IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 23, 2018, accepted September 24, 2018, date of publication October 1, 2018, date of current version October 29, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2872931

Anomalous Sound Detection Using Deep Audio Representation and a BLSTM Network for Audio Surveillance of Roads

YANXIONG LI, (Member, IEEE), XIANKU LI, YUHAN ZHANG, MINGLE LIU, AND WUCHENG WANG

School of Electronic and Information Engineering, South China University of Technology, Guangzhou 510640, China

Corresponding author: Yanxiong Li (eeyxli@scut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61771200 and Grant 61101160 and in part by the Open Project Program of the National Laboratory of Pattern Recognition under Grant 201800004.

ABSTRACT Surveillance systems based on image analysis can automatically detect road accidents to ensure a quick intervention by rescue teams. However, in some situations, the visual information is insufficiently reliable, whereas the use of a sound detector can greatly improve the overall reliability of the surveillance system. In this paper, we focus on detecting two classes of anomalous sounds for audio surveillance on roads, i.e., tire skidding and car crash, whose occurrences are an evidently acoustic indication of road accidents or disruptions. In the proposed method, we extract a feature of deep audio representation (DAR) and then use a classifier of a bidirectional long short-term memory network to determine the class of the sound to which each test audio segment belongs. We propose a framework based on multiple-stage deep autoencoder network (DAN) to extract the DAR, which fuses complementary information from several input features and thus can be more discriminative and robust than those input features. In the experiments, we discuss the influences of the parameter settings of the DAN's hidden layers on the performance of DAR and compare the DAR with other features. Furthermore, the proposed method is compared to the state-of-the-art methods. In evaluating the data with various signal-to-noise ratios, the results show that the DAR outperforms other features, and the proposed method is superior to the state-of-the-art methods for detecting anomalous sounds on roads.

INDEX TERMS Deep audio representation, bidirectional long short-term memory network, accident detection, audio surveillance.

I. INTRODUCTION

With the high-speed development of the economies of developing countries such as China, the number of private cars and public vehicles has been rapidly increasing on roads. There is one death every four minutes due to road accidents in developing countries [1]. People in these countries have made great efforts to ensure the security and safety of both people and goods on roads. Currently, surveillance systems based on image analysis have been widely adopted for monitoring road traffic [2]–[4]. Road traffic surveillance mainly includes the detection of accidents or road disruptions for quickly ensuring the intervention of rescue teams [5].

It has been reported that decreasing the time between the moment when an accident occurs and the moment when the rescue team is dispatched significantly reduces the mortality rate (by approximately 6%) [6], [7]. Currently, cameras are generally adopted to control the behavior of vehicles through tracking the traces of vehicles [8]–[11] and thus can monitor different traffic conditions on roads, such as accidents and long queues [12], [13]. Nevertheless, in some situations, the visual information is insufficient to reliably infer the activities of vehicles or to discover possibly dangerous conditions. For example, a tire skidding on the road is definitely evidence of an accident or a dangerous condition and has a quite distinctive acoustic indication, but it is hard to identify from images. In addition, accidents can occur out of the view scope of cameras (i.e., the blind area of the camera) or occur when the light is quite dim. In these situations, neither a human operator nor an image analysis-based surveillance system can detect the accidents based on the visual information only. In contrast, audio analysis-based surveillance systems have no blind areas and are not

influenced by illumination variations and thus can normally work during both day and night. Hence, the processing of audio signals acquired by a microphone as a complementary mode to image processing can definitely enhance the detection abilities of automatic surveillance systems [14], [15]. In addition, it is currently simple to deploy audio analysis-based surveillance since IP cameras adopted for road surveillance are generally equipped with embedded microphones for audio signal acquisition.

Although audio analysis is critical for detecting accidents in some situations, the problem of anomalous sound detection for audio surveillance on roads is challenging in open environments. Currently, there are some difficulties in applying anomalous sound detection to road surveillance. The first difficulty is that anomalous sounds are generally superimposed on a high level of background noise and sometimes occur at a great distance from the microphones. As a result, the signal-to-noise ratio (SNR) is very low, and thus the detection of such anomalous sounds becomes a complicated problem. The second difficulty is that intraclass variations of time-frequency characteristics for each class of anomalous sounds are quite significant due to complex road conditions and various types of vehicles. For example, the duration of a tire skidding can range from less than one second to several seconds, and the spectrogram of car crash has different shapes because the crash can occur among various types of vehicles, at different places, at different speeds, and so on. Due to these aforementioned difficulties, audio analysis-based intelligent surveillance systems are currently in the research stage and not widely deployed for road surveillance. Anomalous sound detection on roads is a key component of the audio surveillance system and has received considerable attention [16], [17]. In this study, we focus on effectively detecting two classes of anomalous sounds, i.e., tire skidding and car crash, for audio surveillance on roads. These two classes of anomalous sounds frequently occur on roads and are an obvious acoustic indication of road accidents or disruptions. This work is motivated by a practical audio surveillance application in which anomalous sounds (i.e., the sounds of interest) generally occur on roads with various SNRs.

A. RELATED WORKS

Over the past decades, great efforts have been made for sound detection or classification in the fields of both signal processing [18]–[34] and intelligent transportation [16], [35]–[42].

Some recent evaluation campaigns for sound detection or classification have been launched in the field of signal processing, such as detection and classification of acoustic scenes and events (DCASE) 2013 [18], and DCASE 2016-2018 [19]–[23]. In addition to these campaigns, some researchers have individually performed studies on the detection or classification of sounds. For example, Lee *et al.* [22] detected sound events via the feature of mel-spectrogram and the ensemble of convolutional neural networks (CNN). Similarly, Xu *et al.* [23] classified audio events using the feature of log-mel-spectrogram and the classifier of gated CNN. McLoughlin *et al.* [24] used a deep neural network (DNN) and a low-resolution overlapped spectrogram as the classifier and input feature for robust sound classification. Gencoglu *et al.* [25] adopted a DNN-based classifier fed by the feature of MFCCs, mel-energy, or log-mel-energy, for identifying isolated acoustic events. Rabaoui *et al.* [26] recognized environmental sounds using a hidden Markov model (HMM)-based classifier with the input of various combinations of time-frequency features, such as zero-crossing rate, spectral centroid, MFCCs, and perceptual linear prediction coefficient (PLPC). Phan *et al.* [27] used a regression forests classifier fed by the feature of acoustic superframes to detect sounds. Küçükbay and Sert [28] detected sounds by combining an MFCC feature with an SVM classifier. Lu *et al.* [29] extracted a sparse feature representation based on the similarity measurement of spectral exemplars and then built an SVM classifier for sound detection. Tran and Li [30] studied kernel techniques of the subband probabilistic distance under the framework of SVM for sound event recognition. With the input feature of MFCCs, their proposed SVM classifier outperformed conventional SVM classifiers. For detecting cheering and applause events in the audio stream of various TV programs, Lu *et al.* [31] proposed a method based on the SVM classifier and audio feature vectors, such as subband energy, PLPC, and pitch. Zhang *et al.* [32] classified sounds using the features of tensor-based sparse approximation with the classifier of a Gaussian mixture model (GMM). Kumar *et al.* [33] detected sounds from acoustic unit occurrence patterns.

In addition to the works above, some research on sound detection or classification have been reported in the field of intelligent transportation. For example, the problem of sound detection for audio surveillance was highlighted in [35]. In addition, some classes of sounds, such as gunshots, were regarded as the detection targets for audio surveillance in different locations [36]–[41]. Recently, Foggia *et al.* [16] focused on detecting two classes of anomalous sounds on roads for audio surveillance, i.e., tire skidding and car crash, by using the classifiers of SVM and KNN that were fed by the features of MFCC, energy ratios in bark sub-bands and temporal-spectral features. The work of Foggia *et al*. is quite similar to our work in this study and is used as one of the baselines for performance comparisons.

As seen from the aforementioned discussions, the features adopted in the previous works are hand-crafted features, e.g., MFCCs, log-mel-spectrogram, sub-band energy, and zero-crossing rate. Although these hand-crafted features performed well in the previous works, they still have shortcomings. For example, MFCC is the most popularly used feature, but it is not suitable for discriminating sounds with nonstationary noise due to its poor robustness [43]. Gabor filter bank is a biologically inspired spectro-temporal feature [44] and can improve the performance in comparison with MFCC under adverse noise conditions, but it cannot deeply characterize the properties of complex sounds and has relatively poor discriminability [43], [45]. In addition, they are shallow

features instead of deep transformed features, and thus they cannot deeply represent the properties of sounds. To overcome the shortcomings of the shallow features adopted in the previous works, we propose a deep transformed feature that is more discriminative and insensitive to noises.

B. OUR CONTRIBUTIONS

Inspired by the success of deep learning for feature representation [46], we propose a framework based on multiple-stage deep autoencoder networks (DAN) to extract a feature to deeply represent different properties among various classes of sounds and integrate the strongpoints of input features. In the proposed framework, the hidden layers of each DAN are trained to generate information relevant for discrimination among sounds. The bottleneck layer is the narrowest hidden layer in a DAN, whose activation signals can be used as a compact representation of the original high-dimensional inputs fed to the input layer of a DAN [47], [48]. Hence, the output of the bottleneck layer of each DAN is adopted as a new feature representation, and the output of the bottleneck layer of the last DAN in the proposed framework is called deep audio representation (DAR) here. To the best of our knowledge, there are no other studies to extract DAR by multiple-stage DANs for detecting anomalous sounds on roads. The feature of DAR not only fuses complementary information among different input features but also identifies new potential information by nonlinear transformation and dimensionality reduction realized by multiple-stage DANs. Hence, it can perform better for anomalous sound detection under different noisy conditions than state-of-the-art features such as MFCC or Gabor filter bank.

In addition, considering the high contextual correlation among sounds and the advantage of neural networks in capturing sound-sequence information, we propose using a classifier of a bidirectional long short-term memory (BLSTM) network instead of traditional classifiers, such as support vector machine (SVM) and K-nearest neighbor (KNN), adopted in the previous works (e.g., [16]) to model sounds. In the proposed method, the BLSTM network is fed by the feature of DAR for realizing the task of anomalous sound detection on roads.

The contributions of this study are as follows. First, we propose a novel framework of feature extraction adopting a multiple-stage deep neural network based on DAN. By unsupervised learning, the robust and discriminative feature, namely, DAR, is extracted by the proposed framework to efficiently characterize the properties of various classes of sounds for accurate detection of anomalous sounds. Second, we propose a method for detecting two classes of anomalous sounds on roads by combining the feature of DAR with a classifier of a BLSTM network. Third, we discuss the impacts of the parameter settings of the DAN's hidden layers on the performance of the DAR and compare the DAR with state-of-the-art features used in previous works. Additionally, we compare the proposed method with state-of-the-art methods adopted in previous works for detecting two classes of anomalous sounds on roads under different SNRs. These contributions have not been addressed in previous works. The ultimate goal of this study is to develop an audio surveillance system on roads, and a module for anomalous sounds detection with high accuracy is the most critical component in the system.

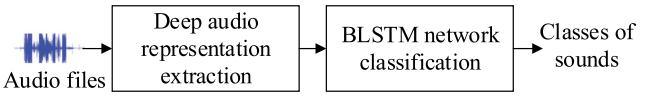


FIGURE 1. The diagram of the proposed method.

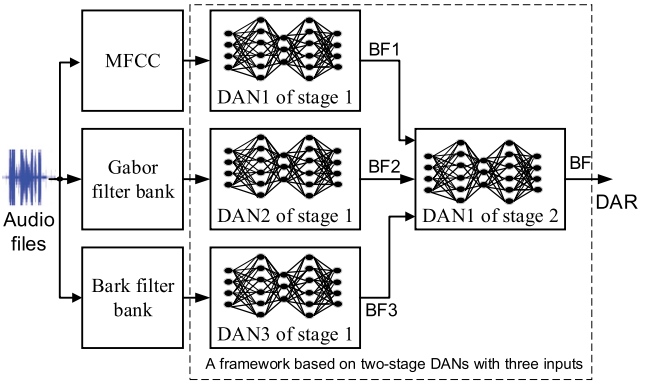


FIGURE 2. The diagram for extracting the feature of deep audio representation (DAR), where BF stands for bottleneck feature.

The rest of the paper is organized as follows. Section II describes the proposed method. Section III presents experiments and discussions, and finally, conclusions are drawn in Section IV.

II. THE METHOD

The diagram of the proposed method is depicted in Fig. 1, which consists of two modules: deep audio representation extraction and BLSTM network classification.

A. DEEP AUDIO REPRESENTATION EXTRACTION

The motivation for designing the proposed feature of DAR is based on two considerations. First, each class of sounds generally possesses a unique time-frequency property, which can be effectively represented by MFCC, Gabor filter bank, or bark filter bank for sound detection [16], [17]. Second, deep learning techniques have a strong ability to learn a compact representation from the high-dimensional input data [46]. Additionally, each type of feature is complementary to some extent and has its own advantages, e.g., strong discriminability of MFCC and bark filter bank [16], [17] or strong anti-noise robustness of a Gabor filter bank [44], [45]. Hence, we propose a framework based on multiple-stage DANs to learn a feature by transforming and fusing the input features, e.g., MFCC, Gabor filter bank and bark filter bank, with the aim of integrating their advantages and thus obtaining a better result.

As shown in Fig. 2, a framework based on two-stage DANs with three input features is taken as an example to illustrate

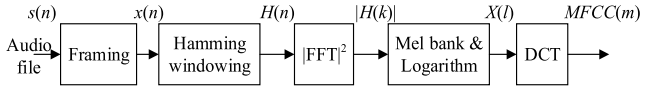


FIGURE 3. The extraction procedure of MFCC.

the extraction of the DAR, which is a realization of the two considerations above. Each audio file is first split into frames for extracting three features, i.e., MFCC, Gabor filter bank and bark filter bank, and then a framework based on two-stage DANs is built for extracting the DAR. Three input features are fed to DAN1, DAN2 and DAN3 in the first stage. Then, the transformed features are output from the bottleneck layers of DAN1, DAN2 and DAN3. The outputs of the bottleneck layer in these three DANs in stage 1 are called bottleneck features (BF) for the input features and marked as BF1, BF2 and BF3. Then, the concatenation of BF1, BF2 and BF3, i.e., [BF1 BF2 BF3], is fed to the DAN in the second stage, i.e., DAN1 of stage 2, whose bottleneck layer output is the feature of DAR. The DAR integrates complementary information of all input features through multiple-stage DANs and thus possesses the benefits of its input features, e.g., MFCC, Gabor filter bank and bark filter bank in Fig. 2.

1) MFCC EXTRACTION

The feature of MFCC is widely used and has been proven to be effective for sound detection [16], [17]. Hence, it is used as an input feature of the proposed framework for enhancing the discriminability of DAR. Fig. 3 shows the extraction of MFCC, which consists of five parts: framing, Hamming windowing, fast Fourier transform (FFT), mel-bank and logarithm, and discrete cosine transform (DCT).

The audio file is first divided into audio frames $x(n)$ of frame length N_s sampling points with half overlapping. Next, a windowing operation is performed by a Hamming window function $w(n)$ which is defined by

$$w(n) = 0.54 - 0.46\cos\left(\frac{2n\pi}{N_s - 1}\right), \quad 0 \le n \le N_s - 1. \tag{1}$$

Thus, each windowed audio frame $h(n)$ is obtained by multiplying $x(n)$ by $w(n)$. To analyze $h(n)$ in the frequency domain, one N_s-point FFT is then carried out for transforming $h(n)$ into the corresponding frequency elements. The value and the magnitude of a frequency element are computed by

$$H(k) = \sum_{n=0}^{N_s-1} h(n)\, e^{-j\frac{2nk\pi}{N_s}}, \quad 0 \le k \le N_s - 1, \tag{2}$$

$$|H(k)| = \sqrt{(\mathrm{Re}\{H(k)\})^2 + (\mathrm{Im}\{H(k)\})^2}, \tag{3}$$

where $\mathrm{Re}\{H(k)\}$ and $\mathrm{Im}\{H(k)\}$ stand for the real and imaginary parts of $H(k)$, respectively. The logarithmic power spectrum on the mel-scale is computed by a filter bank with L_f filters [49],

$$X(l) = \log\left(\sum_{k=k_{ll}}^{k_{lu}} |H(k)|\, W_l(k)\right), \tag{4}$$

where $l = 0, 1, \ldots, L_f\text{-}1$; $W_l(k)$ is the l^{th} mel-scale filter, and k_{ll} and k_{lu} are the lower bound and the upper bound of the l^{th} filter, respectively. The lower bound and the upper bound of a filter are determined by considering the relationship between the frequency and the mel-scale [5]. The mel-scale is perceptually motivated by the human auditory system [50]. It emphasizes the spectra in the low frequency. The mel-scale frequency is computed by

$$Mel(f) = 1125\ln\left(1 + \frac{f}{700}\right), \tag{5}$$

where f denotes the linear frequency in Hz. Finally, a DCT is performed on $X(l)$ to obtain the MFCC, i.e., $MFCC(m)$:

$$MFCC(m) = \sum_{l=1}^{L_f} X(l)\cos\left(\frac{m\,(l - 0.5)\,\pi}{L_f}\right), \tag{6}$$

where $m = 1, \ldots, M$, and M denotes the dimension of MFCC.

2) GABOR FILTER BANK EXTRACTION

It has been shown that the feature of Gabor filter bank can improve the robustness of speech recognition and sound detection under noisy conditions [43], [45], [51]. The usage of Gabor filters [52] is motivated by their similarity to spectro-temporal patterns of neurons in the auditory cortex of mammals [53]. Therefore, a Gabor filter bank is used as an input feature of the proposed framework for enhancing the anti-noise robustness of the DAR. A Gabor filter is a product of a two-dimensional Hanning-shaped envelope function with a two-dimensional sinusoidal carrier. The Hanning-shaped envelope function is defined by

$$h_b(n) = \begin{cases} 0.5 - 0.5\cos\left(\dfrac{2\pi n}{b}\right) & -\dfrac{b}{2} < n < \dfrac{b}{2} \\ 0 \qquad\qquad else, \end{cases} \tag{7}$$

where b stands for the width of the envelope, multiplied by a sinusoidal carrier function with frequency ω

$$s_\omega(n) = \exp(j\omega n). \tag{8}$$

The Gabor filter can be expressed by the frequency index m and frame index n, the central frequency channel m_0 and the central time frame n_0, the spectral modulation frequency ω_m and temporal modulation frequency ω_n, and the number of semicycles under the envelope ν_m and ν_n, i.e.,

$$\begin{aligned} &g(m_0, n_0, \omega_m, \omega_n, m, n, \nu_m, \nu_n) \\ &\quad = s_{\omega_m}(m - m_0) \cdot s_{\omega_n}(n - n_0) \cdot h_{\frac{\nu_m}{2\omega_m}}(m - m_0) \\ &\quad \cdot h_{\frac{\nu_n}{2\omega_n}}(n - n_0). \end{aligned} \tag{9}$$

The extraction procedure of the Gabor filter bank feature is as follows. The audio file is first divided into audio

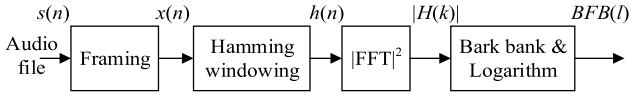


FIGURE 4. The extraction procedure of the bark filter bank.

frames $x(n)$ of frame length of N_s sampling points with half overlapping. Next, the windowed audio frame $h(n)$ is obtained through multiplying $x(n)$ by a Hamming window function $w(n)$ and is then transformed to the frequency domain by a discrete Fourier transformation (DFT). The absolute value $|Y_{n,k}|$ of the resulting spectrogram is mel-warped by triangular-shaped mel-filters $F_{k,m}$ in a frequency region between 64 Hz and 8 kHz and logarithmized, resulting in a log-scaled mel-spectrogram with mel-bands m

$$\tilde{Y}_{n,m} = \log\left(\sum_{k=0}^{N_s-1} \left|Y_{n,k}\right| \cdot F_{k,m}\right), \quad 0 \le m \le M' - 1, \quad (10)$$

where M' stands for the number of mel-filters. Finally, the Gabor filter bank feature is obtained by filtering the log-scaled mel-spectrogram $\tilde{Y}_{n,m}$ with the real part of the Gabor filters defined by (9) that are sensitive to frequency changes over time, i.e.,

$$G_{n,m}(m_0, n_0, \omega_m, \omega_n, v_m, v_n) = \sum_u \sum_\lambda \tilde{Y}_{n,m} \cdot \mathrm{Re}\{g(m_0, n_0, \omega_m, \omega_n, u+m, \lambda+n, v_m, v_n)\}, \quad (11)$$

where $\mathrm{Re}\{g(\cdot)\}$ denotes the real part of $g(\cdot)$.

3) BARK FILTER BANK EXTRACTION

The feature of the bark filter bank is also used as an input feature for extracting the DAR since its extraction procedure is different from that of MFCC and Gabor filter bank. Thus, they have complementary information for representing the property of sounds. As shown in Fig. 4, there are four components for its extraction: framing, Hamming windowing, FFT, and bark bank and logarithm.

The bark filter bank has similar preprocessing to MFCC, i.e., framing, Hamming windowing and FFT. However, instead of a mel-scale filter bank, a bark-scale filter bank is used [50]. Bark-scale is also perceptually motivated by human hearing. The bark-scale frequency is calculated by

$$Bark(f) = 13 \arctan(0.00076f) + 3.5 \arctan\left(\left(\frac{f}{7500}\right)^2\right), \quad (12)$$

where f denotes linear frequency in Hz.

The extraction procedure of the bark filter bank feature is as follows. The audio file is first divided into audio frames $x(n)$, and then the operation of Hamming windowing is performed to obtain windowed audio frame $h(n)$. Next, one N_s-point FFT is used to transform $h(n)$ to the linear-frequency spectrum $H(k)$. The logarithmic power spectrum on the bark-scale

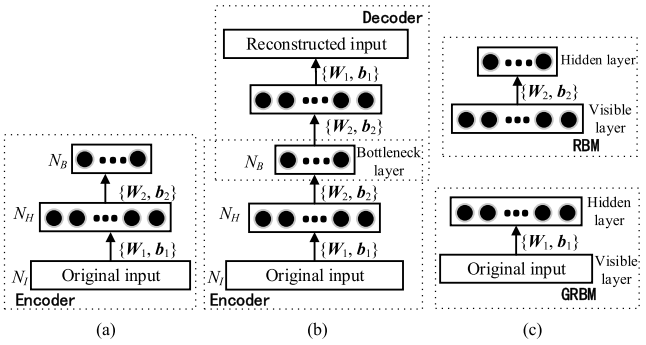


FIGURE 5. (a) A neural network with three layers. (b) Unrolling the network. The encoder and decoder are inside two different dashed rectangles. (c) The network is initially built as a restricted Boltzmann machine (RBM) and a Gaussian RBM (GRBM). N_B, N_H and N_I are neuron numbers of the bottleneck, hidden and input layers, respectively.

is finally computed by a filter bank with L_f filters [50],

$$BFB(l) = \log\left(\sum_{k=k_{ll}}^{k_{lu}} |H(k)|\, W_l(k)\right), \quad (13)$$

where $l = 0, 1, \ldots, L_f$-1; $W_l(k)$ is the l^{th} bark-scale filter; and k_{ll} and k_{lu} are the lower and the upper bounds of the l^{th} filter, respectively. The lower and the upper bounds of a filter are determined by considering the relationship between the frequency and the bark-scale.

4) BOTTLENECK FEATURE EXTRACTION

The network for bottleneck feature extraction used in the proposed framework is a DAN. The reason for using the DAN to extract the bottleneck feature is that the DAN can be trained without the training sample labels (i.e., in an unsupervised way) with higher performance. As a result, it is very convenient to design the system of anomalous sound detection, since the training sample labels are not always available in practice, and data annotation is time-consuming.

In this study, the output of the bottleneck layer is used as a compact representation of the input features. We extract a feature representation from the neuron activations of the bottleneck layer, called the bottleneck feature. In the DAN, an adaptive and multilayer encoder network is adopted to transform high-dimensional inputs to a low-dimensional code, whereas a decoder network is used to recover the inputs from the code [54]. Fig. 5 shows a DAN with one hidden layer (in decoder and encoder networks), which is taken as one example for demonstrating the extraction of the bottleneck feature.

In Fig. 5 (a), an encoder network is used to transform the original input features (e.g., Gabor filter bank) to a concise representation (i.e., a bottleneck feature with N_B dimensions). It is assumed that $\boldsymbol{F} = \{\boldsymbol{f}_i : \boldsymbol{f}_i \in \boldsymbol{R}^{D\times 1}\}_{i=1,2,..,I}$, $\boldsymbol{W} = \{\boldsymbol{W}_i\}_{i=1,2}$ and $\boldsymbol{b} = \{\boldsymbol{b}_i\}_{i=1,2}$ stand for the sets of: input feature vector, weight matrix and bias vector of the encoder network, respectively. The encoder network defines a transformation $\Phi(\cdot)$: $\boldsymbol{R}^{D\times 1} \rightarrow \boldsymbol{R}^{Q\times 1}$, which transforms an input feature vector $\boldsymbol{f}$

with D dimensions into a Q-dimensional representation $\Phi(\boldsymbol{f})$:

$$\Phi(\boldsymbol{f}) = \boldsymbol{W}_2 \cdot \psi(\boldsymbol{W}_1 \cdot \boldsymbol{f} + \boldsymbol{b}_1) + \boldsymbol{b}_2, \tag{14}$$

where $\psi(\cdot)$ is an activation function: $\psi(\boldsymbol{f}) = 1/(1 + e^{-\boldsymbol{f}})$.

The encoder network in Fig. 5 (a) is unrolled, and then a DAN with a decoder network is obtained as depicted in Fig. 5 (b). The decoder network creates a transformation $\hat{\Phi}(\cdot)$: $\boldsymbol{R}^{Q\times 1} \rightarrow \boldsymbol{R}^{D\times 1}$, which uses the transformed representation $\Phi(\boldsymbol{f})$ to rebuild the original input feature vector $\boldsymbol{f}$. The reconstructed input is defined as

$$\hat{\Phi}(\boldsymbol{f}) = \boldsymbol{W}_1 \cdot \psi(\boldsymbol{W}_2 \cdot \Phi(\boldsymbol{f}) + \boldsymbol{b}_2) + \boldsymbol{b}_1. \tag{15}$$

After obtaining the reconstructed input, an objective function O_r is defined as

$$O_r = \sum_{i=1}^{I} \|\boldsymbol{f}_i - \hat{\Phi}(\boldsymbol{f}_i)\|^2, \tag{16}$$

where $||\cdot||$ represents the Euclidean norm.

A gradient descent algorithm is adopted to learn the set of weight matrices $\boldsymbol{W} = \{\boldsymbol{W}_i\}_{i=1,2}$, and the backpropagation algorithm is used to calculate the derivatives of the objective function with respect to the weights [54]. The construction of a DAN includes two steps: a pretraining process to initialize the network's parameters and a fine-tuning process. As shown in Fig. 5 (c), the restricted Boltzmann machine (RBM) is one basic unit for pretraining the DAN [54], which consists of a visible layer and a hidden layer. Each neuron in the visible layer is connected to every neuron in the hidden layer, and the values of the neurons are binary. The energy function of the RBM is defined by

$$E(\boldsymbol{v}, \boldsymbol{h}) = -\boldsymbol{v}^{\mathrm{T}}\boldsymbol{W}_{ij}\boldsymbol{h} - \boldsymbol{b}_i\boldsymbol{v} - \boldsymbol{b}_j\boldsymbol{h}, \tag{17}$$

where T stands for the transpose of a matrix (or vector); $\boldsymbol{v}$ and $\boldsymbol{h}$ denote the neuron vectors of the visible and hidden layers, respectively; $\boldsymbol{W}_{ij}$ represents the weight matrix between the visible and hidden layers; and $\boldsymbol{b}_i$ and $\boldsymbol{b}_j$ denote the bias vectors of the visible and hidden layers, respectively.

To process the real-valued data, a Gaussian RBM (GRBM) is used in the first layer of the DAN. The energy function of the GRBM is defined as

$$E(\boldsymbol{v}, \boldsymbol{h}) = \sum_i \frac{(\boldsymbol{v}_i - \boldsymbol{b}_i)^2}{2\sigma_i^2} - \sum_i \sum_j \frac{\boldsymbol{v}_i}{\sigma_i}\boldsymbol{W}_{ij}\boldsymbol{h}_j - \sum_j \boldsymbol{b}_j\boldsymbol{h}_j, \tag{18}$$

where $\boldsymbol{W}_{ij}$, $\boldsymbol{b}_i$ ($\boldsymbol{b}_j$), $\boldsymbol{v}_i$ and $\boldsymbol{h}_j$ stand for the weight matrices, the bias vectors, the neuron vectors of the visible and hidden layers of the GRBM, respectively; and σ_i is the standard deviation of the Gaussian noise for visible neuron i. The joint probability distribution of the neurons is defined by

$$P(\boldsymbol{v}, \boldsymbol{h}) = \frac{1}{Z}\exp\left(-E(\boldsymbol{v}, \boldsymbol{h})\right), \tag{19}$$

where Z is a normalization factor for scaling P ($\boldsymbol{v}$, $\boldsymbol{h}$) to the range of [0 1]. The parameters of RBM are iteratively updated by minimizing the negative log-likelihood $-\sum_{\boldsymbol{h}} \log P(\boldsymbol{v}, \boldsymbol{h})$ by the stochastic gradient descent algorithm. The contrastive divergence [55] is used for approximating the intractable calculation of the gradients. During pretraining the DAN, two adjacent layers are used as an RBM, and the RBMs are trained bottom-up for obtaining better initial parameters. As shown in Fig. 5 (c), the weight matrix $\boldsymbol{W}_1$ and bias vector $\boldsymbol{b}_1$ are trained by treating the bottom two layers as a GRBM, and the weight matrix $\boldsymbol{W}_2$ and bias vector $\boldsymbol{b}_2$ are trained in the same way by treating the next two layers as an RBM. After the pretraining process, the parameters of the DAN are fine-tuned using the backpropagation algorithm [54].

All DANs in the proposed framework (as shown in Fig. 2) are individually generated using their corresponding input features, i.e., MFCC, Gabor filter bank, bark filter bank, [BF1 BF2 BF3]. Then, the DAR is extracted for each audio file by the framework. To model the dynamic properties of sounds, a context with T adjacent frames are generally taken into account. Hence, the neuron number of the input layer of each DAN is $T \times D$, where D is the dimension of the input feature. The parameter settings (i.e., layer number and neuron number per layer) of the hidden layers have a direct influence on the performance of the DAR, and thus, their settings will be discussed in the experiments. The output of the DAN is the reconstructed value of its original input, and thus, the neuron number of the output layer is equal to that of the input layer.

B. BLSTM NETWORK CLASSIFICATION

A recurrent neural network (RNN) possesses feedback connections, and thus works flexibly and efficiently with time-series signals, e.g., audio signals. Because of the problem of the exploding and vanishing gradient, a simple RNN is unable to address long-duration dependencies [56]. Hidden units of the gated RNN are gate based. The LSTM network is one ordinary class of gated RNNs and is widely applied. The detailed introduction to the LSTM network is described in [57].

The LSTM network is flexible for modeling sequential data and is adept at utilizing and storing information for long periods of time [57]. A basic LSTM block consists of three gates (input gate i_n, output gate o_n, and forget gate f_n), one cell c_n, block input I_n, and output activation function O_n, three peephole connections ($\boldsymbol{p}_i, \boldsymbol{p}_f, \boldsymbol{p}_o$) among cells and three gates. The detailed diagram of the LSTM block is illustrated in Fig. 6.

The input and output gates control whether the input signals have an impact on the cell and whether the cell can influence other neurons. The forget gate controls whether the state should be remembered or not, while the peephole connections scale the state of the three gates with the cell state. The output of the LSTM block is recurrently connected back to the block input and all gates. The forward pass of the LSTM layer, including the block input I_n, input gate i_n, forget gate f_n, memory cells c_n, output gate o_n, and block output O_n,

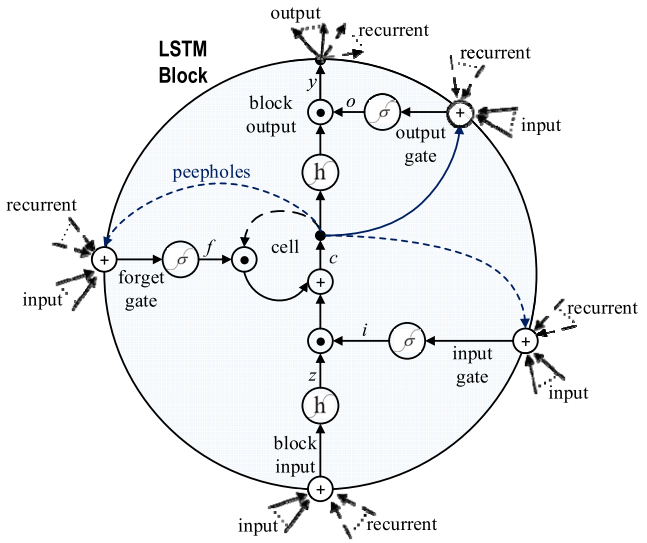


FIGURE 6. The diagram of the LSTM block [58].

are defined by

$$\begin{cases} I_n = h\left(\boldsymbol{W}_I x_n + \boldsymbol{R}_I O_{n-1} + \boldsymbol{b}_I\right) \\ i_n = \sigma\left(\boldsymbol{W}_i x_n + \boldsymbol{R}_i O_{n-1} + \boldsymbol{p}_i \odot c_{n-1} + \boldsymbol{b}_i\right) \\ f_n = \sigma\left(\boldsymbol{W}_f x_n + \boldsymbol{R}_f O_{n-1} + \boldsymbol{p}_f \odot c_{n-1} + \boldsymbol{b}_f\right) \\ c_n = i_n \odot I_n + f_n \odot c_{n-1} \\ o_n = \sigma\left(\boldsymbol{W}_o x_n + \boldsymbol{R}_o y_{n-1} + \boldsymbol{p}_o \odot c_n + \boldsymbol{b}_o\right) \\ O_n = o_n \odot h\left(c_n\right), \end{cases} \tag{20}$$

where n and x_n stand for the order number of the sequential data and the input feature, respectively. $\boldsymbol{W}$, $\boldsymbol{R}$, and $\boldsymbol{b}$ are the weight matrix, the recurrent weight matrix, and the bias vector, respectively. $\boldsymbol{p}$, σ, h, and $\odot$ represent the peephole weight vector, the logistic sigmoid activation function, the hyperbolic tangent activation function, and the pointwise product with the gate value, respectively. Finally, the subscripts I, i, f, and o stand for the block input, input gate, forget gate, and output gate, respectively. Similarly, the corresponding backward pass that is required in the training stage is given in [58].

Though the LSTM network can utilize context for long periods of time, it can gain information from the previous context only and does not have access to information from the future context. As far as the task of anomalous sound detection is concerned, it is needed to utilize information in both directions. Bidirectional RNN (BRNN) realizes this objective through processing the sequential data with two separate hidden layers in both forward and backward directions [59]. The BLSTM network is a combination of the LSTM network and BRNN [60]. Hence, it not only has access to the context for long periods of time but can also utilize the context in both forward and backward directions with two separate hidden layers that are connected to the same output layer [61], as shown in Fig. 7.

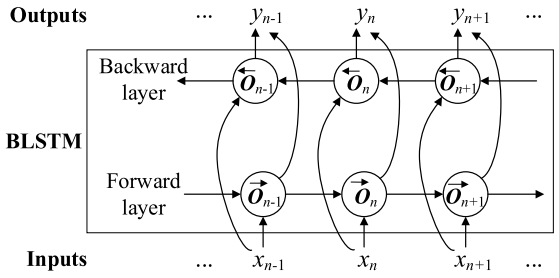


FIGURE 7. The diagram of the BLSTM network.

The BLSTM network can be realized by

$$\begin{cases} \vec{O}_n = \Gamma\left(\boldsymbol{W}_{\vec{I}} x_n + \boldsymbol{W}_{\vec{O}} O_{n-1} + \boldsymbol{b}_{\vec{O}}\right) \\ \overleftarrow{O}_n = \Gamma\left(\boldsymbol{W}_{\overleftarrow{I}} x_n + \boldsymbol{W}_{\overleftarrow{O}} O_{n+1} + \boldsymbol{b}_{\overleftarrow{O}}\right) \\ y_n = \boldsymbol{W}_{\vec{y}} \vec{O}_n + \boldsymbol{W}_{\overleftarrow{y}} \overleftarrow{O}_n + \boldsymbol{b}_y, \end{cases} \tag{21}$$

where $\vec{O}_n$ is the forward hidden sequence, $\overleftarrow{O}_n$ is the backward hidden sequence, and Γ is a sigmoid function.

Considering the advantage of the BLSTM network in capturing sequence information, we propose its use as the classifier for anomalous sound detection in this study and compare it with other classifiers adopted in the previous works. The parameter settings of the BLSTM network will be given in Section III as shown in Table 2.

III. EXPERIMENTS AND DISCUSSIONS

This section begins to introduce experimental data, and then presents experimental setups including the definitions of two evaluation metrics (i.e., *Accuracy*, $F1$ score), the parameter settings for extracting features and for building classifiers. Next, the impacts of the parameter settings of the DAN's hidden layers on the performance of DAR are discussed. Finally, the performance comparison of different features, classifiers and methods for anomalous sound detection is evaluated on the data with different SNRs.

A. EXPERIMENTAL DATA

The experimental data adopted in this study is a public dataset created by Foggia *et al.* [16], which is publicly available at *http://mivia.unisa.it*. The dataset contains two classes of hazardous road events: tire skidding and car crash. The audio clips are saved as WAV files with a sampling frequency of 32 kHz and 16-bit quantization. An audio-based surveillance system needs to detect anomalous sounds in different kinds of background sounds. Hence, the anomalous sounds are not isolated but superimposed to different typical background sounds of roads and traffic jams to consider the occurrence of such abnormal events in real-world conditions.

As similarly done in [16], we adopt a procedure to combine the anomalous sounds with background noises for obtaining different SNRs. The audio file $s(n)$ is first normalized so that they have the same overall energy:

$$\bar{s}(n) = \frac{s(n)}{s_r(n)}, \tag{22}$$

TABLE 1. The details of the experimental data.

Event type	Sample number	Length (s)
Background sound	-	2732
Tire skidding	200	522.5
Car crash	200	326.3

where $s_r(n)$ is the value of the root mean square of the audio file $s(n)$. A background noise file $b(n)$ is randomly chosen from the typical traffic background noises. Then, N_a foreground sounds were randomly selected and superimposed on the background noises for simulating the occurrence of sounds in a real-world condition. The selected sounds were mixed with the background noises as defined by:

$$s_j^{'}(n) = \sum_{i=1}^{N_a} \left\{ b_j(n) \oplus_{[sp_i, ep_i]} (A \cdot \bar{s}_i(n)) \right\}, \tag{23}$$

where $\oplus_{[sp_i, ep_i]}$ is an operation that combines sound $\bar{s}_i(n)$ with background noise $b_j(n)$ in the interval ranging from start point sp_i to end point ep_i of the sound. The end point ep_i is separated from the start point of the next sound $\bar{s}_{i+1}(n)$ by several seconds in which only background noise is present. The value of coefficient A is tuned for obtaining different SNRs in the experiments.

The final experimental dataset consists of 57 audio files with a duration of approximately one minute. Each of the files has a sequence of anomalous sounds. Two-hundred samples per class in total are distributed in these audio files. In the experiments, the audio files are randomly divided into training, validation and test data as 80%, 10% and 10%, respectively. Ten-fold cross-validation is performed, and the final result is the averaged scores of all 10 folds. The details of the experimental data are presented in Table 1.

B. EXPERIMENTAL SETUP

The experiments are implemented on a computer with an Intel(R) Core(TM) i7-6700, 3.10 GHz CPU, 48 GB RAM, and a NVIDIA 1080 TI GPU. Both the *Accuracy* and $F1$ score are used here to evaluate the overall performance of different methods since they have been popularly adopted as performance metrics for sound detection or classification [20], [21], [62]. The higher their values are, the better the performance of the method is. *Accuracy* is defined by

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \tag{24}$$

where *TP*, *FP*, *TN* and *FN* stand for true positive, false positive, true negative and false negative, respectively. For calculating *TP*, *FP*, *TN* and *FN*, the samples of one class of sound are regarded as *positive* while the samples of the other classes of sounds are considered *negative*, and they are alternated in turn. If the outcome from a prediction for one sample is *positive* and the actual value is also *positive*, then it is called a *TP*; however, if the actual value is *negative,* then it is called an *FP*. Conversely, a *TN* occurs when both the prediction outcome and the actual value are *negative* for one sample, and *FN* is when the prediction outcome is *negative* while the actual value is *positive*.

TABLE 2. The parameters for extracting DAR and training the BLSTM network.

Type	Parameters settings
Input feature	Frame length/overlapping: 40/20 ms, 13-D MFCC, 311-D Gabor filter bank (GFB), 20-D bark filter bank (BFB).
DANs	Bottleneck layer: 50 neurons, input layer fed by MFCC: 91 neurons, input layer fed by GFB: 2177 neurons, input layer fed by BFB: 140 neurons, input layer fed by [BF1 BF2 BF3]: 1050 neurons, neuron number of the output layer is equal to that of the corresponding input layer, learning rate: 0.001, maximum iteration: 3000, batch size: 256, context size: 7 frames.
BLSTM network	Cells: 400, weight decay: 0.1, dropout: 0.8, maximum iterations: 3000, batch size: 256, unrolled step: 10, training algorithm: backpropagation via time, initial forget bias: 1.

The $F1$ score is defined by

$$F1 = \frac{2 \times PR \times RR}{PR + RR}, \tag{25}$$

where *PR* and *RR* denote the precision rate and recall rate, respectively. *PR* and *RR* are defined by

$$PR = \frac{TP}{TP + FP}, \tag{26}$$

$$RR = \frac{TP}{TP + FN}. \tag{27}$$

The main parameter configurations for the DAR extraction and BLSTM network building are given in Table 2. Based on the experimental setup above, the training of BLSTM network is usually finished within 1000 iterations and the corresponding learning time ranges from three to four hours. The initial loss value is set to $\ln(N)$, where N is the type number of anomalous sound, i.e., $N = 2$ here. Hence, the loss values change from 0.693 to approximately 0.060 during network learning. Because the parameter settings of the DANs' hidden layers in the proposed framework have direct impacts on the performance of DAR and the key novelty of this study is to propose a feature of DAR, the parameter settings of the DANs' hidden layers will be discussed in subsection III.*C* for obtaining better results.

C. HIDDEN LAYERS SETTINGS OF DANS

In this subsection, we discuss the parameter settings of the DANs' hidden layers in the proposed framework for extracting DAR. We tune the parameters of the hidden layers. The results of anomalous sound detection under the conditions of different settings of hidden layers are listed in Table 3, in which the digit 50 denotes the number of neurons in the bottleneck layer, and both 100 and 200 stand for the numbers of neurons in the other hidden layers. For example, [50, 100] represents that the DAN has 2 hidden layers (including the

TABLE 3. Impacts of hidden layer's settings on the performance of the DAR (in %).

Parameters	*Accuracy*	$F1$
[50]	82.53	81.25
[50, 100]	84.54	83.63
[100, 50, 100]	86.64	85.23
[200, 50, 100, 200]	89.75	87.77
[200, 100, 50, 100, 200]	**92.75**	**91.86**
[200, 100, 50, 100, 100, 200]	90.82	89.75
[200, 100, 100, 50, 100, 100, 200]	88.87	87.94

bottleneck layer), and the bottleneck layer and another hidden layer have 50 and 100 neurons, respectively. As shown in Table 3, the proposed method achieves 92.75% of *Accuracy* and 91.86% of $F1$ score, i.e., the highest values of both the *Accuracy* and $F1$ score, when the parameters of the hidden layer of the DANs are set to [200, 100, 50, 100, 200]. This parameter setting of the hidden layer is fixed and adopted for the proposed method in the following subsections.

D. COMPARISON OF DIFFERENT FEATURES

Three state-of-the-art features (i.e., MFCCs [16], [17], [28], bark filter bank (BFB) [28], [51], and Gabor filter bank (GFB) [43], [45], [51]) are first extracted from each audio file according to the settings in Table 2. To compare the performance of different features under the same condition, the back-end classifier is the BLSTM network (the same) for all features in this experiment. The results obtained by different features with various SNRs are listed in Table 4.

As shown in Table 4, under the same SNRs, the values of both the *Accuracy* and $F1$ score achieved by DAR are constantly higher than that obtained by MFCCs, bark filter bank, Gabor filter bank, any transformed bottleneck feature of one input feature and any combination of the transformed bottleneck features. These results highlight the noise-robust discriminative abilities of the proposed feature of DAR. As far as three individual input features are concerned, the Gabor filter bank outperforms other two features, whereas the bark filter bank is the worst one in terms of both *Accuracy* and $F1$ score under different SNRs. Similar results can be observed for their corresponding transformed bottleneck features, i.e., GFB_T, $MFCC_T$ and BFB_T. For the combinations of two transformed bottleneck features, the feature of $MFCC_T + GFB_T$ outperforms other combinations of two features in all SNRs conditions. In addition, the lower the SNRs are, the larger the improvements obtained by the Gabor filter bank (or GFB_T) are. For example, when SNR is -10 dB, Gabor filter bank achieves the largest improvement of (*Accuracy*, $F1$ score) by (6.03%, 4.02%) and (9.46%, 9.59%) compared to MFCC and the bark filter bank, respectively. Evaluated on clean data (highest SNR), the performance of MFCC approximates that of the Gabor filter bank.

In conclusion, DAR can integrate the advantages of the input features and outperforms other features in terms of *Accuracy* and *F1* score under all SNRs conditions. Hence, the proposed framework based on two-stage DANs is proven to be effective for feature transformation and fusion.

TABLE 4. Results obtained by different features under various SNRs (in %).

SNRs	Feature	*Acc.*	$F1$	Feature	*Acc.*	$F1$
clean	MFCC	85.13	84.53	$MFCC_T + BFB_T$	87.87	87.21
	BFB	82.36	81.53	$BFB_T + GFB_T$	88.12	87.85
	GFB	85.74	85.34	$MFCC_T + GFB_T$	88.74	88.13
	$MFCC_T$	85.64	84.75	$MFCC_T$+BFB_T+GFB_T	90.13	89.20
	BFB_T	83.38	83.12	**DAR**	**92.15**	**91.32**
	GFB_T	86.85	86.59			
20 dB	MFCC	84.42	82.21	$MFCC_T + BFB_T$	85.36	85.75
	BFB	80.41	79.35	$BFB_T + GFB_T$	86.63	86.53
	GFB	84.33	83.35	$MFCC_T + GFB_T$	87.94	87.58
	$MFCC_T$	84.57	83.32	$MFCC_T$+BFB_T+GFB_T	89.86	88.73
	BFB_T	81.24	81.36	**DAR**	**92.02**	**91.15**
	GFB_T	84.98	84.56			
10 dB	MFCC	82.01	80.63	$MFCC_T + BFB_T$	83.67	83.34
	BFB	79.67	78.08	$BFB_T + GFB_T$	85.52	84.33
	GFB	82.16	81.16	$MFCC_T + GFB_T$	86.56	85.97
	$MFCC_T$	82.32	81.31	$MFCC_T$+BFB_T+GFB_T	88.53	87.61
	BFB_T	79.91	80.26	**DAR**	**90.76**	**89.52**
	GFB_T	83.14	82.35			
0 dB	MFCC	80.24	79.02	$MFCC_T + BFB_T$	82.31	81.82
	BFB	79.43	75.45	$BFB_T + GFB_T$	85.21	82.89
	GFB	81.45	80.37	$MFCC_T + GFB_T$	85.76	84.65
	$MFCC_T$	80.63	79.64	$MFCC_T$+BFB_T+GFB_T	86.97	85.85
	BFB_T	78.87	78.50	**DAR**	**89.62**	**88.03**
	GFB_T	81.95	80.93			
-10 dB	MFCC	73.73	74.12	$MFCC_T + BFB_T$	75.24	75.50
	BFB	70.30	68.55	$BFB_T + GFB_T$	82.23	80.86
	GFB	79.76	78.14	$MFCC_T + GFB_T$	84.01	82.63
	$MFCC_T$	74.67	75.31	$MFCC_T$+BFB_T+GFB_T	85.35	84.18
	BFB_T	71.15	73.36	**DAR**	**88.11**	**86.52**
	GFB_T	80.21	79.38			

$MFCC_T$: Transformed bottleneck feature of MFCC; BFB_T: Transformed bottleneck feature of BFB; GFB_T: Transformed bottleneck feature of GFB; *Acc.*: *Accuracy*.

E. COMPARISON OF DIFFERENT CLASSIFIERS

To compare different classifiers under the same conditions, the front-end feature is the DAR (the same) for all classifiers in this experiment. The parameter settings of the BLSTM network are given in Table 2. The parameter settings of HMM, GMM, SVM, KNN, DNN and LSTM network are optimally determined, and their main parameters are experimentally set as follows. HMM: 3 states with 64 Gaussian components per state; GMM: 64 Gaussian components; SVM: radial basis kernel function, one-vs-one multiclass training; KNN: $K = 5$, Euclidean distance for distance calculation; DNN: 3 hidden layers, 100 neurons per hidden layer; and LSTM network: 400 cells. The results obtained by different classifiers on the data under various SNR conditions are presented in Table 5.

The classifier of the BLSTM network adopted in this study achieves (*Accuracy*, $F1$ score) of (92.15%, 91.32%), (92.02%, 91.15%), (90.76%, 89.52%), (89.62%, 88.03%) and (88.11%, 86.52%) when evaluated on the data with no added noise (i.e., clean), SNR of 20 dB, SNR of 10 dB, SNR of 0 dB, and SNR of -10 dB, respectively. As shown in Table 5, under the same SNRs, the values of *Accuracy* and $F1$ score obtained by the BLSTM network are always higher than those yielded by HMM, GMM, SVM, KNN, DNN and LSTM network that were used in the previous works. These results highlight the

TABLE 5. Results of different classifiers under various SNRs (in %).

SNRs	Classifiers	*Accuracy*	$F1$
Clean	HMM	85.43	84.65
	GMM	81.24	82.37
	SVM	80.61	83.06
	KNN	77.97	77.78
	DNN	88.51	87.24
	LSTM	89.63	88.31
	BLSTM	**92.15**	**91.32**
20 dB	HMM	82.87	82.14
	GMM	77.50	73.28
	SVM	77.45	75.65
	KNN	76.54	75.87
	DNN	86.53	86.13
	LSTM	88.37	87.24
	BLSTM	**92.02**	**91.15**
10 dB	HMM	80.12	78.98
	GMM	74.01	72.56
	SVM	75.54	74.34
	KNN	72.29	70.62
	DNN	84.63	83.37
	LSTM	86.47	85.31
	BLSTM	**90.76**	**89.52**
0 dB	HMM	75.34	73.23
	GMM	72.40	70.46
	SVM	69.53	68.45
	KNN	65.76	67.28
	DNN	83.73	81.83
	LSTM	84.97	83.72
	BLSTM	**89.62**	**88.03**
-10 dB	HMM	70.76	69.87
	GMM	65.90	64.60
	SVM	64.32	65.74
	KNN	61.43	62.83
	DNN	82.86	80.16
	LSTM	83.48	81.85
	BLSTM	**88.11**	**86.52**

TABLE 6. Results of different methods under various SNRs (in %).

SNRs	Methods	*Accuracy*	$F1$
Clean	MFCC-HMM [26]	80.18	82.47
	MFCC-GMM [19], [32]	82.42	81.17
	MFCC-SVM [16]	78.82	82.35
	MFCC-KNN [16]	76.76	77.78
	MFCC-DNN [20], [24], [25]	84.87	84.06
	LMS-CNN [21]	87.64	87.12
	DAR-BLSTM	**92.15**	**91.32**
20 dB	MFCC-HMM [26]	76.03	74.27
	MFCC-GMM [19], [32]	75.97	73.59
	MFCC-SVM [16]	74.32	71.33
	MFCC-KNN [16]	74.35	70.12
	MFCC-DNN [20], [24], [25]	79.72	79.13
	LMS-CNN [21]	86.34	85.36
	DAR-BLSTM	**92.02**	**91.15**
10 dB	MFCC-HMM [26]	73.89	73.12
	MFCC-GMM [19], [32]	72.95	71.18
	MFCC-SVM [16]	69.83	67.38
	MFCC-KNN [16]	66.38	64.47
	MFCC-DNN [20], [24], [25]	76.83	75.38
	LMS-CNN [21]	84.62	83.76
	DAR-BLSTM	**90.76**	**89.52**
0 dB	MFCC-HMM [26]	69.83	67.13
	MFCC-GMM [19], [32]	67.36	65.89
	MFCC-SVM [16]	64.97	61.64
	MFCC-KNN [16]	64.62	61.84
	MFCC-DNN [20], [24], [25]	72.15	69.83
	LMS-CNN [21]	83.12	82.29
	DAR-BLSTM	**89.62**	**88.03**
-10 dB	MFCC-HMM [26]	58.83	57.47
	MFCC-GMM [19], [32]	54.42	55.15
	MFCC-SVM [16]	48.25	48.43
	MFCC-KNN [16]	46.80	48.14
	MFCC-DNN [20], [24], [25]	63.92	62.86
	LMS-CNN [21]	81.16	80.22
	DAR-BLSTM	**88.11**	**86.52**

strong ability of the BLSTM network to capture sequence information for anomalous sound detection.

F. COMPARISON OF DIFFERENT METHODS

In this subsection, we compare the proposed method (i.e., DAR-BLSTM as shown in Table 6) to some representative methods in the previous works. The methods proposed by Foggia *et al.* [16] are the newest and most relevant concerning the problem of anomalous sounds detection on roads, in which MFCC was proven to be the most effective feature by combining with the classifiers of SVM and KNN. Hence, the methods in [16] are used as two baselines and marked as MFCC-SVM and MFCC-KNN as listed in Table 6. In other representative works, MFCC was also used as one of the predominant features together with the classifiers of HMM [26], GMM [19], [32] and DNN [20], [24], [25]. These methods are also used as baselines and marked as MFCC-HMM, MFCC-GMM and MFCC-DNN as given in Table 6. The method recommended by the DCASE 2018 [21] uses log-mel-spectrogram (LMS) and a CNN as the input feature and classifier, respectively, which is also adopted as a baseline and marked as LMS-CNN. The parameters of these state-of-the-art methods are set according to the suggestions in the corresponding references and optimally tuned on the experimental data.

Table 6 shows that the proposed method outperforms other methods under all SNR conditions. The lower the SNRs are, the larger the improvements obtained by the proposed method are. For example, when the SNR is equal to -10 dB, the improvement of (*Accuracy*, $F1$ score) attain the highest values, i.e., (29.28%, 29.05%), (33.69%, 31.37%), (39.86%, 38.09%), (41.31%, 38.38%), (24.19%, 23.66%) and (6.95%, 6.30%), compared to the methods of MFCC-HMM, MFCC-GMM, MFCC-SVM, MFCC-KNN, MFCC-DNN and LMS-CNN, respectively. In other words, the proposed method is insensitive to background noises, since the decreases of both the *Accuracy* and $F1$ score are quite small (maximum decreases in *Accuracy*: 4.04% = 92.15% - 88.11%, and $F1$: 4.80% = 91.32% - 86.52%) with the decrease of SNRs. Conversely, other methods are not robust for background noises because the differences of both the *Accuracy* and $F1$ score for these methods are significant when SNRs change. For example, for the method of MFCC-KNN, the maximum decrease in *Accuracy* and $F1$ score are: 29.96% = 76.76% - 46.80%, and $F1$: 29.64% = 77.78% - 48.14%, respectively.

TABLE 7. Confusion matrix for the proposed method evaluated on the data with no added noise (in %).

	Tire skidding	Car crash	Background
Tire skidding	**92.64**	2.02	5.34
Car crash	1.65	**93.60**	4.75
Background	4.71	5.08	**90.21**

TABLE 8. Confusion matrix for the proposed method evaluated on the data with SNR of 20 dB (in %).

	Tire skidding	Car crash	Background
Tire skidding	**92.36**	2.16	5.48
Car crash	2.03	**93.68**	4.29
Background	3.71	6.27	**90.02**

TABLE 9. Confusion matrix for the proposed method evaluated on the data with SNR of 10 dB (in %).

	Tire skidding	Car crash	Background
Tire skidding	**91.24**	1.73	7.03
Car crash	1.83	**92.26**	5.91
Background	3.73	7.49	**88.78**

TABLE 10. Confusion matrix for the proposed method evaluated on the data with SNR of 0 dB (in %).

	Tire skidding	Car crash	Background
Tire skidding	**89.97**	2.79	7.24
Car crash	2.51	**91.24**	6.25
Background	5.19	7.16	**87.65**

TABLE 11. Confusion matrix for the proposed method evaluated on the data with SNR of −10 dB (in %).

	Tire skidding	Car crash	Background
Tire skidding	**88.45**	2.40	9.15
Car crash	2.53	**89.65**	7.82
Background	5.63	8.14	**86.23**

To identify the confusion details among different sounds obtained by the proposed method, Tables 7 to 11 present the confusion matrices when the proposed method is evaluated on the data with no added noise (i.e., clean), with SNRs of 20 dB, 10 dB, 0 dB and −10 dB. The confusion matrix [63] contains information about actual and predicted detection obtained by a method and is often used to describe the performance of a method. As shown in Tables 7 to 11, the sound with the best detection result is *Car crash*. In contrast, the sound with the worst detection result is *Background*, since it has larger intraclass varieties in terms of acoustic properties.

IV. CONCLUSIONS

In this work, we have addressed the detection problem of two classes of anomalous sounds on roads. Because our work is motivated by a practical audio surveillance application, it is essential to be able to detect anomalous sounds under heavy noise degradation situations. Thus, it is demonstrated that by extracting a feature of DAR using the proposed framework based on multiple-stage DANs and then combining the feature of DAR with the classifier of a BLSTM network, a better performance of anomalous sounds detection is achieved, even under quite low SNRs.

Based on the details of the proposed method and results, these following conclusions are evident.

1) In terms of both *Accuracy* and $F1$ score, the proposed method greatly outperforms state-of-the-art methods adopted in the previous works. Additionally, it still performs quite well with the decrease of SNRs. Thus, it is suitable for deployment for audio surveillance on roads where the SNR is very low.
2) The proposed framework for extracting the feature of DAR can integrate complementary information contained in several input features by deep nonlinear transformation of multiple-stage DANs. The proposed DAR captures the properties of various classes of sounds and can be used as an effective feature for sound detection. Additionally, in terms of the *Accuracy* and $F1$ score, it is superior to state-of-the-art features such as MFCC, bark filter bank, Gabor filter bank.

Future work will include: 1) considering other kinds of networks (e.g., denoising autoencoder networks) and increasing the numbers of both input features and stages of networks for extracting DAR; 2) exploring more effective features and classifiers to improve the performance of the methods for sound detection under noisy conditions, especially heavy, noisy situations on roads; 3) extending the class of anomalous sounds from 2 to more, with the aim of providing more effective cues for intelligent audio surveillance on roads.

REFERENCES

[1] S. S. Thomas, S. Gupta, and V. K. Subramanian, "Event detection on roads using perceptual video summarization," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 9, pp. 2944–2954, Sep. 2018, doi: 10.1109/TITS.2017.2769719.

[2] S. R. E. Datondji, Y. Dupuis, P. Subirats, and P. Vasseur, "A survey of vision-based traffic monitoring of road intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 10, pp. 2681–2698, Oct. 2016.

[3] B. Maaloul, A. Taleb-Ahmed, S. Niar, N. Harby, and C. Valderrama, "Adaptive video-based algorithm for accident detection on highways," in *Proc. IEEE SIES*, Jun. 2017, pp. 1–6.

[4] C.-M. Huang, H.-L. Wang, H. Zhou, S. Xu, and D. Ren, "EVAC-AV: The live road surveillance control scheme using an effective-vision-area-based clustering algorithm with the adaptive video-streaming technique," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1228–1238, Sep. 2017.

[5] T. Gandhi and M. M. Trivedi, "Pedestrian protection systems: Issues, survey, and challenges," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 413–430, Sep. 2007.

[6] J. White, C. Thompson, H. Turner, B. Dougherty, and D. C. Schmidt, "WreckWatch: Automatic traffic accident detection and notification with smartphones," *Mobile Netw. Appl.*, vol. 16, no. 3, pp. 285–303, 2011.

[7] S. Rauscher, G. Messner, and P. Baur, "Enhanced automatic collision notification (ACN) system—Improved rescue care due to injury prediction—First field experience," in *Proc. ESV*, 2009, pp. 1–10.

[8] L. Brun, A. Saggese, and M. Vento, "Dynamic scene understanding for behavior analysis based on string kernels," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 10, pp. 1669–1681, Oct. 2014.

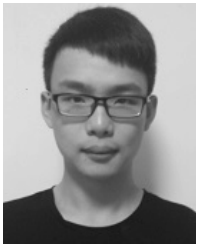
[9] S. Sivaraman, B. Morris, and M. Trivedi, "Observing on-road vehicle behavior: Issues, approaches, and perspectives," in *Proc. IEEE ITSC*, Oct. 2013, pp. 1772–1777.

[10] L. Wang, N. H. C. Yung, and L. Xu, "Multiple-human tracking by iterative data association and detection update," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 1886–1899, Oct. 2014.
[11] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1773–1795, Dec. 2013.
[12] A. Abadi, T. Rajabioun, and P. A. Ioannou, "Traffic flow prediction for road transportation networks with limited traffic data," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 653–662, Apr. 2015.
[13] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.
[14] P. Marmaroli, M. Carmona, J.-M. Odobez, X. Falourd, and H. Lissek, "Observation of vehicle axles through pass-by noise: A strategy of microphone array design," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1654–1664, Dec. 2013.
[15] M. Cristani, M. Bicego, and V. Murino, "Audio-visual event recognition in surveillance video sequences," *IEEE Trans. Multimedia*, vol. 9, no. 2, pp. 257–267, Feb. 2007.
[16] P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio, and M. Vento, "Audio surveillance of roads: A system for detecting anomalous sounds," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 279–288, Jan. 2016.
[17] Y. Li *et al.*, "Unsupervised detection of acoustic events using information bottleneck principle," *Digit. Signal Process.*, vol. 63, pp. 123–134, Apr. 2017.
[18] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Trans. Multimedia*, vol. 17, no. 10, pp. 1733–1746, Oct. 2015.
[19] M. Valenti, A. Diment, G. Parascandolo, S. Squartini, and T. Virtanen, "DCASE 2016 acoustic scene classification using convolutional neural networks," in *Proc. Detection Classification Acoustic Scenes Events Workshop (DCASE)*, Sep. 2016, pp. 95–99.
[20] A. Mesaros *et al.*, "DCASE 2017 challenge setup: Tasks, datasets and baseline system," in *Proc. DCASE Workshop*, Nov. 2017, pp. 85–92.
[21] (2018). *DCASE2018 Challenge*. [Online]. Available: http://dcase.community/challenge2018/
[22] D. Lee, S. Lee, Y. Han, and K. Lee, "Ensemble of convolutional neural networks for weakly-supervised sound event detection using multiple scale input," in *Proc. DCASE Workshop*, 2017, pp. 74–79.
[23] Y. Xu, Q. Kong, W. Wang, and M. D. Plumbley, "Large-scale weakly supervised audio classification using gated convolutional neural network," in *Proc. IEEE ICASSP*, Oct. 2018, pp. 121–125.
[24] I. McLoughlin, H. Zhang, Z. Xie, Y. Song, and W. Xiao, "Robust sound event classification using deep neural networks," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 3, pp. 540–552, Mar. 2015.
[25] O. Gencoglu, T. Virtanen, and H. Huttunen, "Recognition of acoustic events using deep neural network," in *Proc. Eur. Conf. Signal Process.*, Sep. 2014, pp. 506–510.
[26] A. Rabaoui, Z. Lachiri, and N. Ellouze, "Using HMM-based classifier adapted to background noises with improved sounds features for audio surveillance application," *Int. J. Signal Process.*, vol. 5, no. 1, pp. 46–55, 2009.
[27] H. Phan, M. Maaß, R. Mazur, and A. Mertins, "Random regression forests for acoustic event detection and classification," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 1, pp. 20–31, Jan. 2015.
[28] S. E. Küçükbay and M. Sert, "Audio-based event detection in office live environments using optimized MFCC-SVM approach," in *Proc. IEEE ICSC*, Feb. 2015, pp. 475–480.
[29] X. Lu, Y. Tsao, S. Matsuda, and C. Hori , "Sparse representation based on a bag of spectral exemplars for acoustic event detection," in *Proc. ICASSP*, May 2014, pp. 6255–6259.
[30] H. D. Tran and H. Li, "Sound event recognition with probabilistic distance SVMs," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 6, pp. 1556–1568, Aug. 2011.
[31] L. Lu, F. Ge, Q. Zhao, and Y. Yan , "A SVM-based audio event detection system," in *Proc. Int. Conf. Electr. Control Eng.*, Jun. 2010, pp. 292–295.
[32] X. Zhang, Q. He, and X. Feng , "Acoustic feature extraction by tensor-based sparse representation for sound effects classification," in *Proc. IEEE ICASSP*, Apr. 2015, pp. 166–170.
[33] A. Kumar, P. Dighe, R. Singh, S. Chaudhuri, and B. Raj, "Audio event detection from acoustic unit occurrence patterns," in *Proc. IEEE ICASSP*, Mar. 2012, pp. 489–492.
[34] H. Phan, M. Maass, L. Hertel, R. Mazur, I. McLoughlin, and A. Mertins, "Learning compact structural representations for audio events using regressor banks," in *Proc. IEEE ICASSP*, Mar. 2016, pp. 211–215.
[35] M. Crocco, M. Cristani, A. Trucco, and V. Murino, "Audio surveillance: A systematic review," *ACM Comput. Surv.*, vol. 48, no. 4, 2016, Art. no. 52.
[36] C. Clavel, T. Ehrette, and G. Richard, "Events detection for an audio-based surveillance system," in *Proc. ICME*, Jul. 2005, pp. 1306–1309.
[37] G. Valenzise, L. Gerosa, M. Tagliasacchi, F. Antonacci, and A. Sarti, "Scream and gunshot detection and localization for audio-surveillance systems," in *Proc. IEEE AVSS*, Sep. 2007, pp. 21–26.
[38] S. Ntalampiras, I. Potamitis, and N. Fakotakis, "Probabilistic novelty detection for acoustic surveillance under real-world conditions," *IEEE Trans. Multimedia*, vol. 13, no. 4, pp. 713–719, Aug. 2011.
[39] P. Foggia, A. Saggese, N. Strisciuglio, and M. Vento, "Cascade classifiers trained on gammatonegrams for reliably detecting audio events," in *Proc. IEEE AVSS*, Aug. 2014, pp. 50–55.
[40] D. Conte, P. Foggia, G. Percannella, A. Saggese, and M. Vento, "An ensemble of rejecting classifiers for anomaly detection of audio events," in *Proc. IEEE AVSS*, Sep. 2012, pp. 76–81.
[41] M. L. Chin and J. J. Burred, "Audio event detection based on layered symbolic sequence representations," in *Proc. IEEE ICASSP*, Mar. 2012, pp. 1953–1956.
[42] P. Foggiaa, N. Petkovb, A. Saggesea, N. Strisciuglioa, and M. Vento, "Reliable detection of audio events in highly noisy environments," *Pattern Recognit. Lett.*, vol. 65, pp. 22–28, Nov. 2015.
[43] M. R. Schädler, B. T. Meyer, and B. Kollmeier, "Spectro-temporal modulation subspace-spanning filter bank features for robust automatic speech recognition," *J. Acoust. Soc. Amer.*, vol. 131, no. 5, pp. 4134–4151, 2012.
[44] J. Schröder, S. Goetze, and J. Anemüller, "Spectro-temporal Gabor filterbank features for acoustic event detection," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 12, pp. 2198–2208, Dec. 2015.
[45] J. Schröder *et al.*, "On the use of spectro-temporal features for the IEEE AASP challenge 'detection and classification of acoustic scenes and events,'" in *Proc. IEEE Workshop ASPAA*, Oct. 2013, pp. 1–4.
[46] D. Yu and L. Deng, *Automatic Speech Recognition: A Deep Learning Approach*. London, U.K.: Springer-Verlag, 2015.
[47] F. Grezl, M. Karafiat, S. Kontar, and J. Cernocky, "Probabilistic and bottleneck features for LVCSR of meetings," in *Proc. IEEE ICASSP*, Apr. 2007, pp. 757–760.
[48] D. Yu and M. L. Seltzer, "Improved bottleneck features using pretrained deep neural networks," in *Proc. INTERSPEECH*, 2011, pp. 237–240.
[49] D. G. Childers, D. P. Skinner, and R. C. Kemerait, "The cepstrum: A guide to processing," *Proc. IEEE*, vol. 65, no. 10, pp. 1428–1443, Oct. 1977.
[50] X. Huang, A. Acero, and H.-W. Hon, *Spoken Language Processing: A Guide to Theory, Algorithm and System Development*. Upper Saddle River, NJ, USA: Prentice-Hall, 2001.
[51] N. Moritz *et al.*, "Noise robust distant automatic speech recognition utilizing NMF based source separation and auditory feature extraction," in *Proc. CHiME Challenge Workshop*, 2013, pp. 1–6.
[52] D. Gabor, "Theory of communication," *Inst. Electron.*, vol. 93, pp. 429–457, 1946.
[53] A. Qiu, C. E. Schreiner, and M. A. Escabí, "Gabor analysis of auditory midbrain receptive fields: Spectro-temporal and binaural composition," *J. Neurophysiol.*, vol. 90, no. 1, pp. 456–476, 2003.
[54] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
[55] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, vol. 14, no. 8, pp. 1771–1800, 2002.
[56] R. Pacanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. ICML*, 2013, pp. 1310–1318.
[57] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE ICASSP*, May 2013, pp. 6645–6649.
[58] K. Greff, R. K. Srivastava, J. Koutnìk, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.
[59] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
[60] L. Sun, S. Kang, K. Li, and H. Meng, "Voice conversion using deep bidirectional long short-term memory based recurrent neural networks," in *Proc. IEEE ICASSP*, Apr. 2015, pp. 4869–4873.

[61] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Netw.*, vol. 18, no. 5, pp. 602–610, 2005.
[62] Y.-X. Li, Q.-H. He, S. Kwong, T. Li, and J.-C. Yang, "Characteristics-based effective applause detection for meeting speech," *Signal Process.*, vol. 89, no. 8, pp. 1625–1633, 2009.
[63] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006.

YANXIONG LI received the B.S. and M.S. degrees in electronic engineering from Hunan Normal University in 2003 and 2006, respectively, and the Ph.D. degree in electronic engineering from the South China University of Technology (SCUT) in 2009. From 2008 to 2009, he was a Researcher with the Department of Computer Science (DCS), City University of Hong Kong. From 2013 to 2014, he was a Researcher with DCS, The University of Sheffield, U.K. In 2016, he was a Visiting Scholar with the Institute for Infocomm Research, Singapore. He is currently an Associate Professor at the School of Electronic and Information Engineering, SCUT. His research interests include audio signal processing, pattern recognition, and audio surveillance.

XIANKU LI received the B.S. degree in electronic engineering from Yangtze University in 2016. He is currently pursuing the master's degree with the South China University of Technology. His research interests include audio signal processing and audio surveillance.



YUHAN ZHANG received the B.S. degree in electronic engineering from the China University of Mining and Technology in 2017. He is currently pursuing the master's degree with the South China University of Technology. His research interests include audio signal processing and acoustic scene classification.

MINGLE LIU received the B.S. degree in electronic engineering from Hunan Normal University in 2018. He is currently pursuing the master's degree with the South China University of Technology. His research interests include audio signal processing and sound event detection.

WUCHENG WANG received the B.S. degree in electronic and information engineering from Northeastern University in 2018. He is currently pursuing the master's degree with the South China University of Technology. His research interests include audio signal processing and sound event classification.