Highly Accurate Machine Fault Diagnosis Using Deep Transfer Learning

Siyu Shao, *Student Member, IEEE*, Stephen McAleer, Ruqiang Yan, *Senior Member, IEEE*, and Pierre Baldi, *Fellow, IEEE*

***Abstract*—We develop a novel deep learning framework to achieve highly accurate machine fault diagnosis using transfer learning to enable and accelerate the training of deep neural network. Compared with existing methods, the proposed method is faster to train and more accurate. First, original sensor data are converted to images by conducting a Wavelet transformation to obtain time-frequency distributions. Next, a pretrained network is used to extract lower level features. The labeled time-frequency images are then used to fine-tune the higher levels of the neural network architecture. This paper creates a machine fault diagnosis pipeline and experiments are carried out to verify the effectiveness and generalization of the pipeline on three main mechanical datasets including induction motors, gearboxes, and bearings with sizes of 6000, 9000, and 5000 time series samples, respectively. We achieve state-of-the-art results on each dataset, with most datasets showing test accuracy near 100%, and in the gearbox dataset, we achieve significant improvement from 94.8% to 99.64%. We created a repository including these datasets located at mlmechanics.ics.uci.edu.**

***Index Terms*—Convolutional neural network (CNN), deep learning (DL), fault diagnosis, machine health monitoring, pretrained model, transfer learning (TL).**

I. Introduction

MACHINE fault diagnosis is concerned with monitoring mechanical machines, identifying when a fault has occurred, and categorizing the fault. To identify and categorize faults, multiple sensors are installed to collect data, such as vibration data or thermal imaging data. These data are then processed to determine whether a fault has occurred and then categorize that fault. Traditionally, machine fault diagnosis contains three main stages: sensor signal acquisition, feature extraction and selection, and fault classification. Sensor signal acquisition involves collecting sensor data while the machine is running. Feature extraction is traditionally done through time-frequency analysis, which includes original sensor data in both the frequency and time domains. For the final fault classification stage, the extracted features are used to train machine learning models to make fault predictions. However, there are several limitations with these traditional fault diagnosis methods.

1) Conventional fault diagnosis methods are based on manually selected features. As a result, fault classification performance can decrease significantly if these manually selected features are inadequate for the task.
2) Handcrafted features are task specific for different classification tasks which means features used to accurately make predictions under certain circumstances are unsuitable for other scenarios. It is difficult to design a set of features that are able to generate reliable predictions among all conditions.

Deep learning (DL) methods provide effective solutions to overcome the above limitations in part due to their powerful feature learning ability. Deep architectures have multiple hidden layers which have the capability to learn hierarchical representations directly from the raw data. Through model training, deep architectures are able to automatically select discriminative representations that are useful for making accurate predictions in subsequent classification stages according to the training data. DL has been successfully applied to many areas of science and technology [1], such as computer vision [2], speech recognition [3], natural language processing [4], games [5], particle physics [6], [7], organic chemistry [8], and biology [9], [10], to name just a few areas and examples.

Not surprisingly, DL has been applied to mechanical fault diagnosis. For instance, Sun *et al.* [11] designed an autoencoder-based neural network for induction motor diagnosis and achieved accurate fault prediction. Ding *et al.* [12] developed a spindle bearing fault diagnosis system using wavelet packet energy as the input to a deep convolutional neural network (CNN) and obtained reasonable diagnostic performance in diagnosing various machine failures. An enhanced gated recurrent network was used to estimate the remaining life of a machine and conduct fault diagnosis in both gearbox and rolling element bearings in [13].

Manuscript received July 20, 2018; accepted August 1, 2018. Date of publication August 10, 2018; date of current version April 3, 2019. The work of Siyu Shao was supported in part by National Natural Science Foundation of China under Grant 51575102 and in part by Fundamental Research Funds for the Central Universities and Research Innovation Program for College Graduates of Jiangsu Province under Grant KYLX16_0191. The work of Siyu Shao, Stephen McAleer, and Pierre Baldi was also in part supported by DARPA under Grant D17AP00002 to Pierre Baldi. Paper no. TII-18-1882. *(Corresponding authors: Ruqiang Yan and Pierre Baldi.)*

S. Shao and R. Yan are with the School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China (e-mail: cathygx.sy@gmail.com; ruqiang@seu.edu.cn).

S. McAleer and P. Baldi are with the Department of statistics, Department of Computer Science, School of Information and Computer Science, University of California, Irvine CA 92697-3435 USA (e-mail: mcaleer.stephen@gmail.com; pfbaldi@ics.uci.edu).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org..

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2018.2864759

Although DL models have achieved successful applications in machine fault diagnosis tasks, there are still problems associated with DL methods. First, deep models implemented by most of the aforementioned papers have less than five hidden layers. Deep models with more than ten hidden layers have not been investigated and their performance in machine fault diagnosis tasks has not been assessed. However, as the number and size of the hidden layers increases, the number of free parameters increases too, and training very large networks from scratch usually requires massive amount of labeled data and considerable computational and time resources. In addition to parameter optimization, hyperparameter tuning (architecture, learning rates, dropout rates, etc.) greatly affects performance and is also very time consuming.

A promising approach to overcome the difficulties of training a deep architecture from scratch uses transfer learning (TL) where instead of fully training a neural network with random initialization, a deep neural network that has been trained from sufficient labeled data in a different application is used and fine-tuned based on the task at hand. TL focuses on using knowledge gained in one problem to solve a different but related problem. It has been used in many applications, such as text classification and spam filtering [14].

TL has been applied to the research on fault diagnosis. For example, Wen proposed a deep TL method for fault diagnosis based on sparse autoencoder that can learn common features across different working conditions [15]. Shen proposed a TL-based approach for bearing fault diagnosis where auxiliary data are transferred to improve diagnostic performance among various operating conditions [16].

These applications of TL require the training of large architectures from scratch. In contrast, this paper focuses on using TL to accelerate the training process of deep architecture and learn hierarchical representations. This is achieved by using pretrained deep CNNs, which have been pretrained using large datasets of natural images. As we shall see, the network architectures, model parameters, and model hyperparameters can be transferred to the target model for machine fault diagnosis. The lower level weights of the target neural network are obtained from the pretrained model and the higher level weights are fine-tuned for the specific fault diagnosis task. In this way, TL gives the target model a reasonable initialization and reduces the number of parameters that need to be updated. Thus, TL greatly improves the training process for deep neural network.

The main contributions of this paper can be summarized as follows.

1) We present a machine fault diagnosis framework based on a deep CNN which uses transfer strategy to improve model training efficiency. Lower-level network parameters are transferred from a previously trained deep architecture, trained using natural images, followed by fine tuning of the high-level parameters and the entire architecture using task-specific mechanical data. The proposed framework uses a deep architecture with more than ten hidden layers to learn hierarchical representations from sensor data and is able to achieve quick and accurate working condition recognition. This paper is the first attempt to use pretrained models for efficient applications in the machine fault diagnosis domain.
2) The performance of the proposed framework using a pretrained deep model is compared with the performance of a CNN trained from scratch using the same data. The proposed approach leads to faster training and better accuracy. In addition, the proposed approach is also compared to other machine learning methods in terms of fault state classification accuracy.
3) We report consistent results across three main mechanical datasets including induction motors, bearings, and gearboxes, demonstrating that the proposed framework leads to a state-of-the-art pipeline for fault diagnosis.

The rest of the paper is organized as follows. Section II introduces the theoretical background of the proposed approach, including time-frequency imaging, deep CNNs, and TL. In Section III, the overall mechanical fault diagnosis system is illustrated in detail. In Section IV, experimental studies on three datasets are carried out to verify the effectiveness of the proposed model, together with performance comparisons to other methods. Conclusions and future work are presented in Section V.

II. Methods

A. Time-Frequency Imaging

We use time-frequency imaging to convert original sensor data to images. Time-frequency imaging is a technique that transforms signal frequency time-series data to the time-frequency domain. This is a useful tool in analyzing mechanical sensor signals for fault diagnosis since it provides an insight into the original data in the context of time [17] and since different time-frequency patterns are specific to different machine working conditions. Time-frequency imaging can be obtained from various methods, including short-time Fourier transform (STFT), continuous wavelet transform (CWT), Wigner-Ville distribution, etc. Among them, CWT is an effective technique to represent signals in multiple resolutions.

The wavelet transform is widely used in feature extraction in fault diagnosis tasks and can be regarded as a mathematical tool to transform time-series to another feature space. We use a CWT to obtain time-frequency distributions, generating representations of the original signal in the time and frequency domains simultaneously [18].

Wavelet transform conducts an inner product operation of the signal and a set of the wavelets. This set of wavelets is a wavelet family realized by scaling and translating the mother wavelet $\psi(t)$, shown as

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-\tau}{s}\right) \tag{1}$$

where s is a scale parameter inversely related to frequency, and τ is a translation parameter.

A CWT of a signal $x(t)$ can be obtained by a convolution operation of a complex conjugate, mathematically defined as follows:

$$W(s,\tau) = \langle x(t), \psi_{s,\tau}\rangle = \frac{1}{\sqrt{s}}\int x(t)\psi^*\left(\frac{t-\tau}{s}\right)dt \tag{2}$$

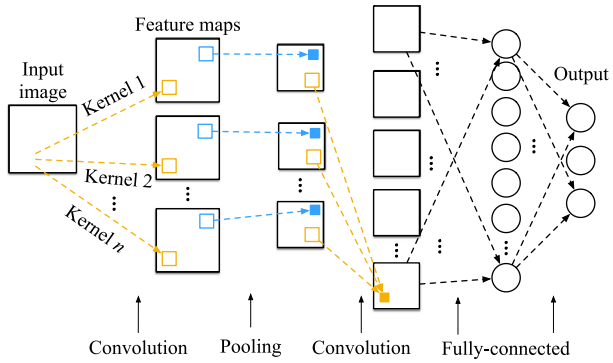

Fig. 1. Typical convolutional neural network architecture.

where $\psi^*(\cdot)$ denotes the complex conjugate of the above function $\psi(\cdot)$. This equation demonstrates that the CWT is similar to the Fourier transform where a signal can be decomposed into the frequencies that it is composed of. Through this operation, the signal $x(t)$ is decomposed into a series of wavelet coefficients where the wavelet family is the basis function. Based on above equations there are two kinds of parameters in family wavelets: s and τ. After the convolution operation, the signal $x(t)$ is transformed by the family wavelets and projected to the two-dimensional (2-D) time and scale dimensions [18]. In this way, one-dimensional time series are converted to time-frequency images.

B. CNN

CNNs are widely used when dealing with image tasks. A deep CNN is able to learn hierarchical features automatically from input images where features from higher level layers are more abstract than those from lower layers. These abstract features are beneficial for accurate classification and are learned automatically. In general, CNNs contain three kinds of layers: convolution layers, pooling layers, and fully connected layers. Convolution and pooling layers are combined to form convolution blocks, and several such blocks are stacked to build a deep architecture. Usually a fully connected layer is used as the last layer to perform classification or regression. Fig. 1 shows a typical CNN architecture.

The main ideas behind CNN are local receptive fields, shared weights, and the pooling operation. Compared with fully connected neural networks, CNNs adopt the concept of a local receptive field where only a small localized area of the input image is connected to each node in a convolution layer. This way of connection dramatically reduces the number of parameters in the deep architecture and reduces the training difficulty of the network. In order to detect the same local feature throughout the whole image, weights and biases within one convolution layer are shared between hidden neurons and each set of shared weights and biases is called one convolution kernel.

Each kernel is convolved across the width and height of the input, computing the dot product between the kernel and the input. The kth feature map before nonlinear transformation has the feature value Z_k, given by

$$Z_k = W_k \otimes x + b_k \tag{3}$$

where W_k denotes kth convolution kernel and b_k represents bias term; x is the input image of this convolution layer; $\otimes$ is a 2-D convolution operation, performing a dot product of the kernel and the input. Adding activation functions $a(\cdot)$ gives nonlinearities to convolution layers, which can be denoted as

$$A_k = a(Z_k) \tag{4}$$

where A_k represents a nonlinear feature value of the kth feature map and activation function $a(\cdot)$ is a rectified linear unit which is widely used in DL architectures.

To generate feature maps through various kernels, convolution operation between kernels and input images are performed as

$$a_{i,j} = a\left(\sum_{d=0}^{D-1}\sum_{m=0}^{F-1}\sum_{n=0}^{F-1} w_{m,n,d}\, x_{i+m,j+n,d} + b\right) \tag{5}$$

where $a_{i,j}$ represents the node in the output feature map with the location of (i,j); x is the input image which has depth of D. $x_{i+m,j+n,d}$ denotes the value at the location of (i,j) in the dth depth; the weight of convolution kernel at the location of (m,n) in the dth depth is denoted as $w_{m,n,d}$. b represents kernel bias and $F \times F$ is the size of convolution kernel.

Convolution layers are followed by a pooling layer which performs down sampling operation to previous feature maps. Mathematically, a pooling operation is defined as

$$y_{i,j,k} = \text{pool}_{(m,n)\in R_{ij}}(x_{m,n,k}) \tag{6}$$

where $\text{pool}(\cdot)$ represents the pooling rule and $y_{i,j,k}$ represents the new value at the location of (i,j) in kth feature map after the pooling operation. R_{ij} is the pooling receptive region around the location (i,j) and $x_{m,n,k}$ denotes the node at location (m,n) within the receptive field. In practice, a max pooling operation is the most commonly used pooling rule applied to image classification task. In max pooling, the maximum value within a pooling region is selected and propagated to the next layer.

After a number of stacked convolution blocks, fully connected layers are used to perform classification or regression [19], [20]. For classification tasks, after a set of fully connected layers, the output layer is a $softmax$ function to predict categories. Mathematically, it can be obtained by

$$y = f\left(Wa + b_c\right) \tag{7}$$

where y represents the predicted labels; W represents the weight matrix between a, neurons in fully connected layer, and connected nodes in output layer; b_c denotes the bias, and f is the $softmax$ activation function.

CNNs used for classification are trained by minimizing the cross entropy loss, which is defined as

$$H(r,p) = -\sum_i r_i \log(p_i) \tag{8}$$

where r is 0 or 1 corresponding to the true label, and p denotes the output probability from the CNN model. Cross entropy loss serves to evaluate errors between true labels and predicted probabilities. The gradients of the weights with respect to this loss function are found by backpropogation and the weights are updated by stochastic gradient descent.

C. TL and Fine-Tuning Strategy

TL is an established area of machine learning research. In simple terms, given a source domain D_s and a target domain D_t, TL tries to apply knowledge learned previously from D_s to D_t. Here, TL is able to help training a target model by initializing the target model with parameters that are transferred from a pretrained model.

Training a deep architecture from scratch is difficult in practice. Large deep neural networks contain a large number of weights which are randomly initialized before the training procedure and iteratively updated based on labeled data and the loss function. Updating all the weights iteratively is extremely time consuming, and with limited training data, deep architectures have the possibility to overfit to the training data.

TL provides a promising alternative that makes use of a pretrained deep CNN which is already trained by another dataset. As shown before, CNNs are able to learn hierarchical representations from images, and the knowledge embedded in the pretrained model's weights can be transferred to the new task. Lower-level convolution layers extract low-level features like edges and curves, which are applicable to common image classification tasks, while operations in later layers can learn more abstract representations that are specific to different application fields. Therefore, lower-level representations can be transferred, and only the higher-level representations need to be learned from the new dataset. The procedure for updating the weights of higher hidden layers is called fine-tuning, and its success partly depends on the "distance" between the source dataset and the target dataset. For similar datasets, one can fine-tune only the fully connected layers, while for datasets that have considerable differences, several convolution blocks need to be updated. Compared with training from scratch, this approach is faster as it essentially reduces the number of parameters that need to be trained.

Training a deep CNN from pretrained weights has already been successfully applied to several different tasks [21], and most of the existing pretrained models are well trained from sufficient natural image data. For instance, in biomedical imaging, although there is often considerable difference between natural images and images from the biomedical domain of interest, several studies have demonstrated the effectiveness of applying pretrained models to medical imaging tasks. Inspired by these achievements in the biomedical and other domains, here we investigate knowledge transfer from natural image to time-frequency imaging of mechanical dataset.

III. Machine Fault Diagnosis using Deep TL

The proposed frame for detecting the working conditions of a mechanical system with high precision is based on deep CNN and time-frequency images are used as the input. TL based on pretrained model helps improve deep model performance.

We propose a machine fault diagnosis pipeline that is able to automatically learn fault signatures and recognize machinery working states directly from the original vibration signals.

The overall procedure of the proposed mechanical fault diagnosis system is shown in Fig. 2, including time-frequency imaging, data preparation, pretrained model building with fine-tuning, and model application. The pretrained model used in this paper is a deep convolutional network created by Oxford Visual Geometry Group (VGG) [22] which is a 16-layer network. This model has achieved accurate classification performance on ImageNet dataset and detailed information about VGG-16 is shown in Table I and Fig. 3.

The pretrained model used in this paper was trained on ImageNet, and the target dataset is the time-frequency imaging of the mechanical dataset. Since the natural images and the time-frequency images are not similar; more convolution blocks need to be fine-tuned for the mechanical dataset compared to images that are more similar to natural images. We transfer the general features from first three convolution blocks. We then fine-tune

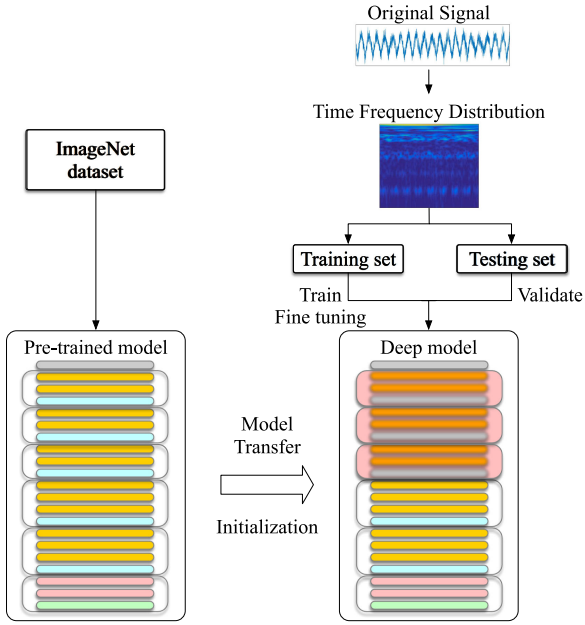

Fig. 2. General pipeline of mechanical fault diagnosis based on fine-tuning pretrained model.

TABLE I
DETAILED CONFIGURATION OF VGG-16 ARCHITECTURE

Layer	Type	Receptive field size - number of channels	Output	Number of parameters
Input	*Data*	-	$224*224*3$	0
$Block1-conv1$	*Convolution*	$3*3-64$	$224*224*64$	1792
$Block1-conv2$	*Convolution*	$3*3-64$	$224*224*64$	36928
$Block1-pool$	*MaxPooling*	$2*2$	$112*112*64$	0
$Block2-conv1$	*Convolution*	$3*3-128$	$112*112*128$	73856
$Block2-conv2$	*Convolution*	$3*3-128$	$112*112*128$	147584
$Block2-pool$	*MaxPooling*	$2*2$	$56*56*128$	0
$Block3-conv1$	*Convolution*	$3*3-256$	$56*56*256$	295168
$Block3-conv2$	*Convolution*	$3*3-256$	$56*56*256$	590080
$Block3-conv3$	*Convolution*	$3*3-256$	$56*56*256$	590080
$Block3-pool$	*MaxPooling*	$2*2$	$28*28*256$	0
$Block4-conv1$	*Convolution*	$3*3-512$	$28*28*512$	1180160
$Block4-conv2$	*Convolution*	$3*3-512$	$28*28*512$	2359808
$Block4-conv3$	*Convolution*	$3*3-512$	$28*28*512$	2359808
$Block4-pool$	*MaxPooling*	$2*2$	$14*14*512$	0
$Block5-conv1$	*Convolution*	$3*3-512$	$14*14*512$	2359808
$Block5-conv2$	*Convolution*	$3*3-512$	$14*14*512$	2359808
$Block5-conv3$	*Convolution*	$3*3-512$	$14*14*512$	2359808
$Block5-pool$	*MaxPooling*	$2*2$	$7*7*512$	0
$Fc1$	$Fully-connected$	$1*1*4096$	4096	102764544
$Fc2$	$Fully-connected$	$1*1*4096$	4096	16791312
$Output$	$Fully-connected$	$1*1*C$	C	$4096*C+C$

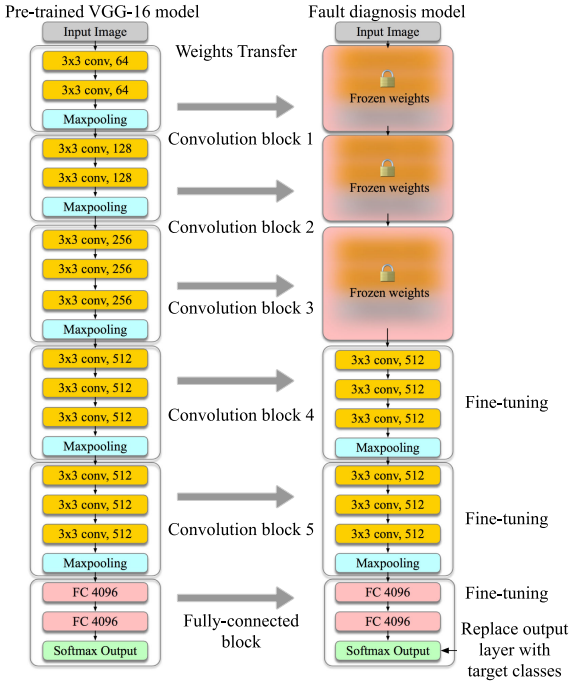

Fig. 3. Transfer learning procedure trains the three highest-level blocks of the pretrained VGG-16 network while leaving the weights of the bottom three blocks frozen.

the last two convolution blocks and the fully connected layers on the time-frequency dataset.

1) *Time-frequency imaging:* Vibration signals are acquired by sensors installed on mechanical machines during operation. These collected sensor signals are then transformed from the time domain to the time-frequency domain through a CWT, forming a set of time-frequency images which are utilized as the input of the following pretrained model.
2) *Data preparation:* In order to train and fine-tune the pretrained deep architecture, RGB images of a specific size are needed, so we must conduct data preparation on the time-frequency images. Since the converted distributions are gray-scale images which only have one-channel, a channel augmentation scheme is performed to achieve 2-D images with three channels by duplicating the gray-scale images into three channels and adding a basis to each channel. After that, the processed images are divided into two parts: the training dataset and the testing dataset. The training dataset is used to train the pretrained model and fine-tune its weights while the testing dataset is only used to verify the performance of deep model and is not used during training process.
3) *Pretrained model building and fine-tuning:* In this stage, we tune a pretrained deep CNN. After removing the top layer of the pretrained CNN and adding an output layer whose size is determined by the number of possible machine working conditions, the newly added output layer's weights are initialized randomly. We set the last two convolution blocks and the fully connected layers to be trainable. During the training procedure, earlier layers are frozen while the weights of trainable layers are updated to minimize errors between predicted labels and the true ones, shown in Fig. 3. After enough epochs, the designed model is fine-tuned and the deep architecture together with all the parameters are saved.
4) *Model application:* The testing dataset is used to validate the precision of the designed model on the mechanical fault diagnosis task and the fine-tuned pretrained deep model can be applied to diagnose machine working states.

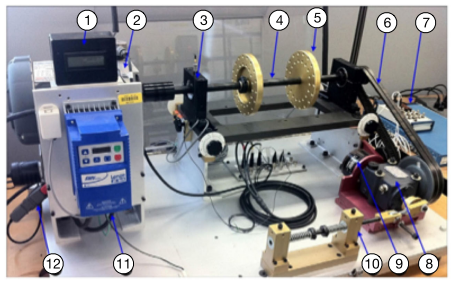

Fig. 4. Experimental facility. (1) Opera meter. (2) Induction Motor. (3) Bearing. (4) Shaft. (5) Loading Disc. (6) Driving Belt. (7) Data Acquisition Board. (8) Bevel gearbox. (9) Magnetic Load.(10) Reciprocating Mechanism. (11) Variable Speed Controller. (12) Current Probe [23].

TABLE II
MOTOR CONDITION DESCRIPTION [23]

	Condition	Description
HEA	Normal motor	Healthy motor without defect
SSTM	Stator winding defect	3 turns shorted in stator winding
UBM	Unbalanced rotor	Unbalance caused by 3 added washers on the rotor
RMAM	Defective bearing	Inner race defect bearing in the shaft end
BRB	Broken bar	Broken rotor bars
BRM	Bowed rotor	Rotor bent in center 0.01

IV. EXPERIMENTAL VERIFICATION

To test the performances of this proposed fault diagnosis system and verify its effectiveness, we conduct experiments on three datasets including induction motors, gearboxes, and bearings. Comparative experiments are also carried out to compare the classification accuracy with existing methods including both conventional feature based methods and DL-based methods. In addition, in order to compare the performances between fine-tuning a pretrained model and a CNN trained from scratch, we build a CNN model with three convolution blocks and train it from scratch with the same data. Due to page limit, detailed performances about each dataset are provided in the supplementary material, including classification accuracy, loss, and confusion matrix.

A. Induction Motor Dataset

The dataset used in these experiments is acquired by a machine fault simulator, shown in Fig. 4. Vibration signals are acquired by an acceleration sensor when the induction motor operates under six different conditions. Six different working conditions during motor operation can be simulated. Descriptions on each working condition simulation are shown in Table II [23].

TABLE III
CLASSIFICATION RESULTS FOR THE INDUCTION MOTOR DATASET

Fault diagnosis method	HEA	SSTM	UBM	BRB	RMAM	BRM	Average	Training time
[11]	92.68%	93.50%	99.55%	99.91%	100%	100%	97.61%	-
[23]	100%	98%	100%	100%	100%	100%	99.67%	-
[24]	-	-	-	-	-	-	99.98%	-
CNN trained from scratch	100%	99.00%	90.00%	99.00%	100%	100%	98.00%	1317s
Pre-trained model	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**397s**

For time-frequency imaging and data preparation, a time window containing 1024 data points is chosen to be one sample and every sample is converted to a time-frequency image with size of 224 × 224 which is suitable for the pretrained VGG-16 network. The gray-scale time-frequency images are extended to have three image channels by duplicating the original images, and the format of the these processed images before sending them to the deep neural network is 224 × 224 × 3. As described above, there are six different working conditions during induction motor operation where each working state is considered to be a separate category. The whole dataset is divided into training and testing subsets where each working condition contains 1000 samples for training and 100 samples for testing. Therefore, the size of the training set is 6000 and size of the testing set is 600.

Fault diagnosis for induction motors can be regarded as a 6-class classification task. The output layer of the pretrained VGG-16 model is replaced with a new layer with six neurons corresponding to six different working states with random weight initialization. To fine-tune the network, we freeze the first three convolution blocks and train the weights in the last two convolution blocks and the fully connected layers. During the training procedure, we use Adam optimizer with learning rate 0.0001 and the batch size is set to be 32. After finishing fine-tuning the deep model, we evaluate model classification performance on induction motor data with the test dataset.

We used the Keras library [25] for model training and fine-tuning. Classification accuracy is calculated to evaluate model performance and experimental results are shown in Table III with comparisons to an induction motor fault classification framework based on a sparse autoencoder architecture introduced in [11], a Deep belief network (DBN)-based fault diagnosis system proposed in [24], and conventional feature-based method [23].

As shown in these results, the proposed method is able to achieve accurate predictions of working states, outperforming all other methods on this induction motor dataset. Compared with traditional feature-based method [23], the proposed model is able to achieve higher accuracy with much less human intervention, especially in recognizing SSTM working state. Tenfold cross validation is carried out and the results are shown in Fig. 5. Training error curves for the proposed method as well as a CNN trained from scratch are shown in supplementary material. For ease of comparison, we calculated the time spent that the models were trained to achieve 99% accuracy, shown in Table III. Results show that the proposed fine-tuned model is able to achieve stable and accurate classification after two epochs, while the CNN trained from scratch needs more than ten epochs to achieve

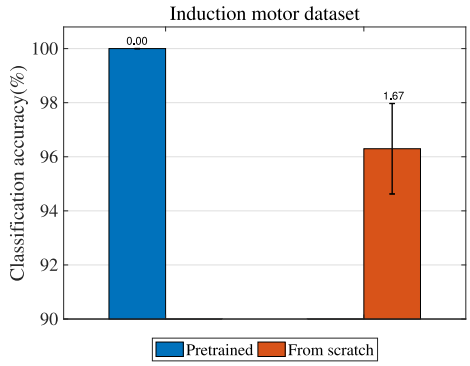

Fig. 5. Classification accuracy of average tenfold cross validation on the induction motor dataset.

99% accuracy. Although CNN model has only three hidden layers and has less parameters than pretrained model, it takes a lot of time to fully train a CNN with random initialization. Compared with training from scratch, the proposed pretrained model has better convergence speed and classification accuracy. Furthermore, the hyperparameters of the CNN trained from scratch need to be determined through trial and error to obtain best performance which is very time consuming, while the proposed pretrained model has already obtained optimal hyperparameters that can be transferred to fault classification tasks.

B. Bearings Dataset

Experimental data for the bearings dataset are provided by the Case Western Reserve University Bearing Data Center [31]. Vibration data were collected using attached accelerometers from the drive end of the motor housing. Vibration signals were collected under four different operational conditions with respect to different bearing loads (load 0–3 hp). Within each operational condition, single point faults were introduced with fault diameters of 0.007, 0.014, and 0.021 in on the rolling element, the inner raceway, and the outer bearings raceway, respectively. Each operational condition has nine different fault categories with one health state, ten kinds of bearing states in total.

In order to test the performance of the proposed framework under various working environments, several subdatasets are created as follows.

1) **A** Training data and testing data are both from vibration signals under working load of 0 hp.
2) **B** Training data and testing data are both from vibration signals under working load of 1 hp.
3) **C** Training data and testing data are both from vibration signals under working load of 2 hp.

TABLE IV
CLASSIFICATION RESULTS FOR THE BEARINGS DATASET

Fault diagnosis method	Dataset under different working loads					
	A	B	C	D	E	F
[26]	88.9%	-	-	-	-	-
[27]	-	-	-	92.5%	-	-
[28]	-	-	-	-	95.8%	-
[29]	-	-	-	-	97.91%	-
[30]	-	-	-	-	99.66%	-
[13]	-	-	-	-	99.6%	-
[12]	98.8%	98.8%	99.4%	99.4%	99.8%	96.8%
CNN trained from scratch	98.30%	99.30%	90.66%	99.72%	98.85%	96.47%
Pre-trained model	**100%**	**100%**	**100%**	**99.96%**	**99.95%**	**98.80%**

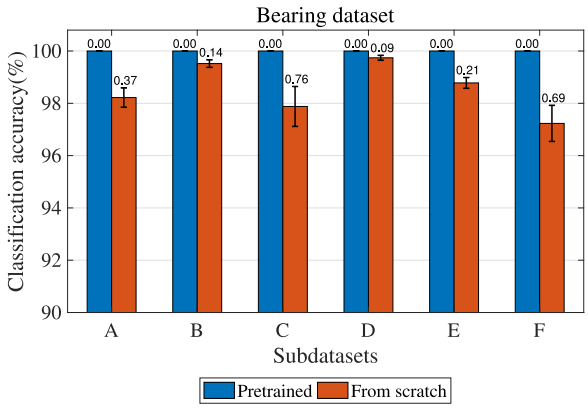

Fig. 6. Classification accuracy of average tenfold cross validation on the bearings dataset.

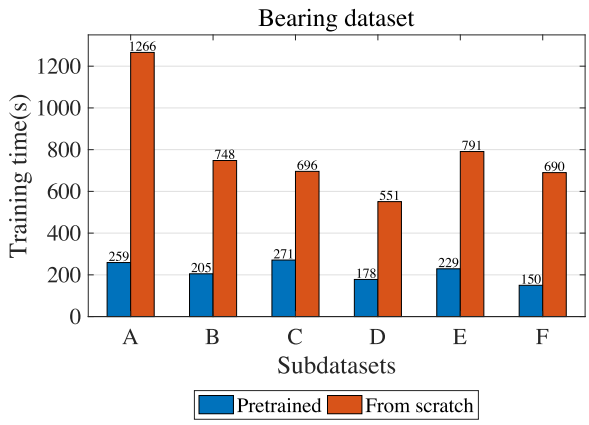

Fig. 7. Training time of CNN trained from scratch and pretrained model on the bearing dataset.

4) **D** Training data and testing data are both from vibration signals under working load of 3 hp.
5) **E** Training data and testing data are both from vibration signals under working loads of 0–3 hp with balanced samples.
6) **F** Training data come from vibration signals under working loads 0–2 hp while testing data are from working load of 3 hp.

Therefore, fault diagnosis for the bearings dataset is considered as a 10-class classification task. For time-frequency imaging, the same transformation as performed on the induction motor dataset is done to these operational bearing vibration signals. For subsets A–D, the training data of each subdataset contain 500 samples for each machine state and the total number of training data is 5000 samples for 10 classes. The testing dataset is set to be the same number as the training dataset. For subset E, each machine state within a certain bearing load contains 100 samples and 10 classes under 4 different bearing loads. The whole set consists of 4000 samples for training and 4000 samples for testing. For subset F, each machine state within a certain bearing load also contains 100 samples, and for the training dataset, it contains 10 classes and 3 different working loads where the total number of training samples are 3000. Testing data are only from working load 3 where the testing set has 3000 samples.

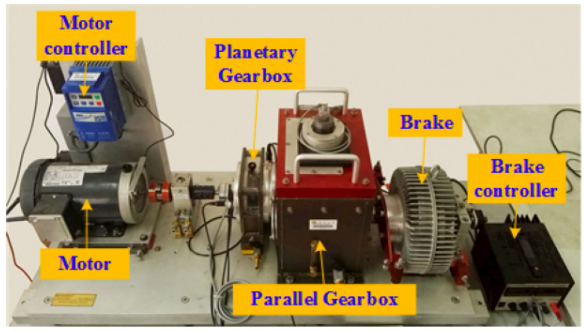

Fig. 8. Experimental setup for gearbox dataset.

Corresponding to the ten different working states, the output layer of the pretrained VGG-16 model is replaced with a new layer with ten neurons with random weight initialization.

Classification accuracy is recorded to compare model performance with other methods, and experimental results are shown in Table IV. We compare our results to the following: bearing fault diagnosis system based on energy-fluctuated multiscale feature learning strategy using designed Deep ConvNet [12], local feature-based gated recurrent unit (LFGRU) networks for machine health monitoring [13], unsupervised feature learning using sparse filtering [30], conventional feature based

TABLE V
CLASSIFICATION RESULTS FOR THE GEARBOX DATASET

Fault diagnosis method		Bearing		Gear		Mixture	
		20-2	30-2	20-0	30-2	20-0	30-2
[13]	SAE-DNN	87.5%	92.1%	92.7%	91.9%	-	-
	GRU	91.2%	92.4%	93.8%	90.5%	-	-
	BiGRU	93.0%	93.6%	93.8%	90.7%	-	-
	LFGRU	93.2%	94.0%	94.8%	95.8%	-	-
CNN trained from scratch		98.90%	98.84%	98.70%	94.14%	98.07%	96.40%
Pre-trained model		**99.94%**	**99.42%**	**99.64%**	**99.02%**	**99.82%**	**99.31%**

TABLE VI
BEARING AND GEARBOX FAULT TYPES DESCRIPTION

Location	Type	Description
Gearbox	Chipped	Crack occurs in the gear feet
	Miss	Missing one of feet in the gear
	Root	Crack occurs in the root of gear feet
	Surface	Wear occurs in the surface of gear
Bearing	Ball	Crack occurs in the ball
	Inner	Crack occurs in the inner ring
	Outer	Crack occurs in the outer ring
	Combination	Crack occurs in the both inner and outer ring

methods [26] based on wavelet features and support vector machine (SVM). [27] is based on nine time-domain statistical features combined with wavelet coefficients energy, [28] uses a self-organizing map based on time-domain and frequency-domain features, and [29] uses ensemble empirical mode decomposition and optimized SVM.

From the classification results shown in the Table IV, the proposed fine-tuned pretrained deep model is able to accurately classify bearing operating states under various working environments, outperforming all other methods in terms of classification accuracy. Compared with conventional approaches based on wavelet features and SVM, our framework is 11.1% more accurate, and compared to [27] and [28], the proposed model achieves a performance gain of 7.4% and 4.1%, respectively. Furthermore, the pretrained model is able to learn representative features from input without manual feature selection. In addition, when compared with machine learning based methods [12], [13], our proposed approach performs better in classification accuracy. The proposed model is able to learn representative features automatically based on different aims. For example, in subdataset F, the architecture is trained using signals from loads 0–2 and is tested using load 3 which is a new working condition to the trained model. Although the trained model is not familiar with signals from load 3, it can still differentiate bearing faults with relatively accurate rates regardless of the working environment. This demonstrates its effectiveness and robustness in the task of bearing fault diagnosis.

In order to compare the performances of proposed method with CNN model trained from scratch, tenfold cross validation is carried out and results are shown in Fig. 6. Besides, time spent that models were trained to achieve 99% accuracy is calculated and shown in Fig. 7. As shown above, together with results in supplementary material, pretrained model is able to achieve accurate prediction after second epoch, while CNN model trained

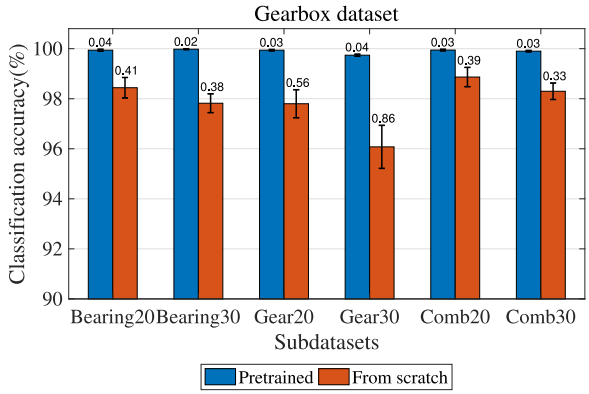

Fig. 9. Classification accuracy of average tenfold cross validation on the gearbox dataset.

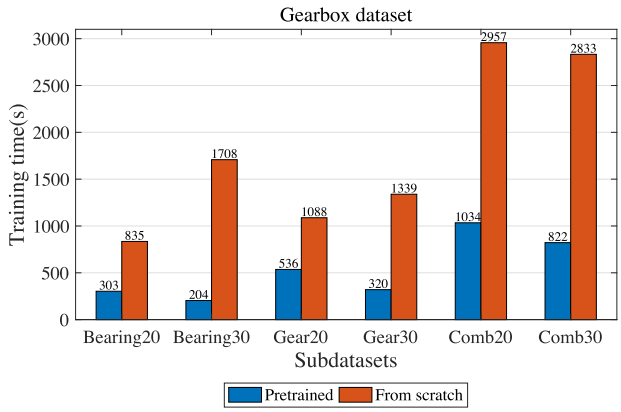

Fig. 10. Training time of CNN trained from scratch and pretrained model on the gearbox dataset.

from scratch needs more than ten training epochs. Compared with training a CNN model from scratch, the proposed method has more accurate classification results and a better rate of convergence, especially in subdataset F where the proposed method shows better generalization ability.

C. Gearbox Dataset

The gearbox dataset was collected from the drivetrain dynamic simulator (DDS) shown in Fig. 8. Two different working conditions are investigated with the rotating speed system load set to be either 20 HZ-0V or 30 HZ-2V. The different types of faults for bearings and gearboxes are shown in Table VI.

The dataset contains five different working conditions for the bearing data and the gearbox data: four failure types and one health state. Therefore, fault diagnosis for DDS is a 5-class classification task. Each fault type consist of 1000 samples for training and the entire gear and bearing training datasets are both 5000 images. Testing datasets are the same size as training datasets. In order to test the proposed method when dealing with mixed faults, gear faults and bearing faults are combined to form a mixture dataset including four kinds of gear failure, four kinds of bearing failure, and one health state. Each working state contains 1000 samples for training and 1000 samples for testing and both the training and testing dataset contain 9000 samples.

Corresponding to five and nine different machine failure types, the output layer of the pretrained VGG-16 model is replaced with a 5-neuron output layer and a 9-neuron output layer separately to predict corresponding labels.

Experimental results are shown in Table V. A comparison is given with the method proposed in [13], where an enhanced gated recurrent unit is adopted in diagnosing gearbox fault and several other methods including stacked autoencoder based deep neural network (SAE-DNN), gated recurrent units network (GRU), and bidirectional gated recurrent units network (BiGRU). From the results, the pretrained model outperforms other machine learning based methods in classification accuracy and achieves approximated 6% performance improvement.

Tenfold cross validation is carried out to compare the performances of pretrained model and CNN trained from scratch and results are shown in Fig. 9. Time spent is calculated and shown in Fig. 10. Among all the subdatasets, pretrained model takes less training time and achieves more accurate predictions.

Detailed results are shown in supplementary material, and the proposed method is more stable during training compared to training from scratch. These results show that the proposed pretrained model is able to achieve accurate predictions for the gearbox dataset.

V. Conclusion

In conclusion, we have developed a deep TL framework for mechanical fault diagnosis and classification, and we have created a repository of several benchmark datasets. Our comparative analysis has shown that the proposed approach achieves state-of-the-art results across each of these datasets. In the future, DL can be expected to continue to play a useful role in fault detection and classification across different mechanical systems, as well as in other areas of control engineering.

References

[1] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, 2015.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., New York, NY, USA: Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[3] G. Hinton *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[4] R. Socher, Y. Bengio, and C. D. Manning, "Deep learning for NLP (without magic)," in *Tutorial Abstracts of ACL 2012*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2012, pp. 5–5.

[5] L. Wu and P. Baldi, "Learning to play GO using recursive neural networks," *Neural Netw.*, vol. 21, no. 9, pp. 1392–1400, 2008.

[6] P. Baldi, P. Sadowski, and D. Whiteson, "Searching for exotic particles in high-energy physics with deep learning," *Nature Commun.*, vol. 5, 2014, Art. no. 4308.

[7] C. Shimmin, P. Sadowski, and e. a. Pierre Baldi, "Decorrelated jet substructure tagging using adversarial neural networks," *Physical Rev. D*, vol. 96, p. 074034, 2017.

[8] D. Fooshee *et al.*, "Deep learning for chemical reaction prediction," *Mol. Syst. Des. Eng.*, vol. 3, pp. 442–452, 2017, doi: 10.1039/c7me00107j.

[9] C. N. Magnan and P. Baldi, "SSpro/ACCpro 5: Almost perfect prediction of protein secondary structure and relative solvent accessibility using profiles, machine learning, and structural similarity," *Bioinf.*, vol. 30, no. 18, pp. 2592–2597, 2014.

[10] P. Baldi, "Deep learning in biomedical data science," *Annu. Rev. Biomed. Data Sci.*, vol. 1, pp. 181–205, 2018.

[11] W. Sun, S. Shao, R. Zhao, R. Yan, X. Zhang, and X. Chen, "A sparse auto-encoder-based deep neural network approach for induction motor faults classification," *Meas.*, vol. 89, pp. 171–178, 2016.

[12] X. Ding and Q. He, "Energy-fluctuated multiscale feature learning with deep convnet for intelligent spindle bearing fault diagnosis," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 8, pp. 1926–1935, Aug. 2017.

[13] R. Zhao, D. Wang, R. Yan, K. Mao, F. Shen, and J. Wang, "Machine health monitoring using local feature-based gated recurrent unit networks," *IEEE Trans. Ind. Electronics*, vol. 65, no. 2, Feb. 2018.

[14] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[15] L. Wen, L. Gao, and X. Li, "A new deep transfer learning based on sparse auto-encoder for fault diagnosis," *IEEE Trans. Syst., Man, Cybern.: Syst.*, 2017, doi: 10.1109/TSMC.2017.2754287.

[16] F. Shen, C. Chen, R. Yan, and R. X. Gao, "Bearing fault diagnosis based on SVD feature extraction and transfer learning classification," in *Proc. Prognostics Syst. Health Manage. Conf.*, 2015, pp. 1–6.

[17] B. Boashash, *Time-frequency Signal Analysis and Processing: A Comprehensive Reference*. Cambridge, Massachusetts, Academic Press, 2015.

[18] R. Yan, R. X. Gao, and X. Chen, "Wavelets for fault diagnosis of rotary machines: A review with applications," *Signal Process.*, vol. 96, pp. 1–15, 2014.

[19] J. Gu *et al.*, "Recent advances in convolutional neural networks," 2015, arXiv:1512.07108.

[20] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," 2012, arXiv:1207.0580.

[21] H. Azizpour, A. Sharif Razavian, J. Sullivan, A. Maki, and S. Carlsson, "From generic to specific deep representations for visual recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, 2015, pp. 36–45.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, arXiv:1409.1556.

[23] X. Yang, R. Yan, and R. X. Gao, "Induction motor fault diagnosis using multiple class feature selection," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf.*, 2015, pp. 256–260.

[24] S.-Y. Shao, W.-J. Sun, R.-Q. Yan, P. Wang, and R. X. Gao, "A deep learning approach for fault diagnosis of induction motors in manufacturing," *Chin. J. Mech. Eng.*, vol. 30, pp 1347–1356, 2017.

[25] F. Chollet *et al.*, *Keras* (2015). [Online]. Available: https://keras.io

[26] W. Du, J. Tao, Y. Li, and C. Liu, "Wavelet leaders multifractal features based fault diagnosis of rotating mechanism," *Mech. Syst. Signal Process.*, vol. 43, no. 1, pp. 57–75, 2014.

[27] X. Jin, M. Zhao, T. W. Chow, and M. Pecht, "Motor bearing fault diagnosis using trace ratio linear discriminant analysis," *IEEE Trans. Ind. Electron.*, vol. 61, no. 5, pp. 2441–2451, May 2014.

[28] W. Li, S. Zhang, and G. He, "Semisupervised distance-preserving self-organizing map for machine-defect detection and classification," *IEEE Trans. Instrum. Meas.*, vol. 62, no. 5, pp. 869–879, May 2013.

[29] X. Zhang, Y. Liang, J. Zhou, and Y. Zang, "A novel bearing fault diagnosis model integrated permutation entropy, ensemble empirical mode decomposition and optimized SVM," *Meas.*, vol. 69, pp. 164–179, 2015.

[30] Y. Lei, F. Jia, J. Lin, S. Xing, and S. X. Ding, "An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data," *IEEE Trans. Ind. Electron.*, vol. 63, no. 5, pp. 3137–3147, May 2016.

[31] Case Western Reserve University Bearing Data Center. [Online]. Available: http://csegroups.case.edu/bearingdatacenter/pages/welcome-case-western-reserve-university-bearing-data-center-website

Siyu Shao (S'14) received the bachelor's degree in automation from Soochow University, Suzhou, China, in 2013. She is currently working toward the Ph.D. degree in instrument science and technology at the School of Instrument Science and Engineering, Southeast University, Nanjing, China.

Her research interests include deep learning models and mechanical fault diagnosis.

Stephen McAleer received the B.S. degree in mathematics and economics from Arizona State University, Tempe AZ, USA, in 2017. He is currently working toward the Ph.D. degree in Statistics at the University of California Irvine, CA, USA, working with Dr. Pierre Baldi.

His research interests include deep learning and reinforcement learning. He recently cocreated an algorithm which is able to solve the Rubik's Cube without any human knowledge. This work has been featured in MIT Technology Review, LA Times, Gizmodo, and Popular Mechanics.

Ruqiang Yan (M'07–SM'11) received the M.S. degree in precision instrument and machinery from the University of Science and Technology of China, Hefei, China, in 2002, and the Ph.D. degree in mechanical engineering from the University of Massachusetts, Amherst, MA, USA, in 2007.

From 2009 to 2018, he was a Professor with the School of Instrument Science and Engineering, Southeast University, China. He joined the School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an, China, in 2018. His research interests include data analytics, machine learning, and energy-efficient sensing and sensor networks for the condition monitoring and health diagnosis of large-scale, complex, dynamical systems.

Dr. Yan is a member of the ASME. He was the recipient of the New Century Excellent Talents in University Award from the Ministry of Education in China, in 2009. He is an associate editor for the IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT.

Pierre Baldi (SM'06–F'11) earned M.S. degrees in mathematics and psychology from the University of Paris, France, in 1980, and the Ph.D. degree in mathematics from the Caltech, CA, USA, in 1986.

He is currently a Distinguished Professor with the Department of Computer Science, Director with the Institute for Genomics and Bioinformatics, and Associate Director with the Center for Machine Learning and Intelligent Systems at the University of California, Irvine, CA, USA. His research interests include understanding intelligence in brains and machines. He has made several contributions to the theory of deep learning, and developed and applied deep learning methods for problems in the natural sciences. He has written 4 books and over 300 peer-reviewed articles.

Dr. Baldi was the recipient of the 1993 Lew Allen Award at JPL, the 2010 E. R. Caianiello Prize for research in machine learning, and a 2014 Google Faculty Research Award. He is an Elected Fellow of the AAAS, AAAI, IEEE, ACM, and ISCB.