A data-driven fault detection and diagnosis method via just-in-time learning for unmanned ground vehicles

Changxin Zhang, Xin Xu, Xinglong Zhang, Xing Zhou, Yang Lu & Yichuan Zhang

Published online: 17 Nov 2022.

Submit your article to this journal ☑

Article views: 1012

View related articles ☑

View Crossmark data ☑

Taylor & Francis
Taylor & Francis Group

REGULAR PAPER

🔓 OPEN ACCESS   Check for updates

A data-driven fault detection and diagnosis method via just-in-time learning for unmanned ground vehicles

Changxin Zhang, Xin Xu 🔟, Xinglong Zhang, Xing Zhou, Yang Lu and Yichuan Zhang

College of Intelligence Science and Technology, National University of Defense Technology, Changsha, People's Republic of China

ABSTRACT

Fault detection and diagnosis technologies for unmanned ground vehicles are important for ensuring safety and reliability. Due to the complexity and uncertainty of unmanned ground vehicles, it is challenging to realize accurate and fast fault detection and diagnosis. For the purpose of solving the data-driven fault detection and diagnosis problems of unmanned ground vehicles, improving the diagnostic accuracy and shortening the training time, a novel fault detection and diagnosis method is proposed, which is called JITGP-ELM. In the proposed method, a model estimator based on the just-in-time Gaussian process is designed for the online residual generation to cope with the dynamics and nonlinearity of systems. A fault classifier using Extreme Learning Machine is designed for fault identification with residuals extracted by the just-in-time Gaussian process modelling. The proposed method has online adaptability, noise-resistant ability, and high generalization. A field test on a real unmanned ground vehicle's steering-by-wire system demonstrates the effectiveness of the proposed method.

1. Introduction

Due to the development of artificial intelligence and computing science in recent years, the level of autonomous driving of unmanned ground vehicles (UGVs) has been significantly promoted, followed by the increasing complexity of their systems. This advancement brings tremendous challenges to system safety and reliability. For highly intelligent UGVs, the consequences caused by their faults could be significant because they may even threaten passengers' lives. Therefore, fault detection and diagnosis (FDD) strategies for UGVs are getting more and more attention in the academic area and industrial applications to ensure reliability [1–5].

In the past decades, model-based and data-driven fault detection and diagnosis have gained increasing attention parallelly [6]. Model-based approaches have been prevalent in the fault detection and diagnosis field and have accomplished many achievements [3,7–9]. The model-based approaches construct residual signals to reflect the inconsistencies between the expected and actual behaviours by a precise physical model and measurements of the system [3]. In [10], a model-based method is proposed for fault detection of vehicle suspension and hydraulic brake systems, and a rule-based approach is used to judge the fault from model residuals. In [11,12], the model-based sensor fault detection and isolation methods for UGVs are researched from theoretical and experimental aspects, respectively.

Şahin et al. [13] propose a fault detection and isolation method for the wheel slippage and actuator faults of a holonomic mobile robot with the robot model. A significant advantage of these methods is that they can provide a description of the dynamic behaviour of the system and a physical understanding of the system, and residuals can eliminate the nonlinearity and dynamics of the system. However, the model-based methods rely on the accurate physical model of systems, which is often difficult to obtain in practice.

Data-driven methods are applicable in practice without prior knowledge of systems [14–16]. Data-driven fault detection and diagnosis approaches mainly consist of multivariate statistical analysis, machine learning (ML), and signal processing. Multivariate statistical methods are popular for effectively analyzing the available signal in UGVs, such as principal component analysis (PCA) [17], partial least squares (PLSs) [18], and independent component analysis [19]. Still, the threshold-based methods will be difficult to deal with multi-fault diagnosis problems that are commonly encountered because these methods can only provide binary classification solutions. The ML-based diagnosis methods for fault classification are proposed to deal with multi-fault classification, such as FDD based on Back Propagation (BP) neural network [20], Support Vector Machine (SVM) [21], and Extreme Learning Machine (ELM) [22]. For UGVs, Lei et al. [23] select the extremely randomized trees model as the best

---

classification model for refrigerant leak detection in heat pump systems of electric UGVs. In [24], a CNN-based fault detection and classification model is used during the V-cycle development process and applied to the gasoline engine system. Similarly, Wallace et al. [25] use a long short-term memory (LSTM) neural network to learn the fault classification boundaries for UGVs. The fault classifier methods based on ML can learn from historical data of the system under normal and various fault statuses, which take classification accuracy as the learning target and have had a wide range of applications. However, the generalization performance of ML-based methods is a concern because the classification accuracy is closely related to the completeness and representativeness of training data. Furthermore, the selection of fault features is also challenging for learning efficiency.

Because model-based and data-driven approaches have their own characteristics, the researchers proposed hybrid methods to improve the fault diagnosis algorithm and achieved impressive results. Tidriri et al. [26] examine the characteristics of both model-based and data-driven approaches, as well as existing hybrid research, and point to the potential of the latter. Besides, Khorasgani et al. [27] demonstrate a unified framework of data-driven and model-based diagnosis methods, which are elaborated on three cascaded tasks of data acquisition, feature extraction and fault diagnosis. More specifically, Jung et al. [28] propose a hybrid diagnostic system design that combines model-based residuals with incremental exception classifiers to identify unknown and multiple faults. Furthermore, Jung et al. [29] design a residual selection algorithm in a hybrid approach to improve the performance of fault detection and isolation. For UGVs, a nominal model is established in [6], and the fault analysis for automotive antilock braking systems is carried out by combining the support vector machine. Wolf et al. [30] use the dynamic Bayesian network with residual generation to combine with the knowledge-based and model-based approaches to realize fault detection, diagnosis and prognosis for autonomous vehicles. From existing works, the hybrid of model-based residual extraction and the use of residual as fault features to be the inputs of classifiers is a practical approach to improve the performance of fault detection and diagnosis. However, in these approaches, the prediction model is assumed as prior knowledge and will not be updated in the process. Therefore, in the presence of environmental changes or interference, the performance of residual extraction may deteriorate, which motivated our study.

Based on the above analysis, we propose a JITGP-ELM method for multi-fault detection and diagnosis for unmanned ground vehicles in this paper. In the proposed method, a residuals generator based on a just-in-time Gaussian process regression (JITGP) is designed to adapt to the dynamics and nonlinearity.
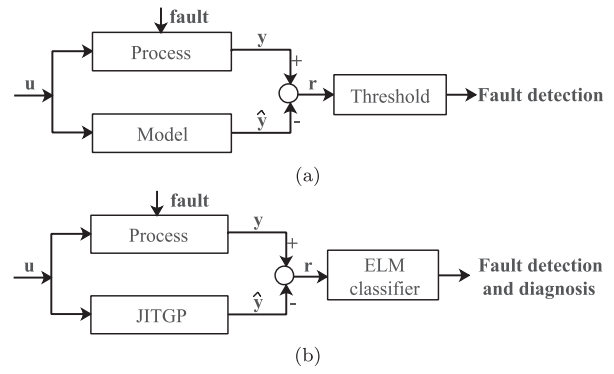

Figure 1. Comparison between the model-based and the proposed methods for fault detection and diagnosis. (a) Model-based method and (b) JITGP-ELM.

Then, a fault classifier using Extreme Learning Machine is designed for fault diagnosis with residuals extracted by just-in-time Gaussian process modelling. The proposed approach adopts JITGP modelling, inheriting its advantages of online adaptation, noise immunity, and low computational cost. Furthermore, the ELM network has advantages in learning speed and generalization ability. In the fault detection and diagnosis problem on the steering-by-wire system of a UGV, the proposed method effectively provides an application solution. The comparison between the conventional methods and the proposed method is shown in Figure 1.

The main contribution of the work includes the following two aspects. First, a fault detection and diagnosis method named JITGP-ELM is proposed to realize adaptive residual extraction and high generalization fault status classification in UGVs. Compared with previous works [6,14,30], our approach is designed for realizing adaptive residual generation, and crucially, JITGP is utilized to improve the flexibility to environmental changes or interference with noise immunity. Second, experiments have been performed using our approach for solving a real-world fault detection and diagnosis problem under three fault statuses on a UGV steering-by-wire system. The experiment results show that the approach is practical and effective. Also, the proposed method is data-driven and does not rely on the physical model of the system. So another advantage of our work is that it could be extended to fault detection and diagnosis problems of other systems like ships or aircraft.

The organizational structure of this paper is as follows. In Section 3, a fault detection and diagnosis problem formulation for a dynamic nonlinear system and the GP model for system modelling are introduced. In Section 4, the JITGP-ELM algorithm is presented. In Section 5, an application to a UGV and the experimental results are provided to illustrate the effectiveness of the proposed JITGP-ELM algorithm for fault detection and diagnosis. Finally, conclusions are drawn in Section 6.

2. Related works

Within fault detection and diagnosis domains, there have been many previous works on UGVs [1–5]. The previous work is mainly divided into the model-based [10–13] and data-driven [17–19,23–25] methods, as well as their hybrid [6,30]. A heuristic idea of the hybrid method is to generate residuals through a prediction model and then diagnose faults with the residuals [26–29]. Our work inherits this idea with data-driven adaptive modelling and fault classification based on machine learning methods.

To realize the above idea, modelling methods mainly include physical formulation and ML-based modelling in the previous works. In the field of UGV fault diagnosis, the residual generators are mainly realized by physical modelling [6,10–12] and offline data training [30,31] with the inability to adapt to changes in vehicle dynamics and environment, which is inevitable in the working process of UGVs. Different from previous works, our work considers the need for such adaptability. As a popular local learning method, JITL establishes local predictive models for each test sample based on similar samples selected from the dataset [32,33], which has the advantage of low computational cost. Nevertheless, conventional JITL with linear local models could be sensitive to noise in the small sample case, leading to a conflict between speediness and accuracy in practice. The just-in-time Gaussian process (JITGP) algorithm, whose local models are developed by the Gaussian process (GP) model instead of linear models, was used in [34–36] to predict industrial processes with noise immunity. Furthermore, the GP model can provide the prediction variance reflecting probabilistic information.

In the second half of the idea, a fault classifier based on machine learning is considered for multi-fault diagnosis. There are many standard classifiers, like BP [20], SVM [21], CNN [24], LSTM [25] and other machine learning methods [6,23,30] for multi-fault detection and diagnosis problems. Among them, the ELM was proposed in [37] with a structure of single-hidden layer feedforward neural networks (SLFNs), which is believed to have the ability to provide high generalization and training speed. Therefore, the JITGP-ELM is proposed in this paper to realize adaptive residual extraction and high generalization fault status classification for fault detection and diagnosis problems of UGVs.

3. Problem formulation and research backgrounds

This section will introduce a fault detection and diagnosis problem formulation for a time-delay dynamic nonlinear system. Then, a brief preliminary on Gaussian process regression for the process estimation with noise will be provided.

3.1. FDD problem for a dynamic nonlinear system

When a fault occurs in a system, the manifestation of the system will be changed by the fault. The changes could include the mapping between input and output, which can be indicated by adding a fault signal in the mapping function. The fault signal varies depending on the fault locations, types, or other properties, which can be described as fault statuses. To model a multi-input multi-output (MIMO) dynamic nonlinear system with a fault, the output is as follows:

$$\mathbf{y}(k) = f(\mathbf{x}(k)) + f_{fault}(\mathbf{x}(k), status) + \varepsilon, \qquad (1)$$

where $\mathbf{x}(k) = [\mathbf{y}^T(k-1), \mathbf{y}^T(k-2), \ldots, \mathbf{y}^T(k-k_y), \mathbf{u}^T(k-k_d-1), \mathbf{u}^T(k-k_d-2), \ldots, \mathbf{u}^T(k-k_d-k_u)]^T$ consisted of control inputs and delay outputs at k, in which $\mathbf{u} = [u_1, u_2, \ldots, u_d]^T$ and $\mathbf{y} = [y_1, y_2, \ldots, y_m]^T$ are respectively the input and output vector, k_y and k_u respectively represent historical steps of $\mathbf{y}$ and $\mathbf{u}$ which constitute the state $\mathbf{x}(k)$, k_d is the time delay of the process, k represents a consecutive number of sample, f_{fault} represents an unknown fault function which could be in any form where the *status* represents a fault state discretely, and ε is white noise. In this paper, the FDD method aims to determine the fault status of the system using its measurable input and output.

3.2. System dynamics modelling with Gaussian process model

The Gaussian process model is a non-parametric model in Bayesian statistics, which has been used for dynamic system estimation in [38]. In general, the system modelling with GP model includes three phases as follows:

3.2.1. GP prior model

As a common type of priors over a function, the GP model assumes that $f(\mathbf{x})$ is a Gaussian random variable $\mathcal{N}(\mu, \sigma^2)$ at any point $\mathbf{x}$, where μ and σ are independent constants. Thus, the joint multivariate Gaussian distribution over a set of variables can be expressed as $f(\mathbf{x}^1), \ldots, f(\mathbf{x}^N) \sim \mathcal{N}(0, \Sigma)$, where $\Sigma \in \mathbb{R}^{N \times N}$ is a matrix and Σ_{pq} can be formulated as $\Sigma_{pq} = C(\mathbf{x}^p, \mathbf{x}^q)$ that denotes the covariance between $f(\mathbf{x}^p)$ and $f(\mathbf{x}^q)$. The $C(\cdot, \cdot)$ is a covariance function whose standard form is

$$C(\mathbf{x}^p, \mathbf{x}^q) = v_1 \exp\left(-\frac{1}{2} \sum_{i=1}^{D} \theta_i \left| x_i^p - x_i^q \right|^2\right). \qquad (2)$$

where D is the dimension of $\mathbf{x}$, $\theta_i > 0$ are parameters denoting the importance of x_i on $f(\mathbf{x})$ and parameter v_1 controls the variational vertical scale.

For an unknown function $y = f(\mathbf{x}) + \varepsilon, \mathbf{x} \in R^D$, where ε is a white noise with a variance of v_0, i.e. $\varepsilon \sim \mathcal{N}(0, v_0)$, we have $y^1, \ldots, y^N \sim \mathcal{N}(0, \mathbf{K})$ with $\mathbf{K}_{pq} =$

$\Sigma_{pq} + v_0 \delta_{pq}$ following the GP prior probabilistic framework, where $\delta_{\mathrm{pq}} = 1$ if $p = q$, 0 otherwise. More details about the GP model can be found in [38,39].

3.2.2. Hyperparameters optimization

Given N observation states $\mathbf{x}^1, \ldots, \mathbf{x}^N$ and their target outputs $y^1, \ldots, y^N$, the probability distribution $p(\mathbf{Y} \mid \mathbf{X}, \Theta)$ gives a likelihood of training data [38]:

$$\mathcal{L}(\Theta) = \log(p(\mathbf{Y} \mid \mathbf{X}, \Theta))$$
$$= -\frac{1}{2} \log(|\mathbf{K}|) - \frac{1}{2} \mathbf{Y}^T \mathbf{K}^{-1} \mathbf{Y} - \frac{N}{2} \log(2\pi),$$
$$(3)$$

where $\mathbf{Y} = [y^1, \ldots, y^N]^T \in \mathbb{R}^N$ is the training target matrix, $\mathbf{X} = [\mathbf{x}^1, \ldots, \mathbf{x}^N]^T \in \mathbb{R}^{N \times D}$ is the training state matrix, $\Theta = [v_0, v_1, \theta_1, \ldots, \theta_D]^T$ is the vector of hyperparameters, and $\mathbf{K} \in \mathbb{R}^{N \times N}$ is the training covariance matrix. The hyperparameters can be estimated by maximizing the likelihood:

$$\Theta^* = \arg \max_{\Theta} \mathcal{L}(\Theta), \qquad (4)$$

where the optimization needs to compute the partial derivative of $\mathcal{L}$ to each of the hyperparameters:

$$\frac{\partial \mathcal{L}(\Theta)}{\partial \Theta_i} = -\frac{1}{2} \mathrm{trace}\left(\mathbf{K}^{-1} \frac{\partial \mathbf{K}}{\partial \Theta_i}\right) + \frac{1}{2} y^T \mathbf{K}^{-1} \frac{\partial \mathbf{K}}{\partial \Theta_i} \mathbf{K}^{-1} y. \qquad (5)$$

3.2.3. Prediction

Once the estimation of hyperparameters Θ^* is given, we can predict the output y^* at any untested state $\mathbf{x}^*$ based on the relevant state-output pairs $(\mathbf{x}^i, y^i)$ for $i = 1, \ldots, N$. Thus, the unbiased prediction of y^* can be obtained by the predictive distribution $p(y^* \mid \mathbf{Y}, \mathbf{X}, \mathbf{x}^*) = p(\mathbf{Y}, y^*)/p(\mathbf{Y} \mid \mathbf{X})$ with mean and variance [38]:

$$\mu\left(y^*\right) = \mathbf{k}\left(\mathbf{x}^*\right)^T \mathbf{K}^{-1} \mathbf{Y}, \qquad (6)$$

$$\sigma^2\left(y^*\right) = k\left(\mathbf{x}^*\right) - \mathbf{k}\left(\mathbf{x}^*\right)^T \mathbf{K}^{-1} \mathbf{k}\left(\mathbf{x}^*\right) + v_0, \qquad (7)$$

where $\mathbf{k}(\mathbf{x}^*) = [C(x^1, x^*), \ldots, C(x^N, x^*)]^T$ is the covariance vector between the test and training states and $k(x^*) = C(x^*, x^*)$ is the covariance between the test state and itself.

4. The JITGP-ELM approach

Aiming at the problem formulated in Section 3.1, we will introduce the proposed JITGP-ELM method in this section. Figure 2 shows the block diagram of the proposed method. The JITGP-ELM method includes two procedures: fault feature extraction by JITGP and fault status determination by ELM with the fault feature, where the fault feature is the estimation residuals generated by JITGP. In the following, we will first present the
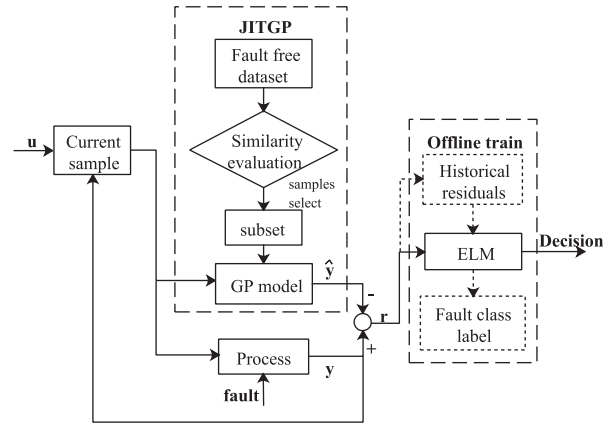

Figure 2. The framework of JITGP-ELM.

framework of JITGP-ELM, then the JITGP modelling algorithm for fault feature extraction, and the offline ELM training process.

4.1. Framework description of the JITGP-ELM approach

The proposed method is under the JITL scheme, which has three steps [40]. (1) Select a set of samples from the dataset based on some similarity criterion. (2) Establish a local model with the selected samples to predict the process output. (3) When a new sample needs to be predicted, we build a new local model by repeating the above two steps. This paper uses the GP model to improve conventional JITL to develop noise-resistant estimation and obtain high-quality residuals as a fault feature.

In conventional methods, the fault status is determined by a threshold with residuals between estimation and observation. Thus, the threshold-based methods could with two problems: the improper design of the threshold value and incompetence for multi-fault diagnosis problems. In the JITGP-ELM approach, the residuals are analyzed by an ELM classifier model that deals with multi-fault diagnosis problems without thresholds and is entirely data-driven. After the process estimation by JITGP, the extracted residuals will be input to the ELM fault classifier model, whose output is the FDD decision of fault status, where the ELM model is trained offline.

Based on the previous discussion, the JITGP-ELM framework can be described in Algorithm 1. In the JITGP-ELM algorithm, the fault feature extraction is integrated into the JITGP algorithm, and the training process of the ELM-classifier is integrated into the ELM-training algorithm, which will be discussed in the following subsections.

In the JITGP-ELM, the updating GP model is for the residual feature generation by estimating the healthy system states accurately and adaptively. Then, the ELM is for the fault classification with the residual features. Therefore, it is considered that when the system state

estimation is relatively accurate, the updated GP model will not have a great impact on the mapping relationship between residual features and fault types. Thus, the offline training ELM model will not be updated during the working process. However, after the process, we could retrain the ELM model with the collected data and the actual fault label.

4.2. JITGP-based fault feature extraction

While the original data increases the useful information for fault diagnosis, it also inevitably brings some redundant information. Redundant features will reduce the efficiency and accuracy of fault diagnosis. Thus, feature extraction is necessary to improve the fault diagnosis performance [22]. Many methods have been proposed for feature extraction in fault diagnosis. In [41], a two-stage feature selection method, namely Hybrid Distance Evaluation technique, was proposed to select a subset of combined features with strong classification ability. Luo et al. [22] extracts three types of fault features to compose the compound feature set based on ensemble empirical mode decomposition. In [14,42], the system dynamics and nonlinearity are circumvented using models, and the residuals will be analyzed using a multivariate statistical method as a fault feature to receive fault detection results. In this paper, JITGP serves as a process model to predict the output for generating residuals, which are used to be the fault feature.

In the JITGP, the GP model is used to establish the local models instead of the autoregressive model (ARX), which is frequently used by the conventional JITL methods [14,34]. In conjunction with the JITL, the computational cost of the GP model is significantly reduced due to the presence of the sample selection step. The details of JITGP-based fault feature extraction are discussed below, including relevant sample selection with a similarity estimation, local model prediction with GP, and a residual vector will be generated according to the output estimation and observation at the last step.

4.2.1. Relevant sample selection
In JITGP, a similarity evaluation is used to be the criterion to select the most relevant samples to the current sample, which needs to be predicted. With reference to [14,33], the similarity between the current sample $\mathbf{x}^q$ and a historical sample $\mathbf{x}^i$ in a dataset is measured as

$$\mathcal{S}_i = \gamma \sqrt{e^{-d^2(\mathbf{x}^q, \mathbf{x}^i)}} + (1 - \gamma) \cos(\alpha_i), \qquad (8)$$

where $\gamma \in [0, 1]$ is a weight parameter, α_i represents the angle between $\Delta \mathbf{x}^q$ and $\Delta \mathbf{x}^i$, in which $\Delta \mathbf{x}^q = \mathbf{x}^q - \mathbf{x}^{q-1}$ and $\Delta \mathbf{x}^i = \mathbf{x}^i - \mathbf{x}^{i-1}$; $\mathcal{S}_i \in [0, 1]$ represents the similarity and the larger $\mathcal{S}_i$ denotes the higher similarity between $\mathbf{x}^q$ and $\mathbf{x}^i$. According to [14], it should

be noted that the $\cos(\alpha_i)$ in (8) has to be larger than 0, which means that the angle $\alpha_i < 90°$. If $\alpha_i > 90°$, the two vectors will be considered dissimilar, and the corresponding $\mathbf{x}^i$ will not be taken into account when establishing a local model. After the similarity evaluation, a subset of data can be constructed with the relevant data $(\mathbf{x}^k, \mathbf{y}^k)$ for $k = 1, \ldots, n$ with the biggest $\mathcal{S}_i$, where n is the number of samples to establish a just-in-time local model.

4.2.2. Local model prediction with GP
Consider an untested state $\mathbf{x}$ needs to be predicted with tested samples $(\mathbf{x}^k, \mathbf{y}^k)$ for $k = 1, \ldots, n$. To simplify modelling computation, we assume that the various components of $\mathbf{y}$ at the current output are independent of each other. Therefore, the prediction of $\mathbf{y} = f(\mathbf{x}) + \varepsilon$ can be decomposed into sub-problems: $y_i = f_i(\mathbf{x}) + \varepsilon_i$, that is a component estimation of $\mathbf{y}$. According to Section 3.2, the prediction of y_i can be obtained by $\hat{y}_i = \mu_i(\mathbf{x})$ from (6) with mean squared error $s_i^2 = \sigma_i^2(\mathbf{x})$ from (7), after the GP modelling and hyperparameters optimization. Thus, the prediction of $\mathbf{y}$ can be obtained: $\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_m]^T$.

4.2.3. Residuals generation
To generate residuals as the fault feature for fault detection and diagnosis, the JITGP simulates the nonlinear and dynamic behaviour of the actual process. The residuals are the difference between the output estimation and the actual observation, which has eliminated the dynamics and nonlinearity of the process and extracted the essential information of faults. Thus, a residual is generated by the following equation:

$$\mathbf{r} = \mathbf{y} - \hat{\mathbf{y}}, \qquad (9)$$

where $\mathbf{y}$ and $\hat{\mathbf{y}}$ is the actual and predictive output of the current state, respectively.

Because of the high computational cost of repeatedly applying GP regression, the proposed method reduces the computation cost by running the GP modelling in the framework of just-in-time learning, i.e. JITGP. In the framework of JITGP, the GP modelling is based on a small dataset by the sample selection process. The implementation of the techniques discussed above requires inversions of the covariance matrix $\mathbf{K}$, with a computational complexity of $\mathcal{O}(n^3)$ and a memory complexity of $\mathcal{O}(n^2)$ with n-size training data. Here, the computational cost could be high since the requirement of the inversion of $\mathbf{K}$ in (5) [43]. This issue can be addressed with the natural advantage of just-in-time learning. Due to $n \ll N$, i.e. the number of training data n is far less than the amount of data in the dataset N, the computational load for matrix inversion of $\mathbf{K}$ will be greatly reduced. Thus, the algorithm can better meet the requirement of real-time, which is one of the superiorities of JITGP modelling.

Algorithm 1 The JITGP-ELM Algorithm

Input: a state-output sample $(\mathbf{x}, \mathbf{y})$ needs fault detection and diagnosis.
Dataset require: fault dataset S_f and fault-free dataset S_N.
Initialization: set the nearest neighbour number n and the ELM hidden layer nodes number L.

1: **Offline fault classifier training:**
2: *ELM-classifier = ELM-training(S_f)*
3: **Online decision making:**
4: Calculate the residual: $\mathbf{r} = JITGP(\mathbf{x}, \mathbf{y})$;
5: Calculate $\hat{\mathbf{t}} = ELM\text{-}classifier(\mathbf{r})$
6: Determine the fault status: $status = \arg\max_{i \in \{1,\dots,o\}} \hat{\mathbf{t}}_i$
7: **Output:** The decision of fault detection and diagnosis: fault *status*.

Algorithm 2 JITGP-based residuals extraction: $JITGP(\mathbf{x}, \mathbf{y}, S_N)$

Input: a state-output pair $(\mathbf{x}, \mathbf{y})$
Dataset require: fault-free sample set, $S_N = [\mathbf{X}_N, \mathbf{Y}_N]$.
Initialization: set the nearest neighbour number n.

1: **for** each $\mathbf{x}^i \in \mathbf{X}_N$ **do**
2:     Calculate the similarity $\mathcal{S}_i$ between $\mathbf{x}$ and $\mathbf{x}^i$ by (8);
3:     **if** $\cos(\alpha_i) \leq 0$ **then**
4:         $\mathcal{S}_i = 0$
    end
5: Construct a subset S_n with n nearest neighbours from $\mathbf{S}_N$ according to $\{\mathcal{S}_i\}$;
6: Construct a GP prior model with the subset S_n;
7: Obtain the optimal hyperparameters Θ^* from (4);
8: Calculate $\hat{\mathbf{y}}$ corresponding $\mathbf{x}$ by (6) with the subset S_n and Θ^*;
9: Calculate the residual $\mathbf{r}$ by (9);

Output: the estimated residual $\mathbf{r}$.

Based on the above discussions, the JITGP-based fault feature extraction algorithm can be summarized in Algorithm 2. In order to establish the JITGP dataset, a large amount of offline fault-free data is required. When a current sample demands a prediction, the most relevant data in the dataset are found for building a local model based on the measure of similarity $\mathcal{S}_i$ from (8). The hyperparameters of GPR are optimized by (4) to build the local model. Then, the output prediction of the current sample is obtained by (6), and a residual could be computed by (9).

Since the JITGP is used to extract the residual features by estimating the output of the health system whose dynamics model is unknown, the GP model in the JITGP should be built based on fault-free data by selecting relevant data in the fault-free dataset. The fault-free dataset will be updated when judging that the system sensor is not faulty, so the GP model in the proposed method is updated based on an updated fault-free dataset for coping with the dynamic change of the system. Furthermore, compared to building a GP model based on a whole fault-free dataset, the selection of samples in just-in-time learning could reduce the computational burden because of the reduction in training data.

After that, a fault classification model will make the fault decision by analyzing residuals. Theoretically, the fault diagnosis can be obtained by a classifier, such as BP or SVM, when it is appropriate. Here, We use ELM as the fault classifier because of some of its excellent characteristics.

4.3. ELM fault classifier training

ELM is a single-hidden layer feedforward neural network whose hidden layer parameters are generated in an arbitrary way [37]. Compared with other learning methods, such as Single Layer Perceptron and SVM, ELM is believed to have advantages in learning speed and generalization ability. Figure 3 shows the network structure of ELM.

With an m-dimensional input $\mathbf{r}$, the output of an SLFN is

$$\hat{\mathbf{t}} = \sum_{i=1}^{L} \beta_i g\left(\mathbf{w}_i \cdot \mathbf{r} + b_i\right) = \sum_{i=1}^{L} \beta_i h_i(\mathbf{r}), \qquad (10)$$

where L represents the number of hidden nodes, $\mathbf{w}_i = [w_{i1}, \dots, w_{im}]^T$ is the input weight vector connecting the input and the ith hidden node, $\beta_i = [\beta_{i1}, \dots, \beta_{io}]^T$ is the output weight vector of ith hidden node in which o is the number of the output nodes, $g(\cdot)$ denotes the activation function of the hidden layer, b_i denotes the threshold of the ith hidden node, and $h_i(\mathbf{r})$ is the output of ith hidden node with input $\mathbf{r}$.
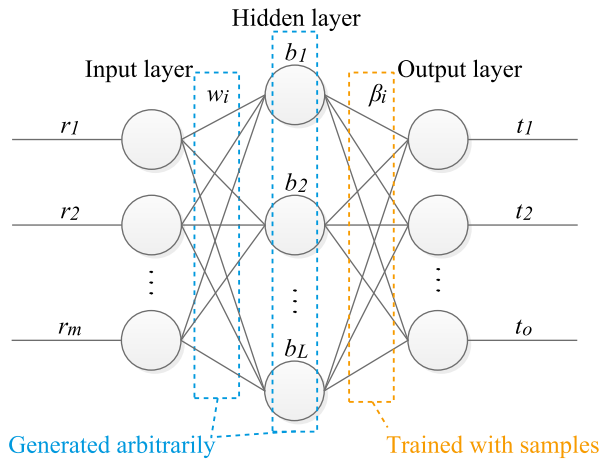
Figure 3. The network structure of ELM.

Figure 4. Application of sensor fault detection and diagnosis in steering-by-wire systems of UGVs.

Thus, given N samples $(\mathbf{r}^i, \mathbf{t}^i)$ for $i = 1, \ldots, N$, where $\mathbf{r}^i = [r_1^i, \ldots, r_m^i]^T \in \mathbb{R}^m$ and $\mathbf{t}^i = [t_1^i, \ldots, t_o^i]^T \in \mathbb{R}^o$, there are N equations in (10) format which can be compacted as

$$\mathbf{H}\beta = \mathbf{T}, \qquad (11)$$

where

$$\mathbf{H} = \begin{bmatrix} g\left(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1\right) & \cdots & g\left(\mathbf{w}_{\tilde{N}} \cdot \mathbf{x}_1 + b_L\right) \\ \vdots & \ddots & \vdots \\ g\left(\mathbf{w}_1 \cdot \mathbf{x}_N + b_1\right) & \cdots & g\left(\mathbf{w}_{\tilde{N}} \cdot \mathbf{x}_N + b_L\right) \end{bmatrix}_{N \times L},$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{L \times o}, \quad \mathbf{T} = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times o}.$$

According to [37], $\mathbf{H}$ is the hidden layer output matrix, in which $\mathbf{H}_{ij}$ is the jth hidden node output with input $\mathbf{r}^i$, that can be calculated after assigning random hidden nodes parameters (w_j, b_j). According to [37,44], the smallest norm least squares solution of (11) can be computed as

$$\hat{\beta} = \mathbf{H}^\dagger \mathbf{T}, \qquad (12)$$

where $\mathbf{H}^\dagger$ is the Moore-Penrose generalized inverse of $\mathbf{H}$. Thus, the ELM algorithm can be summarized as follows:

For classification, the output of ELM could have o nodes with o equal to the number of classes. If the class label is k, the expected output vector will be $\mathbf{t} = [0, \ldots, 0, \overset{k}{1}, 0, \ldots, 0]^T$ where the kth element is 1, and the rest of the elements are 0. For multiclass problems, the predictive class of a testing input is the index number of the output node with the highest value. Thus, the predictive fault status of $\mathbf{r}$ is

$$status = \underset{i \in \{1, \ldots, o\}}{\arg\max} \hat{\mathbf{t}}_i, \qquad (13)$$

where $\hat{\mathbf{t}}(\mathbf{x}) = [\hat{\mathbf{t}}_1(\mathbf{x}), \ldots, \hat{\mathbf{t}}_o(\mathbf{r})]$ and $\hat{\mathbf{t}}_j(\mathbf{r})$ is the output of the jth nodes.

Therefore, the major steps for training the ELM classifier based on residual extraction are summarized in Algorithm 3. In order to train the ELM network, the dataset $S_f = [\mathbf{X}_f, \mathbf{Y}_f, \mathbf{T}_f]$ is required, where $[\mathbf{X}_f, \mathbf{Y}_f]$ is the state-output data under normal and various fault conditions, and $\mathbf{T}_f$ is the corresponding target label of fault statuses. Then, the training dataset $[\mathbf{R}_f, \mathbf{T}_f]$ is constructed by combining process residuals $\mathbf{R}_f$ calculated by JITGP, with the corresponding fault status label $\mathbf{T}_f$. After determining the parameters and activation function, the ELM training process will then complete and obtain a fault classifier.

5. Experimental verification

In order to evaluate the performance of the proposed method, the following experiments were completed on fault detection and diagnosis problem for the steering-by-wire system in a real UGV. The experimental platform is a wheeled electric vehicle with a steering-by-wire system, as shown in Figure 5(a). As a new generation of the steering system, the steering-by-wire system achieves autonomous steering, relying on the electrical system instead of the traditional mechanical transmission mechanism that becomes the basis and guarantee of assisted or unmanned driving. However, faults could occur in the sensors and actuators that may affect the reliability and safety of the electrical systems. The experiment considers the fault detection and diagnosis for the steering wheel angle sensor with yaw rate and lateral acceleration sensors' signals in unmanned ground vehicles (Figure 4).

5.1. Data preparation

The experimental design of fault detection and diagnosis includes three sensor faults, which of steering wheel angle, yaw rate, and lateral acceleration. The input of the system is the steering wheel angle (SW), and the outputs are the yaw rate (YR) and the lateral acceleration (LA). The purpose is to detect and identify different sensor faults by analyzing the observable manifestation of the UGV. Due to the requirement to estimate the yaw rate and the lateral acceleration from observing the steering wheel angle, a set of data that can reflect the UGV's dynamics and nonlinearity needs to be collected for

---

Algorithm 3 ELM fault classifier training based on JITGP: *ELM-training*(S_f)

Dataset require: $S_f = [\mathbf{X}_f, \mathbf{Y}_f, \mathbf{T}_f]$

Initialization: set ELM hidden layer nodes number L; set a residual dataset $\mathbf{R}_f$ with empty; select an activation function $g(\cdot)$ that is infinitely differentiable.

1: **for** each $\mathbf{x}^q \in \mathbf{X}_f, \mathbf{y}^q \in \mathbf{Y}_f$ **do**
2:      $\mathbf{r}^q = JITGP(\mathbf{x}^q, \mathbf{y}^q)$
3:      Store $\mathbf{r}^q$ in the dataset $\mathbf{R}_f$;
4: **end**
5: Construct a training dataset $[\mathbf{R}_f, \mathbf{T}_f]$;
6: Set network nodes numbers of input layer m and output layer o with $m = numel(\mathbf{r}^i)$ and $o = numel(\mathbf{t}^i)$;
7: Assign the connection weight w of the input layer and the hidden layer and the threshold b of the hidden nodes randomly;
8: Calculate the hidden layer output matrix $\mathbf{H}$ with $w, b, g(\cdot)$ and $\mathbf{R}_f$;
9: Calculate the weight of the output layer $\hat{\beta} = \mathbf{H}^\dagger \mathbf{T}_f$;

Output: the ELM classifier network with $w, b, g(\cdot)$ and $\hat{\beta}, \hat{\mathbf{t}} = ELM\text{-}classifier(\mathbf{r})$.
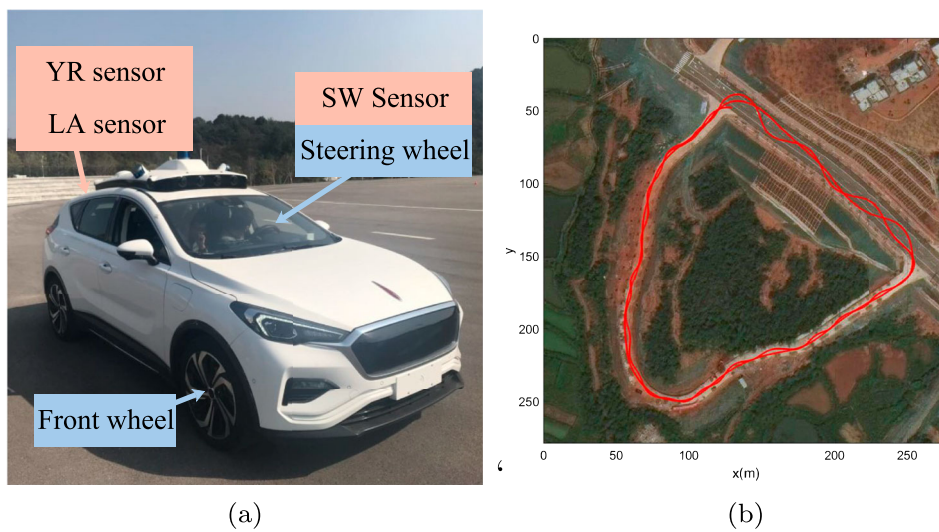

Figure 5. The experimental platform and environment. (a) UGV platform and (b) The location of the data recorded and the trajectory of vehicle test.

the proposed data-driven method. The location of the experimental data collection and the trajectory of the vehicle test is shown in Figure 5(b), collecting a total of 10,800 samples of input-output data with a frequency of 10 Hz. Furthermore, the data on sharp turns, braking, and accelerating, was collected into the dataset to ensure the persistence of excitation. When a sensor in the system failed, the sensor's signal was a white noise close to zero.

From the collected data of 10,800 samples, the first 6000 samples were chosen to construct the fault-free dataset, which supported the online JITGP algorithm to build local models. The training dataset of fault classification was constructed from the 6001st to 10,000th sample, which consisted of 4000 samples. These 4000 samples were divided into four categories, with 1000 samples in each category. The categories' labels were the fault statuses, which were fault-free, fault 1 (yaw rate sensor fault), fault 2 (lateral acceleration sensor fault), and fault 3 (steering wheel angle sensor fault), respectively. Finally, the remaining 800 samples were

used to form a test set for algorithm testing, which was also divided into four groups with 200 samples in each group, respectively corresponding to the above four fault statuses.

5.2. Fault feature extraction of steering-by-wire system

The experiments were conducted in the test set of 800 samples. In order to compare the estimation abilities of JITGP and conventional JITL, Figure 6 shows the output estimations of yaw rate and lateral acceleration for 200 fault-free samples. The JITL's local models are developed by conventional ARX models, while the GP model develops the local model of JITGP. The principle of parameter selection is to minimize the cost of calculation to ensure the performance of the algorithm. The parameters were set as follows: the amount of relevant data for JITGP and JITL local model, $n = 15$, using a small sample for speediness; the number of steps constructing local model states, $l = 2$.
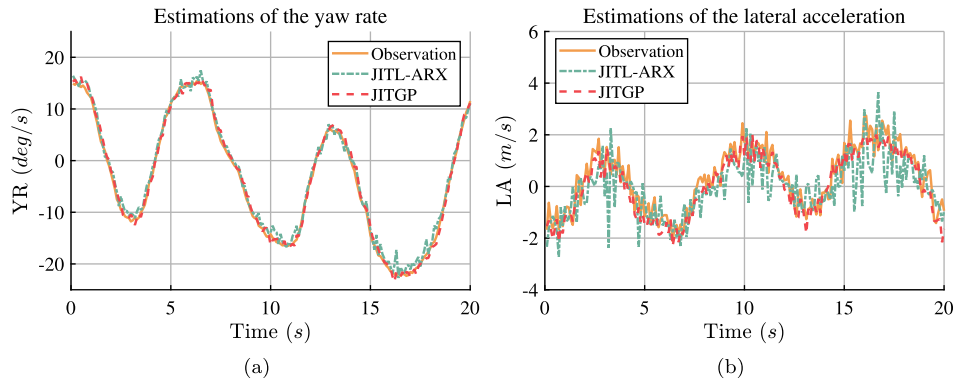
Estimations of the yaw rate

Estimations of the lateral acceleration

Figure 6. Estimations of the yaw rate (YR) and the lateral acceleration (LA) by conventional JITL (JITL-ARX) and JITGP under the fault-free condition. (a) shows the estimations of yaw rate where the mean-square errors of conventional JITL and JITGP are 1.141 and 0.779, respectively; (b) shows the estimations of lateral acceleration where the mean-square errors of conventional JITL and JITGP are 0.816 and 0.179, respectively.

The estimated results comparison between conventional JITL and JITGP under the fault-free condition are shown in Figure 6. Figure 6(a) shows the estimations of yaw rate where the mean-square errors of JITL-ARX and JITGP are 1.141 and 0.779, respectively. Both perform well in estimating the yaw rate when the actual observation curve is relatively smooth, where JITGP estimation curve has fewer fluctuations than JITL-ARX. However, the actual observation curve of lateral acceleration shook violently due to the uneven road surface. Figure 6(b) shows the estimations of lateral acceleration where the mean-square errors of JITL-ARX and JITGP are 0.816 and 0.173, respectively. When estimating the lateral acceleration, the performance of JITL-ARX is unsatisfactory, and JITGP still maintains good performance. The results show that JITL-ARX is sensitive to noise and interference, while JITGP can obtain a smoothing and close estimated result with the observation. Hence, the estimation results show that JITGP improves conventional JITL effectively.

Furthermore, 800 samples were divided into four groups to simulate different fault statuses. In the simulation, the 201st–400th samples (20–40 s) are in fault 1; the 401st–600th samples (40–60 s) are in fault 2; the 601st–800th samples (60–80 s) are in fault 3. The residuals as fault features generated by JITL-ARX and JITGP in different fault statuses are shown in Figures 7–9. From the residual results, we can find that if a fault occurs to one of the sensors of yaw rate and lateral acceleration, the residuals between the observations and estimations of corresponding output will increase (Figures 7 and 8). Another case is when a fault occurs to the steering wheel angle's sensor, which means that the input of the predictive model is a fallacious value observed by the sensor with fault. In this case, the two outputs' residuals will both increase (Figure 9). The residual results in Figures 7–9 show that the faults and fault-free characteristics are separated more obviously with JITGP than with JITL-ARX, which is even more pronounced in lateral acceleration residuals. Predictive

precision is the essential factor affecting the quality of residual information, and the extracted features are beneficial to the fault classification task in the following procedure.

5.3. Fault diagnosis of steering-by-wire system

5.3.1. Fault diagnosis based on JITL modelling

After modelling the system with JITLs, the residuals are generated as fault features, and the following work uses the extracted fault features for fault diagnosis. Six methods are BP, ELM, JITL-BP, JITL-ELM, JITGP-BP, and JITGP-ELM, whose performance was compared by being tested on the fault detection and diagnosis problem mentioned above. Among them, The BP and ELM networks were compared with fault classification, where the fault features to train the networks differed. In the JITL-based and JITGP-based methods, the fault features are residuals generated by the JITL-ARX and JITGP output estimations, respectively. On the other hand, the fault feature in BP and ELM only was original input and output data.

The parameters in algorithms should be set consistently in order to compare the performance of the algorithm. For the three-layer networks, there is a relationship between the number n_1 of input layer nodes and the number n_2 of hidden layer nodes: $n_{hadden} = 2 \times n_{input} + 1$. Thus, if we consider 10 historical steps, the network structures of BP and ELM are 20-41-4 for the residual-based methods and 30-61-4 for the direct method. The network performance function of BP is the mean square error. The learning algorithm adopted the steepest descent method, and the error target of learning was 0.001. The transfer function of the hidden and output layer are the hyperbolic tangent S-type transfer function (tansig) and the linear transfer function (purelin), respectively. And the activation function used in ELM is a simple sigmoid function $g(x) = 1/(1 + \exp(-x))$. Before training the networks, all data samples need to be normalized to avoid the
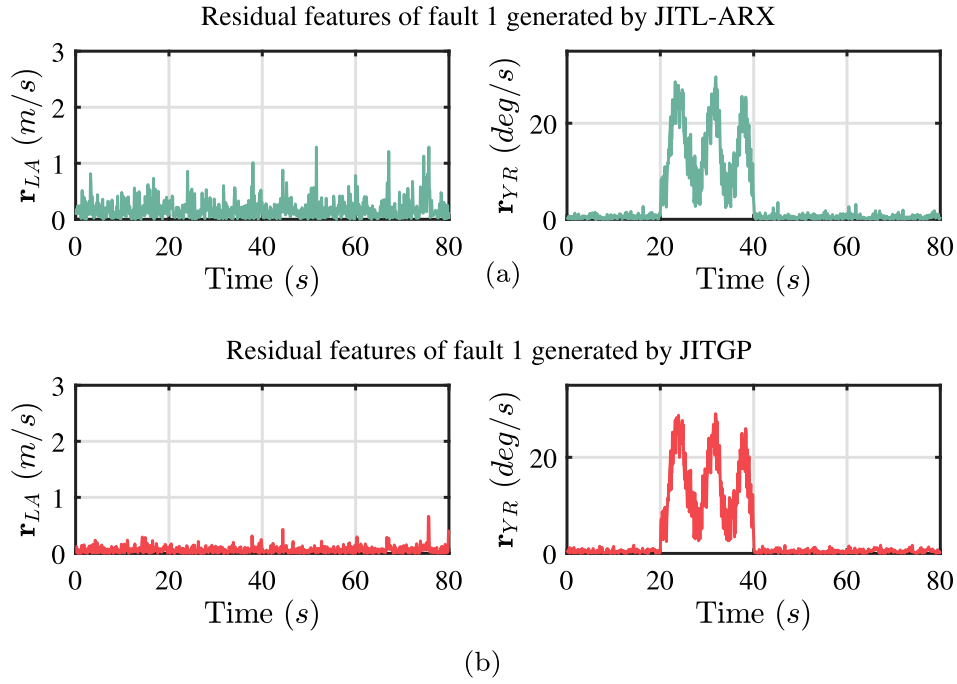
Figure 7. Residual features of fault 1 (20–40 s) extracted by JITL-ARX (a) and JRTGP (b).
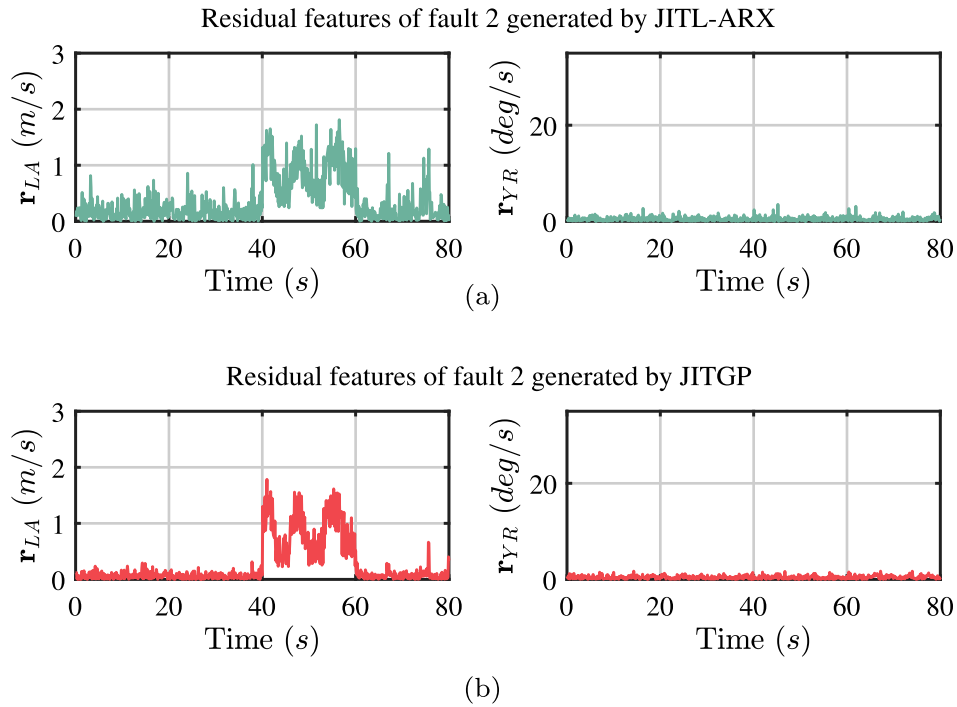

Figure 8. Residual features of fault 2 (40–60 s) extracted by JITL-ARX (a) and JRTGP (b).

error of fault diagnosis structure caused by the differences between different data sets.

The comparison of methods' accuracy of fault diagnosis based on JITLs is shown in Figure 10. The results are the classification accuracies, which is the ratio of correctly classified samples to the total number of samples. The diagnostic accuracies of BP, ELM, JITL-BP, JITL-ELM, JITGP-BP, and JITGP-ELM are 84.88%, 84.12%, 95.55%, 97.78%, 96.20%, and 99.46%, respectively, where JITL-BP, JITL-ELM, JITGP-BP, and

JITGP-ELM are in the proposed scheme with different model estimates and fault classifiers for comparing the performance; BP [20] and ELM [22] are existing fault classifier methods. The inputs of these methods are UGV state values, and outputs are fault status. The results show that the accuracy of methods in the proposed scheme is higher than methods in the conventional scheme. Furthermore, the JITGP-ELM has the best performance among methods in the proposed scheme. The experimental results show that BP only
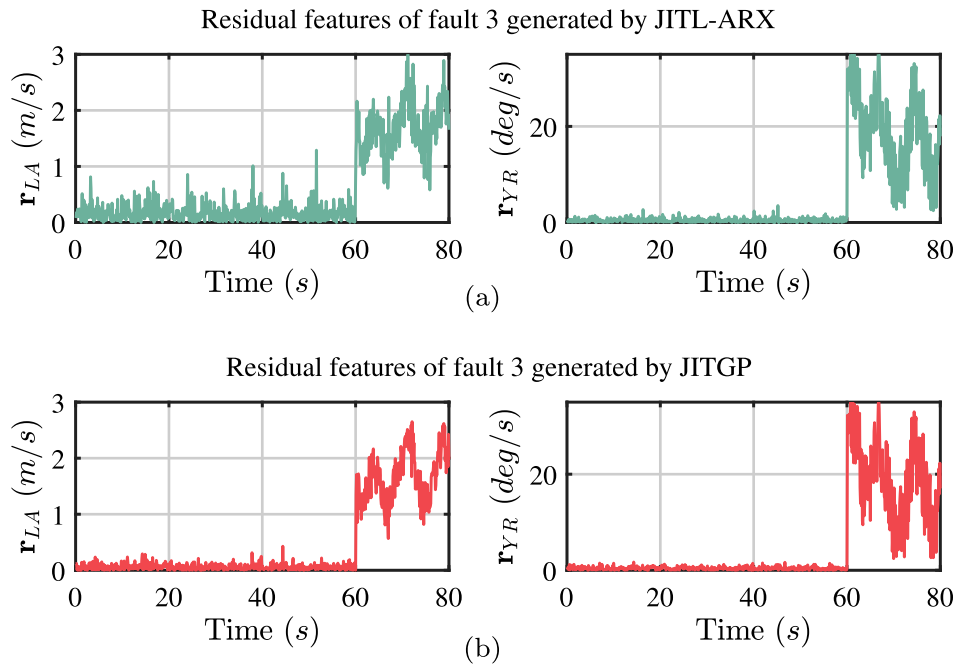
Figure 9. Residual features of fault 3 (60–80 s) extracted by JITL-ARX (a) and JRTGP (b).
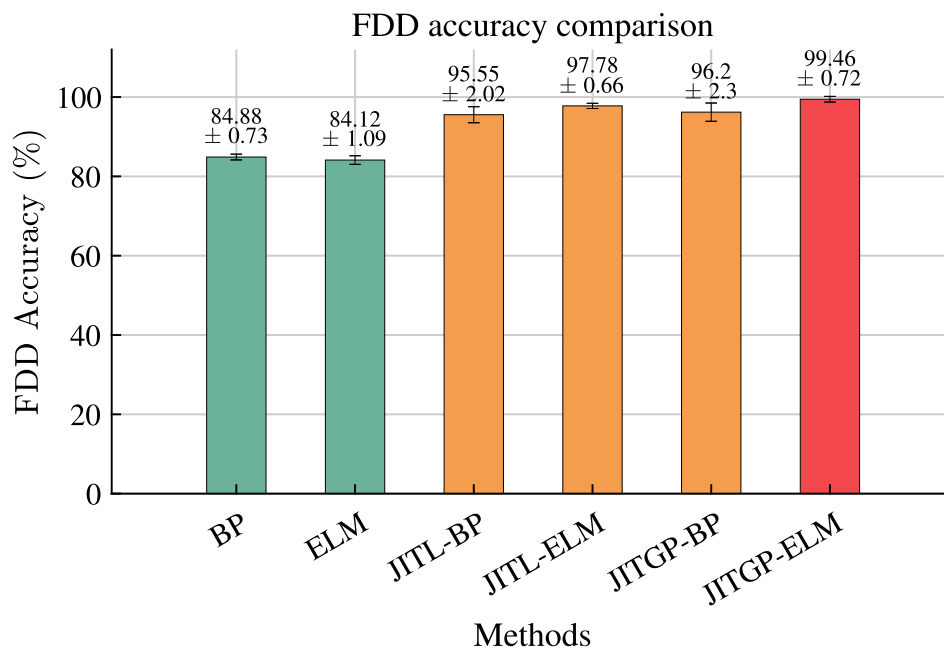

Figure 10. Methods comparison: the diagnostic accuracies of BP, ELM, JITL-BP, JITL-ELM, JITGP-BP, and JITGP-ELM are 84.88 ± 0.73%, 84.12 ± 1.09%, 95.55 ± 2.02%, 97.78 ± 0.66%, 96.20 ± 2.30%, and 99.46 ± 0.72%, respectively, where JITL-BP, JITL-ELM, JITGP-BP, and JITGP-ELM are in the proposed scheme with different model estimates and fault classifiers for comparing the performance; BP [20] and ELM [22] are existing fault classifier methods. The error bar represents the standard deviations of the results in 30 experiments.

is better than ELM, and ELM is better than BP when analyzing residuals as classification input. In the case of BP and ELM only, information needs to be extracted from more complex data in the classification process. BP has more parameters to optimize in this case, which is more conducive to extracting fault information from the original data. In the case with residuals as fault features extracted by JITL and JITGP, the fault features are more pronounced because the residuals have

eliminated the system dynamics and nonlinearity; in this case of small samples, BP may suffer overfitting, while ELM has an advantage in the generalization ability to prevent overfitting. The proposed method has a mean accuracy of 99.46%, which is 1.68% higher than 97.78% for JITL-ELM. The main reason is that the ELM has a high generalization to deal with the noisy residual features, which illustrates the significance of the selection of the fault classifier. With the ELM as the fault
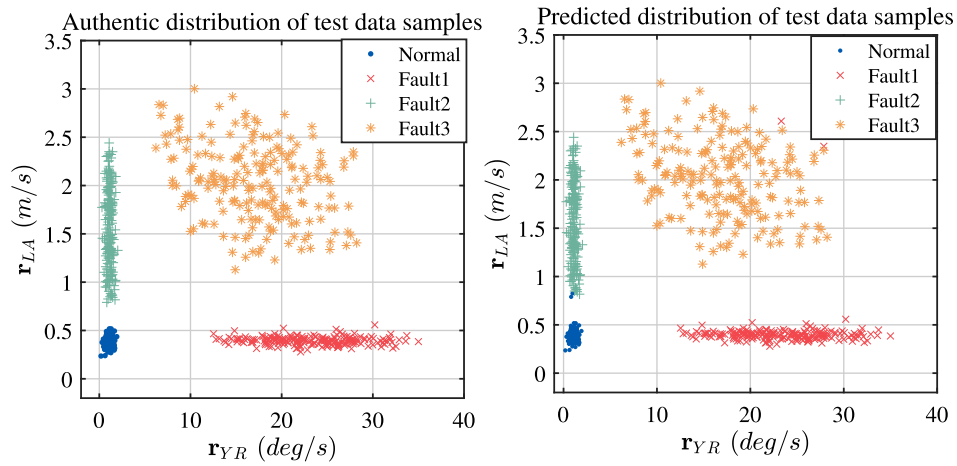
Figure 11. Fault features and diagnosis results of JITGP-ELM.

classifier, the proposed method has further performed the highest accuracy diagnosis with the advantage of the fault feature.

Figure 11 shows the fault diagnosis results of JITGP-ELM, and it can be seen that the residual is valid as a fault feature. The high-quality fault features depend on the accuracy of the system's prediction, and JITGP provides this online adaptive prediction at a low computational cost. The separation of fault features influences FDD performance significantly, and the residual as fault feature is effective for fault diagnosis. In summary, the data-driven fault detection and diagnosis method proposed in this paper has characteristic high classification accuracy. The diagnostic accuracy of the test set is 99.46%. Two samples of fault 2 are miss diagnostics, and two samples of fault 3 are wrongly diagnosed as fault 1.

5.3.2. Comparison with the data-driven diagnosis approaches

To further prove the reliability novel method, it is necessary to compare it with the existing algorithms. One-against-one SVM [21], BP neural network [20], and ELM [22] were selected as the control group in these algorithms. To ensure the fairness of algorithm comparison, the novel method and the control group were tested with the same data set of fault diagnosis and the same computer (CPU at 3.2 GHz, RAM at 16 GB) environment. The parameters in algorithms should be set consistently. For the three-layer networks of BP and ELM, the network structures are 20-41-4 for the proposed scheme (ELM in JITGP-ELM) and 30-61-4 for the direct method (BP and ELM) with the $n_{hadden} = 2 \times n_{input} + 1$ principle. For ELM, the activation function is the sigmoid function. For BP, the transfer function of the hidden and output layer are the hyperbolic tangent S-type transfer function (tansig) and the linear transfer function (purelin), respectively; the network performance function is the mean square error (MSE); the learning algorithm adopts the steepest descent method; the error target of learning is set as 0.001; the learning rate is set as 0.02; the neural weights

are initialized with random values in the range [−0.1, 0.1]; the maximum epochs of learning is set as 1000; the learning rate is set as 0.1. For SVM, there are six one-against-one models with the same parameters. For each of them, the kernel function is the radial basis function (RBF); the best penalty parameter c and the best kernel parameter g are searched by the cross-validation approach; the error target of learning is set as 0.001; the learning rate is 0.02. The training set and test set are consistent. The input and output of these models are 10 historical steps of vehicle state data and fault status, respectively. The classifier directly classifies these data in the direct method (ELM, SVM and BP), where the features are the states of the UGV, i.e. the data of steering wheel angle, yaw rate and lateral acceleration. In the proposed method, yaw rate and lateral acceleration are predicted according to steering wheel angle so that residuals are generated, and then residual features are used for fault classification.

The comparison of the results of four kinds of diagnosis methods is given in Table 1. The performances are expressed by the means and standard deviations(if have) of the results in 30 experiments. The precision of a fault diagnosis method is the ratio of the samples with correctly predicted a fault to the total number of detections of this fault; the recall is the ratio of the samples with correctly predicted a fault to the total number of samples containing the fault. From Table 1, the proposed method has the highest precisions and recalls, except the precision of fault 1, among the comparison methods. The precision indicates the reliability of a method when it announces a fault, and the recall indicates the reliability of detecting the faults. Moreover, the proposed method has the highest diagnostic accuracy of test samples, exceeding other methods by more than 10%. The reason is that the action of fault feature extraction by JITGP effectively enhances the fault diagnosis, while others were original input and output signals. The comparison shows that the proposed method benefits the fault diagnosis of the steering-by-wire system.

Table 1. Comparison of the performances of four fault diagnosis methods.

Methods	Training time (s)	Running time (s)	Precision (%)			Recall (%)			Accuracy (%)
			Fault1	Fault2	Fault3	Fault1	Fault2	Fault3	
SVM [21]	0.24	0.013	76.41 ± 1.79	87.11 ± 1.52	83.90 ± 2	95.85 ± 1.97	76.53 ± 2.79	68.55 ± 2.52	79.05 ± 1.17
BP [20]	6.34	0.006	88.77 ± 0.36	88.85 ± 3.63	88.56 ± 0.22	90.67 ± 2.84	89.07 ± 2.01	75.13 ± 3.71	84.88 ± 0.73
ELM [22]	0.036	0.012	98.93 ± 1.81	76.81 ± 3.07	89.23 ± 2.06	69.72 ± 3.16	93.18 ± 1.29	88.98 ± 0.7	84.12 ± 1.08
Proposed	0.017	0.036	98.28 ± 2.46	99.97 ± 0.18	99.89 ± 0.68	100 ± 0	99.62 ± 1.39	98.50 ± 1.83	99.46 ± 0.72

The BP neural network has the longest training time, while the proposed method has the shortest. Because the hidden layer bias matrix and input weight matrix of the ELM algorithm is generated randomly, the problems of repeatedly training and modification of connection weights and thresholds are avoided. Therefore, the ELM algorithm can significantly reduce network training time and has a high training speed. Moreover, the proposed method requires fewer neurons than ELM because of the reduction in fault feature dimensions. The proposed method's running time is the longest because of the action of extracting residuals as fault features. However, the running time of the proposed method is 0.036 s and not more than 0.1 s; that is, it can meet the real-time requirement of 10Hz. The practical consequences show that the JITGP-ELM devised in this paper can significantly improve recognition accuracy while ensuring real-time computation.

6. Conclusion

This paper proposes a data-driven fault detection and diagnosis method called JITGP-ELM for unmanned ground vehicles with unknown nonlinear dynamics. In the proposed method, the model estimator based on the just-in-time Gaussian process is designed for online residuals generation to cope with the dynamics and nonlinearity, which has online adaptability and noise-resistant ability. Based on the model estimation residuals, the fault classifier using Extreme Learning Machine is then designed for fault identification. The proposed method can solve multi-fault diagnosis problems without needing to determine the residual thresholds, which combines fault detection, isolation, and identification to improve the efficiency of the diagnosis framework. Finally, the proposed method is tested in a real UGV's steering-by-wire system with sensor faults, and the test results show its effectiveness. For future works, fault-tolerant control based on the proposed JITGP-ELM method can be considered.

Funding

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Xin Xu http://orcid.org/0000-0003-3238-745X

References

[1] Monteriu A, Asthana P, Valavanis K, et al. Experimental validation of a real-time model-based sensor fault detection and isolation system for unmanned ground vehicles. In: 2006 14th Mediterranean Conference on Control and Automation; Jun. IEEE; 2006. p. 1–8.
[2] Meskin N, Khorasani K. Actuator fault detection and isolation for a network of unmanned vehicles. IEEE Trans Automat Contr. 2009 Apr;54(4):835–840.
[3] Jeppesen BP, Cebon D. Application of observer-based fault detection in vehicle roll control. Veh Syst Dyn. 2009;47(4):465–495.
[4] Zhang H, Wang J. Active steering actuator fault detection for an automatically-steered electric ground vehicle. IEEE Trans Veh Technol. 2016;66(5):3685–3702.
[5] Lu X, Zhiqinag F, Zengliang L, et al. Fault diagnosis for SBW systems of unmanned vehicles. China Mech Eng. 2017;28(22):2689–1770.
[6] Luo J, Namburu M, Pattipati KR, et al. Integrated model-based and data-driven diagnosis of automotive antilock braking systems. IEEE Trans Syst Man Cybern A. 2010;40(2):321–336.
[7] Demetriou M, Polycarpou M. Incipient fault diagnosis of dynamical systems using online approximators. IEEE Trans Automat Contr. 1998;43(11):1612–1617.
[8] Wang W, Li L, Zhou D, et al. Robust state estimation and fault diagnosis for uncertain hybrid nonlinear systems. Nonlinear Anal. 2007 Mar;1(1):2–15.
[9] Jesussek M, Ellermann K. Fault detection and isolation for a nonlinear railway vehicle suspension with a hybrid extended Kalman filter. Veh Syst Dyn. 2013;51(10):1489–1501.
[10] Börner M, Straky H, Weispfenning T, et al. Model based fault detection of vehicle suspension and hydraulic brake systems. Mechatronics. 2002;12(8):999–1010.
[11] Monteriu A, Asthan P, Valavanis K, et al. Model-based sensor fault detection and isolation system for unmanned ground vehicles: theoretical aspects (part I). In: Proceedings 2007 IEEE International Conference on Robotics and Automation; Apr. IEEE; 2007. p. 2736–2743.
[12] Monteriu A, Asthan P, Valavanis K, et al. Model-based sensor fault detection and isolation system for unmanned ground vehicles: experimental validation (part II). In: Proceedings 2007 IEEE International Conference on Robotics and Automation; Apr. IEEE; 2007. p. 2744–2751.
[13] Sahin ON, Dede MIC. Model-based detection and isolation of the wheel slippage and actuator faults of a Holonomic mobile robot. Ind Rob. 2022 May;49(6):1202–1217.
[14] Yin S, Gao H, Qiu J, et al. Fault detection for nonlinear process with deterministic disturbances: a just-in-time learning based data driven method. IEEE Trans Cybern. 2017 Nov;47(11):3649–3657.

[15] Liu J, Li Q, Chen W, et al. A fast fault diagnosis method of the pemfc system based on extreme learning machine and Dempster–Shafer evidence theory. IEEE Trans Transp Electrif. 2019 Mar;5(1):271–284.

[16] Liu X, Huang H, Xiang J. A personalized diagnosis method to detect faults in gears using numerical simulation and extreme learning machine. Knowl Based Syst. 2020;195:Article ID 105653.

[17] Namburu S, Wilcutts M, Chigusa S, et al. Systematic data-driven approach to real-time fault detection and diagnosis in automotive engines. In: 2006 IEEE Autotestcon; Sep. IEEE; 2006. p. 59–65.

[18] Avila Okada KF, Silva de Morais A, Oliveira-Lopes LC, et al. A survey on fault detection and diagnosis methods. In: 2021 14th IEEE International Conference on Industry Applications (INDUSCON); Aug. IEEE; 2021. p. 1422–1429.

[19] Abid A, Khan MT, Iqbal J. A review on fault detection and diagnosis techniques: basics and beyond. Artif Intell Rev. 2021 Jun;54(5):3639–3664.

[20] Li J, Yao X, Wang X, et al. Multiscale local features learning based on BP neural network for rolling bearing intelligent fault diagnosis. Measurement. 2020 Mar;153:Article ID 107419.

[21] Martínez-Morales JD, Palacios-Hernández ER, Campos-Delgado DU. Multiple-fault diagnosis in induction motors through support vector machine classification at variable operating conditions. Electr Eng. 2018 Mar;100(1):59–73.

[22] Luo M, Li C, Zhang X, et al. Compound feature selection and parameter optimization of ELM for fault diagnosis of rolling element bearings. ISA Trans. 2016;65:556–566.

[23] Lei Q, Zhang C, Shi J, et al. Machine learning based refrigerant leak diagnosis for a vehicle heat pump system. Appl Therm Eng. 2022 May;215:Article ID 118524.

[24] Abboush M, Bamal D, Knieke C, et al. Intelligent fault detection and classification based on hybrid deep learning methods for hardware-in-the-loop test of automotive software systems. Sensors. 2022 May;22(11): 4066.

[25] Wallace C, Ackels S, Benavidez P, et al. Real-time distributed ensemble learning for fault detection of an unmanned ground vehicle. In: 2020 IEEE 15th International Conference of System of Systems Engineering (SoSE); Jun. IEEE; 2020. p. 463–468.

[26] Tidriri K, Chatti N, Verron S, et al. Bridging data-driven and model-based approaches for process fault diagnosis and health monitoring: a review of researches and future challenges. Annu Rev Control. 2016;42:63–81.

[27] Khorasgani H, Farahat A, Ristovski K, et al. A framework for unifying model-based and data-driven fault diagnosis. In: Proceedings of the Annual Conference of the Prognostics and Health Management Society. PHM; 2018. p. 1–10.

[28] Jung D, Ng KY, Frisk E, et al. Combining model-based diagnosis and data-driven anomaly classifiers for fault isolation. Control Eng Pract. 2018 Aug;80:146–156.

[29] Jung D, Sundstrom C. A combined data-driven and model-based residual selection algorithm for fault detection and isolation. IEEE Trans Control Syst Technol. 2019;27(2):616–630.

[30] Gomes IP, Wolf DF. Health monitoring system for autonomous vehicles using dynamic Bayesian networks for diagnosis and prognosis. J Intell Rob Syst. 2021 Jan;101(1):19.

[31] Jaenisch HM, Handley JW, Hicklen ML. Real time prediction and diagnostics for unmanned ground vehicle (UGV) mobility. 2007 Apr. p. 65611Z.

[32] Bittanti S, Picci G. Identification, adaptation, learning: the science of learning models from data. Vol. 153. Berlin: Springer Science & Business Media; 1996.

[33] Cheng C, Chiu MS. A new data-based methodology for nonlinear process modeling. Chem Eng Sci. 2004;59(13):2801–2810.

[34] Liu Y, Chen T, Chen J. Auto-switch gaussian process regression-based probabilistic soft sensors for industrial multigrade processes with transitions. Ind Eng Chem Res. 2015 May;54(18):5037–5047.

[35] Yang K, Jin H, Chen X, et al. Chemometrics and intelligent laboratory systems soft sensor development for online quality prediction of industrial batch rubber mixing process using ensemble just-in-time gaussian process regression models. Chemometr Intell Lab Syst. 2016;155:170–182.

[36] Kanno Y, Kaneko H. Ensemble just-in-time model based on Gaussian process dynamical models for nonlinear and dynamic processes. Chemometr Intell Lab Syst. 2020 Aug;203:Article ID 104061.

[37] Huang GB, Zhu QY, Siew CK. Extreme learning machine: theory and applications. Neurocomputing. 2006;70(1–3):489–501.

[38] Kocijan J, Girard A, Banko B, et al. Dynamic systems identification with Gaussian processes. Math Comput Model Dyn Syst. 2005 Dec;11(4):411–424.

[39] O'Hagan A. Curve fitting and optimal design for prediction. J R Stat Soc Ser B (Methodol). 1978 Sep;40(1):1–24.

[40] Cheng C, Chiu MS. Nonlinear process monitoring using JITL-PCA. Chemometr Intell Lab Syst. 2005 Mar;76(1):1–13.

[41] Van M, Kang HJ. Bearing-fault diagnosis using nonlocal means algorithm and empirical mode decomposition-based feature extraction and two-stage feature selection. IET Sci Meas Technol. 2015;9(6):671–680.

[42] Yin S, Zhu X, Kaynak O. Improved PLS focused on key-performance-indicator-related fault diagnosis. IEEE Trans Ind Electron. 2015 Mar;62(3):1651–1658.

[43] Rasmussen CE. Gaussian Processes in Machine Learning, In Summer school on machine learning. Heidelberg: Springer Berlin Heidelberg; 2004. p. 63–71.

[44] Huang G-B, Zhou H, Ding X, et al. Extreme learning machine for regression and multiclass classification. IEEE Trans Syst Man Cybern B (Cybern). 2012 Apr;42(2):513–529.