Renewable Energy 185 (2022) 916–931

Contents lists available at ScienceDirect

Renewable Energy

journal homepage: www.elsevier.com/locate/renene

Convolutional neural network fault classification based on time-series analysis for benchmark wind turbine machine

Reihane Rahimilarki [a], Zhiwei Gao [a, *], Nanlin Jin [a], Aihua Zhang [b]

[a] Faculty of Engineering & Environment, Northumbria University, Newcastle upon Tyne, NE1 8ST, UK
[b] College of Engineering, Bohai University, Jinzhou, China

A R T I C L E I N F O

Article history:
Received 7 April 2021
Received in revised form
19 November 2021
Accepted 13 December 2021
Available online 22 December 2021

Keywords:
Deep learning
Convolutional neural networks
Fault classification
Time-series analysis
Wind turbines
Offshore wind turbines

A B S T R A C T

Fault detection and classification are considered as one of the most mandatory techniques in nowadays industrial monitoring. The necessity of fault monitoring is due to the fact that early detection can restrain high-cost maintenance. Due to the complexity of the wind turbines and the considerable amount of data available via SCADA systems, machine learning methods and specifically deep learning approaches seem to be powerful means to solve the problem of fault detection in wind turbines. In this article, a novel deep learning fault detection and classification method is presented based on the time-series analysis technique and convolutional neural networks (CNN) in order to deal with some classes of faults in wind turbine machines. To validate this approach, challenging scenarios, which consists of less than 5% performance reduction (which is hard to identify) in the two actuators or four sensors of the wind turbine along with sensors noise are investigated, and the appropriate structures of CNN are suggested. Finally, these algorithms are evaluated in simulation based on the data of a 4.8 MW wind turbine benchmark and their accuracy approves the convincing performance of the proposed methods. The proposed algorithm are applicable to both on-shore and off-shore wind turbine machines.

1. Introduction

As environmental pollution and the concerns on global warming are increasing year by year, the importance of using renewable and green energy becomes more vital. As a result, as one of the cleanest energy resources, wind turbine industries, receive considerable attention and budgets. Gradually, wind energy has become an integrated component in the grid in the world. In the UK, wind turbines are reliably responsible for over 20% of the consumed electricity in 2020 [1]. Precisely speaking, by August 2020, there are 10 911 installed wind turbines with the capacity of 24 000 Mega Watts, from which 13 600 Mega Watts onshore and the rest offshore. This production puts the United Kingdom at the sixth-largest producer of wind power in the world [1]. The plan for expanding this potential is to increase the capacity of wind power up to 50 000 Mega Watts by 2030 [1]. Therefore, regarding this plan, careful consideration should be made to have a reliable energy resource in the grid (see Fig. 24).

However, wind turbines just like any other electro-mechanical system, may encounter several faults. Faults in wind turbines can be both in electrical parts and mechanical ones. The nature of the faults can be based on environmental factors, such as high fluctuations of the wind, or based on physical aspects of the components, e.g aging, saturation, or thermal problems. The most occurred faults and their ratios is illustrated in Fig. 1 [2].

Apart from the faults with the basis of high wind fluctuations, many of the faults can be prevented or effectively decreased by a suitable monitoring system. If a typical and low-risk fault occurs and it is not detected and resolved in proper time, it may lead to a severe failure and probable breakdown. Breakdown and maintenance can be even more oppressive and expensive in an offshore wind turbine and it needs a more accurate fault diagnosis system to prevent extra costs. In addition, substandard reliability directly decreases the availability of wind power in the grid. Based on the mentioned issues, fault detection plays an important role in increasing the reliability of wind turbines.

There are three major categories of fault detection in the literature: **model-based**, **signal-based**, and **knowledge-based** [3]. Based on the accurate physical model of the system, model-based

* Corresponding author.
E-mail addresses: reihane.rahimilarki@northumbria.ac.uk (R. Rahimilarki), zhiwei.gao@northumbria.ac.uk (Z. Gao), nanlin.jin@northumbria.ac.uk (N. Jin), zhangaihua@qymail.bhu.edu.cn (A. Zhang).

https://doi.org/10.1016/j.renene.2021.12.056
0960-1481/

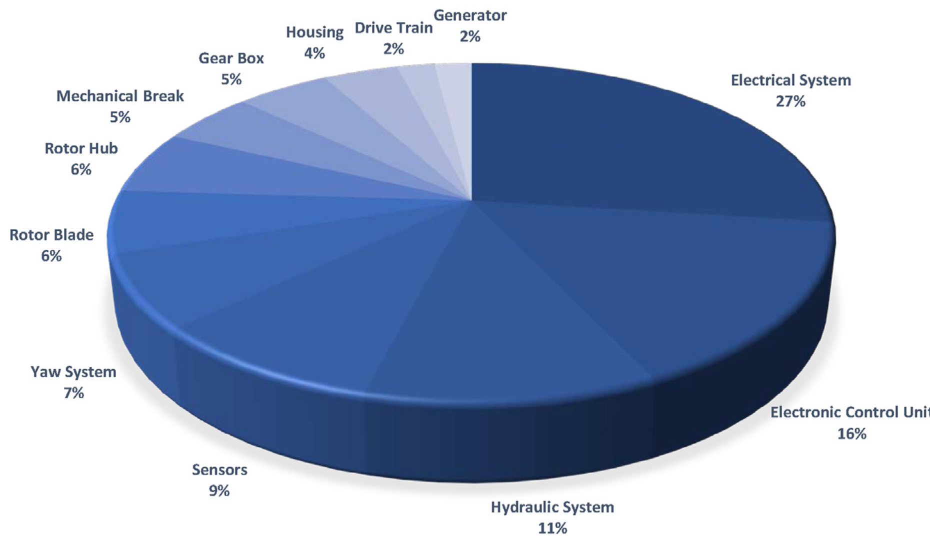


Fig. 1. The ratio of occurred faults in wind turbines [2].

monitoring approaches are proposed [4]. It is worth to mention that the accuracy of the proposed approach is dependent on the accuracy of the used model [5]. Two cascade steps are usually applied in a typical model-based fault diagnosis: *Residual Generation*, and *Residual Evaluation*. In the step of residual generation, the output of the designed model is compered to the output of the real system based on the Equ. 1 [6].

$$r(t) = y - \hat{y} \tag{1}$$

in which, y is the output of the real system and $\hat{y}$ is the output of the model. After this step, the step of residual evaluation should be designed. In this step, based on the expert decision, some thresholds are set to reflect the conditions of the system, which can be healthy or faulty. This step may also include some mathematical or statistical approaches to design a model-based fault observer. The problem of accuracy and sensitivity of the fault diagnosis method was investigated based on scalar quantisation hidden Markov model in Ref. [7]. [8] addressed the problem of performance degradation using regression model and classifier adapting. In Ref. [9], an interval-based observer with analytical redundancy relations was studied to design a fault diagnosis method. A method consists of a model-based fault detection technique, and the proportional-integral observer was presented in Ref. [10] to confront the problem of noise filtering and to achieve a suitable convergence. In Ref. [11], a fault detection and isolation method was presented to attenuate the disturbance's effects based on the LMI approach. In Ref. [12], a model-based fault diagnosis was proposed based on time-varying adaptive thresholds and an approximation estimator for floating offshore wind turbines. The problem of parameter-varying modelling was investigated in Ref. [13] in a 4.8 MW wind turbine and was solved by proposing an adaptive observer.

On the contrary to the model-based approaches, in signal-based ones, it is not essential to have the accurate input-output relation of the model. However, it is vital to analyse the measured signals and to make the decision based on their attributes [14]. Knowing about the featured signals and how an occurred fault has a reflection on the output signal, needs a technical aspect of view.

In general, the signals that are used for fault diagnosis can be analysed in time-domain or frequency-domain. In time-domain fault diagnosis, some time-domain features, such as slope, and root mean square are analysed [15,16]. In frequency-domain fault diagnosis, the frequency-based parameters are studied to carry out a signal-based fault diagnosis approach. One of the examples of this category is using wavelet for denoising of the vibration of the machinery systems [17]. There are also some researches on having Fourier Transform in signal-based fault diagnosis [18,19].

Signal-based fault diagnosis is also vastly studied in the application of wind turbines. In order to solve the problem speed-varying fault diagnosis in wind turbine gearboxes, a multiscale filtering construction fault diagnosis approach was addressed in Ref. [20]. Two fault classification methods were proposed in Refs. [21,22] based on Principle Component Analysis technique, to deal with the problem of sinusoidal fault and actuator effectiveness loss in a wind turbine benchmark. In Ref. [23], a fault diagnosis approach based on time-frequency maps was considered in order to solve rotor asymmetry faults in the wind turbine generator unit. In Ref. [12], after establishing an adaptive threshold estimator for an offshore wind turbine, a signal-based algorithm was developed in order to detect mooring lines faults.

The third category, knowledge-based fault diagnosis, is the methods using the former data in complex models. Usually in this category, the accurate dynamics of the system is hard to achieve and the analysis is developed on a large amount of historical data.

Alternatively, knowledge-based methods are particularly suitable for the cases with a large amount of historical data, and the direct relationships of the system dynamics are challenging to derive [24,25]. To serve this purpose, knowledge-based fault diagnosis is also called *data-driven* approach [26]. Data-driven methods can be very rewarding, due to the complexity of nonlinear dynamics and unexpected faults in industrial area [27], such as wind turbines. To give examples on its advantageous applications, in Ref. [28], a data-driven fault diagnosis method was proposed, to cope with noisy signals and uncertainty in models for wind turbine systems. In order to monitor nonlinearity, a data-driven method was suggested in Ref. [29], developed on available measurements. For the problem of sensor fault diagnosis [30], studied a method based on extreme learning machine.

Computational methods, such as Fuzzy [31,32], Support Vector Machine (SVM) [33,34], K-nearest Neighbourhood [35], Long Short-term Memory network (LSTM) [36,37], Artificial Neural Networks (ANN) [38], and Convolutional Neural Networks (CNN) [39,40] have been widely used in the field of fault diagnosis and detection due to their spectacular power to predict unknown parameters and identify the nonlinear systems. For instance, in Ref. [41], a fault detection method based on LSTM was proposed in a wind turbine benchmark. An LSTM based fault diagnosis was studied by Ref. [42] by analysing frequency data [28]. proposed a fault diagnosis and isolation approach in order to handle uncertain models and noisy signals, using a fuzzy method in wind turbine systems. [43], and [44] addressed a robust method based on Takagi-Sugeno Fuzzy systems for the problem of unknown fault diagnosis. The problem of fault classification for vibration signals was studied in Ref. [45] and a novel solution based on Fuzzy and SVM were proposed. By using the current signal of the generator, a fault identification method was considered based on SVM in order to classify different faults occurred in the generator. Neuro-Fuzzy fault diagnosis approach for the problem of bearing failures in wind turbines was suggested in Ref. [46]. [47] developed a supervised ANN algorithm combined with Kalman filter to detect faults in a hydraulic blade pitch system in an offshore wind turbine. In Ref. [48], a CNN based approach was proposed for fault classification in a wind turbine.

Convolutional neural network (CNN) is one of the most potent means in computational methods. It has been grown rapidly in the area of classification in recent years [49]. The problem of fault classification in bearings in wind turbines were investigated in Refs. [50–52] by developing CNN techniques. Moreover, CNN was used in the area of time-series data in the literature [53]. For instance, in Ref. [54], a CNN structure was studied in order to develop a fault diagnosis method, using time-series data conversion.

In this paper, the aim is to propose a data-driven fault detection method based on CNN structure to classify even the small anomalies in the wind turbine benchmark. The novel contribution in this manuscript, is proposing a time series data-to-image conversion stage and proposing a suitable deep learning structure to handle this problem. For that, the first challenge would be working with time-series signals, which is tricky due to the nature of the high volatility of the signals. Therefore, a preprocessing stage will be discussed to prepare the raw data into 2-D greyscale images. The second challenge would be that data from the benchmark not only contains actuator and sensor faults, it also contains sensor noises or environmental disturbances. Here in this paper, by proposing suitable and novel CNN structures along with a pre-processing technique of converting time-series signals into 2-D images, both challenges are addressed and solved efficiently.

Moreover, it is worth mentioning that due to the fact that the proposed algorithms in this article are completely data-driven and are trained based on historical data from the SCADA system, they can be extended to both onshore and offshore wind turbine machines as well as other dynamics and applications. It is also necessary to point out that the designing of the fault classification approach in this paper is not based on dynamics of the system. The wind turbine dynamics is discussed in section 2, just to give the idea of the nature of the wind turbines and the challenges and is used in the simulations.

The rest of this paper is organised as follows: In section II, the model of the wind turbine is discussed. In section III, the CNN method is explained along with the required mathematical fundamentals. The proposed method and the required pre-processing stages are defined in section IV. The novel methods and CNN structured are simulated and trained with the data from the wind turbine benchmark and the results are illustrated in section V.

2. The model of wind turbine benchmark

Over the past few years, wind energy has received significant attention owing to the concerns on global warming, environmental issues, and fossil fuels reduction. During the past decade, numerous investments have been aimed to wind energy industries and the wind turbine installed capacity had a constant increase. Designing the structure and caging of the wind turbines have been improved due to the more accurate engineering and more robust composites. However, the designing of the generators, drive and control systems roughly remain the same.

Wind turbines have been built horizontally or vertically. As it can be seen in Fig. 2 a, in vertical wind turbines, the blades are installed vertically [55]. One of the advantages of this kind of wind turbines is that the conversion systems and the gearboxes are located on the ground, while, the disadvantage of this type is that the maintenance is somehow complicated as it normally requires rotor removal [56]. Besides, the efficiency of converting wind energy to electrical one is not high in compared to horizontal turbines. For these reasons, nowadays, the modern wind turbines usually have been built horizontally, as shown in Fig. 2 b [57].

In this paper, only the three-blade horizontal wind turbine is discussed. However, the approaches can be extended for any other structure by some little adjustments.

The schematic structure and different components of a typical wind turbine are presented in Fig. 3 [58]. Wind energy rotates the blades and produces mechanical power, transmitted to the system via a shaft, which is connected to these blades. A generator converts mechanical energy to electrical one based on the rotation of the blades. The blades angles can vary to handle the wind speed variations. Meanwhile, the yaw structure is designed to align the whole wind turbine based on the direction of the anemometer.

In [59], a benchmark was investigated, in regard to a three-blade horizontal wind turbine driven by variable speed. The rated power is 4.8 MW. The subsystems of this benchmark can be listed as, *Blade and Pitch Systems*, *Drive Train*, *Generator and Converter*, and the *Controller*. The relation between them can be seen in Fig. 4 [60].

As you can see in Fig. 4, each subsystem is being fed by the signals of other subsystems and generates a signal, required by the others. As it is illustrated in the left side of this figure, wind velocity, v_w, is the input of the Blade and Pitch subsystem. The output of this subsystem is rotor torque, τ_r and is fed to the Drive Train subsystem. It can produce two states of the system, ω_r and ω_g, rotor speed and generator speed, respectively, by receiving generator torque feedback, τ_g. By applying the desired generator torque, $\tau_{g,r}$ and ω_g, the output of Generator and Converter subsystem can be τ_g and P_g, which is the generator power. The Controller in this model is responsible for producing the desired pitch angle β_r and the desired generator torque, $\tau_{g,r}$ by getting feedback from all the subsystems. The dynamics of each subsystem, engaging parameters and the physical relations between them are fully discussed in 2.1.

2.1. Wind model

The wind can be modelled in four parts: the mean wind (slow wind variations) $v_m(t)$, a stochastic part $v_s(t)$, the wind shear $v_{ws}(t)$ (which is the effect of wind energy lost at the surface of the earth, commonly resulting in an increasing wind speed as the distance to earth surface increases), and the tower shadow $v_{ts}(t)$. The combined wind model is given by:

$$v_w(t) = v_m(t) + v_s(t) + v_{ws}(t) + v_{ts}(t). \tag{2}$$

The variation of the wind velocity in this benchmark can be seen in Fig. 5. As it can be seen in this figure, the velocity of the wind is varied between 4 and 20 m/s, with some spike of 25 m/s. As it is discussed in Ref. [59], it is an acceptable range, in which, the generator can operate normally.

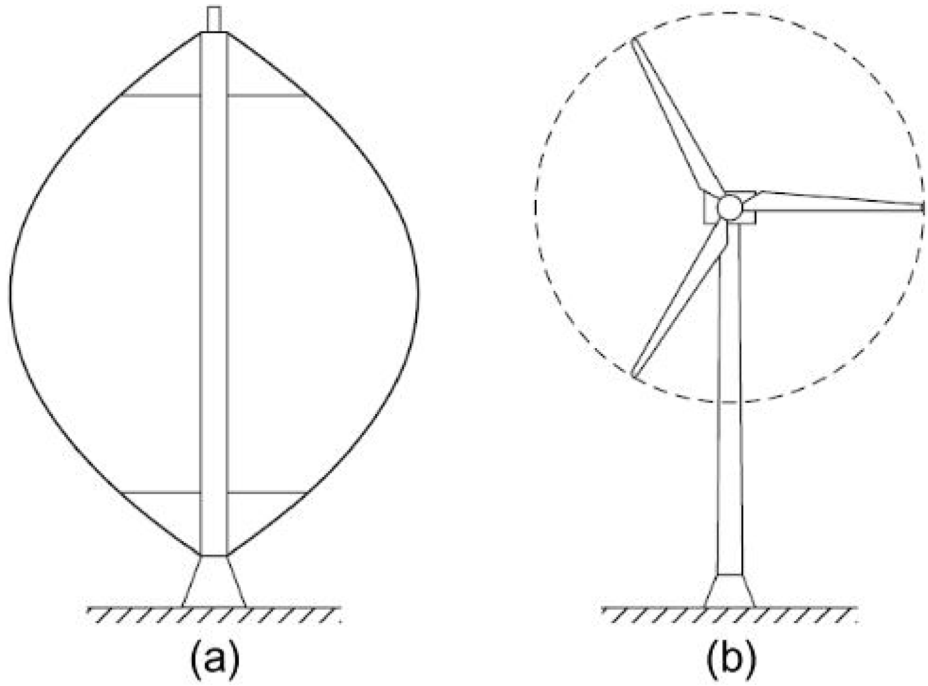


Fig. 2. (a) Vertical-axis structure. (b) Horizontal-axis structure [57].

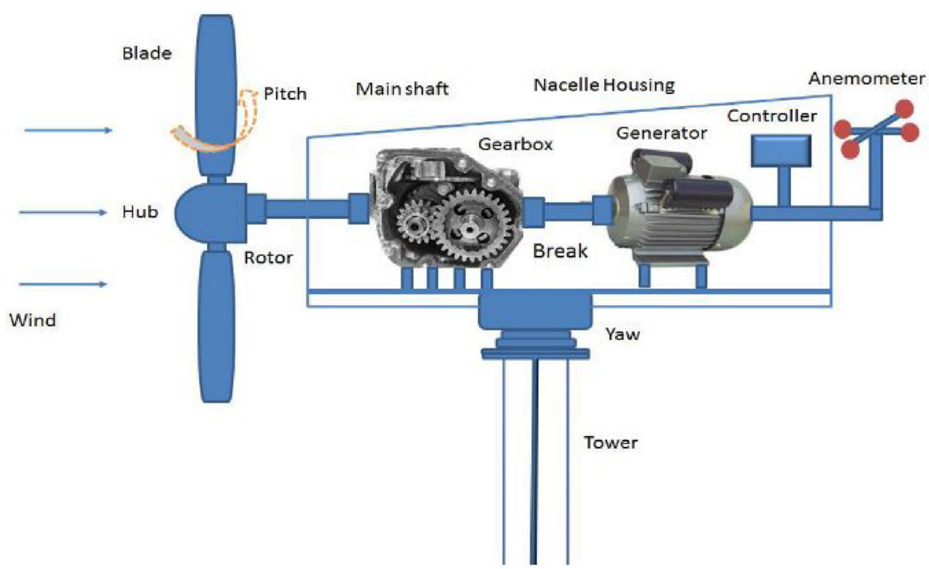


Fig. 3. A typical schematic of a wind turbine [58].

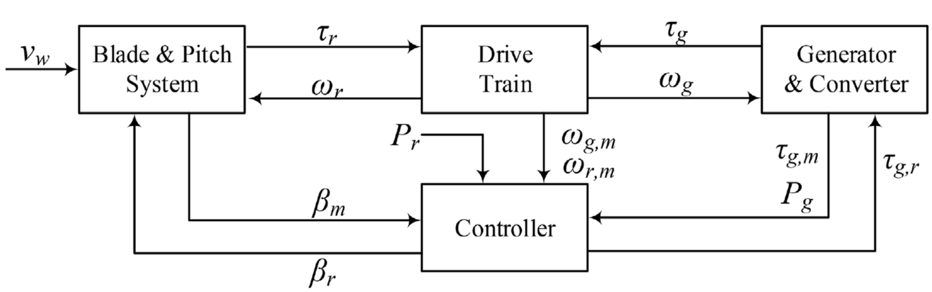


Fig. 4. Wind turbine subsystems [60].

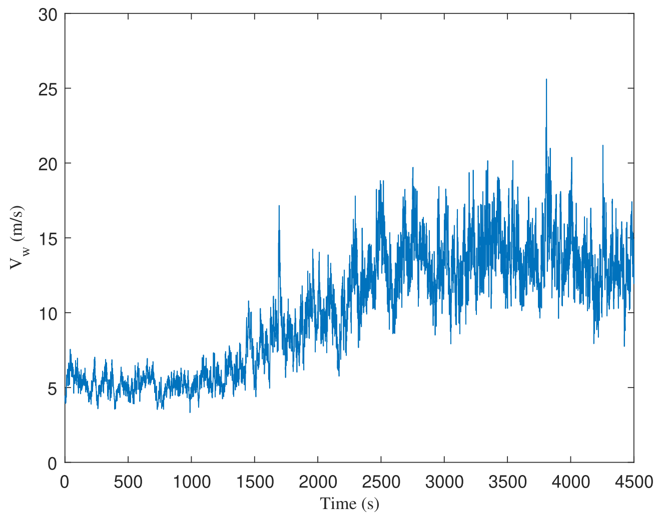


Fig. 5. The variation of wind in the model.

2.2. Blade and pitch system

This subsystem has two main parts: the aerodynamic model and pitch model.

2.2.1. Aerodynamic model

The equation of aero torque applying to the rotor can be seen in 3:

$$\tau_r(t) = \frac{1}{2}\rho\pi R^3 C_q(\lambda(t), \beta(t)) v_w^2, \tag{3}$$

in which, ρ is the air density, R is the radius of the rotor, and C_q is the torque applied to the rotor coefficient which can be modelled based on the pitch angle β and tip-speed-ratio λ given by:

$$\lambda(t) = \frac{R\omega_r(t)}{v_w(t)}, \tag{4}$$

where ω_r is the turbine rotor angular speed.

2.2.2. Pitch system model

The pitch system, here, is modelled based on hydraulic equations. As it is simplified in Ref. [59], closed-loop dynamic of the pitch system is modelled by a second-order transfer function:

$$\frac{\beta(s)}{\beta_r(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega ns + \omega_n^2}, \tag{5}$$

in which, β_r is the pitch reference.

2.3. Drive train subsystem

The drive train subsystem can be modelled as follows:

$$\begin{bmatrix} \dot{\omega}_r(t) \\ \dot{\omega}_g(t) \\ \dot{\theta}_\delta(t) \end{bmatrix} = A_{DT} \begin{bmatrix} \omega_r(t) \\ \omega_g(t) \\ \theta_\delta(t) \end{bmatrix} + B_{DT} \begin{bmatrix} \tau_r(t) \\ \tau_g(t) \end{bmatrix}, \tag{6}$$

in which, θ_δ is the torsion angle of the drive train. The state-space matrices are:

$$A_{DT} = \begin{bmatrix} -\frac{B_{dt}+B_r}{J_r} & \frac{B_{dt}}{N_g J_r} & -\frac{K_{dt}}{J_r} \\ \frac{\eta_{dt} B_{dt}}{N_g J_g} & \frac{-\frac{\eta_{dt} B_{dt}}{N_g^2} - B_g}{J_g} & \frac{\eta_{dt} K_{dt}}{N_g J_g} \\ 1 & -\frac{1}{N_g} & 0 \end{bmatrix}, B_{dt} = \begin{bmatrix} \frac{1}{J_r} & 0 \\ 0 & -\frac{1}{J_g} \\ 0 & 0 \end{bmatrix},$$

in which, B_{DT} is the torsion damping coefficient, B_r and B_g are the rotor and generator external damping, J_r and J_g are the rotor and generator moment of inertia, N_g and η_{DT} are the gear ratio and efficiency of drive train, and K_{DT} is the torsion stiffness.

2.4. Generator and converter subsystem

The generator and converter dynamics can be modelled as a first-order transfer function in wind turbines:

$$\frac{\tau_g(s)}{\tau_{g,r}(s)} = \frac{\alpha_{gc}}{s + \alpha_{gc}} \tag{7}$$

in which, α_{gc} is the parameter of generator and converter model. The output power of the generator is obtained by:

$$P_g = \eta_g \omega_g(t) \tau_g(t) \tag{8}$$

in which, η_g is the generator efficiency.

2.5. Controller

In a typical wind turbine, the control system usually operates in four zones [61]:

- Zone 1: The turbine is not operating.
- Zone 2: The turbine is operating partially until it reaches the desired power.
- Zone 3: The turbine is producing constant power.
- Zone 4: The turbine has to be stopped due to high wind speed.

As the goal of this paper is fault classification, zone 1 and 4 are not discussed any more. In zone 2, the goal of the controller is to reach to the optimal power when the pitch angle is zero or near zero and the tip speed ratio is at optimal constant. Then, the controller is switched to power regulation, which is in zone 3. More details on designing the parameters of this controller can be found in Ref. [59].

2.6. Overall model

By integrating the above subsystems, the overall benchmark can be written in state-space equations [44]:

$$\begin{aligned} \dot{x} &= A(x)x + Bu, \\ y &= Cx, \end{aligned} \tag{9}$$

in which $x = [\omega_r \quad \omega_g \quad \theta_\delta \quad \dot{\beta} \quad \beta \quad \tau_g]^T$ is the state vector and $u = [\tau_{g,r} \quad \beta_r]^T$ is the control input from the pre-designed controller. The system matrices can be illustrated in (10) and (11):

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} & 0 & 0 & 0 \\ A_{21} & A_{22} & A_{23} & 0 & 0 & -\frac{1}{J_g} \\ 1 & -\frac{1}{N_g} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2\zeta\omega n & -\omega_n^2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\alpha_{gc} \end{bmatrix}, \tag{10}$$

$$\begin{aligned} B &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \alpha_{gc} \\ 0 & 0 & 0 & \omega_n & 0 & 0 \end{bmatrix}^T, \\ C &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \end{aligned} \tag{11}$$

in which, $A11 = -\frac{B_{dt}+B_r}{J_r} + \frac{1}{2J_r\lambda^2}\rho\pi R^5 C_q(\lambda,\beta)\omega_r$, $A_{12} = \frac{B_{dt}}{N_g J_r}$, $A_{13} = -\frac{K_{dt}}{J_r}$, $A_{21} = \frac{\eta_{dt}B_{dt}}{N_g J_g}$, $A_{22} = \frac{-\frac{\eta_{dt}B_{dt}}{N_g^2} - B_g}{J_g}$, and $A_{23} = \frac{\eta_{dt}K_{dt}}{N_g J_g}$. The physical meaning and the numerical quantity of each parameter can be found in Table 1 [44,59].

3. Introduction to convolutional neural networks

CNN is one of the deep learning methods, which can be classified as a multi-layer perceptrons ANN. Although this method has been first developed to model the behaviour of visual cortex [62], it has been proved to be a very sturdy tool in both regression and classification problems.

In Fig. 6, a very simple structure of a CNN with three layers is illustrated. Any CNN structure consists of three types of layers: *convolutional layer*, *pooling layer*, and *fully connected layer* [54]. In a convolutional layer, just similar to a neural network, each neuron is connected to a bunch of neurons in previous layer [54]. The controlling of connecting the neurons is with a parameter, called *kernel*. In any convolutional layer, there are numbers of kernels, which act as filters to extract the features of the input signals. There is also a pooling layer in Fig. 6, which is responsible to down-sample the input in order to prevent overfitting and help to reduce the dimension of the space. At the last stage, there is a fully connected neural network (FCNN), which can be used to classify or regression in supervised learning [54].

Table 1
Numerical quantity and their physical meanings of wind turbine parameters.

Param.	Physical Meaning	Value
B_{dt}	Torsion Damping Coefficient	$775.49\frac{Nm.s}{rad}$
B_r	Rotor External Damping	$7.11\frac{Nm.s}{rad}$
J_r	Rotor Moment of Inertia	$55\times10^6\ kg\ m^2$
ρ	Air Density	$1.225\frac{Kg}{m^3}$
R	Rotor Radius	$57.5\ m$
N_g	Gear Ratio	95
K_{dt}	Torsion Stiffness	$2.7\times10^9\frac{Nm}{rad}$
J_g	Generator Moment of Inertia	390 kg m^2
η_{dt}	Efficiency of Drive Train	0.97
B_g	Generator External Damping	$45.6\frac{Nm.s}{rad}$
ζ	Damping Ratio	0.6
ω_n	Natural Frequency	$11.11\frac{rad}{s}$
α_{gc}	Generator and Converter Parameter	$50\frac{rad}{s}$

In the field of designing a CNN structure, some crucial concepts are worth to explain in detail.

Kernel: As mentioned earlier, the kernels behave like matrices of the filters in a convolutional layer. They are applied to the a small portion of the data matrix and produce a convoloved matrix. They are using in order to sharpen the features of the input data, e.g. sharpening the edge in an image. If kernels have the size of $K \times K$, and they are applied to an input data of $N \times N$, the output result will be in size of $((N-K)/S+1) \times ((N-K)/S+1)$ [63], where S is the stride of moving of kernel in input matrix. You can see the visual demonstration of this concept in Fig. 7.

The output result is a matrix, T, with the size of $N \times N$, which its entry of i, j can be calculated as follows:

$$T^{i,j} = \sum_{n=1}^{K} \sum_{m=1}^{K} \left(M_{K\times K} \circ Z_{K\times K}^{i,j}\right)^{n,m}, \tag{12}$$

in which, $M_{K\times K}$ is the kernel matrix with the size of $K \times K$, $\circ$ is Hadamard (element-wise) matrix product, and $Z_{K\times K}^{i,j}$ is a sub-matrix of the input data of Z with the size of $K \times K$, which can be obtained as below:

$$Z_{K\times K}^{i,j} = \begin{bmatrix} Z_{i-h,\,j-h} & \cdots & Z_{i-h,\,j+h} \\ \vdots & \begin{matrix}\cdots \\ Z_{i,j} \\ \cdots\end{matrix} & \vdots \\ Z_{i+h,\,j-h} & \cdots & Z_{i+h,\,j+h} \end{bmatrix} \tag{13}$$

in which, $h = \frac{K-1}{2}$. It is noted that in many cases, the size of the kernel, K, is assumed to be an odd number, to have a symmetry of the output. However, it is not obligatory. In this paper, as it is obvious in Equation (13), it is considered to be odd.

ReLu: This is an activation function for each neuron that can be used in he output of the convolutional layer. ReLu is defined as (14) [63]:

$$R^{i,j} = max(0, T^{i,j}), \tag{14}$$

in which, $R^{i,j}$ is the output of ReLu function, and $T^{i,j}$ is the input of it, which is the output of convolution layer in 12. There are alot of functions that can be used as the activation function, from sigmoid to tangent. However, the most benefit of using ReLu compared to the others is its calculation simplicity.

Max Pooling: As it is mentioned earlier, the aim of using this layer is to downsample the data from its previous layer. The reason for using a downsampling layer is the huge size of the space. The more the size of the space is, the more complicated and higher computational cost. Therefore, it is wise to decrease the size of the matrices. One of these methods is omitting the weaker neurons and keeping the strongest one by using maxpooling layer. In max pooling of $P \times P$, the maximum cell is selected and conveyed to the next layer. This operation is simply showed in Fig. 8 by having a maxpooling layer of 2×2.

Softmax Layer: As the last layer of a typical CNN should be FCNN, an activation function at the end of this layer is necessary. This function can vary based on the goal of the CNN structure, which is regression or classification. Here, in this paper it is calculated as 15. The advantage of using this function is to help the

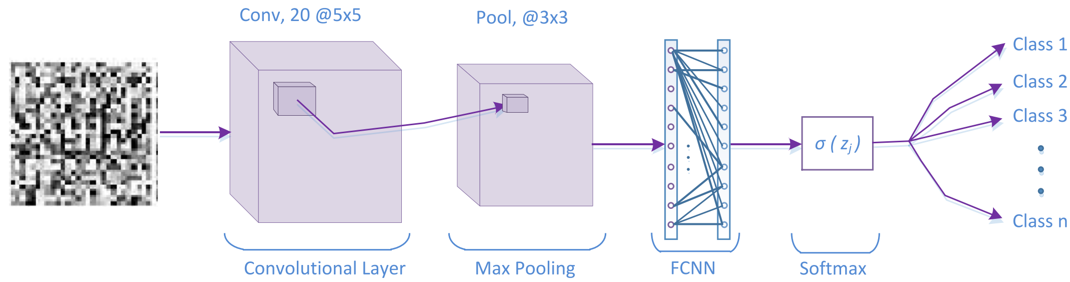


Fig. 6. A typical CNN, consists of three main layers.

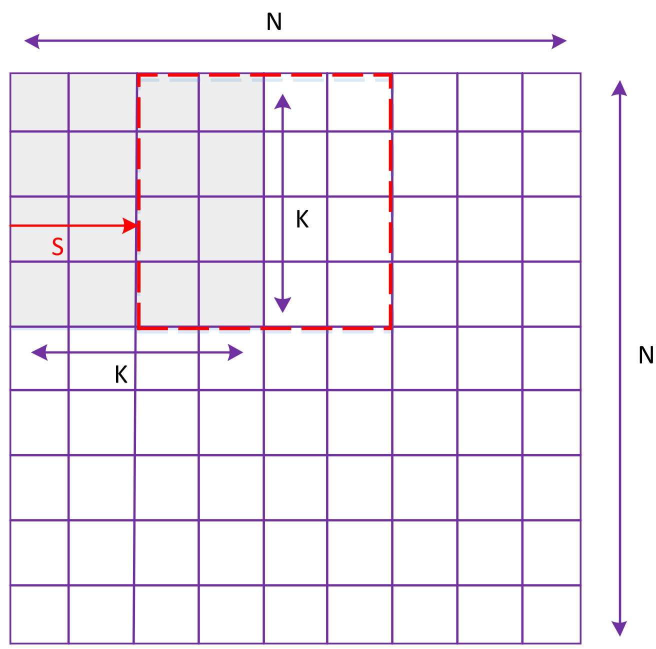


Fig. 7. The illustration of how a kernel would work regarding to an input matrix of *N*.

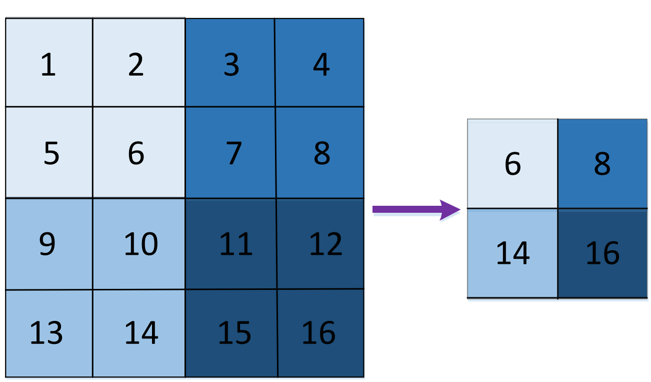


Fig. 8. The process in a max pooling layer.

network, converge to the desired class more smoothly and accurately [63].

$$\sigma(j) = \frac{\exp(z_j)}{\Sigma_{p=1}^{P} exp(z_p))} \tag{15}$$

in which, $\sigma(j)$ is the jth output of this softmax layer, z_j is the output of jth neuron of FCNN, and P is the number of neurons in the last layer of FCNN.

4. Proposed method for fault classification

In this section, the goal is to achieve an approach to diagnose even small anomalies in the output of the wind turbine system. As it is discussed earlier, the challenges, addressed in this paper, are dealing with time-series signals, which are outputs of the benchmark and existing noises and disturbances in the raw data. To achieve this goal in presence of aforementioned difficulties, four CNN structures are proposed in order to classify the data from the benchmark which contains sensor noises and actuator faults on generator torque and pitch angle and sensor faults on the states of the system.

4.1. Preprocess the input data

Understanding the nature of the data is the first step in any deep learning approach. In this case, the outputs of the wind turbine benchmark in section 2, are all time-series signals. One way to deal with this kind of problems, is converting the raw signals into 2-D images [54]. This task can be done by filling row by row of the image matrix with the queue of the data from the time-series data, as it is clearly illustrated in Fig. 9.

As it is shown in Fig. 9, the 2-D image is filled with the time-series signal with M samples and the result is a $n \times n$ matrix, in which $n = \sqrt{M}$.

In this paper, the converted images are presented in grey-scale, which means they are 2-D images, rather than 3-D, which is stated for RGB images. Therefore, they have just one channel. It means that the input images are in $n \times n \times 1$ dimension, in which 1 is the number of the channel.

In addition, it is worth to mention that in CNN approaches, the information is extracted from the image by analysing the relationship between the adjacent pixels. However, in this paper, the adjacent pixels might not show any meaning. Despite of this fact, it is crucial that in every image, the correlation between any two adjacent pixels should be similar, excluding the faulty ones.

4.2. CNN structures

The next step after finishing the preprocessing of the raw data, is designing the CNN structure. Based on the need and the type of the problem, this structure can be varied from a simple one with small number of layers to a very complicated one, with hundreds of layers.

A lot of CNN structures can be found in the literature, which have been effectively used in image processing deep learning, such as AlexNet [64], and GoogLeNet [65]. In this subsection, four CNN structures are investigated to solve the problem of the classification of time-series signals. A comparison will be given in order to

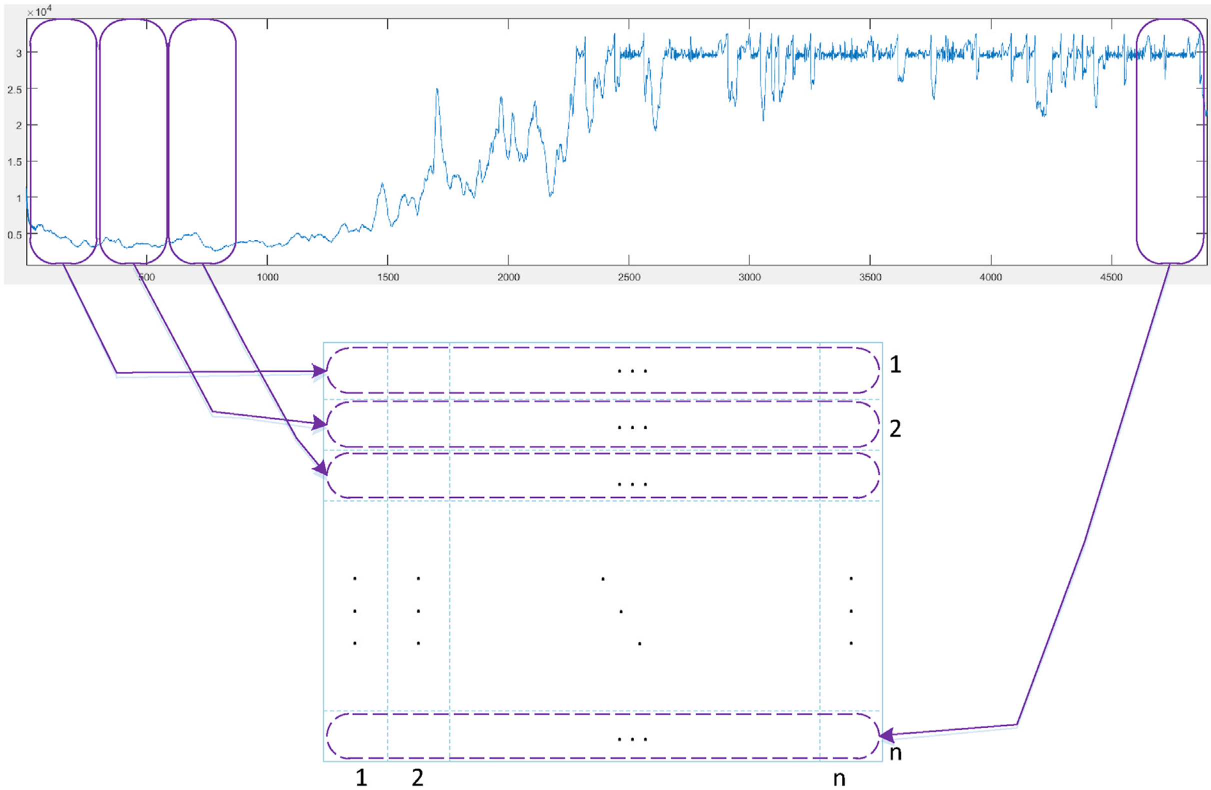


Fig. 9. Filling an image matrix of $n \times n$ with a time-series signal.

evaluate their performances in section 5.

4.2.1. CNN1-20 structure, with one convolutional layer

The first structure is consist of one convolutional layer, a max pooling and an FCNN. The schematic of this structure can be seen in Fig. 10. As it is shown in the left side of this figure, the input images (which are the results of the preprocessing in subsection 4.1) are applied to a convolutional layer with the 20 kernels of 5×5. As it is mentioned earlier, the size of the kernels are considered odd, in all of the structures in this article. The outputs from convolved kernels are routed through a ReLu activation function. The next layer is a max pooling with the pools of 3×3. The responsibility of this layer is to extract the strongest features. The last layer is an FCNN with a softmax function, which is used to classify the data.

4.2.2. CNN2-20 structure, with two convolutional layers

The second structure is similar to the first one, with the difference of one extra convolutional layer. The sequences of this structure can be studied in Fig. 11. The main advantage of adding another convolutional layer is increasing the nonlinearity of the network, so it can be led to increasing the accuracy of classification in more complicated systems. It can also help in the problems with noisy conditions [63].

4.3. CNN3-32 structure, with three convolutional layers

This structure of CNN3-32 is similar to the previous one, with adding another convolutional layer and a very important factor of dropout. The reason for adding another layer is that, by adding a signal, which has high-frequency variation such as pitch angle in the wind turbine benchmark, it is better to increase the nonlinearity of the model in order to cope with this problem. The sequences of this structure can be seen in Fig. 12.

As you can see a Dropout layer is added to this structure. In Dropout layer, some of the neurons are simply *droped out*, since they are very similar to the other ones. This layer is a simple solution to prevent CNN from overfitting [63].

4.4. CNN4-128 structure, with four convolutional layers

Based on the structure of CNN3-32, CNN4-128 structure is introduced by adding a fourth layer of convolution and increasing

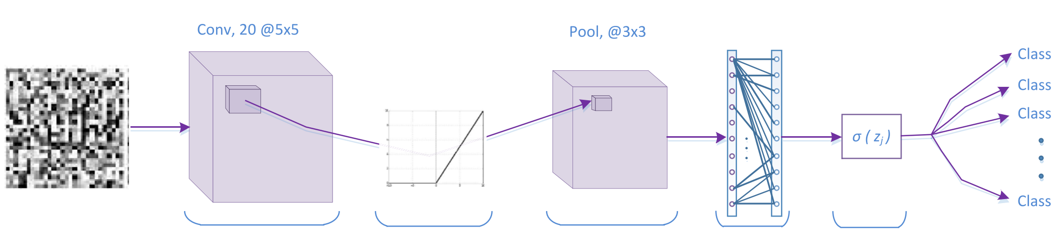


Fig. 10. The schematic of the first proposed CNN with one convolutional layer.

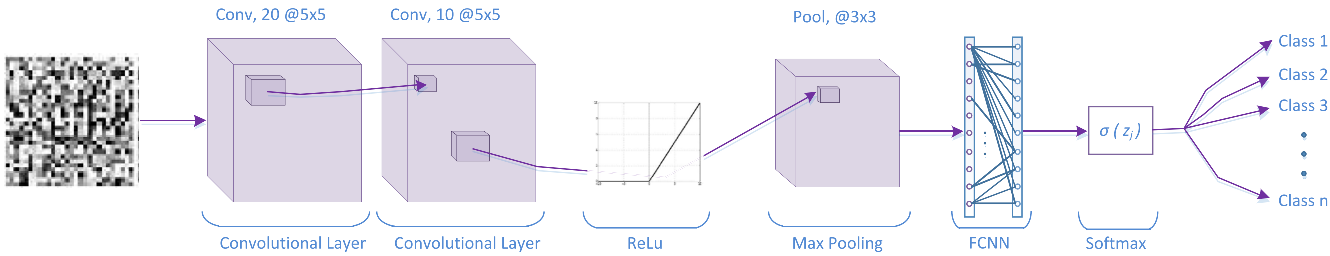


Fig. 11. The schematic of CNN2-20 with two convolutional layers.

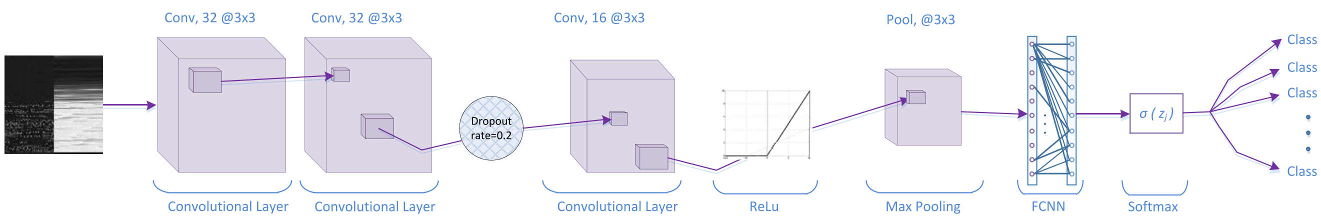


Fig. 12. The schematic of CNN3-32 with three convolutional layers.

the number of kernels in the first layer. As it is mentioned earlier, increasing the nonlinearity of the model, helps the accuracy of training for a dataset with high-frequency variation. The sequences of CNN4-128 can be seen in Fig. 13.

As you can see in this figure, another convolutional layer is added to the structure. Increasing the number of kernels can also help coping with the problem of high-frequency variation of the dataset.

5. Application to 4.8 MW wind turbine benchmark system

In this section, the proposed CNN structures in section 4 are trained with the dataset from the wind turbine benchmark in section 2. The internal dynamics of the system is not very essential in this case, as the proposed approach is fully data-driven. However, knowing the nature of signals can be very helpful to understand how changing the CNN structure can be beneficial in different circumstances, such as noise, high-frequency variation, and environmental disturbances. The trained CNN structures are simulated in three different scenarios to validate the effectiveness of the proposed algorithm. The first one includes fault detection based on just one actuator and in the second one, the approach is developed for two actuators of the wind turbine system. In each scenario, various CNN structures are compared based on the accuracy criteria. In the last scenario, the dataset consists of four different classes of faults, happened on each sensor of the benchmark. Besides, some vital actions that are needed before CNN is also studied in this section.

5.1. Preparing the dataset

The dataset prepared for this simulation, comes from the simulation benchmark, introduced in section 2. The benchmark have been simulated in Matlab, several times to gather the data in different conditions and faults. The outputs of the system encountered with 2–5% effectiveness loss in any of further scenarios. In addition, a Gaussian noise with a variance of 0.3% and mean value of 0 is applied to all the required signals. It is noted that, noisy data would make the classification harder for complex systems. It is also worth to mention that the sampling time is 1 s for all the recordings. One important point in preparing the dataset, is CNN is among in supervised learning. Therefore, the desired output vector should also be prepared, based on the class of each record.

For each scenario, the following step-by-step algorithm is considered:

5.2. Scenario 1: one actuator fault

For the first scenario, the fault situation in Ref. [66] is used, in which there are 2–5% faults on generator torque reference. The dataset in Ref. [66], consist of two classes, healthy and faulty records, are expanded to 4000 records of τ_g. All of them are recorded in times-series format and each of them contains 4900 samples. 2000 records belong to the faulty class and the rest, which are 2000, belong to the healthy one. From each class, 80% of the records are being chosen randomly for the training dataset and 20% go to the testing dataset. As it is essential in any deep learning approach, the records for training and testing sections are completely separable (see Fig. 14).

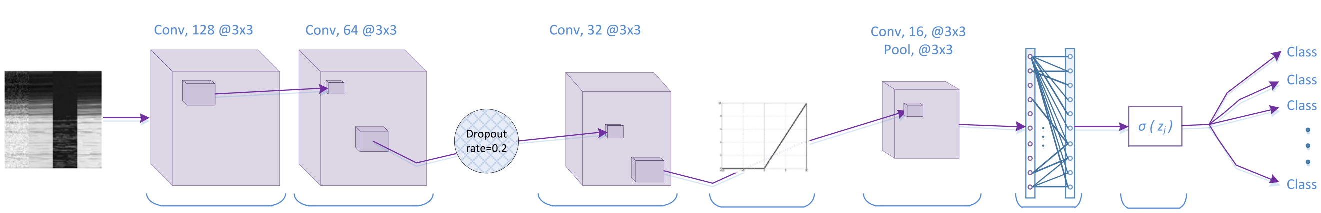


Fig. 13. The schematic of CNN4-128 with four convolutional layers.

Fig. 14. Step-by-step algorithm to simulate each scenario.

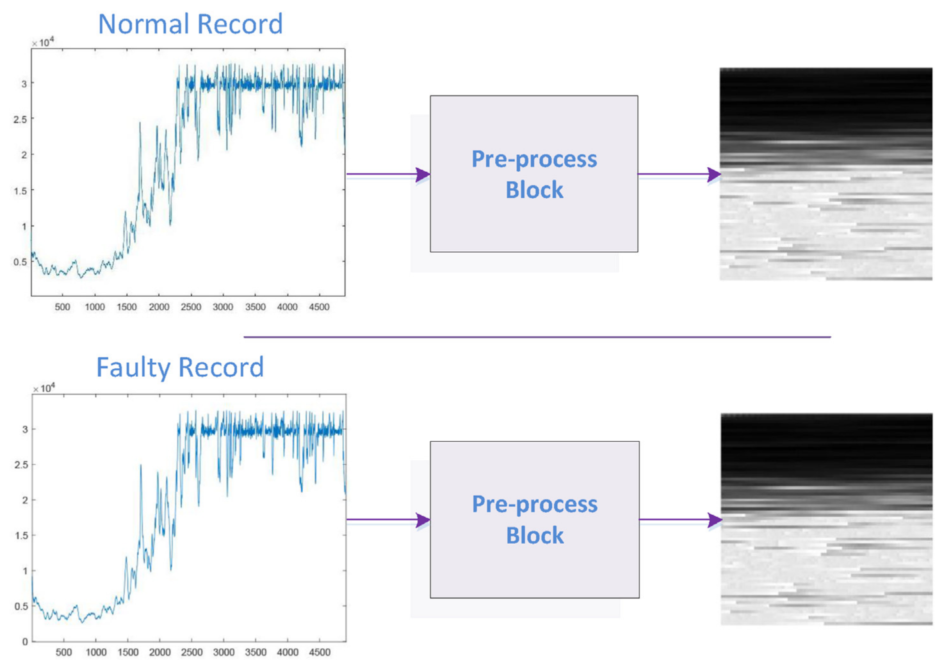


Fig. 15. The conversion of raw data to 2-D grey scale image [66].

As it is discussed in 4.1, a very essential step is to preprocess the data. Each time-series record should be converted to a 2-D image. A random conversion from each class are shown in Fig. 15. As it is obvious, due to the very small fault, the difference between faulty and healthy records are not distinguishable by human eye, neither in the time domain nor in 2-D image. Therefore, it is necessary to develop a method to classify them.

After preparing the dataset, CNN1-20 and CNN2-20 are trained with the input of the image dataset. In addition, each of the structures of CNN1 and CNN2 are trained with different number of kernels, 30 and 45, CNN1-30, CNN2-45, respectively, in order to figure out the effects of increasing the number of the kernels in classification.

In Fig. 16, the classification results for a test dataset of 800 records (20% of the records) are presented. To keep the symmetry of the classes and prevent overfitting of the model, from 800 records, 400 of them are healthy and 400 belong to the faulty class. F_D and H_D are the desired output for faulty and healthy classes and F_R and H_R are the result from the trained model as faulty and healthy. For example in CNN2-20, the CNN structure with two convolutional layers and 20 kernels can estimate just 392 out of 400 faulty records accurately. It classifies the other 8 records, as healthy, which is absolutely incorrect.

The accuracy of different types of CNN structures is compared, in Table 2. The output results verify that increasing the number of kernels and number of convolutional layer can lead to a better accuracy. To see deeply, each kernel is responsible for one feature extractor. Consequently, it is logical to see increasing the features can result in an improvement in the accuracy of the model.

5.3. Scenario 2: two actuators fault

In this scenario, the approach of CNN fault detection in previous subsection is further developed into two actuator faults, the faults on τ_g, and β. The healthy form of the two actuators of the wind turbine benchmark are illustrated in Fig. 17. As can be seen in this figure, the natures of the two signals are completely different. Working with a signal like pitch angle, which has harshly variation, makes the classification much more difficult.

Based on the idea of converting time-series sequences into images in 4.1, we can convert each pair of actuators' signals into one image. The process of converting each pair is illustrated in detail in Fig. 18. Consequently, the image related to the healthy signals of 17 can be seen in Fig. 19.

After preparing the dataset, the proposed structures of CNN2-45 and CNN3-32 are trained and simulated with a dataset consists of 4000 records of two actuator signals, τ_g and β. The reason for choosing just CNN2-45 from previous scenario is that this structure had the best accuracy with one class of faults. Now, we want to see its performance confronting two classes of faults.

To make this scenario more challenging, it is considered to have both faults occurring simultaneously. Therefore, the number of classes are four. The first one is when both of the signals are healthy (H). The second one is when the generator torque is faulty (F1). The third class is when the pitch angle contains a faulty interval (F3). The last class is when both of the signals are faulty (F3). It is noted that, in this scenario, in some of the records, generator torque fault and pitch angle fault happen at the same intervals, and in some others not. Each of the classes contains 500 records. From each

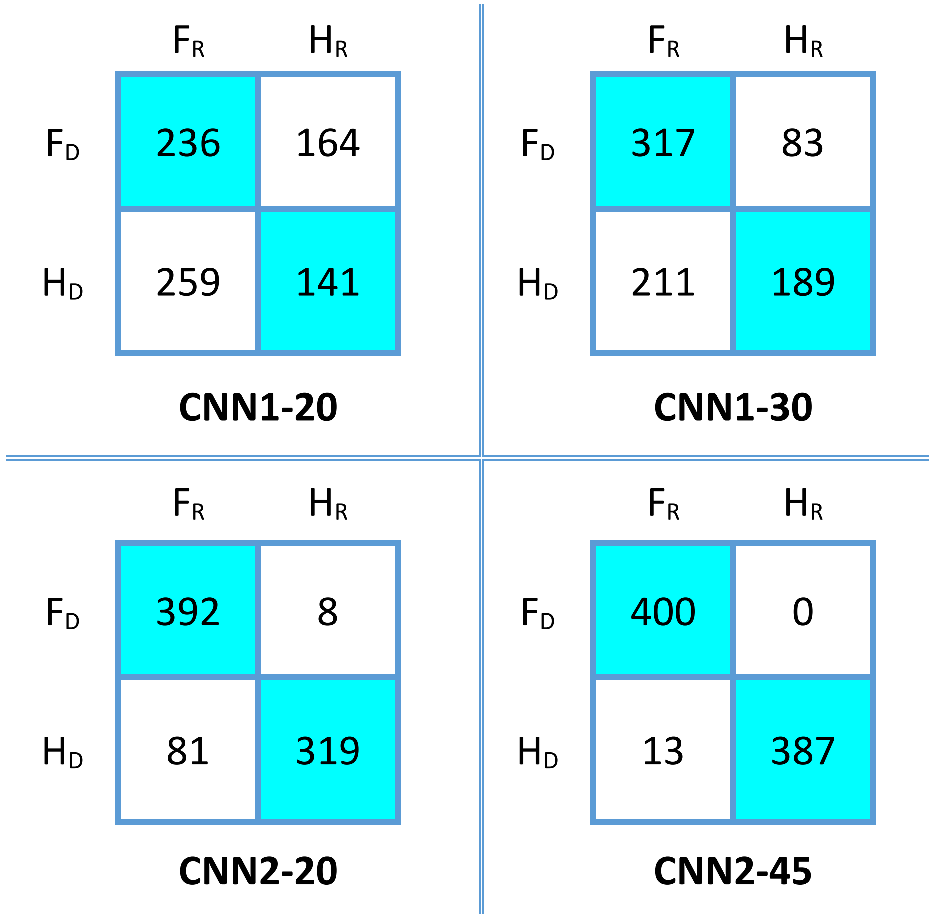


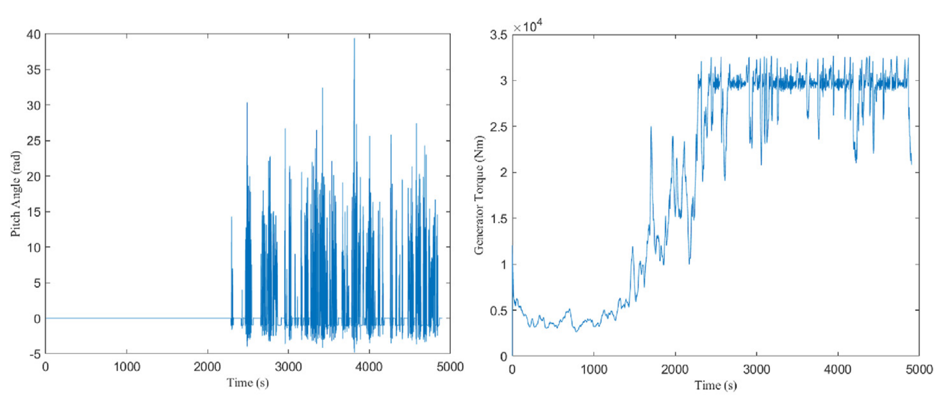
Fig. 16. Classification of testing dataset of scenario 1.

Table 2
Comparison results between different CNN structures in Scenario 1.

Methods	Accuracy (%)
CNN1-20	47.12
CNN1-30	63.25
CNN2-20	88.87
CNN2-45	98.37

category, 80% is being separated randomly for the training dataset and 20% goes to the testing dataset. Again, similar to the previous scenario, the records for training and testing are completely different.

After converting each record to an image, as depicted in 19, two CNN structures of CNN2-45 and CNN3-32 are trained by the prepared dataset. The results of 800 records (20% of the records), which belong to the testing part is brought in 20. The accuracy of



(a) Healthy Pitch Angle Signal.	(b) Healthy Generator Torque Signal.

Fig. 17. Healthy form of the two actuators' signals.

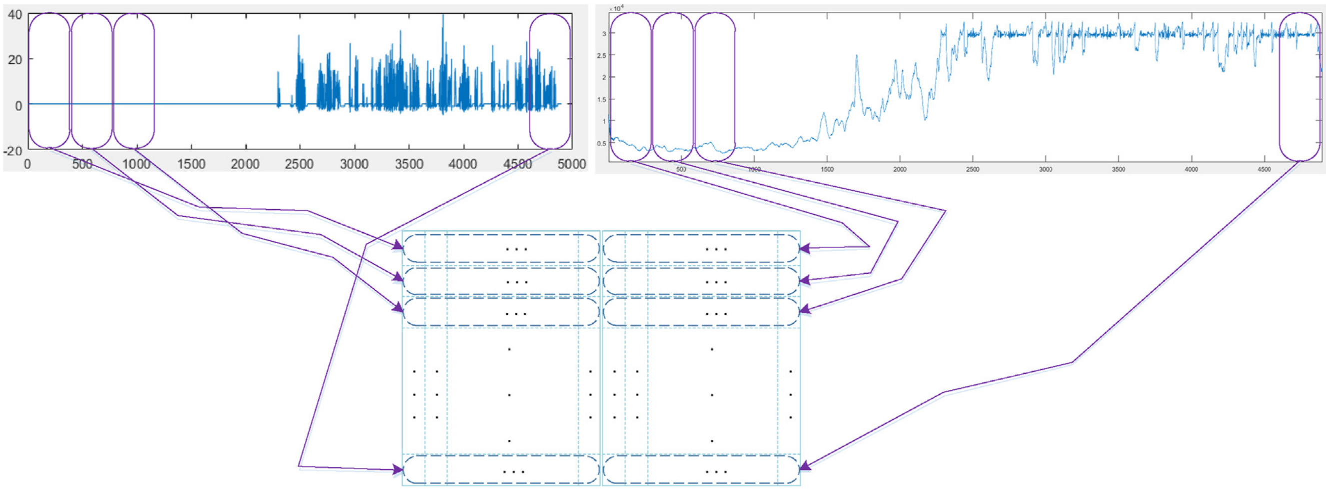


Fig. 18. The process of converting a pair of signals into one image.

Fig. 19. The converted image of a pair of health actuators' signals.

Table 3
Comparison results between different CNN structures in Scenario 2.

Methods	Accuracy (%)
CNN2-45	88.25
CNN3-32	98.87

each structure is also compared in Table 3.

As it is obvious in Table 3, the accuracy of the CNN3-32 is much better than the previous structure. It is noted that although CNN2-45 has a great performance in scenario 1, its performance decreases when having a signal with a high-frequency variation. It is also worth to mention that a more complicated structure and using the Dropout layer can increase the accuracy.

5.4. Scenario 3: four sensors faults

Following the previous subsections, in this subsection, CNN fault detection method is further developed into four sensor faults. The procedure for achieving this goal, is similar to the other two scenarios just by little adjustments. In addition to CNN2-45 and CNN3-

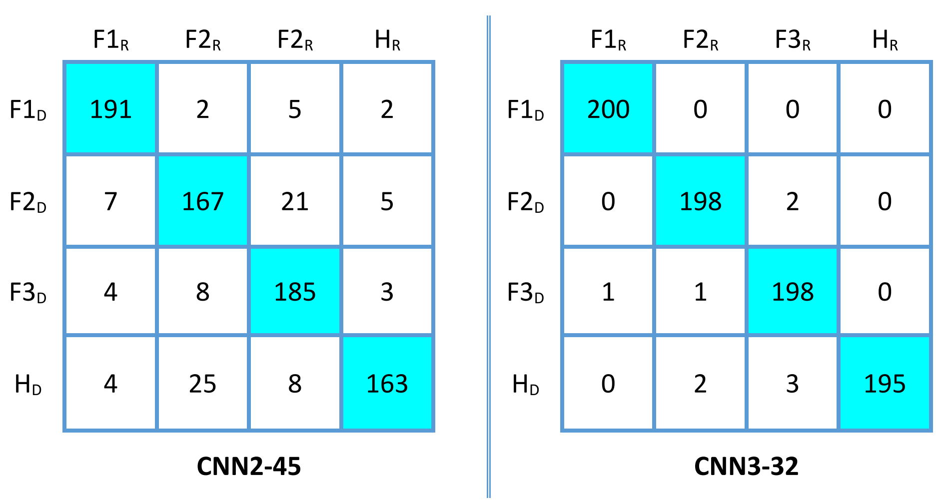
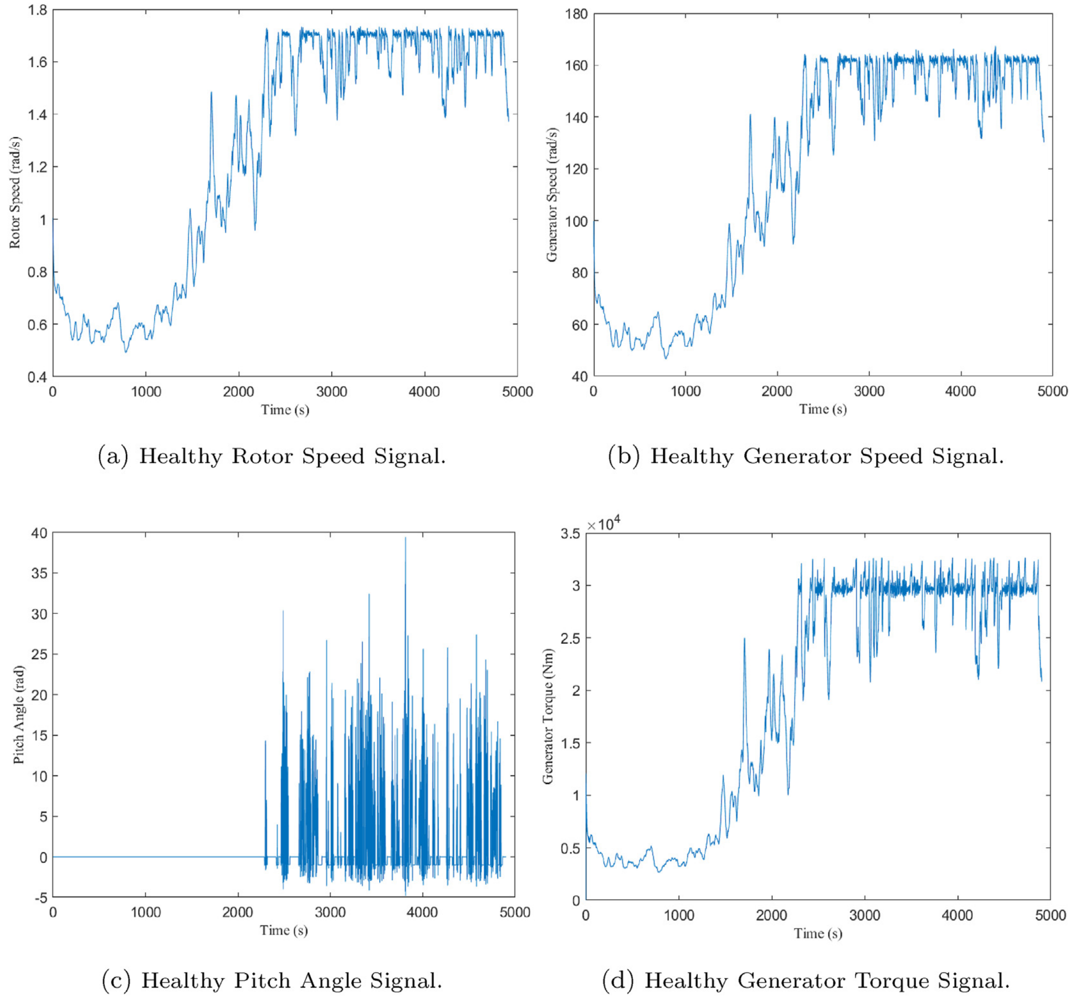


Fig. 20. Classification of testing dataset of scenario 2.



(a) Healthy Rotor Speed Signal.

(b) Healthy Generator Speed Signal.

(c) Healthy Pitch Angle Signal.

(d) Healthy Generator Torque Signal.

Fig. 21. Healthy form of the four sensors' signals.

32, a new and more complicated structure, CNN4-128, is also trained for this dataset (see Fig. 20).

The healthy signals of the four sensors of the wind turbine benchmark can be seen in Fig. 21.

As introduced in 4.1, we converted all of the output data into greyscale images. The format of converting can be seen in Fig. 22. The greyscale result for a sample healthy record is presented in Fig. 23.

The CNN structures are trained and simulated with a dataset consists of 5000 records of four sensor signals, ω_r, ω_g, β, and τ_g. For this scenario, it is considered to have five classes. The first one is when all four signals are healthy (H). The second one is when the rotor speed (ω_r) is faulty (F1). The third class is when the generator speed (ω_g) is faulty (F2). The fourth class contains the fault on pitch angle (β) (F3). And, the last class is when there is a fault on generator torque (τ_g) (F4). It is noted that, in this scenario, it is assumed that there is not simultaneous faults on two or more sensors. Each of the classes contains 1000 records. From each category, 80% is being separated randomly for the training dataset and 20% goes to the testing dataset. Again, similar to the previous scenarios, the records for training and testing are completely different.

After converting each record to an image, as depicted in 23, three CNN structures of CNN2-45, CNN3-32, and CNN4-128 are trained by the prepared dataset. The results of 1000 records (20% of the records), which belong to the testing part is brought in 24. The

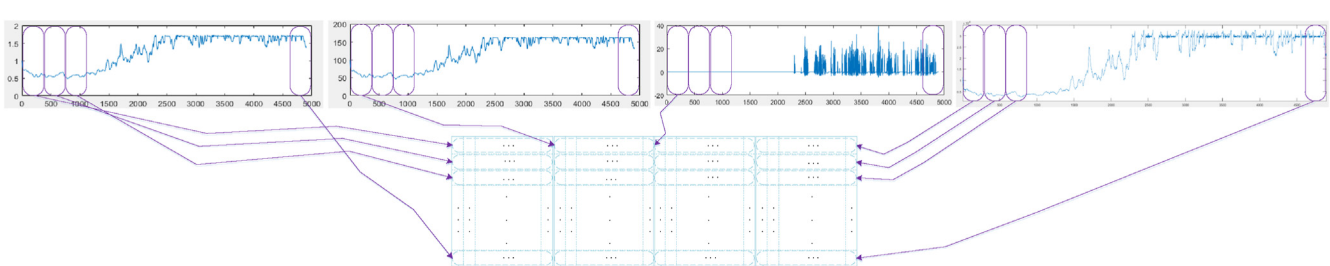


Fig. 22. The process of converting four signals into one image.

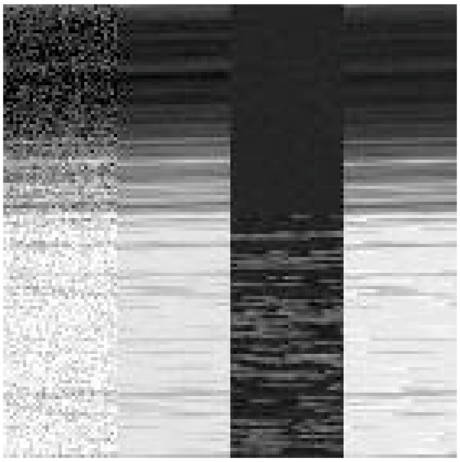


Fig. 23. The converted image of four healthy sensors' signals.

accuracy of each structure is also compared in Table 4.

As it is well-illustrated in Table 4, the accuracy of the CNN4-128 is much better than the previous structures. It is noted that by complicating the input images, the necessity of complex structures is increasing. More convolutional layers, more kernels, and also Dropout layers, are all helpful to reach the higher accuracy in fault detection in complex system with high-variation signals.

Table 4
Comparison results between different CNN structures in Scenario 3.

Methods	Accuracy (%)
CNN2-45	71.4
CNN3-32	91.1
CNN4-128	96.2

6. Conclusion

In this article, four CNN-based structures have been proposed to solve the problem of deep learning fault detection in wind turbines. The first challenge in this field is that working with a time-series signal is so difficult. Therefore, a data preprocessing stage is considered to convert the raw data into 2-D greyscale images. It is a novel approach and very helpful in the case of CNN methods. Then, three different scenarios are investigated. In the first scenario, the dataset consists of the faulty and healthy signals with Gaussian noises. It is vivid that classification would become more difficult, existing noises. However, the simulations showed that adding a convolutional layer to the model can increase the accuracy of the

	$F1_R$	$F2_R$	$F3_R$	$F4_R$	H_R
$F1_D$	105	41	18	23	13
$F2_D$	12	162	9	8	9
$F3_D$	7	15	119	36	31
$F4_D$	25	12	2	154	7
H_D	10	5	8	3	174

CNN2-45

	$F1_R$	$F2_R$	$F3_R$	$F4_R$	H_R
$F1_D$	187	5	0	5	3
$F2_D$	8	183	2	7	0
$F3_D$	2	1	179	2	16
$F4_D$	10	10	0	173	7
H_D	7	1	1	2	189

CNN3-32

	$F1_R$	$F2_R$	$F3_R$	$F4_R$	H_R
$F1_D$	191	2	1	3	3
$F2_D$	0	193	1	0	6
$F3_D$	3	0	188	3	6
$F4_D$	3	0	0	195	2
H_D	1	1	3	0	195

CNN4-128

Fig. 24. Classification of testing dataset of Scenario 3.

training and validation. It is also concluded, by increasing the number of kernels in each structure, the accuracy increased and reached to 98.37% with 45 kernels.

In the second scenario, the proposed approach is developed to cope with two faulty signals, which may have faults in the same intervals. The simulation validates the effectiveness of adding another convolutional layer and also a Dropout layer by having 98.87% accuracy in a dataset of 4000 records.

In the last scenario, a CNN structure was studied to have a fault detection method for four sensor signals. As the simulation accurately showed, adding a layer of convolution had effects on increasing the accuracy to 96.2% in 5000 records.

As it was discussed in the first section, the main goal in this paper is to classify even the smallest anomalies and faults in presences of noises. Based on the simulation and the mathematical structures, it is obvious that proposed algorithm can effectively addressed the challenges such as volatile time-series data in the presence of noises and validate the effectiveness of the novel approach.

As it was stated before, the proposed algorithm in this paper is based on the historical data; therefore it can be used to all kind of wind turbines including onshore and offshore as well as other applications. As a matter of fact, since the offshore wind turbines are dealing with more unmodelled fluctuations and disturbances, it is recommended to benefit from the capabilities of data-driven methods and especially CNN, as one of the most powerful techniques in computational methods.

For the future work, a fault estimation scheme can be proposed based on deep learning techniques to have a regression for future output of each sensor in the occurrence of sensor losses. This idea helps to assure the control system works properly even though the sensors are faulty for a short period of time.

CRediT authorship contribution statement

Reihane Rahimilarki: Methodology, Software, Formal analysis, Validation, Writing – original draft. **Zhiwei Gao:** Conceptualization, Supervision, Writing – review & editing, Funding acquisition. **Nanlin Jin:** Software, Supervision. **Aihua Zhang:** Software, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to thank the research support from the E & E faculty at University of Northumbria (UK), the National Nature Science Foundation of China (NNSFC) under grant 61673074, and the Alexander von Humboldt Foundation under the grant GRO/1117303 STP.

References

[1] E. I. S, Department for Business, Digest of united kingdom Energy Statistics, National Statistics Publication, 2020.
[2] B. Hahn, M. Durstewitz, K. Rohrig, Reliability of wind turbines, in: Wind Energy, Springer, 2007, pp. 329–332.
[3] Z. Gao, C. Cecati, S.X. Ding, A survey of fault diagnosis and fault-tolerant techniques part i: fault diagnosis with model-based and signal based approaches, IEEE Trans. Ind. Electron. 62 (6) (Jun. 2015) 3757–3767.
[4] Z. Gao, S. Sheng, Real-time monitoring, prognosis, and resilient control for wind turbine systems, Renew. Energy 116 (2018) 1–4.
[5] J. Su, W.-H. Chen, Model-based fault diagnosis system verification using reachability analysis, IEEE Trans. Syst. Man Cybernet.: Systems 49 (4) (2017) 742–751.
[6] R. Rahimilarki, Z. Gao, Grey-box model identification and fault detection of wind turbines using artificial neural networks, in: IEEE 16th International Conference on Industrial Informatics (INDIN), IEEE, 2018, pp. 647–652.
[7] Y. Zhao, Y. Liu, R. Wang, Fuzzy scalar quantisation based on hidden markov model and application in fault diagnosis of wind turbine, J. Eng. 2017 (14) (May 2017) 2685–2689.
[8] W. Lu, B. Liang, Y. Cheng, D. Meng, J. Yang, T. Zhang, Deep model based domain adaptation for fault diagnosis, IEEE Trans. Ind. Electron. 64 (3) (Mar. 2017) 2296–2305.
[9] H. Sanchez, T. Escobet, V. Puig, P.F. Odgaard, Fault diagnosis of an advanced wind turbine benchmark using interval-based arrs and observers, IEEE Trans. Ind. Electron. 62 (6) (2015) 3783–3793.
[10] B. Shafai, C. Pi, S. Nork, Simultaneous disturbance attenuation and fault detection using proportional integral observers, in: American Control Conference (IEEE Cat. No. CH37301), vol. 2, IEEE, 2002, pp. 1647–1649.
[11] W. Tang, Z. Wang, Y. Shen, Fault detection and isolation for discrete-time descriptor systems based on h/l-inf observer and zonotopic residual evaluation, Int. J. Control 93 (8) (2020) 1867–1878.
[12] Y. Liu, R. Ferrari, P. Wu, X. Jiang, S. Li, J.-W. van Wingerden, Fault diagnosis of the 10mw floating offshore wind turbine benchmark: a mixed model and signal-based approach, Renew. Energy 164 (2021) 391–406.
[13] H. Shao, Z. Gao, X. Liu, K. Busawon, Parameter-varying modelling and fault reconstruction for wind turbine systems, Renew. Energy 116 (2018) 145–152.
[14] H. Chen, S. Lu, Fault diagnosis digital method for power transistors in power converters of switched reluctance motors, IEEE Trans. Ind. Electron. 60 (2) (Feb. 2013) 749–763.
[15] N. Wassinger, E. Penovi, R.G. Retegui, S. Maestri, Open-circuit fault identification method for interleaved converters based on time-domain analysis of the state observer residual, IEEE Trans. Power Electron. 34 (4) (2018) 3740–3749.
[16] H. Chen, S. Lu, Fault diagnosis digital method for power transistors in power converters of switched reluctance motors, IEEE Trans. Ind. Electron. 60 (2) (2012) 749–763.
[17] N. Su, X. Li, Q. Zhang, Fault diagnosis of rotating machinery based on wavelet domain denoising and metric distance, IEEE Access 7 (2019) 73262–73270.
[18] J. Burriel-Valencia, R. Puche-Panadero, J. Martinez-Roman, A. Sapena-Bano, M. Pineda-Sanchez, Short-frequency fourier transform for fault diagnosis of induction machines working in transient regime, IEEE Trans. Instrum. Meas. 66 (3) (2017) 432–440.
[19] K. Satpathi, Y.M. Yeap, A. Ukil, N. Geddada, Short-time fourier transform based transient analysis of vsc interfaced point-to-point dc system, IEEE Trans. Ind. Electron. 65 (5) (2017) 4080–4091.
[20] J. Wang, F. Cheng, W. Qiao, L. Qu, Multiscale filtering reconstruction for wind turbine gearbox fault diagnosis under varying-speed and noisy conditions, IEEE Trans. Ind. Electron. 65 (5) (May 2018) 4268–4278.
[21] Y. Fu, Z. Gao, Y. Liu, A. Zhang, X. Yin, Actuator and sensor fault classification for wind turbine systems based on fast fourier transform and uncorrelated multilinear principal component analysis techniques, Processes 8 (9) (2020) 1066.
[22] Y. Fu, Y. Liu, Z. Gao, Fault classification in wind turbines using principal component analysis technique, in: IEEE 17th International Conference on Industrial Informatics (INDIN), vol. 1, IEEE, 2019, pp. 1303–1308.
[23] I. Zamudio-Ramirez, J. A. Antonino-Daviu, R. A. Osornio-Rios, R. de Jesus Romero-Troncoso, H. Razik, Detection of winding asymmetries in wound-rotor induction motors via transient analysis of the external magnetic field, IEEE Trans. Ind. Electron..
[24] R. Rahimilarki, Z. Gao, A. Zhang, R. Binns, Robust neural network fault estimation approach for nonlinear dynamic systems with applications to wind turbine systems, IEEE Trans. Ind. Inf. 15 (12) (2019) 6302–6312.
[25] R. Rahimilarki, Z. Gao, N. Jin, R. Binns, A. Zhang, Data-driven sensor fault estimation for the wind turbine systems, in: IEEE 29th International Symposium on Industrial Electronics (ISIE), IEEE, 2020, pp. 1211–1216.
[26] Z. Gao, C. Cecati, S.X. Ding, A survey of fault diagnosis and fault-tolerant techniques part ii: fault diagnosis with knowledge-based and hybrid/active approaches, IEEE Trans. Ind. Electron. 62 (6) (Jun. 2015) 3768–3774.
[27] Y. Jiang, S. Yin, O. Kaynak, Data-driven monitoring and safety control of industrial cyber-physical systems: basics and beyond, IEEE Access 6 (Aug. 2018) 47374–47384.
[28] S. Simani, S. Farsoni, P. Castaldi, Fault diagnosis of a wind turbine benchmark via identified fuzzy models, IEEE Trans. Ind. Electron. 62 (6) (Jun. 2015) 3775–3782.
[29] S. Yin, C. Yang, J. Zhang, Y. Jiang, A data-driven learning approach for nonlinear process monitoring based on available sensing measurements, IEEE Trans. Ind. Electron. 64 (1) (Jan. 2017) 643–653.
[30] B. Gou, Y. Xu, Y. Xia, G. Wilson, S. Liu, An intelligent time-adaptive data-driven method for sensor fault diagnosis in induction motor drive system, IEEE Trans. Ind. Electron..
[31] B. Xu, X. Yin, X. Yin, Y. Wang, S. Pang, Fault diagnosis of power systems based on temporal constrained fuzzy petri nets, IEEE Access 7 (2019) 101895–101904.
[32] H. Yan, Y. Xu, F. Cai, H. Zhang, W. Zhao, C. Gerada, Pwm-vsi fault diagnosis for a pmsm drive based on the fuzzy logic approach, IEEE Trans. Power Electron. 34 (1) (2018) 759–768.
[33] H. Malik, S. Mishra, Proximal support vector machine (psvm) based imbalance

fault diagnosis of wind turbine using generator current signals, Energy Proc. 90 (2016) 593–603.
[34] Q. Shi, H. Zhang, Fault diagnosis of an autonomous vehicle with an improved svm algorithm subject to unbalanced datasets, IEEE Trans. Ind. Electron..
[35] V. Pashazadeh, F.R. Salmasi, B.N. Araabi, Data driven sensor and actuator fault detection and isolation in wind turbine using classifier fusion, Renew. Energy 116 (2018) 99–106.
[36] R. Yang, M. Huang, Q. Lu, M. Zhong, Rotating machinery fault diagnosis using long-short-term memory recurrent neural network, IFAC-PapersOnLine 51 (24) (2018) 228–232.
[37] L. Cao, J. Zhang, J. Wang, Z. Qian, Intelligent fault diagnosis of wind turbine gearbox based on long short-term memory networks, in: IEEE 28th International Symposium on Industrial Electronics (ISIE), IEEE, 2019, pp. 890–895.
[38] G. Iannace, G. Ciaburro, A. Trematerra, Fault diagnosis for uav blades using artificial neural network, Robotics 8 (3) (2019) 59.
[39] W. Huang, J. Cheng, Y. Yang, G. Guo, An improved deep convolutional neural network with multi-scale information for bearing fault diagnosis, Neurocomputing 359 (2019) 77–92.
[40] Y. Chang, J. Chen, C. Qu, T. Pan, Intelligent fault diagnosis of wind turbines via a deep learning network using parallel convolution layers with multi-scale kernels, Renew. Energy 153 (2020) 205–213.
[41] M. Li, D. Yu, Z. Chen, K. Xiahou, T. Ji, Q. Wu, A data-driven residual-based method for fault diagnosis and isolation in wind turbines, IEEE Trans. Sustain. Energy 10 (2) (2018) 895–904.
[42] J. Lei, C. Liu, D. Jiang, Fault diagnosis of wind turbine based on long short-term memory networks, Renew. Energy 133 (2019) 422–432.
[43] X. Liu, Z. Gao, A. Zhang, Observer-based fault estimation and tolerant control for stochastic takagi–sugeno fuzzy systems with brownian parameter perturbations, Automatica 102 (2019) 137–149.
[44] X. Liu, Z. Gao, M.Z. Chen, Takagi–sugeno fuzzy model based fault estimation and signal compensation with application to wind turbines, IEEE Trans. Ind. Electron. 64 (7) (Jul. 2017) 5678–5689.
[45] J. Hang, J. Zhang, M. Cheng, Application of multi-class fuzzy support vector machine classifier for fault diagnosis of wind turbine, Fuzzy Set Syst. 297 (2016) 128–140.
[46] M. Kordestani, M. Rezamand, R. Carriveau, D.S. Ting, M. Saif, Failure diagnosis of wind turbine bearing using feature extraction and a neuro-fuzzy inference system (anfis), in: International Work-Conference on Artificial Neural Networks, Springer, 2019, pp. 545–556.
[47] S. Cho, M. Choi, Z. Gao, T. Moan, Fault detection and diagnosis of a blade pitch system in a floating wind turbine based on kalman filters and artificial neural networks, Renew. Energy 169 (2021) 1–13.
[48] S. Afrasiabi, M. Afrasiabi, B. Parang, M. Mohammadi, M.M. Arefi, M. Rastegar, Wind turbine fault diagnosis with generative-temporal convolutional neural network, in: IEEE International Conference on Environment and Electrical Engineering and IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe), IEEE, 2019, pp. 1–5.
[49] G. Liang, H. Hong, W. Xie, L. Zheng, Combining convolutional neural network with recursive neural network for blood cell image classification, IEEE Access 6 (2018) 36188–36197.
[50] W. Gong, H. Chen, Z. Zhang, M. Zhang, R. Wang, C. Guan, Q. Wang, A novel deep learning method for intelligent fault diagnosis of rotating machinery based on improved cnn-svm and multichannel data fusion, Sensors 19 (7) (2019) 1693.
[51] D. Wang, Q. Guo, Y. Song, S. Gao, Y. Li, Application of multiscale learning neural network based on cnn in bearing fault diagnosis, J. Signal Process. Syst. 91 (10) (2019) 1205–1217.
[52] M. Xia, G. Han, Y. Zhang, J. Wan, et al., Intelligent fault diagnosis of rotor-bearing system under varying working conditions with modified transfer cnn and thermal images, IEEE Trans. Ind. Inf..
[53] L. Eren, T. Ince, S. Kiranyaz, A generic intelligent bearing fault diagnosis system using compact adaptive 1d cnn classifier, J. Signal Process. Syst. 91 (2) (2019) 179–189.
[54] L. Wen, X. Li, L. Gao, Y. Zhang, A new convolutional neural network-based data-driven fault diagnosis method, IEEE Trans. Ind. Electron. 65 (7) (2018) 5990–5998.
[55] E. Hau, Wind Turbines: Fundamentals, Technologies, Application, Economics, Springer Science & Business Media, 2013.
[56] B. Chen, reportAutomated On-Line Fault Prognosis for Wind Turbine Monitoring Using Scada Data, PhD Dissertation, Durham University.
[57] F.D. Bianchi, H. De Battista, R.J. Mantz, Wind Turbine Control Systems: Principles, Modelling and Gain Scheduling Design, Springer Science & Business Media, 2006.
[58] Z. Gao, X. Liu, An overview on fault diagnosis, prognosis and resilient control for wind turbine systems, Processes 9 (2) (2021) 300.
[59] P. Odgaard, J. Stoustrup, M. Kinnaert, Fault tolerant control of wind turbines: a benchmark model, IEEE Trans. Contr. Syst. Technol. 21 (4) (2013) 1168–1182.
[60] H. Sanchez, T. Escobet, V. Puig, P. Odgaard, Fault diagnosis of advanced wind turbine benchmark using interval-based arrs and observers, IEEE Trans. Ind. Electron. 62 (6) (2015) 3783–3793.
[61] R. Gao, Z. Gao, Pitch control for wind turbine systems using optimization, estimation and compensation, Renew. Energy 91 (2016) 501–515.
[62] D.H. Hubel, T.N. Wiesel, Receptive fields of single neurones in the cat's striate cortex, J. Physiol. 148 (3) (1959) 574–591.
[63] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT press, 2016.
[64] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[65] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.
[66] R. Rahimilarki, Z. Gao, N. Jin, A. Zhang, Time-series Deep Learning Fault Detection with the Application of Wind Turbine Benchmark, vol. 1, IEEE, 2019, pp. 1337–1342.