

# Intelligent Fault Detection, Diagnosis and Health Evaluation for Industrial Robots

Huan-Kun HSU\*, Hsiang-Yuan TING\*\*, Ming-Bao HUANG\*\*\*, Han-Pang HUANG\*\*\*\*

\*National Taiwan University, Taipei 10617, Taiwan, E-mail: D00522029@ntu.edu.tw

\*\*National Taiwan University, Taipei 10617, Taiwan, E-mail: D98522035@ntu.edu.tw

\*\*\*National Taiwan University, Taipei 10617, Taiwan, E-mail: D00522011@ntu.edu.tw

\*\*\*\*National Taiwan University, Taipei 10617, Taiwan, E-mail: hanpang@ntu.edu.tw

**crossref** <http://dx.doi.org/10.5755/j02.mech.24401>

## 1. Introduction

To develop an intelligent health diagnosis system for industrial robots, it is crucial to promote equipment utilization and then maintain high throughput. A concept explored here is the employment of an intelligent maintenance system (IMS) [1] aimed at preventing failure and degradation-phase defectiveness through the unremitting monitoring, prediction and estimation of equipment and production performance to enable appropriate diagnosis and maintenance. Condition-based maintenance (CBM) is an important part of an IMS. According to [2], intelligent CBM includes several fundamental elements: condition monitoring, fault diagnosis and prediction of a trend of equipment deterioration.

There are three main approaches to fault detection and diagnosis (FDD), including the knowledge-based, model-based, and data-driven approaches [3]. Knowledge-based approaches generally imitate the behavior of a human expert in checking a system's operation. This kind of approach can be easily implemented but difficult to use in a sophisticated multivariable robotic system. Model-based approaches are completely dependent on prior knowledge about the system model. By comparing the behaviors produced by the current system with the normal operation model, this approach checks for and further diagnoses any fault types. The major disadvantage of this method is that it is very difficult to model complex robotic systems precisely and quickly. Contrary to model-based approaches, data-driven approaches rely only upon sampled data to extract useful information that enables system condition monitoring and diagnosis. Moreover, the primary advantage of data-driven approaches is that they are model free.

Compared to a general mechanical system, robot manipulators usually have more degrees of freedom (DOFs) and need more sensors to monitor the system state, which means that the data-driven approaches are most suitable for complex robotic systems. Data-driven approaches can be further subdivided into two categories. One category is the machine learning method, such as demonstrated by [4], who used a neural network (NN) to model the fault function for fault detection, [5], who used an NN to model residual torque for fault detection, and [6], who used not only an adaptive neuro fuzzy inference system to model the system but also an NN to diagnose faults. However, the obvious limitations with those methods are the quantity and quality of training data.

The other category is the statistics-based method, as seen with [7] showing several outlier detection methods

that can be used for fault detection, [8], who used the Wiener process to predict the fault of the bearings, [9], who used partial least squares to estimate the dynamic parameters for fault detection, [10], who used SPC charts to detect faults for an industrial robot. While outlier types are intuitive and easy to implement, a suitable threshold setting is critical with these methods.

Because an industrial robot typically has a number of parts, numerous kinds of performance degradation faults can occur, such as from a lack of lubrication in transmission mechanisms, a timing belt coming loose, and wear and tear on gears. These faults can lead to industrial robot instability and damage. To avoid such situations, a health evaluation should be added to the maintenance management system. Accordingly, the proposed system presented in this study includes fault detection, fault diagnosis and health evaluation. The proposed fault detection method is based on the statistical process control (SPC) and uses principal component analysis (PCA) [11] to reduce the variable dimensions for multivariate monitoring. Furthermore, the Nelson rules [12] are used to more effectively and sensitively set fault detection thresholds. In applying the predominant statistical theory in a small sample population and abundant nonlinear kernel function of support vector machines (SVMs), the proposed fault diagnosis method is developed based on the multi-class SVM [13] to classify the faults of an industrial robot. Additionally, in the proposed health evaluation method, the robot health index (RHI) generator uses fuzzy logic to evaluate the health of the industrial robot.

This study is organized into four sections. Section 2 proposes the architecture for the online fault detection, diagnosis and health evaluation system. Section 3 presents the proposed method for the complete system. Then Section 4 discusses the results of experiments with an actual industrial robot to verify the effectiveness of the proposed system's functions. Finally, conclusions are discussed in Section 5.

## 2. System architecture

As shown in Fig. 1, the system is proposed to be an online implementation of a solution to the problem of industrial robotic faults. The structure consists of three subsystems: a fault detection subsystem, a fault diagnosis subsystem and a health evaluation subsystem. The fault detection subsystem, designed for continuous monitoring of all kinds of robot tasks, uses online sensor data to check the operation status of the industrial robot in real time. After pre-processing and being multiplied by the projection model, online

sensor data can be transformed into representative information that can be used by the SPC to identify variations in robot performance. When the operation status of a robot reaches a warning threshold, the fault diagnosis subsystem is launched after the robot stops its current task. Moreover, the pre-processing weights, projection model and control limits of the SPC are first determined by the historical normal data of the current task. If the warning conditions are not met, the next step will be the conditions of health evaluation process determined by either the duration of work or the user specification. If these conditions are met, the health evaluation subsystem is launched after the robot stops its current task. Oppositely, the fault detection subsystem returns to the starting point to proceed with the next cycle.

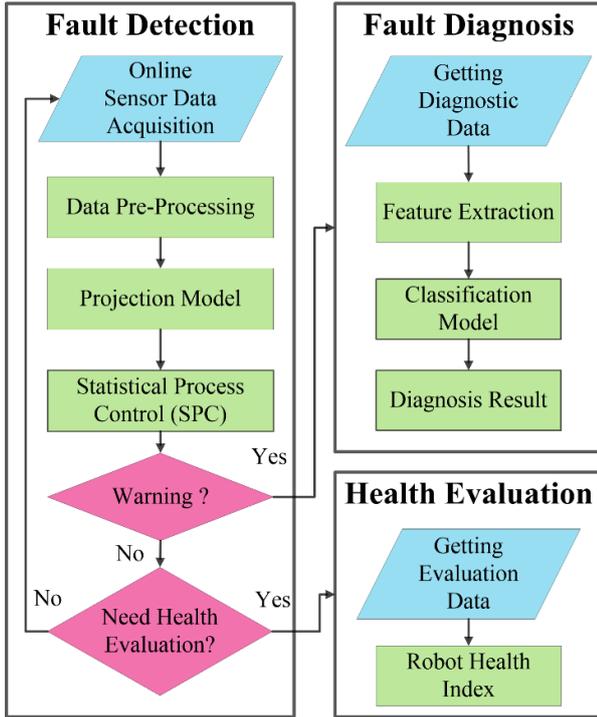


Fig. 1 Intelligent online fault detection, diagnosis and health evaluation system

In the fault diagnosis subsystem, the robot will first perform a diagnostic operation of each joint, and the features of the testing data are then extracted. These features will be input to the classifier to diagnose the robot's current faults. Based on the diagnosis results, the appropriate maintenance can then be performed to prevent serious damage to the industrial robot. Moreover, the classifier is trained by the historical normal features acquired from the original diagnostic operation.

In the health evaluation subsystem, the robot will also perform an evaluation operation of each joint which is the same as the diagnostic operation to acquire the evaluation data. Then the *RHI* can be calculated to evaluate the health condition of the industrial robot. The required calculation distributions are also decided by the historical normal data of original evaluation operations. These three subsystems will be introduced and discussed in detail in the next section.

### 3. Fault detection, diagnosis and health evaluation

In this section, the fault detection, fault diagnosis and health evaluation will be described in more detail.

#### 3.1. Fault detection

Assuming the industrial robot performs a repetitive task, the subsystem will try to identify the variations seen by all of the sensors online. Before online fault detection, the subsystem will build the required pre-processing weights, projection model and SPC control limits from historical normal data of the current task, as shown in Fig. 2. The related parts of the offline building process and the online fault detection process are described below.

##### 3.1.1. Sensor data pre-processing

In total, the sensors include six motor encoders, six motor current sensors and one external single-axis accelerometer. Because of the importance of the industrial robot's performance, errors of historical normal operations and running operations in a task are identified using the following equations:

$$q_{err,i,j} = q_{cmd,i,j} - q_{fb,i,j}, \quad (1)$$

$$\dot{q}_{err,i,j} = \dot{q}_{cmd,i,j} - \dot{q}_{fb,i,j} \quad \text{and} \quad (2)$$

$$\tau_{err,i,j} = \tau_{ref,i,j} - \tau_{fb,i,j}, \quad (3)$$

where:  $q$ ,  $\dot{q}$  and  $\tau$  denote the joint angle, joint velocity and joint torque values of each motor. The subscript *cmd* means the variable is the command value of the desired task. The subscript *fb* means the variable is a feedback value. The subscript *ref* means the variable is the average of the historical normal operation data. The subscript  $i$  refers to the joint index of the robot, and the subscript  $j$  refers to the sample number index of the task. The joint velocity data is calculated from the joint angle data. These error type variables can clearly represent the performance of an industrial robot and are not easily affected by the contents of the task. Ideally, the smaller these errors are, the better the results are. When the operation status of a robot becomes abnormal, the errors of motion will likely be greater. In addition, vibration data from an external accelerometer will be input directly. Therefore, there are total of 19 channels of input data in this subsystem. Because these values from different sensors have different units and scales, a weight tuning process is required. The normalization processes of data shifting and scaling are executed for all historical normal operation data to generate two weight vectors with a length of 19. Then the two vectors are used to pre-process data for the offline building process and online fault detection, as shown in Figs. 1 and 2.

##### 3.1.2. Projection model

With sensor-based machine fault detection, it is not uncommon for data to be collected on dozens of process variables. Clearly, this calls for an approach that can visually display important quantities and still have the sensitivity to detect special causes of variation.

Because of the ability to discriminate direction with the largest variance in a data set that consists of a large number of interrelated variables, PCA is usually used to identify the most representative features [11]. Hence, the operation status of a machine can be interpreted against

those projected data over the first few principal components easily and sensitively. This can save many efforts to observe all the original parameters.

The projection model can be built by a PCA of pre-processed historical normal operation data offline, as shown in Fig. 2. Let the original data matrix be:

$$\mathbf{X} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m,1} & x_{m,2} & \cdots & x_{m,n} \end{bmatrix}, \quad (4)$$

where:  $m$  is the sensor variable's number, which is 19 here, and  $n$  is the data number, which is a multiple of the sample number of the task here. In addition, each row has been pre-processed for normalization. Then the covariance matrix of  $\mathbf{X}$  can be calculated as:

$$\begin{aligned} \boldsymbol{\Sigma}_X &= E(\mathbf{X}\mathbf{X}^T) = \\ &= \frac{1}{n} \begin{bmatrix} \sum_{k=1}^n x_{1,k}^2 & \sum_{k=1}^n x_{1,k}x_{2,k} & \cdots & \sum_{k=1}^n x_{1,k}x_{m,k} \\ \sum_{k=1}^n x_{2,k}x_{1,k} & \sum_{k=1}^n x_{2,k}^2 & \cdots & \sum_{k=1}^n x_{2,k}x_{m,k} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{k=1}^n x_{m,k}x_{1,k} & \sum_{k=1}^n x_{m,k}x_{2,k} & \cdots & \sum_{k=1}^n x_{m,k}^2 \end{bmatrix}, \quad (5) \end{aligned}$$

where: the operator  $E$  denotes the expected value of its argument. The principal components can be acquired by eigenvectors of  $\boldsymbol{\Sigma}_X$  and, according to the size of the corresponding eigenvalues, one or two representative principal components can be chosen to be the projection model. This projection model is used for transforming pre-processed historical normal operation data and running data into projected values, as shown in Figs. 1 and 2, where the former projected values are used to define the control limits, and the latter ones show the state of a running machine. The projected values will form trajectories in a single task.

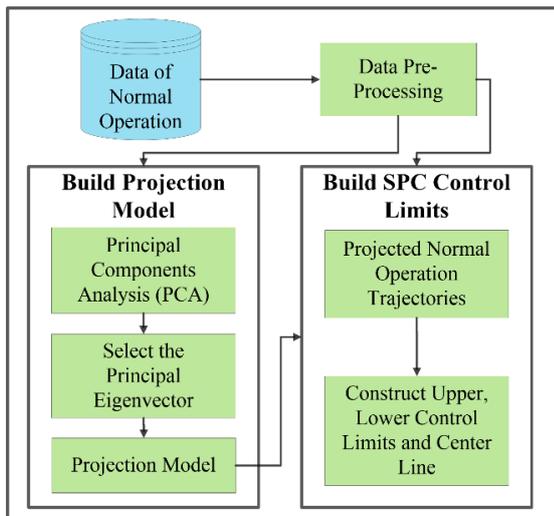


Fig. 2 The offline building process of the projection model and the control limits of the SPC

### 3.1.3. Statistical process control

After acquiring the projected trajectories from historical normal operations, the SPC is adopted to construct

the control limit, as shown in Fig. 2. Then the detection rules of the fault detection subsystem can also be built. SPC [14] is an optimization philosophy applied to continuous process improvement, using a collection of statistical tools for data and process analysis. It seeks to maximize profit by improving product quality and productivity, streamlining process, and reducing scrap rate, among other goals.

In the proposed subsystem, it is assumed that the relationship between normal trajectories in the same task is the normal distribution if there are enough normal trajectories. Next, the control charts are chosen to monitor the system variables. According to the system characteristics, the Xbar-R chart is chosen for continuous sensor data; it is used to assess the stability of a task and includes the process mean (Xbar) and process range ( $R$ ). The corresponding control limits of the  $R$  chart are calculated with the inputs of the normal process spread as:

$$UCL = \bar{R}D_4, \quad (4)$$

$$LCL = \bar{R}D_3 \text{ and} \quad (5)$$

$$CL = \bar{R}, \quad (6)$$

where:  $UCL$ ,  $LCL$ , and  $CL$  denote the upper control limit, lower control limit and central line, respectively.  $\bar{R}$  denotes the average of ranges of sample groups.  $D_3$  and  $D_4$  are sample size specific factors for control limits [14]. The control limits of the Xbar chart are then calculated with the inputs of the normal process mean as:

$$UCL = \bar{\bar{x}} + A_2\bar{R}, \quad (7)$$

$$LCL = \bar{\bar{x}} - A_2\bar{R} \text{ and} \quad (8)$$

$$CL = \bar{\bar{x}}, \quad (9)$$

where:  $\bar{\bar{x}}$  denotes the average of averages from the sample groups.  $A_2$  is also a sample size specific factor for control limits [14]. In the proposed method, the sample groups are formed by several consecutive projection values, and their size is between 2 and 10. For example, in Fig. 3, there are two normal projection trajectories with a sample group size of five, and the sample window will shift with time. The control limits at time  $T$  are then built with all of the normal sample groups at time  $T$ . Therefore, in different time steps, there are different control limits in these charts.

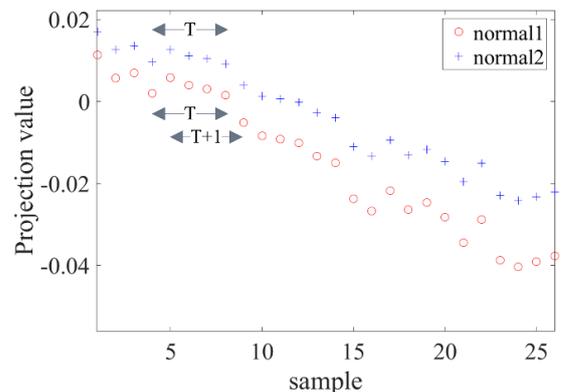


Fig. 3 An illustration of sample groups with a size of five

The range between  $UCL$  and  $LCL$  is six standard deviations. The general fault detection is only based on these two boundaries or additional boundaries of positive and negative two standard deviations [10], so in order to detect more subtle fluctuations and shifts, the Nelson rules [12] are used. Sample graphs of the eight rules are shown in Fig. 4. The central line is  $CL$ , and the sigmas represent the standard deviation of the data. The running projection values are also formed for the same size sample groups to online monitor status using these rules, with individual

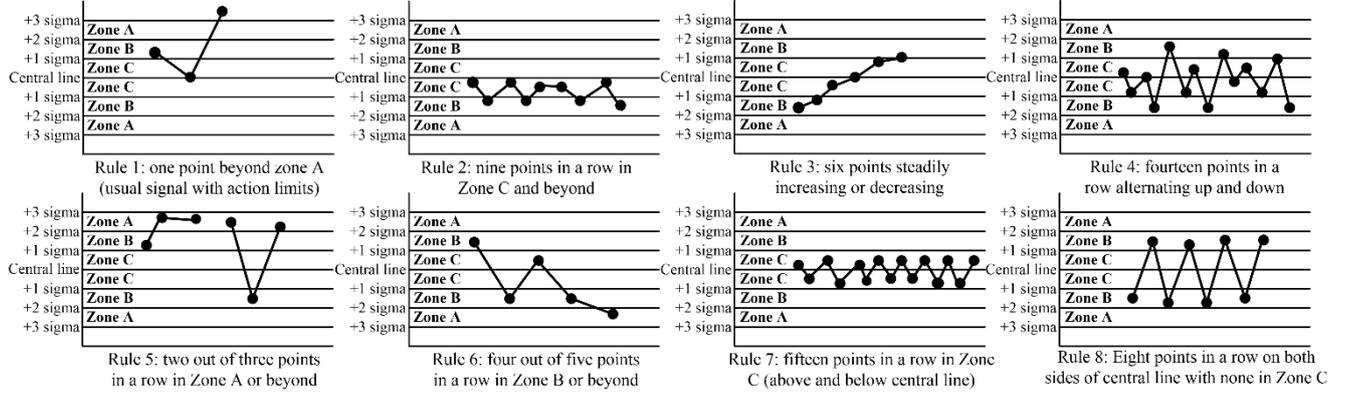


Fig. 4 The violation rules of control chart

### 3.2. Fault diagnosis

In this subsystem, the identifiable faults of a chosen robot should be decided, and the robot will perform a set of diagnostic operations with normal and different fault conditions to collect corresponding features first. These features can be used to train and validate the fault diagnosis model by using machine learning technology offline, after which the model can be used to diagnose faults online, as shown in Fig. 1.

#### 3.2.1. Fault types and diagnostic operation of an industrial robot

The proposed method focuses on possible performance degradation problems of the industrial robot, including increasing joint friction (lack of lubrication) and timing belt loosening (belt aging or an abnormal tension adjustment mechanism). The baseline information is very important for diagnosis, so the initial healthy features of the robot should be fully and effectively collected from the original diagnostic operations. The diagnostic operation is designed with the following characteristics. First, the robot will rotate only one joint at a time, keeping other joints locked. This means that the diagnostic information for each axis can be generated as independently as possible. Second, the diagnostic operation is conducted at a constant speed of rotation back and forth several times. Then the robot operating frequency can be fixed in order to be easily analyzed.

#### 3.2.2. Fault diagnosis features

In the proposed method, an industrial robot's existing sensors, including motor encoders and motor torque sensors, are the preferred choice. Further, because the vibration data is also important for analyzing robot motion, an additional accelerometer is needed.

There are two kinds of features in the proposed method: the time domain and the frequency domain. In the

thresholds in these charts.

In this study, by integrating the aforementioned approaches, the PCA-based SPC using Nelson rules is realized to rapidly find the process variation of an industrial robot equipped with many sensors using only a few control charts. This method can handle effectively the multivariate monitoring problem and it will significantly reduce the burden on monitoring. As soon as an abnormal variation behavior is detected, the fault diagnosis subsystem will be started for further fault classification.

time domain, the mean and variance are used as the features shown below:

$$\mu_s = \frac{1}{N_T} \sum_{j=1}^{N_T} x_j \quad \text{and} \quad (10)$$

$$\rho_s^2 = \frac{1}{N_T} \sum_{j=1}^{N_T} (x_j - \mu_s)^2, \quad (11)$$

where:  $\mu_s$  and  $\rho_s^2$  denote the mean and the variance of the data series, respectively;  $x_j$  represents the data series of one diagnostic operation and  $N_T$  is the number of data points. Therefore, the two features of the joint angle error, as Eq. (1), and the joint torque of the diagnostic operation can be extracted. In the frequency domain, the fast Fourier transform (FFT) is used to analyze the spectrum, and the size of specific frequencies are chosen as features through observation of the spectrum of normal diagnostic operation data. These features will be used for an effective pattern recognition method to train the fault diagnosis model.

#### 3.2.3. Multi-class support vector machines

An SVM is a machine learning technique that has been in use since the 1990s. SVMs are becoming increasingly important in machine learning, especially with data of large dimensions. The advantage of an SVM is its fast convergence in learning and fewer parameters to be tuned. It can solve a number of small-sample problems in practice, and also present a theoretical framework for machine learning.

In the proposed method, because a multi-class classifier is needed, a one-against-one multi-class SVM [13] is adopted. If there are  $p$  classes of data, this kind of SVM constructs  $p(p-1)/2$  two-class SVMs, but each one is trained

with data from only two different classes. Given  $v$  training data  $(z_u, l_u)$ , where  $z$  and  $l$  mean the training input vector and corresponding label and  $u = 1, \dots, v$ , from the  $q$ th and the  $r$ th classes, the following binary classification problem can be solved:

$$\begin{aligned} \min_{\omega^{qr}, b^{qr}, \xi^{qr}} & \frac{1}{2} (\omega^{qr})^T \omega^{qr} + C \sum_{u=1}^v \xi_u^{qr} \\ (\omega^{qr})^T \phi(z_u) + b^{qr} & \geq 1 - \xi_u^{qr}, \text{ if } l_u = q, \\ (\omega^{qr})^T \phi(z_u) + b^{qr} & \leq -1 + \xi_u^{qr}, \text{ if } l_u = r, \\ \xi_u^{qr} & \geq 0, \end{aligned} \quad (12)$$

where:  $\omega$  and  $b$  denote the parameter vector and bias of the support hyperplane,  $\xi$  denotes the slack variable,  $\phi$  is the function for mapping  $z$  to a higher dimension space and  $C$  is the penalty parameter. After all of the two-class SVMs are constructed, each training input  $z$  can be classified by:

$$\text{sign}((\omega^{qr})^T \phi(z) + b^{qr}), \quad (13)$$

where  $\text{sign}$  is the sign function. If the predict result equals positive one, the vote for the  $q$ th class is added by one. Otherwise, the  $r$ th class is increased by one. Finally, we can predict  $z$  is in the class with the highest vote, and the strategy is also called the ‘‘max wins’’ voting strategy.

### 3.3. Health evaluation

Construction of a quantitative index is an intuitive and effective way to understand the health of the industrial robot. In general, the health of the robot will affect its performance and the performance is therefore best at the beginning of use. So the index which is between 0 and 1 can be designed based on the overlapping area of two normal distributions with the same standard deviation [15], and these two distributions can be used to show the current performance and the original performance of the robot, as shown in Fig. 5. In the proposed health evaluation subsystem, the evaluation operation is the same as the diagnostic operation of the fault diagnosis subsystem. The evaluation operation will be executed at the beginning of use to collect at least 30 sets of original operation data, which include the joint angle error and torque of each joint. These sets of data are used to calculate the mean and standard deviation at each sampling time to represent the original performance. When the evaluation process is triggered online, the subsystem will collect several sets of evaluation operation data and average them to minimize fluctuations of the position controller to represent the current performance. Assuming that the standard deviation of current evaluation operation data at every sampling time are the same as those of the original evaluation operation data, the overlapping area of the two normal distributions of current and original evaluation operation data, at one sampling time and with only mean deviation, can be calculated. Then the joint angle error index ( $JA EI$ ) and the joint torque index ( $JTI$ ) for each joint, along with the robot angle error index ( $RA EI$ ) and robot torque index ( $RTI$ ) for the whole robot, are calculated as follows:

$$JA EI_i = \frac{\sum_{j=1}^{N_T} 2 \int_{-\infty}^{\frac{mao_{i,j} + mac_{i,j}}{2}} \frac{1}{\sqrt{2\pi}\sigma_{a,i,j}} e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma_{a,i,j}}\right)^2} dt}{N_T}, \quad (14)$$

$$\text{where} \begin{cases} \mu = mac_{i,j}, \text{ if } ma o_{i,j} < mac_{i,j} \\ \mu = ma o_{i,j}, \text{ if } ma o_{i,j} \geq mac_{i,j} \end{cases}$$

$$JTI_i = \frac{\sum_{j=1}^{N_T} 2 \int_{-\infty}^{\frac{mto_{i,j} + mtc_{i,j}}{2}} \frac{1}{\sqrt{2\pi}\sigma_{t,i,j}} e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma_{t,i,j}}\right)^2} dt}{N_T}, \quad (15)$$

$$\text{where} \begin{cases} \mu = mtc_{i,j}, \text{ if } mt o_{i,j} < mtc_{i,j} \\ \mu = mt o_{i,j}, \text{ if } mt o_{i,j} \geq mtc_{i,j} \end{cases}$$

$$RA EI = \min\{ JA EI_i \mid i = 1 \sim N_j \} \text{ and} \quad (16)$$

$$RTI = \min\{ JTI_i \mid i = 1 \sim N_j \}, \quad (17)$$

where the subscript  $i$  denotes the joint index, and the subscript  $j$  denotes the sample number index of the task.  $N_T$  means the total sample number in the task, and  $N_j$  is the number of joints. The means of the original and current evaluation operations are  $mao$  and  $mac$ , respectively, for the joint angle error;  $mto$  and  $mtc$  are the means of the original and current evaluation operations, respectively, for joint torque;  $\sigma_a$  and  $\sigma_t$  are standard deviations for joint angle error and joint torque, respectively. These six parameters can be calculated according to different values for  $i$  and  $j$ . The proposed four indexes are real numbers between 0 and 1, and the larger the value, the healthier. This study considers that a performance deviation in any joint will severely affect the performance of the whole robot, and the  $RA EI$  and  $RTI$  are minimum values of all  $JA EI$  and  $JTI$ , respectively.

The  $RHI$  generator is a Mamdani fuzzy inference system. There are two input variables,  $RA EI$  and  $RTI$ , and one output variable,  $RHI$ . Fig. 6 shows the proposed architecture for the  $RHI$  generator. Fig. 7 shows the designed membership functions for  $RA EI$ ,  $RTI$  and  $RHI$ . The  $RA EI$  and  $RTI$  are divided into three levels, high, medium and low, and the  $RHI$ , which is also between 0 and 1, is divided into four levels: good, slight aging, medium aging and severe aging. The joint torque values are more significantly affected by faults than the joint angle error values because the robot position controller can increase the torque output to reduce a joint angle error. Therefore, the acceptable deviation of the  $RTI$  is larger than that of the  $RA EI$ , as shown in Fig. 7. The system consists of nine AND rules with min operator for evaluating appropriate robot health, for instance:

$$\text{If } RA EI \text{ is High and } RTI \text{ is High, then } RHI \text{ is Good.} \quad (18)$$

The whole rule base is shown in Table 1 and the fuzzy implication for these rules is the min operator. Then the rule outputs are combined by max aggregation method to the fuzzy set of the output variable,  $RHI$ . Finally, to produce a nonfuzzy action that best represents the possibility distribution of an inferred fuzzy action, the center of area

(COA) method is used for the defuzzification strategy. After defuzzification, the value of *RHI* ranges from 0 to 1; and the higher is better.

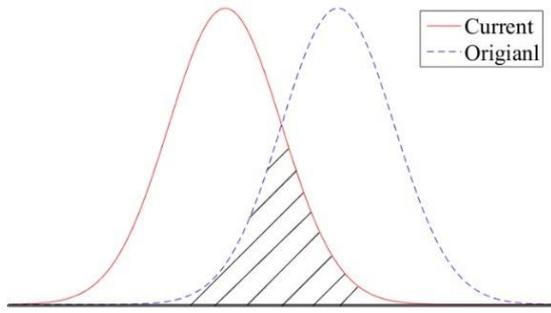


Fig. 5 Overlap of statistical distributions of current and original operations for defining the index

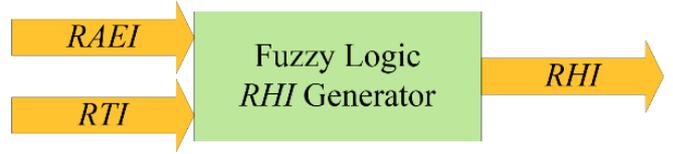


Fig. 6 The proposed architecture for the RHI generator

Table 1

Fuzzy rule base for the *RHI* generator

<i>RTI</i> \ <i>RAEI</i>	LOW	MEDIUM	HIGH
LOW	Severe aging	Severe aging	Medium aging
MEDIUM	Severe aging	Medium aging	Slight aging
HIGH	Medium aging	Slight aging	Good

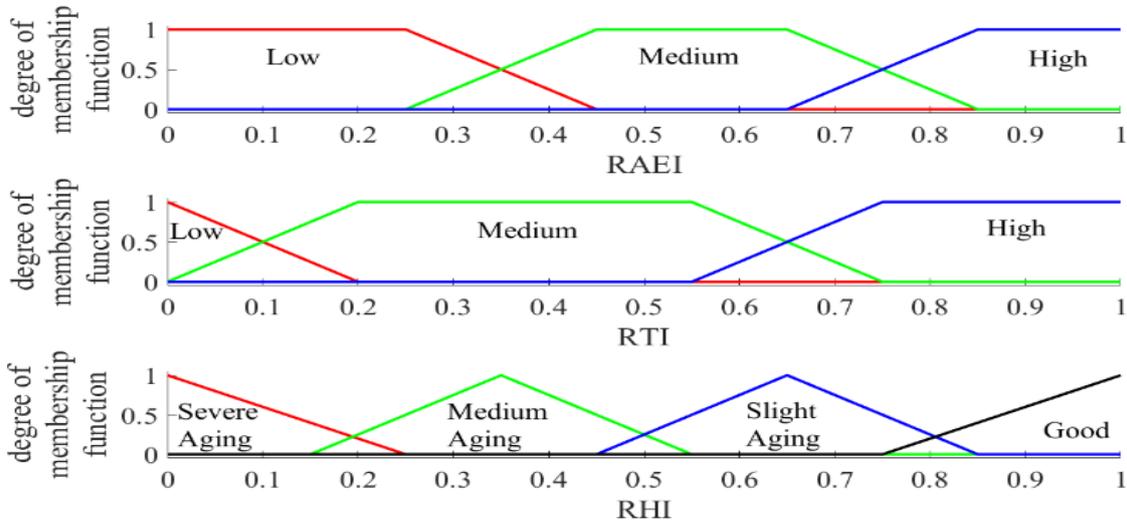


Fig. 7 Membership functions for the fuzzy logic system

#### 4. Experiments

This study covers three experiments corresponding to three subsystems. The chosen industrial robot is a six-axis robot, DRV90L7A6213N by Delta Electronics.

##### 4.1. Experimental settings

Due to the design of the robot’s mechanisms, only the fifth and sixth joints can be used for fault-related experiments. For this experiment, the fifth joint is chosen as the experimental area. Fig. 8 shows how the simulation of increasing joint friction is implemented by applying normal force on the wheel in the solid red box, and the simulation of timing belt looseness is implemented by tuning the length of the screw in the dotted red box. This screw is loosened about 2 mm from its original position to simulate a loose belt. The wheel pressing device is shown in Fig. 9. This device has a screw that moves the white metal block to press the wheel, and there is a flexible pressure sensor to estimate the normal force and friction added to the wheel. The normal force on the pulley is about 60 N in a condition of increasing friction.

##### 4.2. Fault detection experiments

The fault detection subsystem is designed to identify variations in any type of cyclic task. The task designed

to simulate object grasping is shown in Fig. 10 with the video included in the attachment as an example. The robot performs cyclic movements in three points, and the sensor data will be monitored throughout that time. There are 2,411 points, with a 6 ms sampling time in one task cycle. For this experiment, 30 sets of normal operation data for 19 variables are collected to build the projection model, with only one principal component at first. Then the control limits of the Xbar-R chart can be built with a sample group size of five, as shown in Figs. 11 and 12.

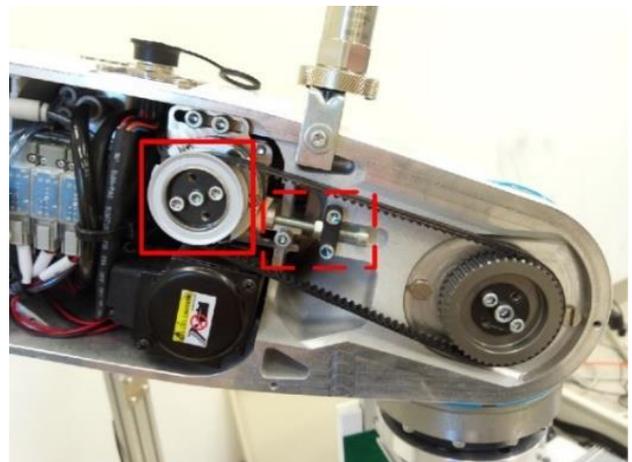


Fig. 8 The experimental area of the fifth joint

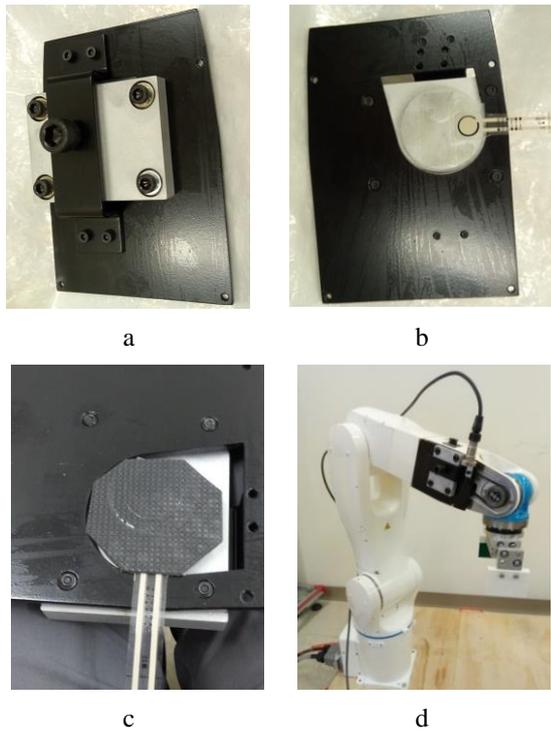


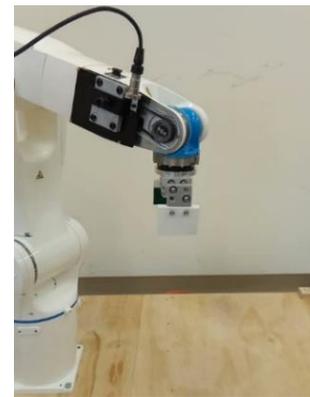
Fig. 9 The wheel pressing device: a – the device body, b – the flexible pressure sensor on the pressing cylinder plane, c – the pressing cylinder plane with a rubber plate, d – the device installation position

There are two tests: one is to check the fault detection ability of the  $\bar{X}$  and  $R$  charts, and the other is to use the Nelson rules for more sensitive fault detection. The first test uses four additional sets of the normal operation data, four sets of the fifth joint's increasing friction operation data and four sets of the fifth joint's loose belt operation data to check how many points are out of the boundaries of positive and negative triple standard deviations, as shown in Figs. 11 and 12. The results are shown in Table 2, where the variations between normal and abnormal operations are obvious. This table shows that the  $\bar{X}$  chart is much more sensitive for fault detection than the  $R$  chart, so the  $\bar{X}$  chart is more suitable for use in fault detection. In the second test, these 12 operations are further checked using Nelson rules in an  $\bar{X}$  chart, and the results are shown in Table 3. Rule 2, used for detecting prolonged bias, shows a difference between normal and abnormal operations, but the number of normal operations is not zero, so this study considers this rule to be unsuitable. Rule 3 is used for detecting existing trends, and there is no obvious difference between normal and abnormal operations. In rule 4, used for detecting oscillation, the numbers in these operations are zero. Rule 7 detects faults with one standard deviation, and if there are any faults, the number will decrease. Therefore, rules 1, 5, 6, and 8 are the most suitable for checking the variations, while rule 1 is the same as the original SPC detection method. Next, the warning thresholds can be set for online implementation at 50, 100, 300 and 200 for rules 1, 5, 6 and 8, respectively. Once the violations for any rule crosses over the threshold, the current task is stopped, and the diagnosis process is launched. Moreover, to make online reaction faster, one original task can be separated into several small tasks, and these small tasks can also use the proposed fault detection approach.

Table 2

The accumulated number of points out of triple standard deviations in the  $\bar{X}$  and  $R$  charts

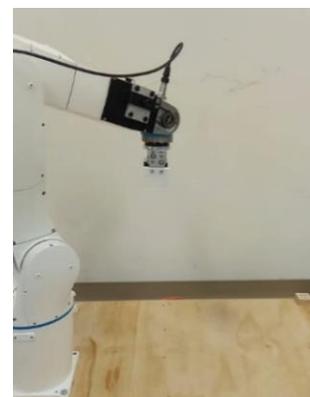
Normal					
ID	1	2	3	4	Mean
$\bar{X}$	0	0	0	0	0
$R$	0	0	0	2	0.5
Friction					
ID	1	2	3	4	Mean
$\bar{X}$	425	281	475	241	355.5
$R$	11	10	4	2	6.75
Belt looseness					
ID	1	2	3	4	Mean
$\bar{X}$	419	437	440	455	437.75
$R$	9	9	10	13	10.25



a



b



c

Fig. 10 The condition monitoring task including three points: a – the first point, b – the second point, c – the third point

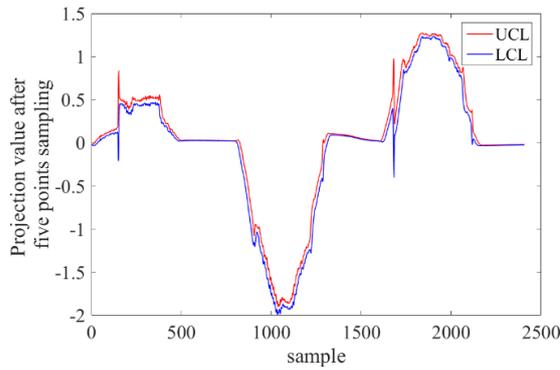


Fig. 11 The upper (red) and lower (blue) boundaries in the Xbar chart

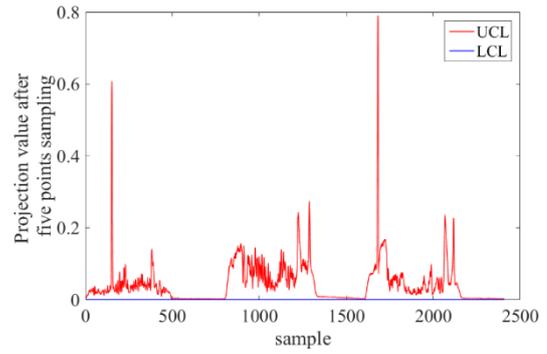


Fig. 12 The upper (red) and lower (blue) boundaries in the R chart. The lower boundary is zero

Table 3

The accumulated number of points that violate the Nelson rules in the Xbar chart

Experimental operations	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8
Normal1	0	1498	641	0	0	0	2393	0
Normal2	0	1449	666	0	0	43	2125	28
Normal3	0	1315	643	0	0	40	2168	8
Normal4	0	1604	612	0	0	189	1576	129
Mean of the normal operations	0	1467	641	0	0	68	2066	41
Friction1	425	1893	605	0	541	886	756	764
Friction2	281	1946	621	0	436	828	843	691
Friction3	475	2001	650	0	612	787	1079	707
Friction4	241	1904	614	0	340	796	765	664
Mean of the increasing friction operations	356	1936	623	0	482	824	861	707
Loose1	419	1891	611	0	625	943	929	839
Loose2	437	2015	627	0	494	768	1090	656
Loose3	440	1975	612	0	503	647	1221	580
Loose4	455	1854	603	0	530	785	891	668
Mean of the loose belt operations	438	1934	613	0	538	786	1033	686

#### 4.3. The fault diagnosis experiment

Table 6

In this experiment, the robot performs the constant-speed diagnostic operation several times in the fifth joint under the aforementioned fault conditions, as shown in Fig. 13 with the video included in the attachment. The motor speed is 5,500 rpm. There are six kinds of features of these diagnostic operations, including the mean and variance of joint angle error and joint torque, as well as two frequencies, 38 Hz and 68 Hz. The data set has three labels: normal, friction and looseness, and each label has 120 data points. The multi-class SVM has three classifiers: normal to friction, normal to looseness, and looseness to friction. Each classifier has 240 data points, and the data are divided into 180 training data points and 60 validating data points. The testing results are shown in Tables 4, 5 and 6. Clearly, the accuracy is 100%, so if one testing sample is input to this system, these classifiers can vote the correct results.

The testing results of looseness to friction

	Predicted looseness	Predicted friction
Actual looseness	26	0
Actual friction	0	34

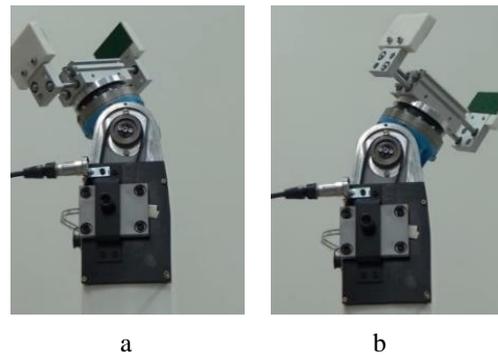


Fig. 13 The snapshots of constant-speed diagnostic motion in the fifth joint: a – positive motion boundary, b – negative motion boundary

Table 4

The testing results of normal to friction

	Predicted normal	Predicted friction
Actual normal	31	0
Actual friction	0	29

Table 5

The testing results of normal to looseness

	Predicted normal	Predicted looseness
Actual normal	35	0
Actual	0	25

#### 4.4. The health evaluation experiment

First, 30 sets of original evaluation operation data of each joint for the whole robot are collected for baseline data. Then 10 sets of current evaluation operation data are collected for each of three different fault conditions in the fifth joint, including normal, increasing friction, and belt looseness, which is the same as previously mentioned. The other five joints are within normal conditions. In this way,

three kinds of current evaluation operation data are acquired for calculating *JA EI*, *JTI*, *RA EI*, *RTI* and *RHI*. The results are shown in Table 7 and 8. In Table 7, it can be seen that the *JA EI* and *JTI* under normal conditions are around 0.92 and 0.93, respectively. This means that there is always some fluctuation of the position controller even if the robot is brand new. By comparing the results for the fifth joint, the differences between the normal and abnormal conditions become very clear, and the *JA EI* and *JTI* decrease with fault conditions. Table 8 shows that there are obvious differences among three fault conditions in *RA EI*, *RTI* and *RHI*. The *RHI* is around 0.92, 0.19 and 0.08 for normal, belt looseness and increasing friction conditions, respectively, so the aging state of the health is clear. This shows that the health evaluation subsystem can discriminate the health status of the industrial robot. The *RTI* changes more dramatically than the *RA EI* because of the regulation of the position controller. The relationship between *RA EI*, *RTI* and *RHI* is shown in

Fig. 14. Because of the COA defuzzification method, the maximum and minimum values of *RHI* are 0.92 and 0.08.

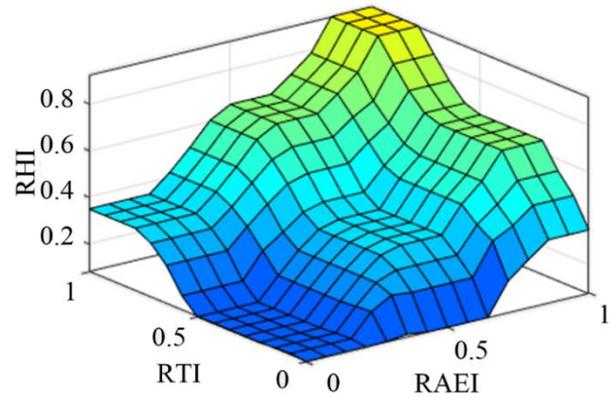


Fig. 14 *RHI* surface plot with *RA EI* and *RTI* inputs

Table 7

The *JA EI* and *JTI* of each joint under different fault conditions

The different fault conditions	The <i>JA EI</i> and <i>JTI</i>	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
Normal	<i>JA EI</i>	0.9082	0.9128	0.9123	0.9156	0.9251	0.9284
	<i>JTI</i>	0.9187	0.9254	0.9237	0.9305	0.9342	0.9367
Looseness	<i>JA EI</i>	0.9013	0.9105	0.9112	0.9143	0.5197	0.9241
	<i>JTI</i>	0.9134	0.9156	0.9294	0.9246	0.0489	0.9306
Friction	<i>JA EI</i>	0.9086	0.9076	0.9062	0.9097	0.1799	0.9214
	<i>JTI</i>	0.9060	0.9098	0.9152	0.9141	1.70E-57	0.9201

Table 8

The *RA EI*, *RTI* and *RHI* under different fault conditions

The different fault conditions	<i>RA EI</i>	<i>RTI</i>	<i>RHI</i>
Normal	0.9082	0.9187	0.92
Looseness	0.5197	0.0489	0.194
Friction	0.1799	1.70E-57	0.08

## 5. Conclusions

This study describes a proposed intelligent fault detection, diagnosis and health evaluation system using a PCA-based SPC with Nelson rules to monitor the multivariate operation condition of industrial robots. After analysis, four suitable Nelson rules are chosen for fault detection. It is shown that this approach can effectively detect subtle variations in a robot's performance. A multi-class SVM is applied to fault diagnosis in this system with several features, and the results of classification are sufficient to recognize faults. Furthermore, if a user wants to know the health status of a robot, the *RHI* can be calculated using the health evaluation process in this system. Finally, the proposed system is successfully validated by two kinds of real fault conditions implemented with an actual six-axis industrial robot.

## References

- Lee, J.; Ni, J.; Djurdjanovic, D.; Qiu, H.; Liao, H. 2006. Intelligent prognostics tools and e-maintenance, *Computers in Industry* 57(6): 476–489. <https://doi.org/10.1016/j.compind.2006.02.014>.
- Yam, R. C. M.; Tse, P. W.; Li, L.; Tu, P. 2001. Intelligent predictive decision support system for condition-based maintenance, *International Journal of Advanced Manufacturing Technology* 17(5): 383–391. <https://doi.org/10.1007/s001700170173>.
- Khalastchi, E.; Kalech, M. 2018. On fault detection and diagnosis in robotic systems, *ACM Computing Surveys* 51(1): 1–24. <https://doi.org/10.1145/3146389>.
- Vemuri, A. T.; Polycarpou, M. M.; Diakouritis, S. A. 1998. Neural network based fault detection in robotic manipulators, *IEEE Transactions on Robotics and Automation* 14(2): 342–348. <https://doi.org/10.1109/70.681254>.
- Cho, C. N.; Hong, J. T.; Kim, H. J. 2019. Neural network based adaptive actuator fault detection algorithm for robot manipulators, *Journal of Intelligent & Robotic Systems* 95(1): 137–147. <https://doi.org/10.1007/s10846-018-0781-0>.
- Yüksel, T.; Sezgin, A. 2010. Two fault detection and isolation schemes for robot manipulators using soft computing techniques, *Applied Soft Computing* 10(1): 125–134. <https://doi.org/10.1016/j.asoc.2009.06.011>.
- Hodge, V. J.; Austin, J. 2004. A survey of outlier detection methodologies, *Artificial Intelligence Review* 22(2): 85–126. <https://doi.org/10.1007/s10462-004-4304-y>.
- Louahem M'Sabah, H.; Bouzaouit, A. 2016. Degradation model of the bearings by Wiener process, *Mechanika* 22(3): 225–228. <https://doi.org/10.5755/j01.mech.22.3.13000>.
- Muradore, R.; Fiorini, P. 2012. A PLS-based statistical approach for fault detection and isolation of robotic manipulators, *IEEE Transactions on Industrial Electronics* 59(8): 3167–3175. <https://doi.org/10.1109/TIE.2011.2167110>.
- Jaber, A. A.; Bicker, R. 2017. Wireless fault detection system for an industrial robot based on statistical control chart, *International Journal of Electrical and Computer Engineering* 7(6): 3421–3435.

- <https://doi.org/10.11591/ijece.v7i6.pp3421-3435>.
11. **You, D.; Gao, X.; Katayama, S.** 2015. WPD-PCA-based laser welding process monitoring and defects diagnosis by using FNN and SVM, *IEEE Transactions on Industrial Electronics* 62(1): 628–636. <https://doi.org/10.1109/TIE.2014.2319216>.
  12. **Nelson, L. S.** 1984. The Shewhart control chart-tests for special causes, *Journal of Quality Technology* 16(4): 237–239. <https://doi.org/10.1080/00224065.1984.11978921>.
  13. **Hsu, C. W.; Lin, C. J.** 2002. A comparison of methods for multiclass support vector machines, *IEEE Transactions on Neural Networks* 13(2): 415–425. <https://doi.org/10.1109/72.991427>.
  14. **Montgomery, D. C.** 2013. *Introduction to Statistical Quality Control*, 7th Edition, John Wiley & Sons, Inc.
  15. **Cheng, F. T.; Chen, Y. T.; Su, Y. C.; Zeng, D. L.** 2008. Evaluating reliance level of a virtual metrology system, *IEEE Transactions on Semiconductor Manufacturing* 21(1): 92–103. <https://doi.org/10.1109/TSM.2007.914373>.

H. K. Hsu, H. Y. Ting, M. B. Huang, H. P. Huang

## INTELLIGENT FAULT DETECTION, DIAGNOSIS AND HEALTH EVALUATION FOR INDUSTRIAL ROBOTS

### S u m m a r y

The focus of this study is development of an intelligent fault detection, diagnosis and health evaluation system for real industrial robots. The system uses principal component analysis based statistical process control with Nelson rules for online fault detection. Several suitable Nelson rules are chosen for sensitive detection. When a variation is detected, the system performs a diagnostic operation to acquire features of the time domain and the frequency domain from the motor encoder, motor current sensor and external accelerometer for fault diagnosis with a multi-class support vector machine. Additionally, a fuzzy logic based robot health index generator is proposed for evaluating the health of the robot, and the generator is an original design to reflect the health status of the robot. Finally, several real aging-related faults are implemented on a six-axis industrial robot, DRV90L7A6213N by Delta Electronics, and the proposed system is validated effectively by the experimental results.

**Keywords:** fault detection and diagnosis, industrial robot, Nelson rules, robot health index, statistical process control.

Received October 15, 2019

Accepted February 17, 2021



This article is an Open Access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 (CC BY 4.0) License (<http://creativecommons.org/licenses/by/4.0/>).