Efficient Fault Diagnosis of Rolling Bearings Using Neural Network Architecture Search and Sharing Weights

MINH TUAN PHAM1, JONG-MYON KIM2, AND CHEOL HONG KIM3
1Department of ICT Convergence System Engineering, Chonnam National University, Gwangju 61186, South Korea
2School of IT Convergence, University of Ulsan, Ulsan 44610, South Korea
3School of Computer Science and Engineering, Soongsil University, Seoul 06978, South Korea
Corresponding author: Cheol Hong Kim (cheolhong@ssu.ac.kr)
This work was supported by the National Research Foundation of Korea (NRF) Grant through the Korea Government [Ministry of Science and ICT (MSIT)] under Grant 2021R1A2C1009031.

ABSTRACT Bearing is one of the most vital components of industrial machinery. The failure of bearing causes severe problems in the machinery. Therefore, continuous monitoring for the bearings is essential rather than regular manual checking, with the requirement for accuracy of prediction and efficiency. This paper proposes a novel intelligent bearing fault condition monitoring and diagnosis method focusing on computation efficiency, which is an important aspect of a continuous monitoring and embedded-based diagnosis device. In the proposed method, acoustic emission signals containing bearing health information are converted into 2-D spectrograms by Constant Q-Transform (CQT) before using a convolutional neural network to infer the bearing state. To reduce the latency while maintaining high accuracy, we propose an efficient search space for neural network architecture search, i.e., a channel distribution search, that automatically obtain the best performing network. Moreover, we present a separation between two processes of condition monitoring and fault diagnosis to save overall computing resources, with a policy of sharing weights in the training process and sharing features in the testing process. The experimental results show that the proposed method reduces about 50% inference time compared to previous methods while achieving an accuracy of 99.82% for eight types of single and compound fault diagnosis for variable rotational speeds.

INDEX TERMS Acoustic emission, bearing fault condition monitoring, bearing fault diagnosis, convolutional neural network, neural network architecture search.

I. INTRODUCTION
The demanding workload of manufacturers nowadays gives rise to the requirements of the trouble-free operation of machinery components. Among such components, bearings are critical; bearings prevent friction in direct metal to metal contact between elements in relative motions by replacing them with low friction rolling. A small bearing crack in complex systems could lead to serious incidents, especially in railways, power plants, machine tools, etc. [1], [2]. However, the percentage of bearing faults occurring in industrial is significantly high, accounting for 50% [3]. Therefore, incipient fault detection may bring considerable benefits, which can be achieved by automatic continuous monitoring and diagnosis processes instead of conventional manual continual maintenance. In general, the condition monitoring task is related to detecting abnormal states, and the fault diagnosis task figures out the existing fault types. In an effort to construct automatic fault monitoring and diagnosis processes, the approaches targeted for embedded systems that require computation optimization increasingly show their importance. Those approaches bring considerable advantages of saving energy for continuous monitoring, quick inference, low equipment purchase costs. In addition, they are not required for Supervisory Control and Data Acquisition (SCADA) systems in Small and Medium Enterprises (SME); moreover, they are independent with network speed and bandwidth. The last advantage also benefits the overall manufacturer’s network system that adopts SCADA because only the information about the state of monitored objectives is transferred to the center instead of transferring the massive influx of redundant information. In addition, implementing in the form...
of handheld devices has greater mobility for monitoring in changing locations.

Over the past decade, many studies in Bearing Fault Diagnosis (BFD) focused on analyzing monitored signals that contain information about bearing health states. Acoustic Emission (AE) is one of the most popular signals used for BFD due to its remarkable ability to capture pure and low-energy fault signals in early degradation cases. The intelligence of diagnosis processes derives from the advent of Deep Neural Networks (DNN) and Machine Learning (ML), which are great tools in exploiting hidden features from acquired signals [4]–[9]. In particular, Convolutional Neural Networks (CNNs) have obtained significantly high fault classification accuracy. For example, Zhang et al. proposed a 1-D CNN for BFD using directly raw collected signals in time-domain, with high performance in noisy environments [10]. Pham et al. proposed a transfer learning from AlexNet (2-D CNN) with input images generated from eight different time-frequency analysis methods [11]. Similarly, Pham et al. showed the effectiveness of time-frequency representation; they utilized VGGNet and EfficientNet with a 2-D spectrogram to achieve high accuracy in complicated bearing faults classification [8], [9]. However, a downside of CNN-based methods in the BFD field is the high overhead of computational resources compared with common traditional methods [11]. Few studies are focusing on this disadvantage. For instance, in order to reduce the number of Multiply-Accumulate (MAC) operations, simple CNN architectures are used [12], [13], or techniques reducing input image size are considered [11]. Currently, in order to maintain the prediction accuracy, the computational cost measured by MACs of previous CNN-based methods is approximately equivalent to the cost consumed by LeNet-5. In addition, reducing the number of MACs, other aspects are still not carefully considered regarding the CNN model’s efficiency designed for BFD.

The practical metrics in evaluating the efficiency of a CNN-based system are throughput and latency. While throughput commonly refers to the number of inferences per second, latency measures the time interval between data arrival in the system and result generation. In common, the efficiency of a CNN algorithm is only evaluated by measuring the number of MAC operations in each CNN model inference. However, the number of MAC is not sufficient for evaluating throughput or latency. The CNN model design decides the number of parameters, the sparsity of weights and activations, and the number of MAC operations per inference by defining the network layers’ shapes. The number of parameters affects the storage requirement and memory access cost, whereas the sparsity referring data repetition could be utilized to reduce the data’s footprint, storage requirements, data movement, and MAC operations. The memory access cost significantly affects the latency, especially in memory-bound systems (bottleneck-effect systems). From these observations, we focus on two major factors affecting the efficiency of our proposed CNN design for intelligent bearing condition monitoring (BCM) and BFD: (1) reducing the number of MAC (number of operations per inference); (2) alleviating the effect of dataflow by reducing the number of parameters and increasing activation sparsity.

In this study, Constant-Q transform (CQT) is used for preliminary feature extraction, which utilizes the advantages of time-frequency representation for nonstationary signals acquired in the condition of variable rotational speeds. This paper contributes two significant improvements of efficiency in CNN design and intelligent BCM and BFD scheme:

1) Defining a new space search for neural architecture search (NAS), namely Channel Distribution Search (CDS) for designing efficient CNN, with the support of efficient CNN design techniques to reduce the number of MAC and dataflow while preserving high accuracy of the classification task.

2) Proposing the sharing of weights and features between two separated models of BCM and BFD. The difference in frequency to perform such two tasks helps to improve the overall efficiency significantly.

This paper is organized as follows: Section 2 presents the related works and background of CQT for time-frequency analysis and techniques for efficient CNN design. In Section 3, the proposed process and model design for BCM and BFD are presented. In Section 4, the experimental procedure is introduced, followed by results and analysis. Finally, the conclusion is presented in Section 5.

II. BACKGROUND

The proposed efficient method initially converts the raw AE signals into time-frequency spectrogram images by using CQT. Then, proposed efficient CNNs established by search is adopted to extract the insight features of those images. Finally, the conditions of bearing and fault types are classified by using fully connected layers. This section reviews previous works in this field of intelligent bearing fault diagnosis. Next, the background about techniques adopted in our proposed method is also introduced (CQT and efficient CNN design techniques) with some highlighted advantages of efficiency.

A. RELATED WORK

The intelligent fault diagnosis process commonly derives from the synergy of digital signal processing (DSP) techniques and the advances of deep learning. DSP methods are applied as preliminary rectifying raw signal directly in single-domain (time-domain, frequency-domain) or time-frequency domain. Single domain DSP methods are often appropriate for BFD simple scenarios such as single fault diagnosis and low-noise environments. Particularly, concerning time-domain analysis, the power of monitored signals is often used as a health indicator, which is proportional to the deterioration of bearings [14]. Moreover, statistical parameters (mean, variance, skewness, kurtosis, etc.) calculated from observed signals are utilized as random variables to infer fault features. On the other hand,
regarding frequency-domain analysis, fault features can be based on the observation of the fault characteristic frequency (FCF) and its harmonics, which reflect the nature of bearings [15], [16]. Although those single-domain methods do not require too many computational resources, they have many downsides caused by the lack of ability to represent time-dependence and frequency-dependence simultaneously. For example, Fourier transform is not appropriate for signals containing non-periodic components or transient signals; however, transient states could be good sources of information about bearing health. Moreover, FCFs depend on rotational speeds; thus, it is not feasible to use spectral analysis in such cases where monitored signals are nonstationary. In contrast, the time-frequency domain causes a significant power of synergy, which can analyze the time-domain and frequency all-round. Common methods in this domain include Short-Time Fourier Transform (STFT) [17] and Wavelet Transforms (WT) [18], [19]. STFT is a simple method; however, it cannot illustrate the rapid change of signal due to its fixed analysis window. WT supports flexible resolution change according to frequency value, which is widely used for addressing the shortcomings of STFT. Yuan et al. used Continuous Wavelet Transform (CWT) to refine the analysis of signal at multiscale [18]. Yan et al. analyzed overall the implementation of various types of WT, which are CWT, Discrete Wavelet Transform (DWT), Second-Generation Wavelet Transform (SGWT), and Wavelet Packet Transform (WPT) [20].

Besides DSP methods, DNN and ML increasingly play a vital role in intelligent fault diagnosis. Classic methods, namely Artificial Neural Network, Bayesian, and especially Support Vector Machine, support classification tasks effectively. However, their shallow structures are likely to require additional preliminary feature extractions. The advent of DNN produces powerful tools for extracting in-depth features, which could handle high-dimensional complicated issues. Therefore, there have been many applications for this field, such as Long Short-Term Memory [21], Deep Auto-encoder [22], Deep Belief Network [23], etc. In particular, CNN-based methods have achieved high accuracy in fault diagnosis. CNN can be adopted not only as a fault classifier but also as feature extraction tools. CNNs are initially applied in this field for classification tasks fed by statistical parameters acting as input features calculated from the signals in the time-domain, such as the study of Bhadane and Ramachandran [24]. After that, Xie and Zhang found that the combination of CNN with traditional methods yield better performance [25]. They combined CNN as a classifier for features extracted by discrete WT. 1-D CNNs are known as efficient methods to directly utilize raw signals for mechanical fault detection [26], [27]. Moreover, CNN has also received attention owing to its sophisticated image processing ability, which can be adopted in BFD with the 2-D signal representations. Yuan et al. combined CNN and SVM to build a network framework for BFD [18]; Chen et al. proposed a combination between Cyclic Spectral Coherence and CNN to enhance fault recognition ability [28]. In an effort to boost the generalization ability and avoid overfitting in cases of lacking labeled samples, Han et al. proposed an adversarial learning framework [29]. In addition, concerning transfer learning with CNN, He et al. proposed a method utilizing multi-channel monitored signals to establish good source models before transferring to target models [30]. With the support of a decision fusion strategy, the method can achieve good performance at the target task only with a few target training samples. Similarly, Shao et al. proposed a CNN-based transfer learning with bearing faults features represented by thermal images [31].

Besides the undeniable CNN advantages of improving accuracy in prediction, a considerable downside of CNN is its huge resource consumption. Therefore, techniques in CNN design to alleviate this problem are increasingly important.

### B. BACKGROUND FOR THE TECHNIQUES USED IN THE PROPOSED METHOD

1) **CQT FOR AE BEARING FAULT SIGNALS ANALYSIS IN THE TIME-FREQUENCY DOMAIN**

CQT time-frequency analysis method is used for feature representation, which converts the original 1-D signals in the time-domain into 2-D time-frequency spectrogram images. In opposite to single domain methods (time-domain, frequency-domain), 2-D representation in the time-frequency domain has been claimed to be compatible with nonstationary signals [11], [9], [19], [32]. In BCM and BFD, most of the frequently used time-frequency methods are quadratical time-frequency distributions (QTFDs) and time-scale analysis (WT: Wavelet Transforms). While QTFDs are based on short-time Fourier transform (STFT) and are suitable for large bandwidth duration (BT) signals; the WT is related to Gabor transform, which is most effective when using low BT and transient signals.

CQT [33], [34] is a variation of WT. It is advantageous to represent a signal in low frequencies and solve the problem of mapping frequency on a logarithmic scale because the buffer size (the length of the sample array to perform the transform) changes with frequency (raising the buffer size for lower frequencies). That solution also mitigates the computational overhead because of reducing the buffer size for high frequencies. Low-frequency components in acquired AE signals contain meaningful information about bearing health rather than high-frequency signal components due to high-frequency noise. CQT depends on the following primary factors: (1) window functions $g_k$, which are real-valued, even functions. In the frequency domain, the Fourier transform of $g_k$ is defined in the interval $[-F_s/2, F_s/2]$; (2) the sampling rate $\omega_s$; (3) the number of bins per octave, b; (4) the minimum and maximum frequencies, $\omega_{\text{min}}$ and $\omega_{\text{max}}$, respectively.

$\omega_{\text{min}}$ and b are initially chosen, followed by a series of spaced frequency:

$$\omega_k = \omega_{\text{min}} \times 2^{k/b}$$  \hspace{1cm} (1)
where $k$ is iteration index satisfying the condition that $\omega_{\text{max}}$ strictly less than a half of Nyquist frequency.

Besides, the bandwidth of $k^{th}$ frequency is configured to the following:

$$\Omega_k = \omega_k + 1 - \omega_k - 1$$  \hspace{1cm} (2)

It is worth noting that the ratio of $k^{th}$ frequency at the center to the window bandwidth is constant. It is related to the name of this method:

$$Q = \frac{\omega_k}{\Delta k} = \frac{1}{2^{1/b} - 2^{-1/b}}$$  \hspace{1cm} (3)

2) EFFICIENT CNN DESIGN TECHNIQUES

a: DEPTH-WISE CONVOLUTION

Depth-wise convolution (DW-Conv) [35] (Fig. 1) is a variant of conventional convolution with single-channel kernels being applied for each input channel. The brief process of this operation is illustrated by the following: (1) separate the channels of input and kernel, (2) calculate the convolution for each input channel with corresponding filter, (3) stacking the calculated outputs together. The simplicity of this variant is compatible with neural networks related to the independence of channels; moreover, this variant helps to reduce computational complexity.

b: BOTTLENECK-LIKE LAYER

Bottleneck-like layer (Fig. 2) is inspired from original bottleneck layer introduced in Inception-v2, Inception-v3 [36] and ResNet [37]. It maintains the same structure of the bottleneck layer with the presence of DW-Conv. Initially, it uses a $1 \times 1$ convolution to reduce the number of features at each channel; then, it uses a $3 \times 3$ convolutional layer; finally, a $1 \times 1$ convolutional layer is used. The Bottleneck-like block helps to retain computation low while exploiting rich insight features. Such a layer is adopted by ShuffleNet [38], [39].

c: ReLU6 ACTIVATION FUNCTION

Activation helps the network learn complex features in the data by keeping the output value from the neuron in a predefined restriction. The ability to rectify non-linearly in a network is an essential characteristic of an activation function.

Among existed activation functions, the simplicity of ReLU (Rectified Linear Units) compared to that of Sigmoid or Tanh produces higher convergence speed; this advantage of ReLU is attributed to eliminating the saturation at pole values [40] and lower computational cost caused by a non-exponential function.

ReLU6 is one of the ReLU variants, which sets all negative values to zero, and higher-than-six values to six, leading to the frequent presence of two values of zeros and sixes, i.e., sparse activation (Fig. 3). In addition, keeping the ReLU bounded by 6 restricts the maximum number of bits to 3 bits. This restriction encourages the network to accumulate sparse features earlier due to the beneficial effect of limiting replicated bias-shifted Bernoulli instead of the infinite amount of values [41]. This is also related to the principle of using fewer bits, which has the benefits of reducing data movement and reducing storage cost. The following formula is used to calculate ReLU6:

$$y = \min(\max(0, \text{features}), 6)$$  \hspace{1cm} (4)

d: EFFICIENT NEURAL NETWORK ARCHITECTURE SEARCH

When there are several hyperparameters needed to be fine-tuned (the number of layers, the number of connections among layers, several layer types, and shapes), manual works can be based on tedious and inefficient trials. Therefore, neural network architecture search (NAS) helps in establishing a neural network. In general, NAS contains three main aspects: (1) search space (the set of all network samples); (2) optimization algorithms (policy to find the best network quickly); (3) performance evaluation (metrics to evaluate network samples).

NAS is commonly performed its task iteratively. For each iteration, the optimization tries to sample some sub-networks from the search space. Then, based on the evaluation results, the algorithm opts for the following network sample. This process is repeated until a termination criterion is satisfied when the best searched neural architecture is obtained (Fig. 4).

Among NAS algorithms, Efficient Neural Architecture Search [42] (ENAS) is a saving-cost approach for designing...
a model automatically. It is an advanced one-shot NAS, which initially defines a mother network containing every candidate as search space. Then, gradually sub-network or child network is trained and evaluated. ENAS utilizes the sharing parameter between sub-networks to accelerate the searching process. NAS is evolved by two common factors – controller and child model. The controller is responsible for controlling and directing the construction of the child network architecture according to a set of instructions based on a defined search tactic. The controller in ENAS is trained with a policy selecting a sub-network that achieved the acceptable reward on the validation sub-set by using a gradient (a policy based on reinforcement learning algorithm). In contrast, the selected sub-network is trained to achieve minimal cross-entropy loss.

III. PROPOSED BCM AND BFD METHODOLOGY

In this study, we established BCM and BFD models. This chapter mainly includes the overall description of the proposed efficient process for the tasks of BCM and BFD and the model construction.

A. OVERALL PROCESS FOR BCM AND BFD

The proposed process contains two major phases – the offline and online phases (Fig. 5). The offline phase establishes neural network models, whereas the online phase utilizes the established models for BCM and BFD. All of the acquired signal segments from sensors are converted to spectrograms (2-dimensional images) using CQT both in the offline and online phases as input data for training and inference, respectively.

The first two steps (Channel Distribution Search and fine-tune the searched architecture) in the offline phase are related to building a model for the bearing fault type classification task, namely the BFD model. While the first step serves to seek the most appropriate architecture (detailed in Channel Distribution Search), the second step utilizes trained weights from the previous search process and retrain the model with few epochs to achieve the highest accuracy. In contrast, the last two steps aim to establish a sub-model, namely the BCM model, which is responsible for classifying normal and abnormal bearing states. This model is built by transferring the first trained convolutional layer from the BFD model and training its classifier (fully connected layer (FC)). That transfer learning helps accelerate the training BCM model and save required computations in the online phase for the reasons below.

In the online phase, the inference process begins with abnormal-state detection. Each spectrogram converted from fixed-length signal acts as an input sample of the BCM model. If an upcoming input sample is diagnosed with the normal state, the inference process will end. Otherwise, the extracted features by shared weights of the BCM model will be utilized as input features for the successive layers of the BFD model where the inference process specifies the type of bearing fault. The large discrepancy between the amount of time the bearings are in normal and abnormal states leads to the difference between the number of normal-state and abnormal-state samples. The most popular type of sample (normal-state samples) consumes the computational resources of BCM inference. In contrast, only abnormal-state samples which rarely occur are investigated further by the BFD model to specify fault types.

B. DESIGN OF BFD AND BCM MODEL

While the BFD model is more complicated to diagnose fault types, the BCM model is simple to detect abnormal states. The BCM model is established mostly by inheriting a part of BFD model’s weights. Therefore, this section mainly focuses on describing the proposed techniques for establishing the model of BFD.

1) ARCHITECTURE BONE DESIGN

We propose a CNN architecture bone (Fig. 6) which is a combination of two models for BCM and BFD tasks. The BFD model consists of Conv header, six consecutive predefined basic blocks, Conv footer, and Classifier 1; the BCM model consists of Conv header and Classifier 2. The Conv header is mutual between the two models. Therefore, their weights and features extracted by themselves can be shared. Table 1 presents the configuration for layers of architecture bone. From the table, it is worth noting that six layers acting...
as basic blocks have structures either type of basic block 1 or type of basic block 2 (detailed in the following section); Conv footer is an additional \(1 \times 1\) convolution layer, which is added right before a globally averaged pooling (AvgPool) to mix up features.

2) BASIC BLOCKS DESIGN

Two kinds of basic blocks in Fig. 7 are designed by referring to practical guidelines for the efficiency of Shuffle-Netv2 [39]. CNN blocks of the architecture bone except for Conv header and Conv footer are basic blocks containing a Channel Distribution (CD) component to determine each branch’s number of channels inside. In each basic block, the motivation for splitting input channels into two branches comes from the feature reusing from DenseNet [43] and CondenseNet [44]; in this method, a part of input channels going directly through the basic block to provide identifying information for the next one. One branch aims to maintain that identity while the other branch consists of three convolutions with the same input and output channels to make sure efficiency. The active function used for CNN layers is ReLU6 which restricts the number of bits consumed by parameters and increases activation sparsity. Bottleneck-like using DW-Conv is also adopted for one branch, enriching useful combinations of features with low computational overhead. We proposed two types of basic blocks, with the type of basic block 2 being the primary basic block type adopting the idea of residual learning (using skip connection) from ResNet [37] to prevent the infection of gradient disappearance when scaling up by remaining an identity. Input channels coming to the type of block 2 are split and distributed throughout two branches (branch 1 and branch of skip connection).

After convolution using Bottleneck-like in branch 1, two branches are concatenated to result in the same number of output channels. In contrast, block 1 is designed similarly but for the circumstances of changing the number of channels. The number of channels will be doubled after going through this type of block. In conventional CNN architecture design,
the later CNN layers tend to an increasingly larger number of channels to learn more complex features. That unavoidable change in the number of channels during go through CNN layers leads to the surge of the number of MACs. Nevertheless, based on an observation that in each basic block, the difference in the number of branches’ channels gives rise to the difference in performance and efficiency, we propose channel distribution search to determine the split ratio of the channel distribution component.

3) CHANNEL DISTRIBUTION SEARCH
Channel Distribution components determine the number of output channels for each branch. With the given number of output channel n, we defined three options for opting for the number of channels distributed for each branch of predefined basic blocks. Each layer-choice has three options (option 1, option 2, and option 3) of the split ratio, as shown in Table 2. According to the defined architecture bone, the search space contains 729 possible sub-networks (six basic blocks acting as layer-choices, three options for each layer-choice). The complexity of search space will be exponentially higher as the number of basic blocks increases. For instance, when the number of basic blocks is increased from six to seven, the number of possible sub-networks increases from 729 to 2187. Therefore, the importance of NAS is undeniable and depends on the property of the dataset (the more complicated a dataset is, the more blocks are required). For the dataset used in this study, we use six blocks to achieve high accuracy while maintaining low computational resource requirements.

The search process is performed by using the ENAS algorithm. From the defined single directed acyclic graph (DAG), ENAS obtains the best sub-graph with efficient computational consumption by utilizing shared parameters among child models. Fig. 8 depicts the search process to find the best combination of layer-choice options from the DAG. With the dataset used in this study, eventually, the final search configuration for six basic blocks is option 2 – option 3 – option 2 – option 3 – option 3 – option 3 (the chosen option in each layer-choice is highlighted) as shown in Fig. 8.

### IV. EXPERIMENTS
To verify the efficiency and accuracy of the proposed models, we use the AE signal dataset used in the literature [8], [11], [45]–[47]. We initially evaluate the proposed process’s efficiency by metrics of the number of MAC, the number of parameters, and latency in various cases. Secondly, we evaluate the proposed channel distribution search efficiency by comparing original models without the searching process and searched models referred to those metrics. Finally, we illustrate the ability to maintain the high accuracy of the proposed method by providing a confusion matrix for classification tasks and a feature visualization via t-SNE.

#### A. EXPERIMENTAL METHODS
For the experimental test tube, the moving power of the system comes from a 3-phase induction motor that can change rotational speed by adjusting a speed controller. A flexible coupling connecting the motor shaft and driven-end shaft is used to prevent shock transmission. A gearbox with a damping ratio of 1.52:1 is used to connect the driven- and non-driven-end shafts where experimental cylindrical bearings are set up (FAG NJ206-E-TVP2). The non-driven-end provides motion to an adjustable blade (acting as a variable load of the system) using components of belt and pulley (Fig. 9 and Fig. 10).

Concerning the data acquisition system, a wideband AE sensor (PAC WSA) is mounted on the bearing used for the non-driven-end shaft. The signals are collected from the AE sensor by a PCI-2 based board at a sampling rate of 250 kHz. The frequency range of the AE sensor is from 100 to 900 kHz; the peak sensitivity is −62 dB; the directionality is ±1.5 dB,
Various bearing faults are created by different sizes of diamond cutting at different locations on bearing (Fig. 11). To illustrate the ability of early damage detection, we use the signal acquired in case of the smallest crack size, which is most challenging to detect and classify in common. The duration of each signal sample is chosen for ensuring both containing enough health information and minimizing the acquisition time. For the determined range of rotational speeds, the empirical formulas are used to calculate fault characteristics frequencies (FCFs), which depend on the bearing’s physical specification and are linearly proportional to shaft rotational speeds [16]. The lowest FCF corresponding to the lowest shaft speed in the range is considered to estimate the duration of each sample to ensure covering the range of FCFs. With the rotational speed $f_r$, the bearing specification and its FCFs are shown in Table 3. Based on that, after repeated trials, we determine the duration of 0.05s for each sample. There are 100 samples for each class used for the training process where the number of training samples is 80 (at rotational speeds of 300, 400, 500) and the remaining samples for validation (at rotational speeds of 250, 350, 450). In the testing, 687 samples in each class are used (at rotational speeds of 250, 350, 450). Table 4 summarizes the experimental dataset. In Fig. 12, the time-frequency images of 8 bearing health conditions (1 normal state and 7 abnormal states) are shown after using CQT. After the conversion into spectrograms, all non-informative borders are removed before scaling to an appropriate image size by bilinear interpolation. The choice of $32 \times 32$ input image size and bilinear interpolation method for resizing are referred to the results in [11], which ensures saving computational resources while keeping the features of bearing faults.

### Table 3. Bearing specification and FCFs.

| Pitch diameter | 46.5 (mm) |
| Rolling element diameter | 9.0 (mm) |
| Contact angle | $0^\circ$ |
| Number of rolling elements | 13 |
| Outer defect frequency | 5.241$f_r$ |
| Roller defect frequency | 2.487$\times 2f_r$ |
| Inner defect frequency | 7.758$\times f_r$ |

### Table 4. Dataset for bearing fault diagnosis.

| Single and compound bearing defects Dataset | Rotational speed (RPM) | Crack size |
|--------------------------------------------|------------------------|------------|
| | | Length (mm) | Width (mm) | Depth (mm) |
| | | | | |
| Training sub-set | 300, 400, 500 | | | |
| Validation sub-set; Testing sub-set | 250, 350, 450 | 3 | 0.60 | 0.30 |

### B. EXPERIMENTAL RESULTS

#### 1) EFFICIENCY OF THE DEFINED NAS AND PROPOSED MODELS

In the proposed NAS scheme, the time for the network establishment is calculated by adding the time spent for CDS (15 minutes), finetuning BFD model (4 minutes), and training classifier-2 of BCM model (2 minutes) (using one GPU GeForce RTX 2080 Ti during model establish process). Trying randomly and individually feasible networks is time-consuming, whereas the NAS scheme reduces the wasted time significantly.

For more convenience in evaluating, we consider BFD and BCM models separately, as shown the Table 5. Note that with the configuration achieved from the search result, searched BFD model can reduce over 20000 MAC operations and 3000 parameters compared to the non-searched model (block’s branches having the same number of channels), which...
FIGURE 12. Spectrograms created by constant-Q transform: (a) outer raceway (BCO); (b) inner raceway (BCI); (c) roller (BCR); (d) inner and outer raceways (BCIO); (e) outer raceway and roller (BCOR); (f) inner raceway and roller (BCIR); (g) inner raceway, outer raceway, and roller (BCIOR); (h) normal state.

TABLE 5. Comparing performance among the proposed method and previous methods.

|                             | BCM model | Searched BFD model | Non-searched BFD model | MobileNet-v2 at scale of 0.01 [11] | LeNet-S-based [50] | LeNet-S-based [51] |
|-----------------------------|-----------|--------------------|------------------------|------------------------------------|-------------------|-------------------|
| Number of MAC operations    | 53874     | 73828              | 94008                  | 589174                             | 668272            | 668272            |
| Number of Parameters        | 490       | 4500               | 7548                   | 34788                              | 72376             | 72376             |
| Average Accuracy            | 100%      | 99.82%             | 98.95%                 | 98.74%                             | 94.20%            |                   |

contributes to accelerating the diagnosis process. In contrast, the BCM model inherits the first CNN layer, which suffers a large amount of MAC, accounting for 53488 shared weights. The reason is that there is a direct transformation between two layers having a considerable difference in the number of channels (3 channels for input image, 8 channels for the first CNN layer). However, the quantity of BCM model parameters is significantly small and negligible, which benefits the dataflow and required memory of the process.

In comparison with previous efficient CNNs used in the field of bearing fault diagnosis, there is a slump in both the number of MAC and the number of parameters. The numbers of MAC of the proposed BFD and BCM models are one-eighth and one-eleventh, respectively, compared to MobileNet-v2 at the scale of 0.01. Similarly, the number of MAC consumed by the BFD model is nearly one-eighth of MobileNet-v2 at the scale of 0.01; in addition, the number of parameters of BCM model is negligible compared to the others. Therefore, the proposed models are remarkably efficient.

The proposed CNN architecture can be scalable thanks to the basic-block-based establishment. The number of blocks has impact on both the performance and efficiency of the diagnosis process. Particularly, for the dataset used in this study, adding one more basic block in the architecture bone (from 6 basic blocks to 7 basic blocks) likely to improve the accuracy in the classification task (100%) but consumes more computing resources. Therefore, in an effort to maintain the efficiency, we adopt the architecture with 6 basic blocks. The architecture with 6 basic blocks decreases the accuracy negligibly (99.82%) while reducing the model inference time and establishment time significantly compared to the architecture with 7 basic blocks. The comparison between these two architectures is provided in Table 6.

To provide more solid evidence for the claim about the efficiency of proposed methods, we measure the inference latency of candidates deployed on ARM-cortex-A-based embedded boards. There are various common embedded boards used, including the Raspberry Pi 3B (ARM Cortex A-53), Raspberry Pi 4 (ARM Cortex A-72), and Jetson Nano (ARM Cortex A-57) (Table 7). In our proposed method, when the upcoming signal is predicted as the normal state (the most popular case), it takes significantly little time by the inference of the BCM model. Otherwise, the BFD model inherits extracted features from the BCM model for its inference, taking up practically one-fourth of the figure for MobileNet-v2 at a scale of 0.01. The number of normal-state samples is considerably larger than the number of abnormal-state samples. Therefore, our proposed process can save required resources and reduce the latency considerably on average.
2) ACCURACY OF THE PROPOSED METHOD
Concerning the accuracy and the performance of the proposed method in abnormal state detection and bearing fault classification, we evaluate the inference results with testing sub-set by using separate models of BCM and BFD. Those experiments are repeated ten times to provide results on average. The overall prediction results of the proposed models are illustrated by the confusion matrix in Fig. 13. Specifically, we use sensitivity as the primary metric to evaluate the accuracy of a diagnosis method. Moreover, we provide a feature visualization that visually represents classification results obtained after important convolution layers of the BFD model. The feature visualization is performed by using t-SNE (T-distributed stochastic neighbor embedding) [49]. It reduces from a high dimensional dataset to a low dimensional graph while retaining much of the original information. This technique tries to find clusters in data that help us to understand more about the experimental data in classification problems.

According to the results in Table 5, it is clear that the task of BCM experiences an absolute accuracy due to the simplicity. The BFD model achieves an accuracy of 99.82%, which outperforms previous competitors in the BFD field. Fig. 14 shows that the t-SNE visualization is initially randomly in chaos before they are gradually clustered throughout the CNN layers. Eventually, the separation of clusters proves the excellent performance of the proposed method.

3) STABILITY OF THE PROPOSED METHOD
The stability of a BFD model should be considered due to real industrial working environments with the presence of noise. Therefore, we evaluated the consistency ability of our proposed BFD model by creating the monitoring signals with white Gaussian noise at different levels. The levels of noise added to monitoring signals are measured by Signal-to-Noise Ratio (SNR), which is defined as the ratio of the power of a signal to the power of background noise.

\[
SNR_{dB} = 20 \log_{10} \left( \frac{P_{signal}}{P_{noise}} \right) \tag{5}
\]
According to the results in Table 8, we can observe that the average accuracy experiences a marginal decrease when the value of SNR is reduced. When the SNR value is 6 (dB) (i.e., the power of signal (desired information) is approximately twice as much as the power of noise (undesired signal)), the average accuracy decreases to 97.19%. When the SNR value is 0 (i.e., the power of noise is equal to the power of signal), several classes (BCO, BCR, BCIO, BCOR) show decreased prediction accuracy, resulting in 89.47% average accuracy. Note that the diagnosis accuracy of the proposed model could be reliable in the working conditions with variable rotational speeds and low levels of noise.

### V. CONCLUSION

In this paper, we proposed an efficient bearing fault monitoring and diagnosis methodology for variable rotational speeds, which comprises a proposed Channel Distribution Search for the model establishment; it separates condition monitoring and fault diagnosis while utilizing sharing weights and features. We initially convert observed signals in the time-frequency domain where most informative features are represented even in complicated conditions (varied rotational speeds, compound faults, early damage, and transient state). The generated 2-D images are initially used by the condition monitoring model to detect whether an abnormal state occurs. Later, these images were analyzed more deeply by the fault diagnosis model. Experimental results indicated that the proposed bearing fault monitoring and diagnosis methods could significantly reduce the number of MAC operations, the number of parameters, and overall inference latency. Furthermore, our methods achieved a classification accuracy of 99.82% for compound fault diagnosis under variable rotational speeds from 250 RPM to 500 RPM.

**TABLE 8. Prediction results of BCM model with noise dataset.**

| SNR (dB) | Prediction Accuracy (%) | Average Accuracy (%) |
|----------|-------------------------|----------------------|
| No Noise | 100.0 | 98.55 | 100.0 | 99.97 | 100.0 | 98.82 |
| 6        | 100.0 | 92.86 | 100.0 | 98.82 | 100.0 | 97.19 |
| 2        | 100.0 | 89.01 | 94.74 | 76.92 | 100.0 | 95.47 |
| 0        | 100.0 | 84.33 | 72.73 | 84.62 | 100.0 | 75.11 |

**REFERENCES**

[1] D. Abboud, M. Elbadawi, W. A. Smith, and R. B. Randall, “Advanced bearing diagnostics: A comparative study of two powerful approaches,” Mech. Syst. Signal Process., vol. 114, pp. 604–627, Jan. 2019, doi: 10.1016/j.ymssp.2018.05.011.

[2] X. Zhang, Z. Liu, Q. Miao, and L. Wang, “Bearing fault diagnosis using a whale optimization algorithm-optimized orthogonal matching pursuit with a combined time–frequency atom dictionary,” Mech. Syst. Signal Process., vol. 107, pp. 29–42, Jul. 2018, doi: 10.1016/j.ymssp.2018.01.027.

[3] S. A. Khan and J.-M. Kim, “Automated bearing fault diagnosis using 2D analysis of vibration acceleration signals under variable speed conditions,” Shock Vibrat., vol. 2016, pp. 1–11, Nov. 2016, doi: 10.1155/2016/8729572.

[4] Z. Meng, X. Guo, Z. Pan, D. Sun, and S. Liu, “Data segmentation and augmentation methods based on raw data using deep neural networks approach for rotating machinery fault diagnosis,” IEEE Access, vol. 7, pp. 79510–79522, 2019, doi: 10.1109/ACCESS.2019.2923417.

[5] G. Jiang, H. He, J. Yan, and P. Xie, “Multiscale convolutional neural networks for fault diagnosis of wind turbine gearbox,” IEEE Trans. Ind. Electron., vol. 66, no. 4, pp. 3196–3207, Apr. 2019, doi: 10.1109/TIE.2018.2848405.

[6] M. Sun, H. Wang, P. Liu, S. Huang, and P. Fan, “A sparse stacked denoising autoencoder with optimized transfer learning applied to the fault diagnosis of rolling bearings,” Measurement, vol. 146, pp. 305–314, Nov. 2019, doi: 10.1016/j.measurement.2019.06.029.

[7] H. Wang, S. Li, L. Song, and L. Cui, “A novel convolutional neural network based fault recognition method via image fusion of multi-vibration-signals,” Comput. Ind., vol. 105, pp. 182–190, Feb. 2019, doi: 10.1016/j.compind.2018.12.013.

[8] M. T. Pham, J.-M. Kim, and C. H. Kim, “Intelligent fault diagnosis method using acoustic emission signals for bearings under complex working conditions,” Appl. Sci., vol. 10, no. 20, p. 7068, Oct. 2020, doi: 10.3390/app10207068.

[9] M. T. Pham, J.-M. Kim, and C. H. Kim, “Accurate bearing fault diagnosis under variable shaft speed using convolutional neural networks and vibration spectrogram,” Appl. Sci., vol. 10, no. 18, p. 6385, Sep. 2020, doi: 10.3390/app10186385.

[10] W. Zhang, G. Peng, C. Li, Y. Chen, and Z. Zhang, “A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals,” Sensors, vol. 17, no. 2, p. 425, 2017, doi: 10.3390/s17020425.

[11] M. T. Pham, J.-M. Kim, and C. H. Kim, “Deep learning-based bearing fault diagnosis method for embedded systems,” Sensors, vol. 20, no. 23, p. 6886, Dec. 2020, doi: 10.3390/s20236886.

[12] Y. Zhang, X. Xing, R. Bai, D. Sun, and Z. Meng, “An enhanced convolutional neural network for bearing fault diagnosis based on time–frequency images,” Measurement, vol. 157, Jun. 2020, Art. no. 107667, doi: 10.1016/j.measurement.2020.107667.

[13] V. Tra, J. Kim, and J.-M. Kim, “Fault diagnosis of bearings with variable rotational speeds using convolutional neural networks,” in Advances in Computer Communication and Computational Sciences (Advances in Intelligent Systems and Computing), vol. 759, 2019, pp. 71–81, doi: 10.1007/978-981-13-0341-8_7.

[14] M. Kang, J. Kim, L. M. Wills, and J.-M. Kim, “Time-varying and multisresolution envelope analysis and discriminative feature analysis for bearing fault diagnosis,” IEEE Trans. Ind. Electron., vol. 62, no. 12, pp. 7749–7761, Dec. 2015, doi: 10.1109/TIE.2015.2460242.

[15] F. Jia, Y. Lei, H. Shan, and J. Lin, “Early fault diagnosis of bearings using an improved spectral kurtosis by maximum correlated kurtosis deconvolution,” Sensors, vol. 15, no. 11, pp. 29363–29377, Nov. 2015, doi: 10.3390/s151129363.

[16] R. B. Randall and J. Antoni, “Rolling element bearing diagnostics—A tutorial,” Mech. Syst. Signal Process., vol. 25, no. 2, pp. 485–520, 2011, doi: 10.1016/j.ymssp.2010.07.017.

[17] D. Zhong, W. Guo, and D. He, “An intelligent fault diagnosis method based on STFT and convolutional neural network for bearings under variable working conditions,” in Proc. Prognostics Syst. Health Manage. Conf. (PHM-Qingdao), Oct. 2019, pp. 1–6, doi: 10.1109/PHM-Qingdao4634.2019.8943026.

[18] L. Yuan, D. Lian, X. Kang, Y. Chen, and K. Zhai, “Rolling bearing fault diagnosis based on convolutional neural network and support vector machine,” IEEE Access, vol. 8, pp. 137395–137406, 2020, doi: 10.1109/ACCESS.2020.3012053.

[19] D. Verstraete, A. Ferranda, E. L. Droguett, V. Meruane, and M. Marodar, “Deep learning enabled fault diagnosis using time-frequency image analysis of rolling element bearings,” Shock Vibrat., vol. 2017, pp. 1–17, Oct. 2017, doi: 10.1155/2017/5067651.

[20] R. Yan, X. O. Gao, and X. Chen, “Wavelets for fault diagnosis of rotary machines: A review with applications,” Signal Process., vol. 96, pp. 1–15, Mar. 2014, doi: 10.1016/j.sigpro.2013.04.015.

[21] M. Qiao, S. Yan, X. Tang, and C. Xu, “Deep convolutional and LSTM recurrent neural networks for rolling bearing fault diagnosis under strong noises and variable loads,” IEEE Access, vol. 8, pp. 66257–66269, 2020, doi: 10.1109/ACCESS.2020.2985617.

[22] S. Haidong, J. Hongkai, L. Xingju, and W. Shuaipeng, “Intelligent fault diagnosis of rolling bearing using deep wavelet auto-encoder with extreme learning machine,” Knowl.-Based Syst., vol. 140, pp. 1–14, Jan. 2018, doi: 10.1016/j.knosys.2017.10.024.
