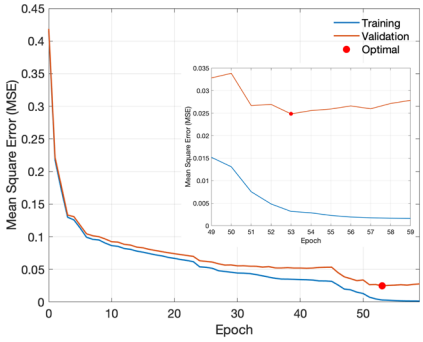
## 3. Comparative study

In this study, the accuracy and learning process of 4 ML models are separately compared including 1) statistic-based pattern recognition network, 2) spectrogram-based convolutional neural network (CNN-1), 3) image-based time history convolutional neural network (CNN-2), and 4) image-based time-frequency hybrid convolution neural network (GoogLeNet). For the statistic-based pattern recognition network, 523 samples per label are utilized for training because of the unbalanced dataset. For CNN-based models, 800 samples per label, of which is augmented twice by flipping responses, is applied for training. This augmented dataset is employed to improve model performance in training process. A pre-processed dataset, which includes all the samples except for ambiguously labeled samples, is utilized for performance comparison. Although insufficient outlier samples enforce some training samples to be included in the test dataset, the test dataset only contains approximately 4% and 10% of training samples for the pattern recognition network and CNN-based models, respectively. Thus, the data anomaly detection performance may be then slightly decreased, in particular of outliers. In addition, hyperparameters (i.e., batch size, initial learning rate, and epoch, etc.) are discussed and tuned for the best performance. Additionally, different input image resolutions are investigated for image-based CNN models to accurately classify data anomalies. The best model is selected for 10-minute online data anomaly classification.

### 3.1 Training process

For the proposed pattern recognition network, the number of input data for each label is 523, which is the lowest number (outlier) across all the labels. This number can balance between different labels in training and validation. The 523 input samples are divided into subsets, i.e., 70% of data are the training subset, while the rest 30% are validation subset. Fig. 10(a) shows the best result of MSE after training. Obviously, both training and validation MSEs simultaneously decrease; however, the validation MSE begins to slightly increase after 53 epochs, meaning that the model tends to overfit the training subset. After 6 sequential increases of the validation MSE, the optimization is terminated. The trained model at 53 epochs is selected as the best result. Fig. 10(b) demonstrates the confusion matrix of the pattern recognition network. As seen in this figure, the accuracy is overall acceptable. The lowest accuracy is around 75% for the outlier classification. The low accuracy partially results from the simplicity of the model architecture. Because the pattern recognition network is established from the shallow neural network architecture, complex nonlinear relationships (e.g., outliers) are much more difficult to be extracted as compared to deep neural network models. Moreover, the minor and the outlier are sometimes miscategorized because these two anomalies share a very similar probability distribution. Both arithmetic mean and standard deviation also provide a limited improvement for distinguishing these two anomalies. Theoretically, the range may help to tell the outlier from the minor but the improvement seems to be insignificant. Therefore, the proposed pattern recognition network is suggested to only detect the normal, square, trend, and drift data anomalies.

For the spectrogram-based CNN model (CNN-1), a total

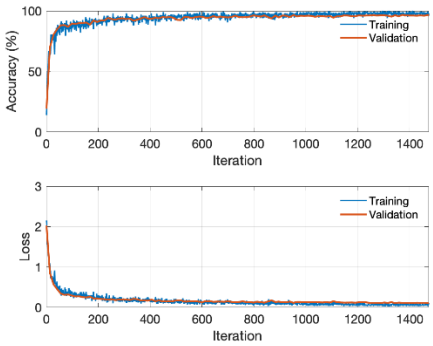



(a) Learning curve

| | Normal | Missing | Minor | Outlier | Square | Trend | Drift |
|---|---|---|---|---|---|---|---|
| Normal | 97.0% | 0.0% | 3.7% | 13.7% | 0.4% | 0.1% | 0.1% |
| Missing | 0.0% | 100% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% |
| Minor | 2.1% | 0.0% | 94.8% | 10.4% | 0.0% | 0.2% | 0.2% |
| Outlier | 0.8% | 0.0% | 1.3% | 74.7% | 0.1% | 0.1% | 0.0% |
| Square | 0.1% | 0.0% | 0.2% | 0.2% | 99.5% | 0.0% | 0.0% |
| Trend | 0.0% | 0.0% | 0.0% | 0.6% | 0.0% | 98.9% | 11.5% |
| Drift | 0.0% | 0.0% | 0.0% | 0.4% | 0.0% | 0.7% | 88.2% |

(b) Confusion matrix

Fig. 10 Training and validation results of pattern recognition network

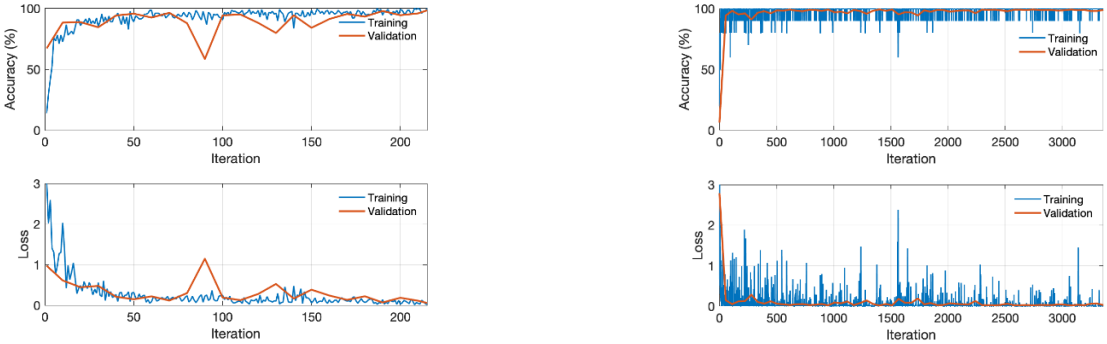



(a) Learning curve

| | Normal | Missing | Minor | Outlier | Square | Trend | Drift |
|---|---|---|---|---|---|---|---|
| Normal | 96.3% | 0.0% | 1.5% | 2.5% | 0.0% | 0.0% | 0.0% |
| Missing | 0.0% | 99.5% | 0.0% | 0.4% | 0.0% | 0.0% | 0.0% |
| Minor | 1.4% | 0.0% | 95.1% | 3.6% | 0.0% | 0.0% | 0.2% |
| Outlier | 2.2% | 0.1% | 3.3% | 93.4% | 0.1% | 0.0% | 0.0% |
| Square | 0.1% | 0.0% | 0.0% | 0.0% | 99.9% | 0.0% | 0.0% |
| Trend | 0.0% | 0.4% | 0.0% | 0.1% | 0.0% | 99.3% | 2.8% |
| Drift | 0.0% | 0.0% | 0.1% | 0.0% | 0.0% | 0.7% | 97.2% |

(b) Confusion matrix

Fig. 11 Training and validation results of CNN-1




(a) Learning curve

(b) Confusion matrix

Fig. 12 Training and validation results of CNN-2 and GoogLeNet

number of 800 samples per label is used as training and 100 samples for validation. The training samples are different from the statistic-based pattern recognition network (i.e., 523 samples) due to the dataset augmentation, where the responses are flipped from left to right and manually selected. This manual selection does not affect the anomaly features but does generate a different image for training. The learning curve is shown in Fig. 11(a). As can be seen, the accuracy rate of training reaches 100%, while the validation accuracy is around 95%. Fig. 11(b) shows the confusion matrix for all types of data anomalies using the pre-processed full dataset. Likewise, to the result from the pattern recognition network, minor datasets are difficult to identify and can be confused with outlier datasets, while other data anomalies have a relatively high accuracy of 95% at least.

Fig. 12 represents the learning curve from the CNN-2 and GoogLeNet models. The time-history based image visualization is utilized for training. The blue curve represents the accuracy and loss from the training dataset (800 samples per label), while the red curve represents those from the validation dataset (100 samples per label). In both models, the losses are observed to converge at a very low value, indicating the effectiveness of the pre-trained

Table 1 Performance comparison when different parameters are employed for one-hour signals

| | Properties | | | | | Type of data anomaly (accuracy, %) | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | Resolution | Resize | Model | Epoch | Learning rate | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | 65×1 | - | Pattern | 60 | 1e-3 | 97.0 | 100 | 94.8 | 74.7 | 99.5 | 98.9 | 88.2 |
| 2 | 103×2400 | - | CNN-1 | 200 | 1e-5 | 96.3 | 99.5 | 95.1 | 93.4 | 99.8 | 99.3 | 97.2 |
| 3 | 256×256 | 28 | CNN-2 | 4 | 1e-2 | 97.2 | 99.8 | 77.0 | 89.0 | 99.4 | 84.8 | 99.8 |
| 4 | 256×256 | 64 | CNN-2 | 4 | 1e-2 | 95.3 | 99.9 | 91.9 | 86.8 | 99.2 | 94.3 | 93.7 |
| 5 | 256×256 | 64 | CNN-2 | 5 | 1e-2 | 97.9 | 99.8 | 93.3 | 93.6 | 99.8 | 98.6 | 99.1 |
| 6 | 256×256 | 224 | GoogLeNet | 4 | 1e-3 | 99.6 | 99.7 | 99.6 | 87.7 | 99.9 | 99.5 | 98.0 |
| 7 | 256×256 | 224 | GoogLeNet | 6 | 1e-3 | 98.5 | 99.7 | 99.1 | 97.0 | 99.9 | 97.1 | 100 |
| 8 | 256×256 | 224 | GoogLeNet | 6 | 1e-5 | 97.8 | 100 | 98.9 | 94.9 | 99.4 | 98.1 | 99.6 |

(note: 1: normal; 2: missing; 3: minor; 4: outlier; 5: square; 6: trend; 7: drift)

model. Moreover, the high validation accuracy implies a high-quality model being obtained.

### 3.2 Individual results based on same-month datasets and discussion

Although the training dataset varies in each model in Section 3.1, the pre-processed dataset (i.e., full dataset without ambiguously labeled samples) is employed to evaluate model performance with respect to the accuracy of the proposed data anomaly classification. Additionally, various hyperparameters (e.g., resolution, epoch, and learning rate, etc.) are tuned to achieve high performance and discussed in the comparative study. As listed in Table 1, a higher resolution results in better accuracy in the CNN-2 models. On the other hand, reducing the learning rate likely enforces the GoogLeNet model to converge at a local optimum. Moreover, the accuracy can be improved for both the CNN-2 and GoogLeNet models by increasing the number of epochs. The consistent trend observed from the CNN-2 and GoogLeNet models also indicates that different CNN architectures can provide a similar characteristic. In addition, with a much more complex model such as GoogLeNet, more features of data anomalies can be extracted as compared to other simple models. This sort of models is then expected to classify the anomalies with higher accuracy. By comparing all proposed ML models, the pattern recognition network (#1) is observed to be less effective because of insufficient trained datasets. As a result, GoogLeNet with an input size of $224 \times 224$ pixels, 6 epochs, and a learning rate of 0.003 yields the best performance among all the models, and the accuracy of all data anomalies is higher than 97%.

## 4. Online data anomaly detection

To obtain precise and efficient data anomaly occurrences, the dataset is divided into 6 portions without flipping the response. The training set only contains the first 10 minute responses. Because the outlier lacks sufficient data (e.g., spikes in time histories), the data flipping is applied to augmenting the data size. The training and testing setups for the 10-minute detection are listed in Table 2.

For online data anomaly detection, a short-period detection should be explored. Instead of using the 1-hour datasets, the datasets are divided into 6 portions without overlapping (i.e., exact 10 minutes) to minimize the detection time. The length of the data is crucial because the drift and trend data anomalies highly depend on the signal length. The ground truth from the one-hour datasets is considered as the labels for each divided portion of the signal. However, for outlier data anomalies, only the peak

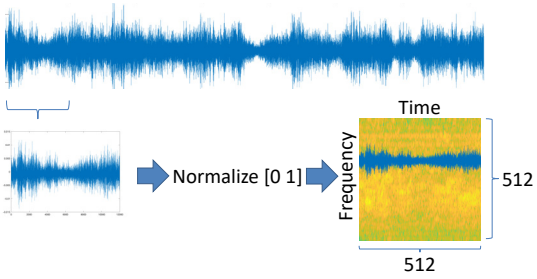



Fig. 13 Flowchart of proposed combination space

responses (outlier response) are extracted to guarantee labeling precision. In this study, the best two models (i.e., GoogLeNet and CNN-2) are utilized for the 10-min data anomaly detection. For both models, the data is divided into 6 portions with no overlapping and normalized by Eq. (2). As compared to the CNN-2, GoogLeNet further presents the time-frequency responses as images through the flowchart illustrated in Fig. 13.

The same parameters, as shown in Table 1, are selected for the two models to investigate the effectiveness while applying short-period detection. In Table 3, the 5$^{th}$ test with 6 epochs and a learning rate of $10^{-3}$ has the best performance. By comparing results between Tables 1 and 3, all data anomalies have similar performance except for the trend data anomaly. This result indicates the importance of the data length, where the short-length trend responses sometimes share similar features with the drift responses. For instance, Fig. 14(a) demonstrates the drifting time history to be explored. When using the full response for image conversion, the detected data anomaly type is correct. However, the trend data anomaly is identified when the data is divided into small portions, as shown in Fig. 14(b). Moreover, the time-frequency trend and drift feature maps are similar (see Fig. 9), and these two types of data anomalies may be misled by the anomaly detection.

To avoid mispresenting anomalies caused by segmenting (i.e., sudden cut of data) during online detection, the 10-minute data is overlapped by 50% (5 minutes). Only the ones that are identified as abnormal twice are considered as data anomalies. As a single signal as an example, Fig. 15 represents the online data anomaly detection using GoogLeNet. The green mask indicates that the model classifies the portion as minor, while the cyan mask indicates the outlier response. The time of occurrence is selected as the midpoint of the detected interval, which is listed in Table 4. In this table, the minor and outlier are successfully classified. To summarize, the proposed Image-Based Time-Frequency Hybrid CNN (GoogLeNet) model is capable of classifying 7 types of data anomalies online with decent accuracy.

Table 2 Setup for training and testing for 10-min detection

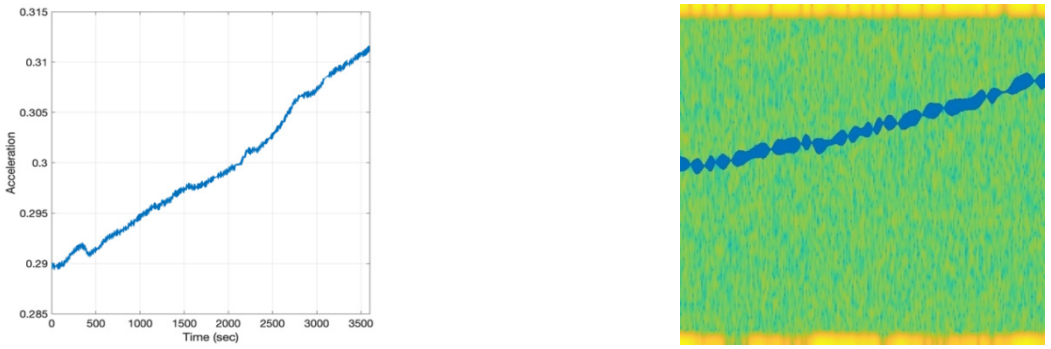
| /type | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Training dataset | | | | 800 | | | |
| Testing(full) dataset | 74838 | 17628 | 9936 | 1024 | 17904 | 30918 | 4062 |

(note: 1: normal; 2: missing; 3: minor; 4: outlier; 5: square; 6: trend; 7: drift)

Table 3 Performance comparison when different parameters are employed for short-period online detection

| | Properties | | | | | Type of data anomaly (accuracy, %) | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | Resolution | Input size | Model | Epoch | Learning rate | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | 512×512 | 28 | CNN-2 | 4 | $10^{-2}$ | 91.2 | 99.7 | 88.2 | 96.9 | 99.5 | 83.2 | 98.0 |
| 2 | 512×512 | 64 | CNN-2 | 4 | $10^{-2}$ | 97.0 | 99.6 | 96.5 | 93.3 | 99.3 | 83.1 | 97.9 |
| 3 | 512×512 | 64 | CNN-2 | 5 | $10^{-2}$ | 97.7 | 99.6 | 97.8 | 94.2 | 99.5 | 88.5 | 95.2 |
| 4 | 512×512 | 224 | GoogLeNet | 4 | $10^{-3}$ | 99.3 | 99.6 | 98.6 | 98.7 | 99.7 | 85.3 | 97.5 |
| 5 | 512×512 | 224 | GoogLeNet | 6 | $10^{-3}$ | 98.6 | 99.6 | 98.8 | 99.6 | 99.7 | 86.7 | 97.1 |
| 6 | 512×512 | 224 | GoogLeNet | 6 | $10^{-5}$ | 98.0 | 99.6 | 97.7 | 98.2 | 99.4 | 86.4 | 94.9 |

(note: 1: normal; 2: missing; 3: minor; 4: outlier; 5: square; 6: trend; 7: drift)




(a) Time-history of drift (b) Feature map using time-history combined space

Fig. 14 Similar features between trend and drift

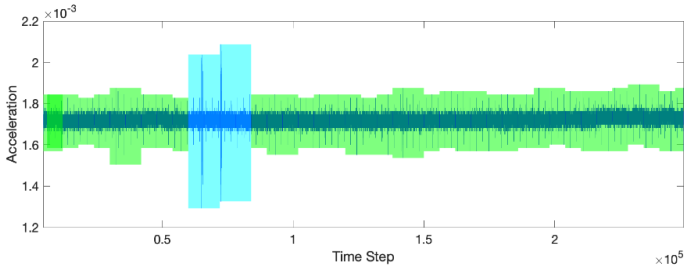



Fig. 15 Performance of channel 1 during 2020-01-01 (green: minor; cyan: outlier)

Table 4 Time of data anomaly occurrence and fault type

| # | Time of Occurrence | Type |
|---|---|---|
| 1 | 2020-01-01-00:05:00 – 2012-01-01-00:45:00 | Minor |
| 2 | 2012-01-01-00:50:00 – 2012-01-01-01:05:00 | Outlier |
| 3 | 2012-01-01-01:10:00 – 2012-01-01-03:20:00 | Minor |

## 5. Ensemble neural network model

### *5.1 Model evaluation*

The best ML model is tested using the original dataset (including the training dataset), and the confusion matrix is presented in Fig. 16(a). As shown in this figure, all types of data anomalies are detected with accuracy over 90%. However, in some cases, normal are classified as minor or outlier, and in some cases trend and drift are misidentified. This is due to the fact that the same features are observed in trend and drift, which is also difficult even by manual classification. In addition, some ground truths are accidentally labeled in wrong categories, resulting in false-positive detections. To enhance the performance, the results based on the models (i.e., pattern recognition network, CNN-1, GoogLeNet,) are merged as an ensemble model. The classified labels will only be considered as true ones if two or more models have the same result. If different labels are obtained from each model, the result using GoogLeNet will be adopted. In Fig. 16(b), the confusion matrix using the ensemble model based on the original dataset demonstrates better performance in most of the data anomalies. By combining three models, performance of the ensemble neural network model significantly outperforms the GoogLeNet alone.

In addition, all datasets are utilized for detection by using the ensemble model as shown in Fig. 17. Different

| | Normal | Missing | Minor | Outlier | Square | Trend | Drift |
|---|---|---|---|---|---|---|---|
| Normal | 90.3% | 0.0% | 4.1% | 0.5% | 1.8% | 0.0% | 0.0% |
| Missing | 0.0% | 99.9% | 0.0% | 0.6% | 0.1% | 1.5% | 0.0% |
| Minor | 3.1% | 0.0% | 94.3% | 1.9% | 0.0% | 0.2% | 0.0% |
| Outlier | 6.6% | 0.1% | 1.6% | 97.0% | 0.0% | 0.0% | 0.0% |
| Square | 0.0% | 0.0% | 0.0% | 0.0% | 98.1% | 0.0% | 0.0% |
| Trend | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 91.5% | 1.0% |
| Drift | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 6.8% | 99.0% |

(a) GoogLeNet

| | Normal | Missing | Minor | Outlier | Square | Trend | Drift |
|---|---|---|---|---|---|---|---|
| Normal | 94.1% | 0.0% | 3.7% | 0.9% | 0.5% | 0.0% | 0.0% |
| Missing | 0.0% | 99.9% | 0.0% | 0.2% | 0.1% | 0.6% | 0.0% |
| Minor | 3.0% | 0.0% | 95.7% | 2.1% | 0.0% | 0.3% | 0.0% |
| Outlier | 2.8% | 0.1% | 0.6% | 96.8% | 0.0% | 0.0% | 0.0% |
| Square | 0.1% | 0.0% | 0.0% | 0.0% | 99.4% | 0.0% | 0.0% |
| Trend | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 94.9% | 2.2% |
| Drift | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 4.2% | 97.8% |

(b) Ensemble neural network

Fig. 16 Confusion matrix of using GoogLeNet and ensemble neural network

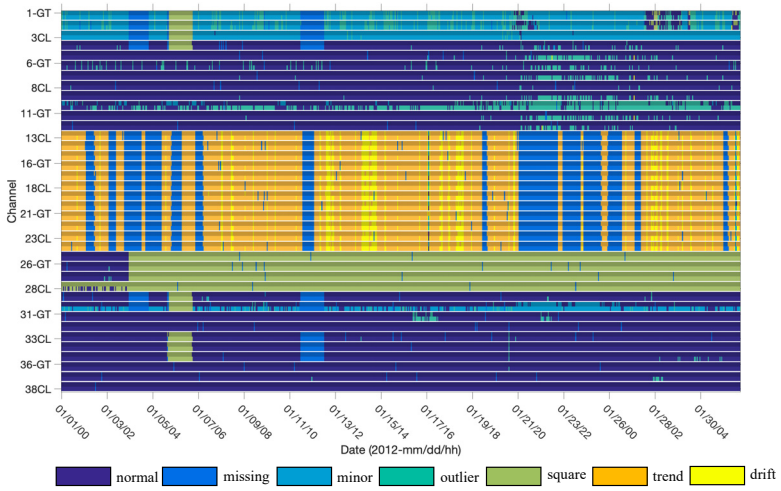


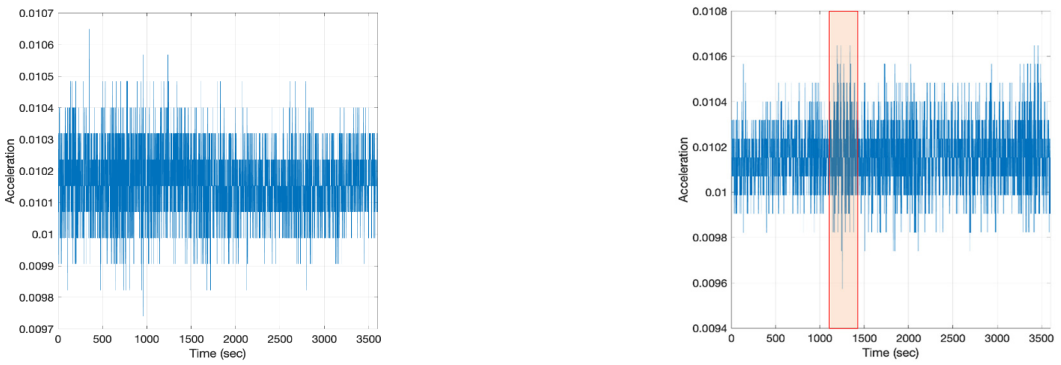

Fig. 17 Data anomaly detection using the proposed ensemble neural network model (GT: ground truth, CL: classification)

colors represent different types of data anomalies. For each row in the figure, both the ground truth and the classification results are displayed. As mentioned previously, normal data is sometimes misidentified as the minor data anomaly such as the 30th channel in Fig. 17. The consistency between the two types indicates that the proposed model is capable of classifying the data anomalies with high accuracy. In addition, the figure also indicates that data anomaly will reoccur to the same sensor in most cases. This repeating behavior provides an opportunity of strengthening the ML model in the future by utilizing time-sequence relationships to the input data.

One important issue observed in this study is that classification errors not only can result from the ML models but sometimes occur due to false-positive ground truths. For instance, Fig. 18 represents the time history of the 30$^{th}$ channel at two different times. The data in Fig. 18(a) is classified as the minor but originally labeled as the normal. The data in Fig. 18(b) is classified as the outlier but also originally labeled as the normal. These results are false-positive and inevitable in the field applications. In addition, multiple data anomalies may occur such as the results in Fig. 18(b). Both minor and outlier responses can be observed.

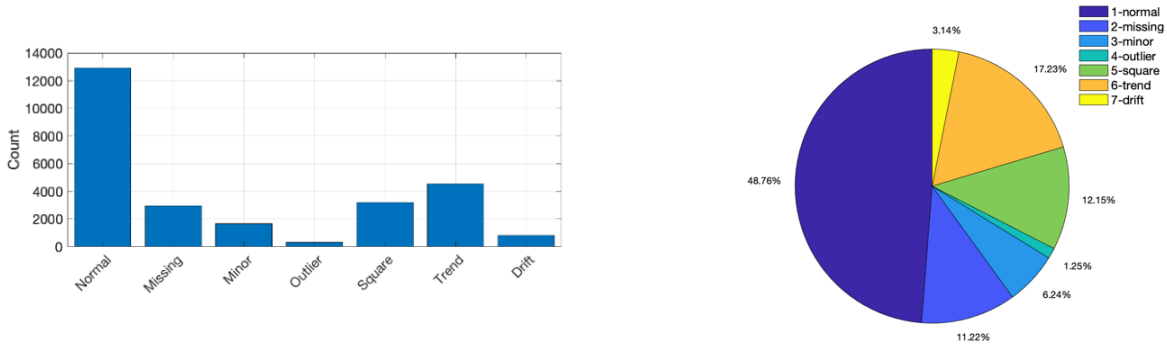
### *5.2 Blind test*

Signal characteristics may change over time, and performance resulted from one month data can overestimate the accuracy as time flies. Thus, monitoring data collected from the same bridge but in another month is utilized as a blind test. The dataset contains 696 acceleration responses with all the 38 channels. Fig. 19 represents the distribution of each type of data anomaly in the blind test. As seen, the blind dataset is highly imbalanced with only 331 outlier samples. Table 5 lists data anomaly classification performance from the blind test dataset using 1) the ensemble neural network, 2) statistic-based pattern recognition network, 3) spectrogram-based convolutional neural network (CNN-1), 4) image-based time history convolutional neural network (CNN-2), and 5) image-based time-frequency hybrid convolution neural network (GoogLeNet). The overall values indicate the averaged detection rate over all anomalies. As seen in the table, the ensemble neural network has comparable performance with the results in Fig. 16 except for outlier signals. GoogLeNet has slightly decreased performance for some anomalies, such as normal, outlier, and trend. For the rest of methods, significant performance reductions are found. To sum up the blind test evaluation, the ensemble neural network still




(a) 2012-01-01-02 (classified as minor) (b) 2012-01-03-20 (classified as outlier)

Fig. 18 Time responses of Channel 30 which ground truth is normal response




(a) Label counts (b) Percentage pie chart

Fig. 19 Distribution of blind test dataset

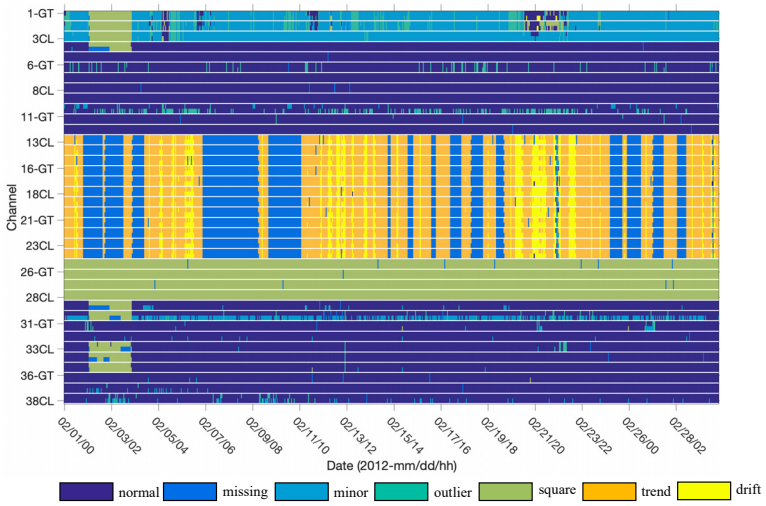



Fig. 20 Blind test detection using the proposed ensemble neural network model (GT: ground truth, CL: classification)

shows the best performance among these five models and demonstrates the capability of classifying anomalies with time-variant features.

Fig. 20 represents the result of data anomaly detection using the ensemble neural network model. Fig. 21 shows the confusion matrix resulted from the blind test. The overall accuracy is 94.4%. Note that false-positive ground truth (e.g., outlier labeled as normal, minor labeled as normal) is also observed at the 10$^{th}$ channel and 30$^{th}$ channel in this dataset. Overall, the consistency between the detection and ground truth (i.e., originally labeled) indicates the high effectiveness of the model. Consequently, the proposed ensemble neural network model is capable of classifying data anomalies with high accuracy regarding time-dependent characteristics.

Table 5 Performance comparison among different models using blind test dataset

| | Type of data anomaly (accuracy, %) | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Overall |
| Ensemble | 93.4 | 100 | 96.2 | 86.1 | 96.7 | 89.6 | 97.7 | 94.4 |
| Pattern | 76.0 | 99.8 | 69.5 | 68.3 | 87.2 | 56.4 | 58.2 | 75.6 |
| CNN-1 | 92.3 | 99.9 | 88.2 | 75.2 | 89.7 | 92.8 | 91.6 | 92.4 |
| CNN-2 | 88.4 | 99.8 | 88.0 | 74.0 | 99.7 | 68.4 | 96.2 | 87.7 |
| GoogLeNet | 90.1 | 99.8 | 96.0 | 87.2 | 99.4 | 88.2 | 97.3 | 92.5 |

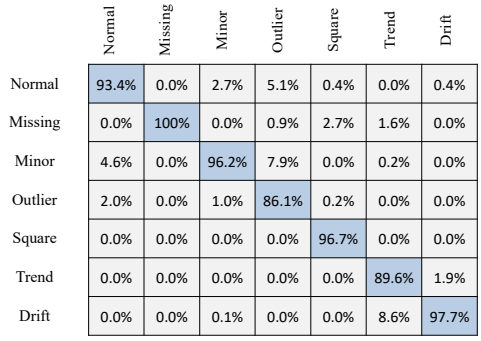



Fig. 21 Confusion matrix based on blind test

## 6. Conclusions

In this study, four machine learning techniques were exploited to detect data anomalies in monitoring data, and these techniques were explored and compared in terms of performance and accuracy. A field dataset from a long-span cable-stayed bridge in China was utilized for training and examining these ML techniques. The results demonstrated that the ML models can successfully detect and classify different data anomalies. As found in the results, data anomaly would reoccur to the same sensor in most cases. When classifying drift and trend, the length of the signal had a large impact on the detection result. In addition, among these ML techniques, the GoogLeNet was tested to have the highest accuracy among the other ML models. Moreover, the GoogLeNet model was also updated and capable of classifying data anomalies online with a time interval of 10 minutes. By combining three different models as an ensemble neural network model, the accuracy was further improved. In accordance with the results and discussions, several brief conclusions can be drawn:

- The image-based CNNs had better performance, as compared with the statistic-based pattern recognition network, due to the increased complexity in the network architecture and the more informative input data. The insignificant features from the original time-history data can be observed and extracted by combining more complicated input construction and network architectures. In addition, the accuracy can be improved by increasing the number of epochs when using a CNN model. The relevance of the image resolutions and learning rate to the accuracy among these CNN models was not obvious in this study.
- The GoogLeNet was tested to have the best performance in 1-hour based classification and exploited for short-period online data anomaly detection. Although the overall accuracy was around 97%, the trend and the drift highly depended on the length of the signal and can sometimes be misclassified. Further studies are needed before the field application to ensure a suitable length.
- Hyperparameters had large impact on model performance. Each parameter should be carefully determined to obtain an optimal model. In addition, this study combined multiple ML models into an ensemble neural network model, and then the accuracy was increased as compared with each model alone.
- The proposed ensemble neural network model was further validated to be effective, even when the signal characteristics were slightly changed over time. This finding was quite essential for this ensemble model to be an autonomous online data anomaly detection method.
- Manually labeling was crucial when applying supervised machine learning techniques. Incorrect labeling (e.g., false-positive labels) would result in a ML model to extract wrong features. Moreover, even if the model was well trained without extracting spurious features, the model accuracy can be misinterpreted. However, labeling was laborious and tedious. In some cases, the data anomaly was equivocal such as in Fig. 18(b) which can be difficult to label even by experts. Therefore, unsupervised machine learning and reinforcement learning techniques can be considered in future studies.

## Acknowledgments

The structural health monitoring data of the long-span bridge are obtained from the organizers of the 1st International Project Competition for Structural Health Monitoring (IPC-SHM), 2020 (http://www.schm.org.cn/#/IPC-SHM, 2020).

## References

Bao, Y., Chen Z., Wei, S., Xu, Y., Tang, Z. and Li, H. (2019a), "The state of the art of data science and engineering in structural health monitoring", *Engineering*, **5**, 234-242. https://doi.org/10.1016/j.eng.2018.11.027

Bao, Y., Tang, Z., Li, H. and Zhang, Y. (2019b), "Computer vision and deep learning–based data anomaly detection method for structural health monitoring", *Struct. Health Monitor.*, **18**(2), 401-421. https://doi.org/10.1177/1475921718757405

Bao, Y., Li, J., Nagayama, T., Xu. Y., Spencer, B.F. and Li, H. (2021), "The 1st International Project Competition for Structural Health Monitoring (IPC-SHM, 2020): A summary and benchmark problem", *Struct. Health Monitor.*, **20**(4), 2229-2239. https://doi.org/10.1177/14759217211006485

Boashash, B. and Ouelha, S. (2016), "Automatic signal abnormality detection using time-frequency features and machine learning: A newborn EEG seizure case study", *Knowledge-Based Syst.*, **106**, 38-50. https://doi.org/10.1016/j.knosys.2016.05.027

Chang, C.M., Chou, J.Y., Tan, P. and Wang, L. (2017), "A sensor fault detection strategy for structural health monitoring systems", *Smart Struct. Syst., Int. J.*, **20**(1), 43-52. https://doi.org/10.12989/sss.2017.20.1.043

Christian, S., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. and Rabinovich, A. (2015), "Going deeper with convolutions", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-9.

Dragos, K. and Smarsly, K. (2016), "Distributed adaptive diagnosis of sensor faults using structural response data", *Smart Mater. Struct.*, **25**(10), 105019. https://doi.org/10.1088/0964-1726/25/10/105019

Fu, Y., Peng, C., Gomez, F., Narazaki, Y. and Spencer Jr., B.F. (2019), "Sensor fault management techniques for wireless smart sensor networks in structural health monitoring", *Struct. Control Health Monitor.*, **26**(7), e2362. https://doi.org/10.1002/stc.2362

Huang, H.B., Yi, T.H. and Li, H.N. (2017a), "Sensor fault diagnosis for structural health monitoring based on statistical hypothesis test and missing variable approach", *J. Aerosp. Eng.*, **30**(2), B4015003. https://doi.org/10.1061/(ASCE)AS.1943-5525.0000572

Huang, H.B., Yi, T.H. and Li, H.N. (2017b), "Bayesian combination of weighted principal-component analysis for diagnosing sensor faults in structural monitoring systems", *J. Eng. Mech.*, **143**(9), 04017088. https://doi.org/10.1061/(ASCE)EM.1943-7889.0001309

Krizhevesky, A., Sutskever, I. and Hinton, G.E. (2012), "ImageNet classification with deep convolutional neural networks", *Adv. Neural Inform. Process. Syst.*, **25**.

Li, L., Liu, G., Zhang, L. and Li, Q. (2019), "Sensor fault detection with generalized likelihood ratio and correlation coefficient for bridge SHM", *J. Sound Vib.*, **442**, 445-458. https://doi.org/10.1016/j.ymssp.2015.05.011

Lo, C., Lynch, J.P. and Liu, M. (2016), "Distributed model-based nonlinear sensor fault diagnosis in wireless sensor networks", *Mech. Syst. Signal Process.*, **66**, 470-484. https://doi.org/10.1016/j.ymssp.2015.05.011

Mahapatro, A. and Khilar, P.M. (2013), "Fault diagnosis in wireless sensor networks: A survey", *IEEE Commun. Surveys Tutorials*, **15**(4), 2000-2026. https://doi.org/10.1109/SURV.2013.030713.00062

Mao, J., Wang, H. and Spencer Jr, B.F. (2020), "Toward data anomaly detection for automated structural health monitoring: Exploiting generative adversarial nets and autoencoders", *Struct. Health Monitor.*, **20**(4), 1609-1626. https://doi.org/10.1177/1475921720924601

MATLAB and Deep Learning Toolbox Release (2020b), The MathWorks, Inc., Natick, MA, USA.

Munir, M., Siddiqui, S.A., Chattha, M.A., Dengel, A. and Ahmed, S. (2019), "FuseAD: unsupervised anomaly detection in streaming sensors data by fusing statistical and deep learning models", *Sensors*, **19**(11), 2451. https://doi.org/10.3390/s19112451

Pan, S.J. and Yang, Q. (2010), "A survey on transfer learning", *IEEE Transact. Knowledge Data Eng.*, **22**(10), 1345-1359. https://doi.org/10.1109/TKDE.2009.191

Pavelka, A. and Proch, A. (2004), "Algorithm for initialization of neural network weights random numbers in MATLAB", *Proceeding: Control Engineering*, **2**, 453-459.

Peng, C., Fu, Y. and Spencer Jr., B.F. (2017a), "Sensor fault detection, identification, and recovery techniques for wireless sensor networks: A full-scale study", *Proceedings of the 13th International Workshop on Advanced Smart Materials and Smart Structures Technology*.

Peng, Y., Qiao, W., Qu, L. and Wang, J. (2017b), "Sensor fault detection and isolation for a wireless sensor network-based remote wind turbine condition monitoring system", *IEEE Transact. Ind. Applicat.*, **54**(2), 1072-1079. https://doi.org/10.1109/TIA.2017.2777925

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C. and Li, F.F. (2015), "ImageNet larger scale visual recognition challenge", *Int. J. Comput. Vision*, **115**, 211-252. https://doi.org/10.1007/s11263-015-0816-y

Sharma, A.B., Golubchik, L. and Govindan, R. (2010), "Sensor faults: detection methods and prevalence in real-world datasets", *ACM Transact. Sensor Networks* (*TOSN*), **6**(3), 23. https://doi.org/10.1145/1754414.1754419

Singla, A., Yuan, L. and Ebrahimi, T. (2016), "Food/non-food image classification and food categorization using pre-trained GoogLeNet Model", *MADiMa 16: Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management*, October, pp. 3-11. https://doi.org/10.1145/2986035.2986039

Smarsly, K. and Law, K.H. (2014), "Decentralized fault detection and isolation in wireless structural health monitoring systems using analytical redundancy", *Adv. Eng. Software*, **73**, 1-10. https://doi.org/10.1016/j.advengsoft.2014.02.005

Tang, Z., Chen, Z., Bao, Y. and Li, H. (2018), "Convolutional neural network-based data anomaly detection method using multiple information for structural health monitoring", *Struct. Control Health Monitor.*, **26**(1), e2296. https://doi.org/10.1002/stc.2296

Yu, C.B., Hu, J.J., Li, R., Deng, S.H. and Yang, R.M. (2014), "Node fault diagnosis in WSN based on RS and SVM", *Proceedings of 2014 International Conference on Wireless Communication and Sensor Network*, Wuhan, China, December, pp. 153-156.

Zhao, C., Sun, X., Sun, S. and Jiang, T. (2011), "Fault diagnosis of sensor by chaos particle swarm optimization algorithm and support vector machine", *Expert Syst. Applicat.*, **38**(8), 9908-9912. https://doi.org/10.1016/j.eswa.2011.02.043