Abstract—Condition monitoring and fault diagnosis in modern manufacturing automation is of great practical significance. It improves quality and productivity, and prevents damage to machinery. In general, this practice consists of two parts: 1) extracting appropriate features from sensor signals and 2) recognizing possible faulty patterns from the features. Through introducing the concept of marginal energy in signal processing, a new feature representation is developed in this paper. In order to cope with the complex manufacturing operations, three approaches are proposed to develop a feasible system for online applications. This paper develops intelligent learning algorithms using hidden Markov models and the newly developed support vector techniques to model manufacturing operations. The algorithms have been coded in modular architecture and hierarchical architecture for the recognition of multiple faulty conditions. We define a novel similarity measure criterion for the comparison of signal patterns which will be incorporated into a novel condition monitoring system. The sensor-based intelligent system has been implemented in stamping operations as an example. We demonstrate that the proposed method is substantially more effective than the previous approaches. Its unique features benefit various real-world manufacturing automation engineering, and it has great potential for shop floor applications.

Note to Practitioners—This paper was motivated by the problem of detecting the complicated sheet-metal processes faults in a high success rate but also applies to the real-world practices. Existing systems adopt traditional approaches fast but cannot adapt to various cases and achieve a diagnosis purpose. Using hidden Markov models and the newly developed support vector techniques, this paper proposes three approaches and develops intelligent learning algorithms to model manufacturing operations. The novel similarity measure criterion is able to implement the signal patterns comparison, overcoming the time-shifting phenomenon. The approach of modular architecture and hierarchical architecture has been presented in the software coding for the recognition of multiple faulty conditions. The sensor-based hardware design facilitates the online condition monitoring system. In this paper, we mathematically characterize the process signal features. We then develop the HMM-based modular fault diagnosis system which makes the modification expansion easy in practice. The hierarchical SVM-based fault diagnosis system uses a small-size training sample but achieves a high successful rate, and overcomes the overfitting and difficulties in structure design of the learning machine. The similarity measure approach using support vector techniques requires training sample sets produced under normal operating conditions and, thus, makes it more feasible than other techniques. Meanwhile, this approach overcomes the time-shifting. The proposed system provides important technologies and valuable reference for research areas and an innovation to manufacturing automation. The system works well in most applications; however, in some cases, certain parameters have to be tuned manually, for example, the parts of the signals to be modeled. The next step is to develop the system so it can learn human skills and set parameters automatically. In the short term, it is important to research ways to improve the system’s intelligent online learning.

Index Terms—Automation, feature extraction, intelligent manufacturing, pattern recognition, similarity measure.

I. INTRODUCTION

Competition markets set high demands on product quality and productivity. However, most manufacturing practices encompass complex processes involving many process variables. The complex interaction among variables determines the performance of machines and product quality. When process variables deviate beyond specified limits, faults may occur. The sheet-metal stamping operation is a typical example, which provides a fundamental tool for modern manufacturing. Millions of parts—ranging from battery caps to clock frames and automobile body panels—are produced via this process daily. Hence, even small improvements in performance could produce significant benefits. However, the stamping process is also rather complicated [1]. It involves more than 40 process variables. Although there have been improvements in performance, eliminating faults (of which there are many) is impossible due to the difficulty of predicting and detecting them. In this context, condition monitoring and fault diagnosis are important aspects of automation manufacturing because they detect abnormal conditions, ensure the quality of products, protect tools and machines against potential damage, and help to increase the availability of machines in the production processes.

The key of condition monitoring and diagnosis, which includes information acquisition, feature extraction, and decision making, is pattern recognition. To make use of the important process information acquired from sensors, feature extraction (i.e., finding the features that would effectively describe the characteristics of the faulty conditions) is crucial. In order to find appropriate features, one needs to understand the monitored process and the monitoring signals. For example, in order to monitor the sheet-metal stamping process, one must understand the stamping operations as well as the sensor signals [2]. With extracted features, statistical approaches (i.e., threshold setting)
are very common in the applications. However, although they are simple, they are also weak. This is because of the many factors that cause sensor signal noise and time-shifting, which can worsen performance.

There have been great efforts to improve detection success rates. Chief among them has been an appropriate sensor signal application. Without doubt, the major factor in the process is the stamping force, which has been studied for related purposes [2], [3]. However, force sensors are expensive and difficult to install. Strain signals, as a substitution, describe the force indirectly with good signal-to-noise ratio (SNR). For example, in [4] and [5], Haar transform of the tonnage signal is used to monitor a large press in automobile body-panel production. In [6], the principal component analysis technique is used to extract features of the tonnage signals.

Meanwhile, advanced approaches (i.e., expert system, fuzzy logic, artificial neural network (ANN), etc.) have been widely applied for condition monitoring and fault diagnosis [7]. In stamping process exploration, for example, [8] applied ANN and finite-element analysis to study strain signals. They improved the performance and made the system intelligent. However, the philosophy underpinning their systems determined that they were not all-purpose. Following the empirical risk minimization (ERM) principle, these systems exhibit a number of drawbacks, such as difficulties related to structure design, overfitting, and so on. Moreover, a large number of samples are required for training [9]. The performance of these systems, therefore, cannot satisfy real-world application requirements and ever-increasing demands. Thus, it is imperative to develop a novel intelligent condition monitoring and diagnosis system.

Three approaches were designed for the objective of the paper: 1) effectively measure and select the features from the appropriate sensors; 2) define an effective and efficient learning algorithm to acquire knowledge from the limited given samples; and 3) generate an accurate and simple criterion of decision making for online applications.

Wavelet packet decomposition outperforms time–frequency analysis in processing nonstationary or transient signals [10]. Unfortunately, a lack of transition-invariant propriety means the coefficients cannot be used directly as features. However, its characteristics may help to define a new feature representation. Using these features as observations, hidden Markov model (HMM), which demonstrates its successful application in voice recognition [11], [12], can characterize the different stamping processes. Moreover, a support vector machine (SVM) has recently gained popularity [13], [14]. Based on the statistical learning theory, it can precisely identify the factors that need to be taken into account to learn successfully. Unlike other learning machines, SVM follows a structural risk minimization (SRM) principle [15]. This means the theory is not only a tool for theoretical analysis but also a tool for creating practical algorithms for estimating multidimensional functions. The learning is based on some simple ideas and provides an obvious intuition of what learning from the examples is about. Additionally, it can lead to high performance in practical applications. Thus, support vector techniques have been applied to many real-world problems, such as pattern identification [16], regression [17], and even nonlinear equalization [18], etc.

Aiming to implement condition monitoring and fault diagnosis, binary classification and function estimation is an alternative to most applications. Through approximating the sensor signal, it is possible to construct a mathematical model for describing the pattern under normal working conditions. Comparing the similarity between two signals based on the model, the degree of deviation shows the difference between the two patterns and, thus, helps to detect fault occurrence.

Several strategies were developed, including a proposal for a novel condition monitoring and diagnosis system by using the support vector techniques and similarity measure. The rest of this paper is structured in the following manner. Section II presents the system design and experiment design. Section III defines marginal energy, while Section IV introduces the HMM-based modular fault diagnosis strategy and its performance. Section V demonstrates the SVM-based hierarchical multiple fault diagnosis architecture, and Section VI defines a new similarity measure and presents the proposed system. Section VII includes results and discussion. Finally, Section VIII contains the conclusion and future work.

II. SYSTEM AND EXPERIMENT DESIGN

A. System Hardware Design

The stamping process is one of the most commonly used processes in modern manufacturing. Aiming to meet the ever-increasing demands for product quality and productivity, a number of monitoring systems have been developed [4]–[6]. Nevertheless, the stamping process is a complicated process involving many factors, such as elastic and plastic deformation of the sheet metals, design, manufacturing and lubrication of dies, static and dynamic behavior of the press, and so on. It is a highly nonlinear transient process, so simple threshold checking and fast Fourier transform (FFT) analysis do not work well. In addition, as new designs improve the productivity of the stamping machines more and more, they also constrain the complex diagnosis methodologies applicable in common devices. As a result, existing systems have achieved only limited success.

As a consequence of rapid development, computer-based systems are now powerful enough to deal with quick responses and heavy computational loads, which make it possible for us to design an intelligent computer-based condition monitoring and fault diagnosis system for metal stamping operations. In this paper, a new computer-based device was developed to comply with the requirements of online monitoring diagnosis for stamping processes.

The device consists of three function blocks. The first one is the data-acquisition block, including one or two strain sensors (Kistler 9232A) mounted on the middle of the press frame and an Axlom data-acquisition card (model: AX5411H) with an analog-to-digital converter function. The scan rate ranges from 1 to 50 kHz. The second block is an embedded central processing unit (CPU) board with a 750-MHz Pentium III CPU. Instead of EEPROM, a 20G electronic hard disk stores the process data to improve the reliability of the system. The third one is an interface block that allows an operator to interact with the system through an liquid-crystal display (LCD) touch screen, and enables the system to upload the data to a remote sever via Internet for further use. Fig. 1 provides the block diagram of the system.
It is known that stamping operations can be divided into three categories: blanking, bending, and drawing. A number of experiments covering these operations have been conducted in the lab and onsite. In the lab, the experiments were carried out on a simple C-frame mechanical press (manufacturer: SEYI; model: SN1-25) shown in Fig. 2, which is a typical press used in mass production of small domestic products and components. Its maximum stamping force is 25 tons and the maximum speed is 110 strokes per minute (SPM). The selection of sampling rate is important for the identification of proper features. The strain gauge signals contain both static and dynamic information. The selection of sampling rate of the system is important to identify the proper features. According to the related work presented in [19], the higher frequency terms of the signal come from the press and, hence, do not relate to the faulty conditions. Considering the stamping speed being 110 SPM and approximately 1/10 of the time it is used in the stamping, the actual stamping time is approximately 0.05 s. Therefore, the sampling rate was set at 5 kHz (sampling time 0.0002 s) to ensure sufficient information is being captured. A total of 5000 data points were sampled per stroke. In order to capture the signals every time at the same time instance, a trigger mechanism was developed. It is a proximity sensor installed near the main shaft. Everytime the key way of the shaft rotated passed the sensor, a pulse was generated and sent to the computer to trigger the start of sampling.

Three experiments are presented here covering a range of situations. The first and second experiments were conducted in a laboratory setting. In the third experiment, data from real-world applications were acquired using a JG21-14 press (manufactured by Jieyang Machine Tool Works) to verify the performance of the system.

### B. System Software Design

Aiming to make the source code easy to read and maintain, the system software adopts a modular architecture comprised of data acquisition, data storage, feature extraction, decision making, interface management, control, and so on. As to the operation system, Linux has many merits. It has stronger network capability and is more robust than the DOS operation system. It also possesses various resources and allows multithread. All functions and tasks are developed using Borland C++ language.

#### TABLE I

<table>
<thead>
<tr>
<th>Condition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>Normal condition</td>
</tr>
<tr>
<td>A</td>
<td>Misfeed: workpiece is not aligned with the dies</td>
</tr>
<tr>
<td>B</td>
<td>Slug: chips left on the surface of workpiece</td>
</tr>
<tr>
<td>C</td>
<td>Too thick: workpiece thickness is thicker than normal</td>
</tr>
<tr>
<td>D</td>
<td>Too thin: workpiece thickness is thinner than normal</td>
</tr>
<tr>
<td>E</td>
<td>Material missing: there is no material</td>
</tr>
<tr>
<td>F</td>
<td>Double stroke: part left in upper module</td>
</tr>
<tr>
<td>G</td>
<td>Crack: occurred in the drawing operation due to bad lubrication, materials or design</td>
</tr>
</tbody>
</table>

The proposed algorithms that benefit system performance are presented in the following sections.

#### C. First Experiment: Single-Step Blanking

This experiment incorporated a blanking operation with a simple mould. The manufactured part is a simple bucket used for mounting input/output (I/O) boards in desktop computers. The workpiece material is steel-plate coldrolled coil (SPCC), the thickness is 1.1 mm, and the blank dimension (the perimeter of the part) is approximately 32 cm. The stamping speed condition is 85 SPM.

The parts that were produced were manufactured under normal conditions. When working conditions changed, so-called faults developed and deviations occurred. Generally, there are eight different conditions in practice as defined in Table I. Six of them were tested in this experiment and the typical strain signals are shown in Fig. 3. The signal contains both static and dynamic information about the stamping process, and is suitable for its condition monitoring. Nevertheless, except that of the last condition, the other signals are all similar in appearance. Thus, the simple threshold crossing check does not work very well.

A total of 6 (condition) × 40 (samples) = 240 samples were collected, and each sample was represented by a 512-dimension vector covering the entire stamping operation.
Fig. 3. Strain signals using single-step blanking under different stamping conditions.

Fig. 4. Strain signals using a multistation module under different stamping conditions.

D. Second Experiment: Multistep Progressive (At the Lab)

This experiment used a multistation mould including the three stamping categories (blanking, bending, and drawing), and the item produced was the cover of the rear part of a small motor. The material from which it is made is SPICE, and the dimensions of the parts are approximately 35 mm × 6.6 mm (diameter × height). Its normal working condition is 80 spm with a 1.0-mm workpiece.

The workpiece for the multistation mould in the experiment is quite long, which moves forward (from left to right) and is driven by the feeder. The fault occurred on one side only and may have caused an unbalanced impact on the punch force. This meant that a single sensor might not detect a fault in the mould; since it occurred on one side, it may not have impacted both sides. Consequently, two strain sensors were mounted on the left and right sides of the press to detect faults. The strain signals from both sensors merging through a simple sensor fusion that is the data for diagnosis were the average of the signals, which provides us with boosted information. Four different conditions were tested, including normal, slug, misfeed, and double stroke. The absence of lubrication caused some scratching on the face of the parts. The typical strain signal under the normal condition is shown in Fig. 5.

Material specifications and the design of the mould caused the battery caps to break easily. Moreover, due to the age and condition of the press, along with adverse working conditions, the SNR of the strain signals in the third experiment was low. As shown in Fig. 5, only serious cracks were detected by traditional condition monitoring systems (i.e., peak value, RMS, kurtosis, and skewness). In all cases, operators are informed of the results too late to ensure reductions in the number of faulty parts manufactured. Reports from operators and QC managers in the factory suggest that at least 20 pieces prior to an operator observing a crack must be discarded. However, no obvious phenomenon can be detected to suggest that cracks will occur, and it is almost impossible to diagnose these faults.

The original strain signal contains 640 data points. One-hundred thirty sets of samples were collected, 90 samples were normal, while the remaining 40 samples were abnormal.

E. Third Experiment: Multistep Progressive (onsite)

The third experiment also used a multistation mould; and the parts produced were battery caps. The material was copper and the approximate dimension of the part was 12 mm × 4.3 mm (diameter × height), designed to work under 150 spm with a 0.7-mm workpiece. It is noted that the normal stamping force of the press is 140 KN. This experiment was conducted in a factory in which there were a number of other presses working in the same area, which may cause a disruption to the process. The press on which the experiment was carried out was timeworn.

Unlike the second experiment, the materials were fed from front to back thus necessitating the use of only one sensor mounted on the left side. Two different conditions were tested, including normal and cracked. The absence of lubrication caused some scratching on the face of the parts. The typical strain signal under the normal condition is shown in Fig. 5.

Material specifications and the design of the mould caused the battery caps to break easily. Moreover, due to the age and condition of the press, along with adverse working conditions, the SNR of the strain signals in the third experiment was low. As shown in Fig. 5, only serious cracks were detected by traditional condition monitoring systems (i.e., peak value, RMS, kurtosis, and skewness). In all cases, operators are informed of the results too late to ensure reductions in the number of faulty parts manufactured. Reports from operators and QC managers in the factory suggest that at least 20 pieces prior to an operator observing a crack must be discarded. However, no obvious phenomenon can be detected to suggest that cracks will occur, and it is almost impossible to diagnose these faults.

The original strain signal contains 640 data points. One-hundred thirty sets of samples were collected, 90 samples were normal, while the remaining 40 samples were abnormal.

F. Experimental Data Generation

All of the data for training or testing purposes in this paper were generated from the actual stamping processes. The data in the third experiment were collected from the real-time applications. The data in the first and second experiments were acquired from different conditions which were created through simulating actual operations progresses. For example, we produce samples with different thickness for “too thick” and “too thin” cases, and we put chip(s) on the surface of workpieces from double stroke, and the other 80 were from slug conditions (the slugs were on Steps 2, 4, 5, and 6).
for the slug fault. In this manner, the experimental data can be treated as the ones in real-time applications.

III. FEATURE REPRESENTATIONS

Developed two decades ago, wavelet transform is an effective method for signal processing and feature extraction. A number of different wavelet transforms have been developed, such as binary wavelet transform and wavelet packet transform [10]. The latter is perhaps the most commonly used wavelet transform producing the wavelet packet coefficients

\[ w_{j,k,n} = \langle f, W_{j,k,n} \rangle \]  

where \( f \) is the signal and \( W \) is the wavelet packet function. The scale parameter \( j \) controls the contraction and dilation not only in the time domain but also in the frequency domain, while the translation parameter \( k \) defines the time location, and the modulation parameter \( n \) denotes the frequency band. It contains the so-called multiresolution characteristics: at different resolution \( j \), the signal is decomposed into \( 2^j \) components and each component represents the projection of the signal at a specific time interval and frequency band. However, this decomposition has a lack of translation-invariant property, causing ineffectiveness for using the wavelet packet coefficients as the features directly. As a result, a new feature extraction method is needed [20].

The proposed features are based on the energy of the wavelets including the time marginal energy (TME) and the frequency marginal energy (FME), as defined below

\[ \text{TME}_{j,k} = \sum_{n=2^j}^{2^{j+1}-1} w_{j,k,n}^2 \]  
\[ \text{FME}_{j,n} = \sum_{k=1}^{2^j-1} w_{j,k,n}^2 \]

where \( j = 1, 2, \ldots, L \) and \( L = 2^J (J \in \mathbb{Z}_+) \) is the length of the original signal. For example, Fig. 6 shows a 3-D view of the wavelet packet decomposition parameters of a typical strain signal acquired from step stamping operation under normal conditions. According to (2) and (3), Fig. 7 demonstrates the marginal energies.

TME and FME have clear physical meanings: The former represents the signal energy in a specific time interval, while the latter reveals the signal energy at a particular frequency band. Deriving from the wavelet packet transform, they inherit the advantages: TME has high time resolution while FME has high-frequency resolution. The final features come out from these marginal energy components according to the particular application, (for example, the top ten components of TME or FME) and they are more robust.

IV. MODULAR SYSTEM FOR FAULT DIAGNOSIS

A. Hidden Markov Model

The HMM is a dynamic modeling method, which was developed to solve real-world problems [11]. It describes a doubly embedded stochastic process using its model parameters \( \lambda = (A, B, \pi) \), where \( A = \{a_{ij}\} \) is the \( N \times N \) state transition probability matrix and \( B_t = \{b_{ij}(k)\} \) is the \( 1 \times N \) observation probability at time \( t \) of each state. These observations are presented in either continuously or discrete formats; \( \pi = \{\pi_i\} \) is the initial state. There are three sets of probabilities: the transition probability \( a_{ij} = P(q_{t+1} = s_j | q_t = s_i) \), where \( \{S\} \) is the state \( i, j = 1, 2, \ldots, N \) (\( N \) is the number of states) and \( t = 1, 2, \ldots, T \) denotes the time; the observation probability \( b_j(k) \); and the initial state probability \( \pi_i = P(q_1 = s_i) \).

In practice, a machine’s working condition cannot be observed directly, just as the hidden states of HMM; however, it can be deduced through describing its process information acquired by the sensors and their probability distribution. Demonstrated by many applications, HMM induces the invariant of the observation, so that it can characterize different working conditions of manufacturing operations. Along with a capacity for learning, the optimal model parameters can also be achieved from training samples. Moreover, the model is developed solely on the basis of condition samples resulting in constructing a modularized diagnosis architecture, which helps the system to deal with the demand and be easily modifiable when new faulty condition appears on the task list in real-world applications.
Therefore, HMM is suitable for condition monitoring and diagnosis applications.

B. HMM-Based Modular Fault Diagnosis System

The modular systems are pursued for meeting the demands in practice. To achieve this goal, the proposed diagnosis architecture was developed and illustrated in Fig. 8. With the training and encoding, a number of independent HMM models are developed. Each model is designed for identifying one particular condition, working in parallel with others. Using Viterbi algorithm [21], the optimal state sequence $Q^{*} = (q_1^{*}, q_2^{*}, \ldots, q_T^{*})$ is determined through the maximum expected probability. Together with $B^{*}$, they are the unique code of the related fault type. Therefore, these HMMs build up a codebook after the training phase. In real-world applications, decoding will be completed using the maximum likelihood criterion, which can be viewed as classification.

Given a new sample, $O^{\text{norm}}$, each model can then calculate the probability $P(O^{\text{norm}}|Q^{*}, \lambda)$ according to the following equation:

$$P_k(O^{\text{norm}}|Q_k^{*}, \lambda_k) = \sum_{Q_k} P(O^{\text{norm}}|q^*, \lambda_k) P(q^*|\lambda_k)$$

$$= \sum_{q_1^{*}q_2^{*}\cdots q_T^{*}} \pi_{q_1^{*}} h_{q_1^{*}}(o_1) a_{q_1^{*}q_2^{*}} h_{q_2^{*}}(o_2) \cdots a_{q_{T-1}^{*}q_T^{*}} h_{q_T^{*}}(o_T)$$

(4)

where $k = 1, 2, \ldots$ is the $k$th HMM. According to maximal likelihood criterion, the unknown condition can be identified by comparing the likelihood calculated by these HMMs. The one with the maximum probability will indicate the faulty condition.

C. Experiment Results

For comparison, an ANN with a back-propagation (BP) algorithm [22] has been applied to each case. In the first experiment, 20 samples from each condition were used as training samples while the remainder was used for verification. In the second experiment, except the double stroke faulty condition, the other samples were divided into two categories: normal and abnormal. For ANN-BP and HMM applications, 20 training samples were derived from normal while another 20 were derived from slugs and misfeeds.

In the implementation of the fault diagnosis using the feature and the proposed method, there are two factors that impact performance: feature representation and states in the HMM. From the experimental results presented in Table II, it is noted that marginal energies characterize the strain signal of stamping processes better than other methods.

Without theoretical support, a number of states of the HMM were attempted from $N = 3$ to $7$ in experiment 1, and $N = 2$ to 4 for experiment 2. These experimental results are depicted in Fig. 9, where Daubechies family wavelet functions db4 and db5 are used.

As pointed out earlier, the states in the HMM have no clear physical meaning; nevertheless, the number of states is somehow related to the number of faulty conditions. For example, in Experiment 1, there are six different conditions and the best number of states is seven. In Experiment 2, there are three conditions and the best number of states is 2. Hence, it may be speculated that the best number of states is about the number of conditions to be identified. This can be used as a guideline for selecting the number of states of the HMM.

The proposed architecture (i.e., one model for one condition) makes the modification/expansion easy. In other words, when new information is acquired, we can simply modify/add a specific model without changing the others. This helps to reduce the retraining time and cost.

![Fig. 8. Architecture of the HMM-based fault diagnosis system.](image)

![Fig. 9. Experimental results using a different number of states.](image)
V. SUPPORT VECTOR APPROACH

A. Support Vector Machine

A support vector machine (SVM) is developed for binary classification. Briefly, with a given training samples set \( \{(x_i, y_i), i = 1, 2, \ldots, n\} \), where \( x_i \in R^m \), \( y_i \in \{-1, 1\} \), the goal of SVM is to define an optimal hyperplane in a high dimensional feature space with maximal margin

\[
y = \text{sgn}(f(x, w) = \text{sgn}(\langle w, \varphi(x) \rangle + b) \tag{5}\]

where \( w \) and \( b \) are the weighting factors deciding the complexity of the hyperplane. The nonlinear mapping \( x \rightarrow \varphi(x) \in Z \) is a feature extraction step for obtaining more information about the data being obtained in the high dimensional space. ERM principles cannot deal with the relationship between learning machine complexity and generalization capacity very well. Following the SRM principle, the upper boundary on the expected risk can be minimized:

\[
R(w) \leq R_{\text{emp}}(w) + \Phi(h/n) \tag{6}
\]

where \( R_{\text{emp}} \) denotes the empirical risk, \( \Phi \) is the confidence interval, and \( h \) is VC dimension of the hyperplane which reflects the classification capacity of the learning machine.

The optimal hyperplane can be obtained by solving the optimization problem

\[
\begin{align*}
\min_{w, b, \xi} & \quad \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i \\
\text{s.t.} & \quad y_i \cdot (\langle w, x_i \rangle + b) \geq 1 - \xi_i, \\
& \quad \xi_i \geq 0, i = 1, 2, \ldots, n
\end{align*} \tag{7}
\]

where \( \xi_i \) is a slack variable giving a soft classification boundary and \( C \) is a constant corresponding to the value \( \|w\|^2 \). In other words, \( C \) gives a tradeoff between two error terms: the range of the fault patterns and the number of the fault samples rejected.

By introducing the Lagrangian approach, the problem is equivalent to solving the quadratic optimization

\[
\begin{align*}
\min L(\alpha) &= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} y_i y_j \alpha_i \alpha_j K(x_i, x_j) \\
\text{s.t.} & \quad \sum_{i=1}^{n} y_i \alpha_i = 0 \\
& \quad 0 \leq \alpha_i < C, \quad i = 1, 2, \ldots, n.
\end{align*} \tag{8}
\]

Aiming to solve the heavy computational load, a kernel function \( K \) was introduced. The solution \( \alpha^* \) is unique. According to Karush–Kuhn–Tucker (KKT) complementary condition \([23], [24]\), only the elements corresponding to the input lying on the margin are nonzero, while the others are zero. In this case, it means

\[
\alpha_i[y_i(\langle w, \varphi(x_i) \rangle + b) - 1] = 0. \tag{9}
\]

The training samples on the margin are so-called support vectors. Thus, the optimal hyperplane is

\[
f(x, \alpha^*, b^*) = \sum_{i \in SV} y_i \alpha_i^* K(x_i, x) + b^*. \tag{10}
\]

B. Hierarchical SVM-Based Fault Diagnosis System

SVM has an exceptional ability for binary classification. To diagnose multiple conditions, the proposed fault diagnosis method is rather straightforward. As shown in Fig. 10, it consists of a number of SVMs. The first SVM is used to detect faults while the others are used to identify the faults. This design is necessary because SVMs can deal with only two classes. It is also because such a structure helps to quickly detect the faults so that appropriate reactions can be taken.

This system has a number of unique features. First, the utilization of kernel functions. The common functions used for
classification are

\[
K(x, z) = ((x \cdot z) + 1)^p \quad (11a)
\]

Radial Basis Function (RBF):

\[
K(x, z) = \exp \left( -\frac{|x - z|^2}{2\sigma^2} \right) \quad (11b)
\]

Sigmoid function:

\[
K(x, z) = \tanh(v \cdot (x \cdot z) + c). \quad (11c)
\]

The kernel function reflects the geometric relationship between the input vector and the support vector, or the similarities of the faults’ features. For example, the polynomial kernel function describes the similarity of the two vectors since the dot product \(\langle x_i, x_j \rangle\) depicts the canonical correlation. Choosing a different order \(p\) would result in different similarity measures and, hence, different results. The RBF kernel function approximates the relationship between the two vectors using a bell-shape function. Tuning the parameter \(\sigma\) would be like tuning the covariance. In the sigmoid kernel function, the parameters \(v\) and \(c\) can be used to adjust the shape of the sigmoid function.

Second, it has an exceptional ability to classify the unknown samples \((x, y)\). The classification work is carried out in the kernel-induced feature space. In order to make the classes linearly separable, normally, the inputs are mapped into a feature space if it is impossible to carry out the task in their original space. Replacing the dot products with the kernel \(K(x_i, x)\), the inputs are mapped into a high dimensional feature space from their original space nonlinearly. The type of kernel decides the dimension in the feature space. For example, if adopting the polynomial kernel of order 2 for a \(d\) dimension vector, the dimension in the feature space is \(d(d + 3)/2\). Thus, it is possible to depict the characteristics of the input more accurately and uniquely by using more information.

Third, the system does not require a large amount of training samples. Reviewing (6), it is noted that when \(h/l\) is big (for example, less than 20, the training sample is small), \(\Phi\) has a slight big value and the performance is poor in representing \(R(\mathbf{w})\) with \(R_{emp}(\mathbf{w})\). As a result, according to the ERM principle, the big training sample size is a premise to acquire a satisfied learning machine. The support vector technique was first introduced by Cortes and Vapnik [13] to solve the classification problems with a small number of training samples. SVM actually suggests a tradeoff between the quality of the hyperplane and the complexity of the classifier with a small training sample size because its formulation embodies the structural risk minimization principle as opposed to the ERM approach commonly employed in statistical learning. Therefore, SVM not only constructs an optimal hyperplane, which has a strong generalization capacity taking into account the small-size training sample, but also overcomes the overfitting and difficulties in structure design of the learning machine.

Finally, after training, the system does not require complicated calculations and, hence, can be done in real time. As shown in (9), the decision is dependent on the kernel function and the SVs. In most applications, the kernel function may be complicated but the number of SVs are limited (e.g., less than 20) and, hence, the decision is usually done in milliseconds.

### Table III

<table>
<thead>
<tr>
<th>Kernel function</th>
<th>Number of SV</th>
<th>Failure cases</th>
<th>Success rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polynomial kernel: (p = 2)</td>
<td>9</td>
<td>4</td>
<td>97.33</td>
</tr>
<tr>
<td>RBF kernel: (\sigma = 0.4)</td>
<td>6</td>
<td>9</td>
<td>94</td>
</tr>
<tr>
<td>Sigmoid kernel: (v = 2, c = 1)</td>
<td>4</td>
<td>7</td>
<td>95.33</td>
</tr>
</tbody>
</table>

### C. Experiment Results

During the implementation, how to set SVM parameters is a difficult problem. \(C\) is the important factor to be defined for the application. From above, it follows:

\[
|y_k| \leq \sum_{i=1}^{n} \alpha_i K(x_i, x) + b \leq \sum_{i=1}^{n} C \xi_i K(x_i, x) + |b|. \quad (12)
\]

To simplify the problem, assume that the RBF kernel function is used. Plus, \(K(x_i, x) \leq 1\); then, the upperbound of the regression function (corresponding to \(K(x_i, x) = 1\)) is

\[
|y_k| \leq C \sum_{i=1}^{n} \xi_i + |b|. \quad (13)
\]

Since \(y_k \in \{-1, 1\}\)

\[
C \geq \left(1 - |b|\right) \frac{1}{\left(\sum_{i=1}^{n} |\xi_i|\right)}. \quad (14)
\]

For the determination of other parameters (i.e., \(p, \sigma, v\)), several independent tests are conducted with different values.

In the first experiment, 10 samples from each condition form the training set. In the second experiment, 10 training samples were from normal while another 15 were from others. Table III presents the best results among the tests.

As pointed out earlier, SVM does not require a large number of training samples. For example, using the polynomial kernel function with \(p = 2\), with just two training samples, SVM achieves a success rate of 87%. The success rate reaches 96.5% with just four training samples. Increasing the number of training samples will further improve the performance but the improvement is not significant.

SVM is a very good tool for fault diagnosis. The success of the SVM is mainly attributed to the fact that it uses a kernel function to map the signals to a higher order feature space, in which additional characteristics of the signal can be detected. Interestingly, the choice of the kernel function is not crucial provided its capacity is adequate.
VI. SIMILARITY MEASURE FOR CONDITION MONITORING

A. Support Vector Regression

Support vector regression (SVR) models the process signal when applying the support vector techniques for function estimation. With predefined loss function, SVR tries to find a function \( f \) which constructs an analog margin in the target space \( y \in \mathbb{R} \) with, at most, certain insensitive for all samples, and is as flat as possible. Simply, using Vapnik’s \( \varepsilon \)-insensitive loss function [9], the problem is

\[
\min_{w, b, \varepsilon, \varepsilon'} \frac{1}{2} ||w||^2 + C \sum_{i=1}^{n} (\varepsilon_i + \varepsilon_i')
\]

s.t.

\[
\begin{align*}
  y_i &= f(x_i) + b - \varepsilon + \varepsilon_i \\
  \langle w, \varphi(x_i) \rangle + b - y_i &\geq \varepsilon + \varepsilon_i' \\
  \varepsilon_i, \varepsilon_i' &\geq 0 \\
  i &= 1, 2, \ldots, n
\end{align*}
\]

where slack variables \( \varepsilon_i, \varepsilon_i' \) cope with additive noise. The parameter \( C \) provides a trade-off between the flatness of \( f \) and the amount to which deviations are larger than \( \varepsilon \). Using the approach similar to SVM, the optimal problem can be solved, resulting in the estimated function

\[
f(x, \alpha, \alpha^*) = \sum_{i \in SV} (\alpha_i - \alpha_i^*) \varphi(x_i, x) + b
\]

(16)

where

\[
\begin{align*}
  \alpha_i (y_i - (\alpha_i - \alpha_i^*) \varphi(x_i, x_j) - b - \varepsilon - \varepsilon_i) &= 0 \\
  \alpha_i^* (\alpha_i - \alpha_i^*) \varphi(x_i, x_j) + b - \varepsilon - \varepsilon_i' - y_i &= 0 \\
  b &= y_i - \sum_{j=1}^{n} (\alpha_j - \alpha_j^*) \varphi(x_j, x_i) - \varepsilon \quad \text{if} \quad \alpha_i \in (0, C) \\
  b &= y_i - \sum_{j=1}^{n} (\alpha_j - \alpha_j^*) \varphi(x_j, x_i) + \varepsilon \quad \text{if} \quad \alpha_i^* \in (0, C).
\end{align*}
\]

(17)

SVR possesses all of the advantages of the support vector techniques: it intuitively maps the input vectors onto a high dimensional feature space, and subsequently locates an optimal function \( f \) that not only reduces the risk but also controls the generalization capability. Meanwhile, only the training samples lying on the \( \varepsilon \)-tube are involved in the regression estimation.

B. Similarity Measure

Aiming to minimize the learning risk, the ideal learning algorithm wants to minimize the loss function

\[
R(w) = \int |y - f(x, w)| dp(x, y)
\]

(19)

where \( p(x, y) \) is the probability distribution of \( (x, y) \). Due to the lack of information on \( p(x, y) \), it is common to use the empirical risk function estimated from the training sample set. In the applications of support vector techniques, loss function plays an important role in signal modeling. To overcome overfitting and enhance generalization, a capacity control term is often added

\[
R_{reg}(w) = R_{emp}(w) + \frac{\lambda}{2} ||w||^2
\]

(20)

where \( c \) denotes a loss function determining how the estimation error based on the training samples will be penalized. \( \lambda > 0 \) is regularization constant. The last term controls the flatness of the function. In this context, the use of the flatness denotes that one seeks small \( w \). This has a significant effect on the generalization capability of the algorithm. Thus, one might consider which loss function should be used. Ideally, it should consist of a simple structure to avoid difficult optimization problems, and it should be suitable for the data.

Meanwhile, the noise affects the samples even after de-noising. A set of training samples is generated by a function \( f \) plus additive noise \( \xi \).

\[
y_i = f(x_i) + \xi_i.
\]

(21)

The likelihood of an estimate \( \mathbf{f}_f = \{(x_i, f(x_i, w))|i = 1, 2, \ldots, n\} \) based on the training sample set is

\[
\mathcal{P}(\mathbf{f}_f | \mathbf{F}) = \prod_{i=1}^{n} \mathcal{P}(f(x_i, w) | (x_i, y_i)) = \prod_{i=1}^{n} \mathcal{P}(f(x_i, w) | y_i)
\]

(22)

\[
\mathcal{P}(\xi) = \prod_{i=1}^{n} \mathcal{P}(\xi_i) = \prod_{i=1}^{n} p(\xi_i)
\]

where \( p(\xi) \) is the noise density. The appropriate loss function can maximize the likelihood, it is equivalent to maximize log

\[
\log \mathcal{P}(\mathbf{f}_f | \mathbf{F}) = \sum_{i=1}^{n} \log \mathcal{P}(y_i - f(x_i, w)).
\]

(23)

Thus, the appropriate loss function is

\[
c(x, y, f(x, w)) = - \log \mathcal{P}(y - f(x, w)) = - \log p(\xi).
\]

(24)

The abnormal condition can be detected by comparing the similarity measure, which is defined by

\[
\log \mathcal{P}(\mathbf{f}_f | \mathbf{F}) = \sum_{i=1}^{n} \log p(\xi_i) = - \sum_{i=1}^{n} c(x_i, y_i, f(x_i, w)).
\]

(25)

The importance of a new similarity measure is twofold: On one hand, it presents a principle that enables the construction of a loss function. Once the noise density of the system is defined, its related loss function can be obtained according to (25). In real-world applications, the standard Gaussian density model \( N(0, 1) \) is commonly used to describe noise. Hence, the
Gaussian density model and its loss function were employed here as outlined in (26)
\[
p(y - f(x, w)) = p(\xi) = \frac{1}{\sqrt{2\pi}} \exp \left( -\frac{1}{2} \xi^2 \right)
\]
\[
c(x, y, f(x, w)) = \frac{1}{2} (y - f(x, w))^2 = \frac{1}{2} \xi^2, \tag{26}
\]

On the other hand, the result provides a theoretical standard by which it can determine whether two signals are generated in the same condition or not. Under normal conditions, the process demonstrates unique characteristics that are reflected in its related signals. The signal patterns in abnormal conditions exhibit deviations, referred to here as noise. The prediction values emerge when substituting the unknown input vectors into the regression models. With a certain noise density model, or the related loss function, the unknown input vector’s likelihood of estimating the function, so-called similarity measure, can be calculated according to (25). As a result, it is possible to implement condition monitoring by comparing the similarity of an unknown signal with the likelihood distribution of normal ones.

C. Novel Condition Monitoring System

Based on the above discussion, a novel condition monitoring system can be constructed according to the system architecture showed as Fig. 11.

There are two paths: an offline training path and an online implementation path, used for learning and detection, respectively. In offline mode, the estimate for the regression will be based on collecting a number of training sample sets in normal condition, which should be already processed if the raw data exhibit poor SNR. Consequently, the mean and variance will be calculated from the likelihood values \(P\), acquired according to the noise density model of the corresponding sample sets. This finally enables us to calculate the threshold of \(P\). In practice, by assessing the likelihood of real data (acquired by following the online implementation path), the system can detect faults and issue appropriate and efficient commands to the actuator.

Gaussian density \(N(0,1)\) is commonly used to describe noise. It defines the loss function as \(c = \xi^2/2\). Although the objective function has little difference with (15), the regression estimation is expressed in the same way as (16). Aiming to estimate the signal pattern, the model is constructed based on samples
\[
y_i = f(x_i) + \xi = f(x_{i-1}, x_{i-2}, \ldots, x_{i-T}) + \xi_i \tag{27}
\]

where \(T\) is the regression step. In this application, \(T = 10\).

Based on the SVM, the system has a series of features: it has an exceptional ability to model signals due to the regression estimation in the kernel-induced feature space. The kernel type determines the dimension in the feature space. Furthermore, the system does not require complicated calculations because of the simplicity of kernel functions and support vectors. It can operate in real time.

Moreover, it possesses unique merits: First, the solution is globally optimal. The quadratic optimization process makes the solution globe minima. Second, the system requires training sample sets produced under normal operating conditions and, thus, makes it more feasible than other techniques. Given a number of training sample sets, the SVR technique will calculate dependency between input vectors and outputs. In addition, its loss function defined by the noise density model helps compute thresholds of the likelihood automatically. Thus, the system has the intelligence to cope with different processes.

D. Experiment Results

In all three experiments, to form the training set, 20 samples were picked up from normal condition only. Table IV lists the experiment results with Fig. 12, where the solid line is the mean value of the training sample, the broken line is the threshold, “*” represents the normal samples, and “+” represents abnormal samples.

Fig. 12 shows that the likelihood of the samples under normal conditions is quite centralized in the first and second experiments, and that samples in abnormal conditions have a wide range of distribution. Due to poor SNR in the third experiment, signal patterns for different working conditions are similar.
which was reflected in the distribution of likelihood. However, the experimental results show that the system has robust performance. The approach extracts the invariant features from the strain signal for normal conditions.

In comparison, the SVR performed perfectly. The SVM and SVR techniques share a common basic technical background. The former requires samples from all conditions for training the system, which results in the acquisition of information richer than the latter. However, unstable signals in the third experiment make it difficult to choose the training samples. By modeling the normal condition signals and considering the characteristics of noise density, the proposed system shows a significant improvement in the third experiment. More appealing though, the proposed system only requires samples from normal operating conditions for training, which makes the approach more ideal for real-world application.

VII. PERFORMANCE ANALYSIS

Comparing the above learning algorithms demonstrates their difference:

1) Learning Mechanism: Developed from the Markov chain method, HMM provides a way to model the stochastic process from observations. It is a tool for solving real-life problems rather than a general model of the learning phenomenon. Whereas, in inductive learning processes, both ANN and SVM determine an indicator for constructing a hyperplane in the feature space. ANN follows the ERM inductive principle, which minimizes errors in the training data, while SVM realizes the SRM principle, which minimizes an upperbound with regard to the expected risk. The latter not only solves many problems of the former (i.e., structure design, local minimum, overfitting, etc., but also overcomes the drawbacks of large samples for training). In the first test, when picking up four samples from each condition as a training set, the success rate reached 96%, so that the small-size training set is large enough for proper classification.

2) System Structure: HMM allows the construction of a feasible modularized system that conveniently modifies or adds diagnosis units. The structure of SVM is similar to ANN. However, its weight values are predetermined by the training patterns, and each node corresponds to a specific supper vector; thus, it is automatically determined after training. ANN can have multiple outputs, while the SVM has only one output. Using a hierarchical structure, the SVM-based system can deal with processes with multiple conditions.

3) Performance: SVM performs better because it realizes SRM principles and is able to map the input from lower dimensional space to a high dimensional kernel-induced feature space. Although the performance of the SVR-based system is not the best, the system has its unique advantages. It can cope with time-shifting, and induce the invariant of the process. Thus, it is suitable for complex cases; for example, the testing results have been greatly improved by this method in the third test. Meanwhile, requiring only normal condition data for training, the system is more feasible.

VIII. CONCLUSIONS AND FUTURE WORK

Based on the discussion above, the following conclusions can be drawn.
1) Marginal energy characterizes the transient or non-stationary process effectively, eliminating noise and enhancing robustness. The coefficients in (2) and (3) can be derived from a wavelet packet transform or from time–frequency distributions. The marginal energy outperforms other approaches at feature representation for transient or non-stationary processes. It has multiresolution and translation-invariant characteristics.

2) HMM helps in modular software architecture to construct a modularized strategy that modifies or adds diagnostic units easily and can induce the invariant.

3) The SVM-based hierarchical structure performs well in identifying different conditions. It solves many practical problems, such as structure design, optimal solution, and small-size training sample learning.

4) The proposed system based on similarity measures and signal modeling using SVR can deal with time-shifting and detect abnormal conditions efficiently and effectively. It has potential in pattern recognition and signal modeling research, as well as other engineering practices. The proposed system provides important technologies and valuable references for research areas, and an innovation to industry. The system works well in most applications; however, in some cases, certain parameters have to be tuned manually, for example, the parts of the signals to be modeled. The next step is to develop the system so it can learn human skills and set parameters automatically. In the short term, it is important to research ways to improve the system’s intelligent online learning.

ACKNOWLEDGMENT

The authors would like to thank Dr. Zhang, Dr. Lee, H. Tong, and Mr. Zhao. Their contribution to this system development is significant.

REFERENCES


Ming Ge received the Ph.D. degree in automation and computer-aided engineering from The Chinese University of Hong Kong (CUHK), Hong Kong, China, in 2003.

He has been with the Hong Kong Productivity Council since 2004, and served as Senior Consultant of Manufacturing. Currently, he is the Adjunct Professor with Sun Yat-Sen University, Guangzhou, China, and the Visiting Professor with Guizhou University, Guiyang, China. From 2003 to 2004, he was the Research Associate Fellow with the University of Wisconsin-Milwaukee and the Research Fellow with CUHK. His research interests include artificial intelligence, pattern recognition, model analysis, decision making, and automation. He has published many papers in international journals.

Dr. Ge had one of his papers in the Best Student Paper Shortlist in IRCA 2004.

Yangsheng Xu received the Ph.D. degree in robotics from the University of Pennsylvania, Philadelphia, in 1989.

He was Chairman of the Department of Automation and Computer-Aided Engineering from 1997 to 2004, where he is currently Chair Professor in the Department. He was a Faculty Member with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, from 1989 to 1999. His research has been in the areas of robotics, automation, and interface.

Dr. Xu is a Fellow of The Hong Kong Institution of Engineers (HKIE). Fellow of the International Eurasian Academy of Sciences, and Advisor of the Chinese National Program on Aerospace.
Ruxu Du was born in China in 1955. He received the M.Sc. degree from the South China University of Technology, Guangzhou, China, in 1983 and the Ph.D. degree from the University of Michigan, Ann Arbor, in 1989.

He has taught at the University of Windsor, Windsor, ON, Canada, and with the University of Miami, Coral Gables, FL. Currently, he is a Professor in the Department of Automation and Computer Aided Engineering at the Chinese University of Hong Kong (CUHK). He is also the Director of the Institute of Precision Engineering of CUHK. His research areas include condition monitoring, fault diagnosis, and manufacturing processes (metal forming, machining, plastic injection molding). He has published many papers in international journals and conferences.

Dr. Du is the Associate Editor of two international journals and has won a number of awards.