Array 19 (2023) 100298

Contents lists available at ScienceDirect

Array

journal homepage: www.sciencedirect.com/journal/array

Fault detection and state estimation in robotic automatic control using machine learning

Rajesh Natarajan [a], Santosh Reddy P [b], Subash Chandra Bose [c], H.L. Gururaj [d,*], Francesco Flammini [e], Shanmugapriya Velmurugan [f]

[a] Information Technology Department, University of Technology and Applied Sciences-Shinas, Al-Aqr, Shinas, 324, Oman
[b] Department of Computer Science & Engineering, BNM Institute of Technology, Bangalore, India
[c] Department of Computer Science, Islamiah College (Autonomous), Vaniyambadi, Tamilnadu, 635751, India
[d] Department of Information Technology, Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education, Manipal, India
[e] IDSIA USI-SUPSI, University of Applied Sciences and Arts of Southern Switzerland, Switzerland
[f] Department of Computer Science, Sri Krishna Arts and Science College, Coimbatore, Tamil Nadu, 641008, India

ARTICLE INFO

Keywords:
Fault detection
State estimation
Automated systems
Robotics systems
Intelligent control

ABSTRACT

In the commercial and industrial sectors, automatic robotic control mechanisms, which include robots, end effectors, and anchors containing components, are often utilized to enhance service quality. Robotic systems must be installed in manufacturing lines for a variety of industrial purposes, which also increases the risk of a robot, end controller, and/or device malfunction. According to its automated regulation, this may hurt people and other items in the workplace in addition to resulting in a reduction in quality operation. With today's advanced systems and technology, security and stability are crucial. Hence, the system is equipped with fault management abilities for the identification of developing defects and assessment of their influence on the system's activity in the upcoming utilizing fault diagnostic methodologies. To provide adaptive control, fault detection, and state estimation for robotic automated systems intended to function dependably in complicated contexts, efficient techniques are described in this study. This paper proposed a fault detection and state estimation using Accelerated Gradient Descent based support vector machine (AGDSVM) and gaussian filter (GF) in automatic control systems. The Proposed system is called (AGDSVM + GF). The proposed system is evaluated with the following metrics accuracy, fault detection rate, state estimation rate, computation time, error rate, and energy consumption. The result shows that the proposed system is effective in fault detection and state estimation and provides intelligent control automatic control.

1. Introduction

Cities throughout the world are evolving into testing grounds for cutting-edge robotic and automation innovations that are used in several fields of business and society. Science fiction is giving way to truth as the 4th industrial transformation involves robotics and automated technology. The potential of robots is growing in all spheres of business and each aspect of daily life, made possible by enormous gains in computer capacity, proliferating data acquired by strong algorithms incorporated in digital systems, new technological advancements, and urban connection. Technological advancements access innovative avenues for the widely expanded use of robotics and automation in fabrication. Beyond the production line and in the city's consultations by corporate data systems, the existence of robotics and intelligent systems structures will change the reasoning, materialism procedures, effects, and methods of the urban sense in a distinctive way [1]. Technology and computing research are combined in the multidisciplinary topic of robotics. The smart interaction between observation and movement is what is referred to as robotics. The Robot University of America defines a robot as a reprogrammable, multifunctional operator created to handle objects such as resources, equipment, or customized equipment using several preprogrammed movements to carry out a range of activities. Robots come in a wide variety of designs and are utilized in a variety of settings and applications. Despite having a wide range of uses and forms, all

* Corresponding author.
E-mail addresses: rajesh.natarajan@shct.edu.om (R. Natarajan), santoshreddy@bnmit.in (S.R. P), bosesubash@islamiahcollege.edu.in (S.C. Bose), gururaj.hl@manipal.edu (H.L. Gururaj), francesco.flammini@supsi.ch (F. Flammini), shanmugapriyav@skasc.ac.in (S. Velmurugan).

https://doi.org/10.1016/j.array.2023.100298
Received 7 April 2023; Accepted 28 May 2023
Available online 1 June 2023

robots have three fundamental constructional features in common: (1) they all possess a few types of robotic development to carry out a specific assignment, (2) they all have electronic systems to energy and regulate the equipment, and (3) they all have some degree of computer coding code that determines when or how to do anything [2]. With the advent of intelligent industrial robots, sensor technology has advanced greatly, and automated robotic control systems have also undergone a considerable shift.

Fig. 1 shows the primary applications of robots. The advancements in artificial intelligence (AI) have a significant impact on developments in advanced automated robotic control. The industrial industry demands greater efficiency, flexibility to item changes, improved safety, decreased costs, etc., thus one of the primary topics of study is the use of AI in autonomous devices. Unfortunately, achieving these objectives using conventional control techniques is neither practical nor affordable. Industrial robots require programs that can adapt to various settings with minimal alterations that may be made by the equipment operators since being adaptable to variations is one of the requirements for success [3].

This might either relate to the precise detection of the presence of a defect in one particular member of the swarm, or to the broad reporting of a fault that has happened elsewhere in the network. After a defect has been discovered, the procedure of determining its root source is referred to as fault analysis. This entails locating the robot's system where the error appeared in the activity analyzed, however, it may even be as detailed as determining how that component exactly malfunctioned. By exchanging power with a group component whose power is getting down, for example, an identified problem may be fixed or minimized to enable the particular component to complete the work [4]. It is essential to encourage technology use and then sustain high efficiency to create autonomous robotics. Moreover, the idea of using an automated management platform to avoid fault and depletion defectiveness is investigated. This platform continuously monitors, forecasts, and estimates output and device efficiency to allow for the proper diagnosis and service. A standard robotic system consists of several components, which makes it susceptible to a variety of efficiency decline defects. These faults include transmitting system maintenance issues, synchronization belt problems, and equipment damage. These flaws may cause imbalance and harm to commercial robots [5].

A detector is a detecting tool that produces an electrical signal from a real physical amount (the so-called state) and delivers measuring information. Errors are introduced throughout the converting procedure. Drift fault and sensor noise are examples of these mistakes [6]. The drift fault can be removed by the sensor's adjustment, but the measuring distortion cannot be completely removed. As a result, the sensor's result cannot be entirely compatible with the condition that is being monitored. One of the most popular techniques for determining a state's most probable value based on observations and a structure is state estimate. It has been extensively employed in mobile robotics, unmanned aerial vehicles (UAVs), monitoring systems, intelligent grids, and healthcare condition monitoring and assessment, and is a need for high-level data collection and administration [7]. The assessment of states, including locations and orientations, is essential for any aerial robot technology. A strong basis is set by the estimating method for more advanced operations like route planning and tracking. The issue gets far more challenging when they consider an automated cluster of drones rather than just a single drone. Every drone in a swarm must determine its ego state as well as the comparative positions of the other drones. Aerial swarm applications in practical settings are now severely constrained by the widespread use of remote systems by robot scientists, such as motion-capturing devices and ultra-wideband (UWB) systems with anchoring to give state estimates [8]. To provide an efficient robot automatic control fault detection and state estimation are necessary. Hence, we suggested the AGDSVM-GF technique for fault detection and state estimation.

The following parts comprise the remaining content of this article. Part II provides a summary of the related literature that is currently accessible. In Part III, the AGDSVM-GF architecture is demonstrated. The design and testing outcomes of the AGDSVM-GF approach are presented in Part IV. In Part V, the conclusion of this work is established.

2. Literature survey

In the article [9], they established a method for real-time fault area recognition and validates closed-loop processing parameters modification for robot-related CFRP AM. The most ground-breaking aspect of that work is the creation of a DL model for accurate real-time fault detection, categorization, and rating. Two distinct kinds of CFRP flaws are detectable using the suggested approach (i.e., misalignment and abrasion). To provide a numerical value to the extent of each fault, they use a technique that combines DL with a geometric evaluation of the degree of misaligned. In the study [10], author examined the strengths and weaknesses of both statistical and ML approaches to failure prediction in robotic systems. More than 5000 robotics were used to

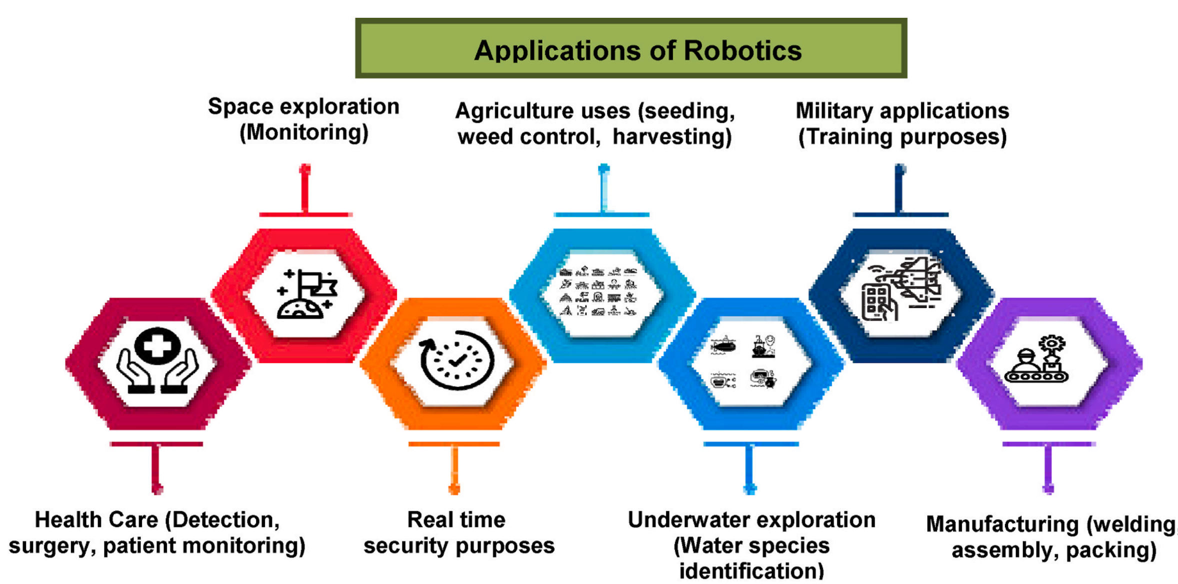

Fig. 1. Applications of robots.

compile the data used in that study. Also, a novel method called hybrid gradient enhancement is suggested, which combines traditional statistics and ML techniques. The study revealed that when contrasted with conventional ML and statistical approaches, hybrid gradient boosting significantly outperforms them. In the article [11], they developed new methods and approaches, all of which may be used with a wide range of physical machinery. Yet, conventional FDD techniques struggle to meet the special demands of the robotics sector. Few studies have been given on the topic of FDD for robotics since it is still a relatively young area of research. Conventional FDD techniques have been the focus of these reviews because of the possibility of their wide applicability to a general kind of robot. Yet, there are defining features of robotic systems that make them distinct from FDD in terms of restrictions and needs. Their goal in writing that piece is to help readers better understand how to use FDD methods that are tailored to the specifics of robotic devices. They go into further detail about the benefits of these methods and the difficulties they must overcome. To achieve that goal, they take a two-pronged approach: first, they describe FDD in detail from the point of view of the various features a robot arm may have, providing instances that demonstrate effective FDD approaches; second, they describe FDD in detail from the point of view of the various FDD strategies, analyzing the benefits and drawbacks of each methodology as they pertain to robotic devices. In the study [12], author provided a self-sufficient camera system for locating power line element problems using DL. Due to its impressive performance in element detection, they developed a DNN based on the YOLOv4 small topology. Those DL models were evaluated by running them on a variety of devices, including the Pic Microcontroller 4, Jetson Nano, Nvidia Jetson TX2, and Nvidia Jetson AGX Xavier, in a real-life setting. In the article [13], they provided a paradigm for the seismically active management of robotic manipulators that is susceptible to joint actuation failures. A problem diagnostic component based on NN and a responsibility controller based on reinforcement learning is both included in the suggested active fault-tolerant management system. An incremental RL controller will provide compensating torques after the transducer problem has been identified and treated, ensuring system security and maintaining controller parameters. In the study [14], author offered an inter-extraction of features and fusing following conditions defect diagnostic system for a robotic system. To start, a component from an LSTM network and a component from an inter-self-attention augmented deep convolution network is used to acquire knowledge about fault-based properties from several angles. In a further step, the global as well as local characteristics retrieved by the abovementioned components are combined for compounded fault identification. In the study [15], they offered a comprehensive overview of carrying problem diagnostics using DL. Classifier, Restricted Boltzmann Model, and CNN, three of the most prominent DL techniques for containing defect diagnostics, are described. Articles and works of study in the field of bearing defect diagnostics are discussed, as are their practical uses. In the article [16], author proposed a DL-based approach called Hierarchical Estimation of Surgical States using Deep Neural Networks, which simultaneously predicts the present super- and perfectly alright states. HESS-DNN utilizes information from the da Vinci Xi operating system's endoscope imaging, robot mechanics, and system changes. Using the HERNIA-20 dataset, which represents actual robotics inguinal hernia reconstruction surgery, HESS-DNN is tested and finds that it accurately guesses the operational federal state and the related perfectly alright operational state. They demonstrate that the hierarchical architecture enhances the state-of-the-art fine-grained estimation methods throughout the whole HERNIA-20 RAS operation. The development of a method in which a suture needle is introduced by the technician into one endoscope and mechanically removed by some other surgical tool is described. In the article [17], they suggested the penetration and pull states of the needles are estimated using YOLOv3 and a CNN. No matter how rigid the item that the sutures needle is placed into, graphics statistics suggest that can classify the state. Additionally, the location may be changed and the computerized surgical tool can reach the needle once again when the pull condition is identified, even if the needle tugging fails. In the study [18], they used robot movements as stimuli to develop and validate a hidden Markov model for assessing human emotional states in real time. Physiological signals including heartbeat, sweating rate, and face muscle contraction are responses to the system. Using a double valence-arousal formulation, the subjective state was assessed. Human volunteers were instructed to respond to moves created by a robotic manipulator that was anticipated during personal interaction. Furthermore assessed was the human body's physiological reaction. Both a conventional potential field generator and a newly published safe movement planner that reduces possible impact pressures along the route were used to create robot movements.

2.1. Problem statement

The result of the system will change depending on whether or not problems occur. The fault will result in the system providing erroneous results. Due to the widespread usage of robots in various industries, a fault in the robotics control system might result in a serious error. To improve the system, it is vital to identify any defects in decentralized robotic technology. It is impossible to directly examine certain machine states, and even if one could, detector error could prevent accurate observations from being made. Automatic control fault detection and state estimation are required to offer an effective robot. Hence, for fault detection and state estimation, we proposed the AGDSVM-GF approach.

3. Methodology

In this section, the proposed method is described. The robot attitude is collected as the dataset and collected from the industrial robots. These data were pre-processed using data correlation. The feature extraction process is done by using linear discriminant analysis (LDA). After this process, fault detection and state estimation are performed by AGDSVM-PF. The proposed flow is depicted in Fig. 2.

The mechanic's structure of the multi-joint robotic system is made up of some linking rods and connections, and at its core is an anthropomorphic robotic arm that is realized by an open chained mechanism with several dimensions of flexibility. The three-axial gyro, three-axial accelerometer, and three-axial magnetometer are always included in the micro-electro-mechanical structure (MEMS)-based orientation sensor. These devices are used to obtain angular speed data, vibratory velocity data, and magnetic field strength measurements. Using this, a multi-joint robot's 12-channel motion data may be collected, with 3-channel aspect angle signals being computed from the other 9-channel signals [19]. By modeling the multi-joint commercial robot in various failure modes, multiple attitude information can now be collected, and this information may then be utilized by the accompanying machine learning technique to build an effective fault detection system.

B. Data preprocessing using data correlation

As part of the data mining and data analysis process, preprocessing converts unstructured data into a format that machines can comprehend and interpret. The organization of the text, images, videos, and other real-world material is imprecise. It often has faults and inconsistencies, is difficult to understand, and lacks a cohesive design. A pre-processing step, a scaling approach, or a mapping technique is normalization.

3.1. Data correlation

Data preprocessing makes use of data correlation. To calculate robot sensor data correlation variables, use the equation below (1).

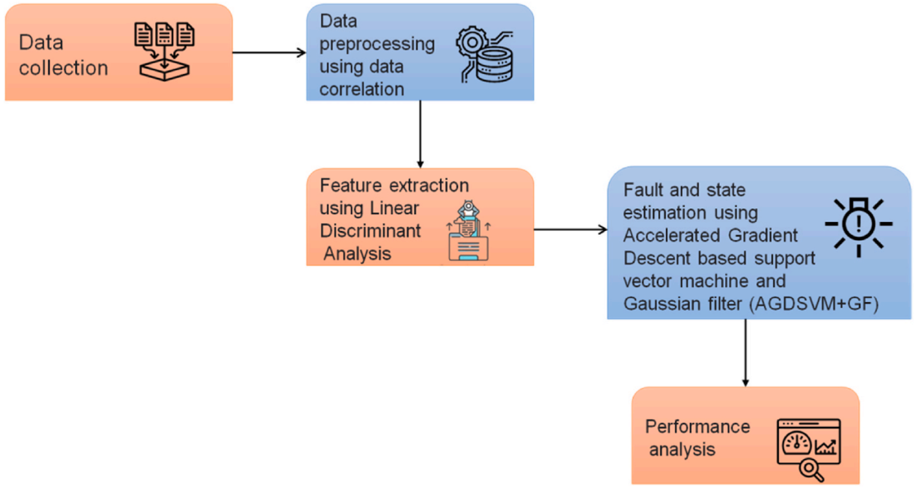

Fig. 2. Flow of the proposed work.

A. Dataset

$$r_y = \frac{\sum_{i=1}^{N-y}(y(x)-y)(y(x+y)-\bar{j})}{\sum_{x=1}^{N}(j(x)-j^2} \tag{1}$$

where N is the duration of the sensor data and y is the variation period. Also, the sensor data for it is represented by $y(x)$, where j is the median of the whole collection of robots sensor data. The closest node component reduction is used to identify the genuine sensor data after a matrix of computed correlation variables in a series of quality. In other words, it identifies ranges as choices if their values are greater than the value with the biggest reduction. The recognized robot data ($n(x)$) then give variance information that may be used as the proper classifier input. An equation (2) may be applied to create variance data at time t. Equation (2) will modify the quantity of different data generated from sensor data dependent on the variation intervals $n(x)$.

$$\Delta_{n(x)}t_0 = j(t) - j(t-n(x)) \tag{2}$$

This is performed to make sure that they are only produced using user data and to prevent bias brought on by situations like $n(x) = 0$ and $n < 0$. The preprocessed data are then utilized as the input data.

C. Feature extraction using Linear Discriminant Analysis

For detection approaches with big and high-dimensional data sets, massive computational complexity and a lack of storage may pose major issues. For the sake of clarity, this part begins by outlining a feature extraction challenge. One of the most basic and effective tools in the area is linear discriminant analysis (LDA), which has given rise to several feature extraction techniques. Because of the inconsistency of the within-class distribution, we might use LDA for feature extraction (FE) problems when the dimension d of the data is greater than the number N of the data. A FE problem is often referred to as a small sample size if d > Nc, where c is the number of classes. More importantly, it is feasible to know for sure that a large body of data has a big, obvious component. In other words, the LDA technique requires that the huge data have an identifiable major component and follow a Gaussian distribution. This suggests that the best data may be extracted and categorized using the LDA technique. Combining the original predictors to create a new variable is the goal of LDA. This is done by increasing the differences between the predetermined groups about the new variable. The two-class classification and the multi-class classification of the LDA technique each have their optimization function. The corresponding base vector is (a1, a2, …, ad), and the base vector component matrix is W, assuming that the function must project onto a less-feature space of integer values. The N x D matrix is what it is. P_a and P_b in Equation (3), according to Equation (4). The main diagonal parts of Y are added to create X. It could result in equation (5) which represents the O(V) optimization process.

$$P_a = \sum_{i=1}^{L} m_i(\mu_i - \mu\Big)(\mu_i - \mu) \tag{3}$$

$$P_b = \sum_{i=1}^{L}\sum_{y=Y} B\Big(b - \mu_i\Big)(b - \mu_i) \tag{4}$$

$$O(V) = \prod_{j=1}^{d} \frac{\omega_j^S T_a \omega_j}{\omega_j^S T_v \omega_j} \tag{5}$$

D. Fault detection and state estimation using AGDSVM + PF

One of the main goals of this robotic platform is fault detection. The sensor data are merged, and a proposed model utilizes the characteristics derived from all of the sensor data to evaluate if any of the instruments are in a defective state. This is because any number of sensors might be experiencing a problem. Support Vector Machine (SVM), which was primarily created for binary categorization and recovery mechanisms, is founded on support vectors. It was created to identify the hyper-plane that would best separate the two groups with the greatest margin. The SVM operates on the vector theory. To separate two classes and select the one with the greatest margin, it tries every feasible separating line, which is shown in Fig. 3.

Basic SVMs were deployed to data that could be separated continuously, however, kernel functions allow for application to non-linear information as well. It was noticed that certain information is not distinguishable in two-dimensional fields but when similar information is examined in higher dimensions plane it was possible to divide them continuously. This transition of the information from the lower dimensions level to the greater plane was performed with the aid of the kernel function. Several kernel functions, including multi and RBO, are accessible. The information was used to determine which kernel operations were necessary. While SVM was created for basic detection but it may be utilized for classification as well. Radius Basic Operator (RBO), also known as the kernel operator, is used by this classifier. Equation (6) contains the RBO's mathematical model.

$$k(i,j) = a^{-\gamma\|i-j\|^2} \tag{6}$$

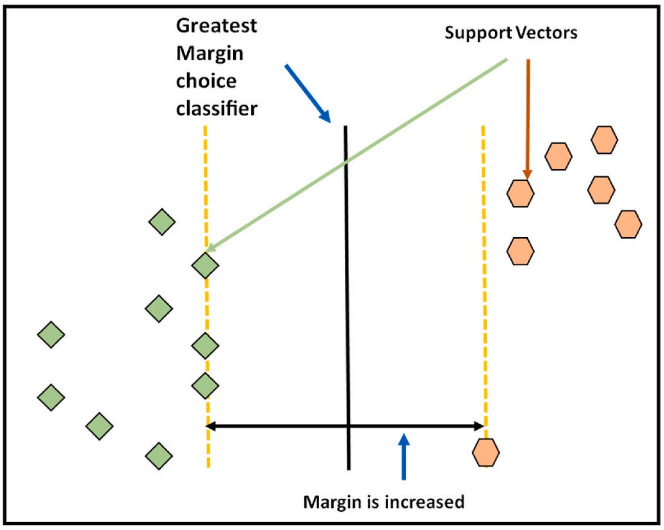

Fig. 3. SVM vector theory.

To identify faults, an SVM classifier is used and it is based on the characteristics analyzed by the DC. By using the one-again-stone categorization technique based on the two-class predictor, the SVM is achieved. This approach was chosen since it has a modest processing overhead and the number of defect types for multi-joint robotic systems is often limited. Because O is the number of fault classes presented in the final layer, the number of SVM processors built in the initial layer of the SVM is D = O(O-1)/2. Although if the source of the data comes from one of the other groups, each SVM classification only allocated it to one of two groups. The popular vote method is then used to allocate a test to the group that it is most commonly categorized into after the output of all the classifications has been merged.

The two-class SVM classifier's design principle is to create a higher dimensional space as the decision plane to most effectively distinguish the two categories. The training dataset collection in the suggested SVM is (x(1),y(1)), (x(2),y(2)),..x(N), y(N), where H is the dimensionality of characteristics obtained by DC. The test space must be converted to a higher dimension using a nonlinear mapping operator since they are linearly detachable. Equations (7) and (8) may then be utilized to represent the optimal solution that was used to build the best dividing the higher dimensional area in this region.

$$\min_{m,d,\varepsilon(i)} \frac{1}{2}\|m\|^2 + p\sum_{i=1}^{N}\varepsilon(i) \tag{7}$$

$$s.t.\begin{cases} y(i)[m.\zeta(x(i)) + d] \geq 1 - \varepsilon(i) i = 1,2,3,..,N \\ \varepsilon(i) \geq 0 \; i = 1,2,3,..,N \end{cases} \tag{8}$$

where "." represents the inner product of vectors, m is the standard matrix indicating the direction of the hyperplane, p is the penalized coefficient, (ε)i is the slacking factor, d is the dislocation defining the separation between the hyperplane and the foundation position. On the premise of the Karush-Kuhn-Tucher (KKT) criterion, the double model of the aforementioned model may be derived by adding a Lagrange multiplier γ (i) for every requirement in equation (9).

$$\max_{\gamma} \sum_{i=1}^{N}\lambda(i) - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\gamma(i)\gamma(j)y(i)y(j)\zeta(x(j)) \tag{9}$$

$$s.t.\begin{cases} \sum_{i=1}^{N} y(i)\gamma(i) = 0 \\ 0 \leq \gamma(i) \leq p, i = 1,2,\ldots..,N \end{cases} \tag{10}$$

where C is a constant that is not equal to 0. Implementing this strategy can guarantee that SVM employs a soft margin rather than a hard margin for identification as long as p is not an unlimited quantity. The decision value in equation (11) may now be obtained by solving the aforementioned quadratic computing framework utilizing the optimizing procedure.

$$f(x) = sgn\left\{\sum_{i=1}^{N}\tilde{\gamma}(i)y(i)\zeta(x(i)).\zeta(x(j)) + \tilde{d}\right\} \tag{11}$$

where the component values for the model's optimum response are $\tilde{\gamma}(i)$ and d. Calculating the inner product $\zeta(x(i)).\zeta(x(j))$ is tricky since the Hilbert space's size might be quite large. To circumvent this difficulty, a kernel process $K(x(i)),(x(j))$ is employed to substitute it, and the selection capability may thereafter be reformulated as in equation (12). This shows the faults in the robotic system.

$$f(x) = sgn\left\{\sum_{i=1}^{N}\tilde{\lambda}(i)y(i)K(x(i)),(x(j)) + \tilde{d}\right\} \tag{12}$$

The noise dispersion features of current technologies are frequently more intricate. Assuming the program's disturbance can be reduced into the sum of many Gaussian noises, which is what we term a multimode issue. Thus, for every Gaussian noise element, we may apply the filtering outlined in the preceding part after first decomposing the noise. If equation 10's representation of p(x) has n Gaussian elements, then it is a mixture of Gaussian functions.

Assume that p(x) has n Gaussian elements and is a hybrid Gaussian dispersion which is shown in equation (13).

$$p(x) = \sum_{i=1}^{n}\varpi_i N(x;\mu_i,P_i) \tag{13}$$

where every Gaussian element, $n(x;\eta_i,v_i)$, and the weight satisfy the conditions $\sum_{i=1}^{n}\omega_i = 1$.

The weight ϖ_i median η_i and variance v_i of every Gaussian element must then be determined. The estimates based on every element with Gaussian noise may then be obtained using the Gaussian filter. The weighted total of all the estimates may finally be calculated by ϖ_i. Typically, it is required to decrease the number of Gaussian elements to prevent finding the following computations too difficult due to an excessive amount of elements. In other terms, it's important to define something as precisely and simply as necessary. By choosing a few of the elements with higher weights and eliminating the ones that are not important, element-reducing techniques involve keeping those essential elements that have an impact on the dispersion pattern. Moreover, the overall number of elements is decreased by merging comparable elements. The procedure of the gaussian filter is as follows

- The overall distortion is a complex one with several Gaussian elements.
- The system's complicated Gaussian distortion is divided into several normal Gaussian disturbances, $P_i = 1, 2, \ldots\ldots$n
- Depending on every Gaussian distortion, an estimation of the state x_i is made.
- Utilizing the estimated outcome and the load, the state prediction outcome depending on the complex mixture of high distortion is produced.

While most optimization work is well-liked and often utilized, the learning phase may sometimes be drawn out. It would be good to do more study on how to modify the learning speed, speed up divergence, and avoid becoming stuck at local optima when searching. The science of dynamics, which replicates an element's momentum, is where the idea of speed originates. Incorporating momentum into the learning rate is meant to maintain a specific amount of the prior updating direction's impact on the next repetition. When working with high distortion, tiny but constant slopes, or noisy contours, the speed approach may accelerate settlement. The velocity variable, introduced by the velocity method, denotes the orientation and the speed of a parameter's motion in the variable field. The average parabolic decline of the negative slope is used to determine the velocity. The velocity change in the gradient descent algorithm is always $.\left(-\frac{\partial P(\theta)}{\partial(\theta)}\right)$. By utilizing the momentum technique, the updated v amount is not merely the gradient descent degree determined by $\varepsilon.\left(-\frac{\partial P(\theta)}{\partial(\theta)}\right)$. Moreover, the friction coefficient is taken into consideration, which is denoted by the prior updating χ^{old} divided by a speed ratio between [0, 1]. Typically, an item's weight is fixed to 1. Equation (14) represents the solution.

$$\chi = \varepsilon.\left(-\frac{\partial P(\theta)}{\partial(\theta)}\right) + \chi^{old}.S \tag{14}$$

S represents the speed component. The prior speed may expedite this search if the present gradient and the prior rate χ^{old} are similar. When the training rate is low, the correct velocity contributes to accelerating the converging. Friction will reduce the derivative's decline to 0, which will cause it to maintain v until it reaches balance. Avoiding the minima during training is advantageous for accelerating the convergence of the search phase. The quantity χ^{old} will slow down this query if the gradient at the moment is the reverse of the gradient at the time of the last upgrade χ^{old}.

When the learning rate is high, the speed approach with the suitable speed component helps to reduce the fluctuation of resolution. Another issue is how to choose the correct speed factor amount. A faster converging rate is difficult to achieve when the speed component is minimal. The present point can leave the ideal rating point if the speed component is high. Many investigations have shown experimentally that putting the speed factor at 0.9 is the best choice. The classic speed approach is significantly enhanced by Accelerated Gradient Descent (ADG). The speed χ^{old} s is increased to, which is represented as e in speed. While refreshing, the gradient of e is utilized. The following are the full updating formulas for the variable defined in 15.

$$\begin{cases} \widetilde{\theta} = \theta + \chi^{old}.s \\ \chi = \chi^{old}.s + \eta.\left(-\frac{\partial P(\widetilde{\theta})}{\partial(\theta)}\right) \\ \theta' = \theta + \chi. \end{cases} \tag{15}$$

The gradient of the upcoming location rather than the present location is updated to represent the increase of speed over velocity. The upgrade equation reveals that the AGD speed technique incorporates more gradient data than the conventional speed approach. When random optimizing is not used, it should be noted that AGD speed enhances the descent from $O\left(\frac{1}{k}\right)$ to $\left(\frac{1}{k^2}\right)$. The magnitude of the learning rate should also be taken into consideration. If the search is conducted near the ideal location, the fluctuation is more probable to take place. As a result, the learning speed has to be changed. The SGD's speed approach often employs the learning level loss factor l, which causes the learning level to decline for each repetition cycle. Equation (16) defines the formula for the learning level decrease.

$$\zeta_t = \frac{\zeta_0}{1 + l.t} \tag{16}$$

where i is an integer in the range [0, 1] and t is the learning speed at repetition r, where 0 is the initial learning level. The equation indicates that the learning speed will decline more slowly the smaller the value of i. When i is equal to 0, the learning level is unaffected, and when i is equal to 1, the learning level starts to decline. In this way, the learning rate of the process is enhanced, and it produces optimal detection.

4. Result and discussion

Many functional procedures are becoming more reliant on high-precision automated robots as a result of the significant expense savings and labor benefits that these robots provide. Methods for fault detection and state estimation are required to increase the efficiency of this robotics performance. This part contains an assessment of the suggested approach. Accuracy, fault detection rate, state estimation rate, computation time, error rate, and energy consumption are the factors taken into consideration while evaluating a system. The existing systems used for comparison are Variable Gain Super Twisting Sliding Mode (VGTWSM) [20], Fault detection and exclusion (FDE) [21], Kinematic state estimator based on the Extended Kalman filter (KSE-EKF) [22], and Multiple-order-holder (MOH) [23].

A. Accuracy

An assessment of accuracy is when the outcome of the measurement agrees with the proper value or guideline. The percentage of predictions that the model correctly predicted is known as accuracy. Any company that must anticipate and identify defects must be evaluated for accuracy. Fig. 4 displays the accuracy of the existing models and the suggested model accuracy. Table 1 shows the results of accuracy. For defect detection and state estimation in autonomous robot control, the proposed method works effectively.

The procedure of detecting faults in actual processes while trying to determine the issue's origin is known as fault detection. In expensive and crucially necessary procedures for security, fault detection is crucial. Earlier procedure fault identification helps prevent the development of abnormal events. The process of fault detection may be carried out in several ways. This demonstrates that the suggested approach is capable of detecting faults in robotic control systems in an efficient manner. Fig. 5 displays the fault detection rate of the existing models and the suggested model accuracy. Table 2 shows the results of the fault detection rate.

State estimation is the action of finding out an efficiency system's intrinsic state utilizing input/output information readings and a computational framework. It is a crucial component of robot system management centers' digital safety assessment capability. It is an essential component of the core hardware and software architecture that underpins the functioning of the power network. This demonstrates that the recommended approach estimates the robotic system appropriately. Fig. 6 displays the fault detection rate of the existing models and the suggested model accuracy. Table 3 shows the results of the state estimation rate.

The computation time is the amount of time required for a calculation to be completed (or running time). The number of times the relevant rules must be applied directly relates to the length of time required to do computations. One may consider these guidelines to be a description of the computation itself. The system will operate more efficiently if the detection can be made in a shorter period. Fig. 7 displays the

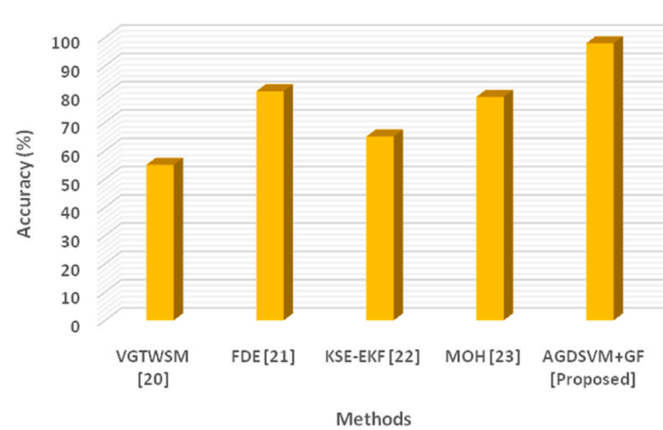

Fig. 4. Accuracy of the existing and proposed model.

Table 1
Accuracy results.
B.Fault detection rate

Methods	Accuracy (%)
VGTWSM [20]	55
FDE [21]	81
KSE-EKF [22]	65
MOH [23]	79
AGDSVM + GF [Proposed]	98

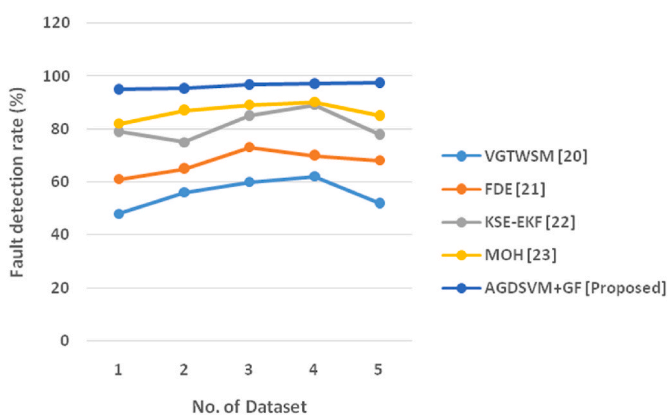

Fig. 5. Fault detection rate of the existing and proposed model.

Table 2
Fault detection rate results.
C.State estimation rate

No. of Dataset	Fault detection rate (%)				
	VGTWSM [20]	FDE [21]	KSE-EKF [22]	MOH [23]	AGDSVM + GF [Proposed]
1	48	61	79	82	95
2	56	65	75	87	95.5
3	60	73	85	89	96.7
4	62	70	89	90	97
5	52	68	78	85	97.5

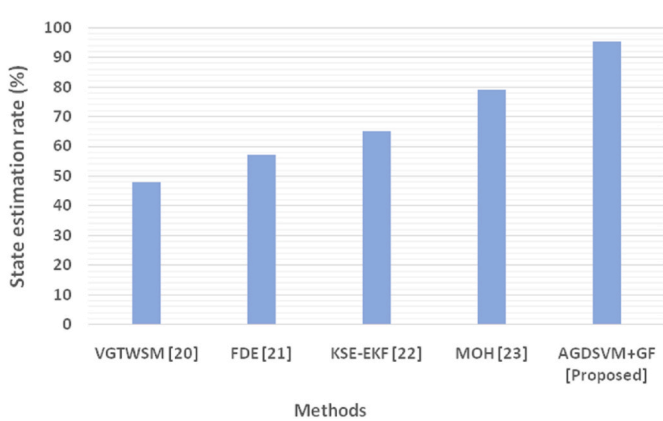

Fig. 6. State estimation rate of the existing and proposed model.

computation time of the existing models and the suggested model accuracy. Table 4 shows the results of computation time. It shows that the proposed modal can identify the state and fault in a less time-consuming manner.

The term error rate refers to a measurement of the amount of a model's prediction error about the real output. The phrase error rate is often used to identify systems. The method result is reliable if the error rate is low. Fig. 8 displays the fault detection rate of the existing models and the suggested model accuracy. Table 5 shows the results of the fault detection rate. It demonstrates that there is less error in the proposed modal detection.

The total amount of energy used by the strategy is referred to as energy consumption. It is the variation between the power structure's

Table 3
State estimation rate results.
D.Computation time

Methods	State estimation rate (%)
VGTWSM [20]	48
FDE [21]	57
KSE-EKF [22]	65
MOH [23]	79
AGDSVM + GF [Proposed]	95.2

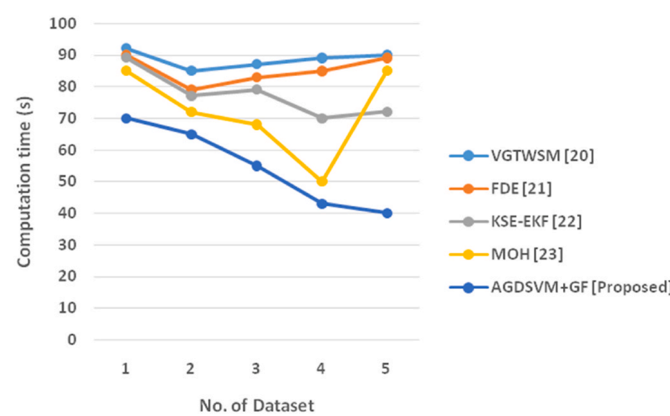

Fig. 7. Computation time of the existing and proposed model.

Table 4
Computation time results.
E.Error rate

No. of Dataset	Computation time (s)				
	VGTWSM [20]	FDE [21]	KSE-EKF [22]	MOH [23]	AGDSVM + GF [Proposed]
1	92	90	89	85	70
2	85	79	77	72	65
3	87	83	79	68	55
4	89	85	70	50	43
5	90	89	72	85	40

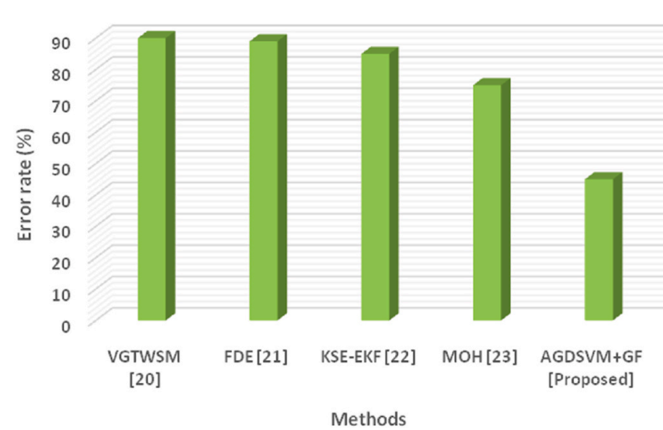

Fig. 8. Error rate of the existing and proposed model.

Table 5
Error rate results.
F.Energy consumption

Methods	Error rate (%)
VGTWSM [20]	90
FDE [21]	89
KSE-EKF [22]	85
MOH [23]	75
AGDSVM + GF [Proposed]	45

original state and its present condition. Examining the Figure demonstrates that the suggested method makes use of inventive technology and uses less power overall as a consequence of this operation. This indicates that the method's energy requirements for fault detection and state estimation are less. Fig. 9 displays the energy consumption of the existing models and the suggested model accuracy. Table 6 shows the results of energy consumption.

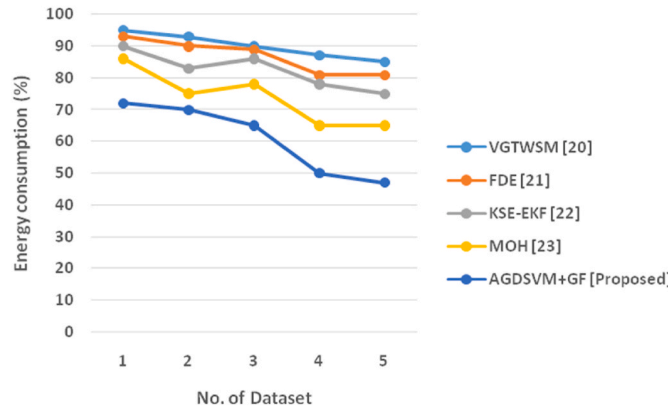

Fig. 9. Energy consumption of the existing and proposed model.

Table 6
Energy consumption results.

No. of Dataset	Energy consumption (%)				
	VGTWSM [20]	FDE [21]	KSE-EKF [22]	MOH [23]	AGDSVM + GF [Proposed]
1	95	93	90	86	72
2	93	90	83	75	70
3	90	89	86	78	65
4	87	81	78	65	50
5	85	81	75	65	47

5. Discussion

In our research, effective methods for providing adaptive control, defect identification, and state prediction for robotic automation machines designed to reliably work in complex contexts are outlined. In this research, we offer a leakage and state-estimation method for autonomous control networks that makes use of the Accelerated Gradient Descent based support vector machine (AGDSVM) and the gaussian filter (GF). (AGDSVM + GF) is the name given to the developed framework, further compared with other methods, in the study [20], they analyzed that the fault-tolerant controllers based on traditional sliding mode control exhibit asymptotic convergence, weaker resilience, shorter transient performance, and a considerable amplitude of chattering, limiting its applicability in real-time applications. They proposed the Variable Gain Super Twisting Sliding Mode (VGTWSM) for this purpose. In the article [21], they suggested the fault detection and exclusion (FDE) technique develops residuals and thresholding utilizing an informative measure known as the Bhattacharya Distance (BD). For state estimation, an expanded information filter is utilized. To identify anomalous sensor measurements, fault signals termed residuals are created by measuring the BD between the spectators's a priori and a probability distribution. A bank of data filters is constructed for system testing, and a set of royalty payments is suggested by calculating the BD among the anticipated distributions and the actual dispersion received from sensor observations. In the study [22], they demonstrate that numbers might be used to distinguish encoder data and developed Kinematic state estimator based on the Extended Kalman filter (KSE-EKF). Unfortunately, due to low-pass filtration, the calculated velocities and accelerations are either erratic or delayed. The use of gyroscopes and altimeters may avoid mathematical translation, however, these devices have some detection flaws and are nonlinear for the required parameters. In the article [23], they demonstrate that some series of shifting gate functionalities are used to simulate the incidence seconds and proposed Multiple-order-holder (MOH) for fault identification. The "occurrence frequency" is parameterized using the lowest and highest interval durations, two exceptional case indices. The targeted outlier may have a norm that is bigger than a predetermined specified limit, which sets it apart from the well-researched standard noise. According to our existing methods, the computation time is high and the error prediction is a much higher rate. As we concluded that our recommended methods achieve good error prediction, and maintain computation time and energy efficiency are much maintained.

6. Conclusion

Robotics and automated systems employ software, control mechanisms, and data systems to operate economic machines and operations, substituting human labor and enhancing effectiveness, productivity, velocity, and reliability. Robotics is utilized in a wide range of applications and improves every industry. Even though it offers many advantages, there are some errors. It's critical to improve fault detection and state estimation in robotics systems. Many computerized systems, including controllers, robotic systems, robotics, etc., employ fault detection and state estimation. Hence, we presented the Accelerated Gradient Descent based support vector machine and gaussian filter in automatic control systems (AGDSVM + GF). Data on robotic attitude were used as the dataset. Preprocessing and feature extraction on the data was done. Using performance measures, the outcomes were assessed and contrasted with previously used methods. Accuracy (98%), fault detection (97.5%), state estimation (95.2%), computation time (40s), error rate (45%), and energy consumption (47%) are the results' outcomes. It demonstrates that the proposed method is effective at fault detection and determining the states of robotic systems. Future versions of the proposed system might add innovative technology to reduce error and improve detection performance.

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.

Data availability

No data was used for the research described in the article.

References

[1] Macrorie R, Marvin S, While A. Robotics and automation in the city: a research agenda. Urban Geogr 2021;42(2):197–217.
[2] Adel S, Zaher A, El Harouni N, Venugopal A, Premjani P, Vaid N. Robotic applications in orthodontics: changing the face of contemporary clinical care. BioMed Res Int 2021;2021:1–16.
[3] Arents J, Greitans M. Smart industrial robot control trends, challenges, and opportunities within manufacturing. Appl Sci 2022;12(2):937.

[4] Miller OG, Gandhi V. A survey of modern exogenous fault detection and diagnosis methods for swarm robotics. Journal of King Saud University-Engineering Sciences 2021;33(1):43–53.

[5] Huan-Kun HSU, Hsiang-Yuan TING, Huang MB, Huang HP. Intelligent fault detection, diagnosis, and health evaluation for industrial robots. Mechanics 2021; 27(1):70–9.

[6] Song S, Xiong H, Lin Y, Huang M, Wei Z, Fang Z. Robust three-phase state estimation for PV-Integrated unbalanced distribution systems. Appl Energy 2022; 322:119427.

[7] Jin XB, Robert Jeremiah RJ, Su TL, Bai YT, Kong JL. The new trend of state estimation: from model-driven to hybrid-driven methods. Sensors 2021;21(6): 2085.

[8] Xu H, Zhang Y, Zhou B, Wang L, Yao X, Meng G, Shen S. Omni-swarm: a decentralized omnidirectional visual–inertial–UWB state estimation system for aerial swarms. IEEE Trans Robot 2022;38(6):3374–94.

[9] Lu L, Hou J, Yuan S, Yao X, Li Y, Zhu J. Deep learning-assisted real-time defect detection and closed-loop adjustment for additive manufacturing of continuous fiber-reinforced polymer composites. Robot Comput Integrated Manuf 2023;79: 102431.

[10] Costa MA, Wullt B, Norrlöf M, Gunnarsson S. Failure detection in robotic arms using statistical modeling, machine learning, and hybrid gradient boosting. Measurement 2019;146:425–36.

[11] Khalastchi E, Kalech M. On fault detection and diagnosis in robotic systems. ACM Comput Surv 2018;51(1):1–24.

[12] Ayoub N, Schneider-Kamp P. Real-time on-board deep learning fault detection for autonomous UAV inspections. Electronics 2021;10(9):1091.

[13] Yan Z, Tan J, Liang B, Liu H, Yang J. Active Fault-tolerant control integrated with reinforcement learning application to robotic manipulator. In: 2022 American control confer*ence (ACC)*. IEEE; 2022, June. p. 2656–62.

[14] Zhou X, Zeng H, Chen C, Xiao H, Xiang Z. An attention-enhanced multi-modal deep learning algorithm for robotic compound fault diagnosis. Meas Sci Technol 2022; 34(1):014007.

[15] Hoang DT, Kang HJ. A survey on deep learning-based bearing fault diagnosis. Neurocomputing 2019;335:327–35.

[16] Qin Y, Allan M, Burdick JW, Azizian M. Autonomous hierarchical surgical state estimation during robot-assisted surgery through deep neural networks. IEEE Rob Autom Lett 2021;6(4):6220–7.

[17] Mikada T, Kanno T, Kawase T, Miyazaki T, Kawashima K. Suturing support by human cooperative robot control using deep learning. IEEE Access 2020;8: 167739–46.

[18] Hu L, Wu C, Pan W. Lyapunov-based reinforcement learning state estimator. 2020. *arXiv preprint arXiv:2010.13529*.

[19] Long J, Mou J, Zhang L, Zhang S, Li C. Attitude data-based deep hybrid learning architecture for intelligent fault diagnosis of multi-joint industrial robots. J Manuf Syst 2021;61:736–45.

[20] Nandanwar A, Tripathi VK, Behera L. Fault-tolerant control for multi-robotics system using variable gain super twisting sliding mode control in cyber-physical framework. In: 2021 IEEE/ASME international conference on advanced intelligent mechatronics (AIM). IEEE; 2021, July. p. 1147–52.

[21] Abci B, Nader J, El Badaoui El Najjar M, Cocquempot V. Fault-Tolerant multi-sensor fusion and thresholding based on the bhattacharyya distance with application to a multi-robot system. on November. In: 15th European workshop on advanced control and diagnosis (ACD 2019) proceedings of the workshop held in bologna, Italy. 21–22. Cham: Springer International Publishing; 2022, June. p. 347–64. 2019.

[22] Fennel M, Driller L, Zea A, Hanebeck UD. Calibration-free IMU-based kinematic state estimation for robotic manipulators. In: 2022 IEEE international conference on multisensor fusion and integration for intelligent systems (MFI). IEEE; 2022, September. p. 1–8.

[23] Zou L, Wang Z, Hu J, Dong H. Partial-node-based state estimation for delayed complex networks under intermittent measurement outliers: a multiple-order-holder approach. IEEE Transact Neural Networks Learn Syst 2022.