

Received March 1, 2020, accepted April 12, 2020, date of publication April 20, 2020, date of current version May 6, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2988986

Acoustic-Based Emergency Vehicle Detection Using Convolutional Neural Networks

VAN-THUAN TRAN^{ID} AND WEI-HO TSAI^{ID}, (Member, IEEE)

National Taipei University of Technology, Taipei 10608, Taiwan

Corresponding author: Wei-Ho Tsai (whtsai@ntut.edu.tw)

This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 108-2221-E-027-064.

ABSTRACT This work investigates how to detect emergency vehicles such as ambulances, fire engines, and police cars based on their siren sounds. Recognizing that car drivers may sometimes be unaware of the siren warnings from the emergency vehicles, especially when in-vehicle audio systems are used, we propose to develop an automatic detection system that determines whether there are siren sounds from emergency vehicles nearby to alert other vehicles' drivers to pay attention. A convolutional neural network (CNN)-based ensemble model (SirenNet) with two network streams is designed to classify sounds of traffic soundscape to siren sounds, vehicle horns, and noise, in which the first stream (WaveNet) directly processes raw waveform, and the second one (MLNet) works with a combined feature formed by MFCC (Mel-frequency cepstral coefficients) and log-mel spectrogram. Our experiments conducted on a diverse dataset show that the raw data can complement the MFCC and log-mel features to achieve a promising accuracy of 98.24% in the siren sound detection. In addition, the proposed system can work very well with variable input length. Even for short samples of 0.25 seconds, the system still achieves a high accuracy of 96.89%. The proposed system could be helpful for not only drivers but also autopilot systems.

INDEX TERMS Audio recognition, convolutional neural networks, emergency vehicle detection, siren sounds.

I. INTRODUCTION

Siren is a special signal sounded by alarm systems or emergency service vehicles such as fire trucks, police cars, and ambulances. When an emergency vehicle performs its task, the siren sound is issued to alert other drivers or pedestrians on the road. However, private cars' drivers may sometimes not listen to nearby siren sounds due to the interference of the in-car audio signal, the modern car's soundproofing ability, or even the distraction of drivers themselves. This problem could lead to a delay in providing emergency services or even traffic accidents because of inappropriate communication and cooperation. Thus, this study proposes an acoustic-based method to detect the presence of emergency vehicles on the road. At this stage, we focus on the detection of siren sounds from standard emergency vehicles including ambulances and fire engines, and police cars. In view of the fact that each country may have itself regulation on the types and frequency band of siren sounds, we aim to develop an

emergency vehicle detection system (EVD) that has an excellent ability of generality, which can work stably under diverse siren types/specifications and traffic conditions. For practical applications, we roughly separate the traffic soundscape into three sub-source of sounds, including siren sounds, vehicle horns generated by ordinary vehicles, and noises. Since vehicle horns and background noises are primary sources of the acoustic signal in the downtown street environment, we formulate the EVD problem as distinguishing siren sounds from vehicle horns and noises.

In general, the siren sound is a sub-type of auditory danger signals standardized by the International Organization of Standard (ISO), and ISO 7731 [1] provides rough guidelines for alarm sounds. However, in reality, the regulation and standard for siren sounds are different from country to country; for instance, New Zealand and the USA generally adopt the similar types of wail, yelp, or phaser sirens while England commonly uses the two-tone pneumatic horn. In Taiwan, the frequency of fire trucks' siren sounds continuously changes from low-frequency 650-750Hz to high-frequency 1450-1550Hz, while the siren sounds of

The associate editor coordinating the review of this manuscript and approving it for publication was Xinyu Du^{ID}.

ambulances consist of two alternating tones, the first tone is 650-750Hz and the second tone is 900-1000Hz [2].

Similarly, Japanese law [3], [6] defines a different specification for the ambulance's siren sound, in which the sirens repeat two tones of 960 Hz and 770 Hz, and the repetition period is 1.3s. In Europe [7], the ambulances and fire vehicles use two tones of 392 Hz and 660 Hz, while 466 Hz and 622 Hz are two tones used for police cars. Fig. 1. shows the spectrograms of the three common types of siren sounds, including a Wail siren whose frequency changes continuously (Fig. 1(a)), a Yelp siren which changes the frequency more quickly (Fig. 1(b)), and a two-tone siren of two alternating tones (Fig. 1(c)). Roughly speaking, Wail, Yelp, and two-tone sirens are the most common types of siren sounds used world-wide. In this work, we build the EVD system on the diverse datasets formed by different types of siren sounds and collected from various countries around the world, rather than training the system on the data recorded in a single country like in [4], [5], [11], so that the system can meet the requirement of generality.

The primary method applied to this work is audio recognition based on convolutional neural networks (CNN). In terms of data processing, we can roughly divide the techniques used in audio recognition into two broad categories: the first one generally applies audio feature engineering techniques to extract useful features in time-domain and/or frequency-domain before performing the recognition task, the second one is to take full advantage of deep neural networks to build an end-to-end recognition system which learns features directly from raw waveforms rather than extracting handcrafted features. Each approach has its advantages and success when being applied to different works; however, for acoustics-based EVD problem, almost all works only rely on the first approach with the use of neural networks or shallow learning algorithms such as support vector machine (SVM), Gaussian mixture models (GMM), and k-nearest neighbors (k-NN). In this work, our idea is to apply both approaches and examine if it is possible to boost the system accuracy by aggregating models of these two approaches; in other words, we also examine if the features extracted by the deep neural network itself can complement the handcrafted features in dealing recognition task or not.

The success in developing an acoustic-based EVD system can pave the way for many applications. The first example of its application is providing aid to hearing-impaired people in driving scenarios and even in daily activities. Specifically, in the real-world environment where the background noise is deafening, such a siren sound detection system can alert them for dangerous situations by converting the warning signals to appropriate messages such as text or flashlight. Apart from helping people who lost the hearing, the EVD system is also useful to alert drivers without hearing-problem of approaching emergency vehicles when they are unintentionally unaware of the warning signals. Besides, the automatic detection of emergency vehicles provides more function and a higher level of safety for driverless vehicles. Another

application of the EVD system is that we can integrate it as a part of the intelligent transport system, for instance, integrating the siren detection into smart traffic light controlling system to give priority to direction with the presence of siren sounds by changing the light status and adjusting passing time accordingly.

The significant contribution of this work is that we provide a comprehensive study on acoustic-based EVD using CNNs. We consider vital aspects of a reliable, stable detection system as follows: (1) In terms of experimental data collection, we consider the diversity of the data in order to achieve the generality of the detection system. We collect siren signals of emergency vehicles in real-world traffic from many countries in America, Europe, and Asia instead of considering a single country such as Taiwan [4], Italy [5], or Japan [11]. Partially, this is because of the difference in the specification of siren sounds among countries. Also, we collect data in various scenarios such as different traffic locations and weather conditions, different siren types, and even overlapped sirens. To the best of our knowledge, the collection and consideration of an extensive siren sound dataset captured in many countries, are first introduced in this work; especially the large dataset is collected in real-life environments where include different levels of noise, collection distances, and the Doppler Effect; (2) We propose a 2-dimensional neural network (2D-CNN) model (referred to as MLNet) for EVD based on the combination of the Mel-frequency cepstral coefficients (MFCC) and log-mel spectrogram features. Our experiment results indicate that MLNet yields higher accuracy compared to the related works, which also proves that the aggregated features are beneficial for acoustic-based EVD; (3) We further develop an end-to-end 1-dimensional convolutional neural network (1D-CNN) model (referred to as WaveNet) which automatically learns from raw waveform the useful features for classification, our experiment results also show the promising accuracy obtained with this model; (4) We propose an ensemble architecture of MLNet and WaveNet to boost the detection accuracy and to prove the complementary relationship between the raw features and handcrafted features in acoustic-based EVD; (5) Last but not least, the success of this work is a good fundamental for the applications listed above.

The rest of this paper is organized as follows. In Section II, we introduce the works related to acoustic-based emergency vehicle detection. Section III analyzes the methods we use for classifying the siren sounds, vehicle horns, and noises. Then, we present the experimental results in Section IV and provide a conclusion in Section V.

II. RELATED WORKS

Till now, there are only a few studies on the recognition of siren sounds, such as [4]–[15]. J. Liaw *et al.* proposed to recognize the ambulance siren sound in Taiwan by the Longest Common Subsequence (LCS) [4]. Such an LCS-based system yielded an accuracy of 85% on a small dataset. Another system based on typical speech recognition techniques was

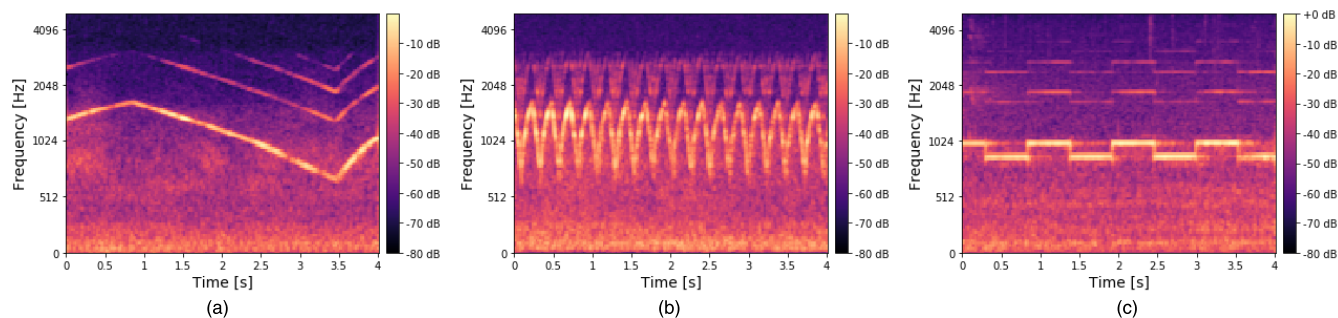


FIGURE 1. Spectrograms of three examples of siren sounds: Wail (a), Yelp (b), and Two-tone siren (c).

introduced in [5], in which MFCC was used together with multi-layer neural networks to detect the siren sounds through the voting method. The system in [5] relatively met the need of low-computational complexity, but it lacked analysis on a diverse dataset, and mainly used the reproduced data. In [6], the detection of alarm sounds was investigated using two approaches including a multi-layer neural network system and a sinusoidal model system, in which the former relied on techniques borrowed from speech recognition, and the latter exploited the structure of alarm sounds and attempted to separate signal from the background to diminish the influence of noise interference. The two systems were tested on a small dataset, and both of them yielded similarly imperfect error rates.

In [7], the authors proposed to employ part-based models (PBMs) in the spectro-temporal domain to detect siren sounds in traffic noise. Their evaluation with self-recorded police sirens and traffic noise collected on-line indicated the potential of applying PBMs to siren-based EVD. The PBMs approach demonstrated better results than hidden Markov models (HMMs) trained on MFCC or log-mel features. Marchegiani and Posner [8] proposed a two-stage approach for acoustic-based EVD in smart vehicles, in which the first stage was to detect the presence of an abnormal sound, and the later stage involved noise reduction and classification. The framework in [8] borrowed the idea from image processing as it analyzed the spectrogram of the incoming signal as an image and employed spectrogram segmentation to isolate and extract the target signal from background noise. It utilized the k-NN classifier on Empirical Binary Masks (EBMs) generated after the noise-reduction step, and yielded the highest accuracy of 83%, which is still far from a requirement for practical applications.

On the other hand, the vehicle classification system in [9] added siren detection as an extra function, and the detection process also heavily relied on the analysis of acoustic signals based on digital signal processing techniques, such as finding the main frequency components in a given frequency band. Since current emergency vehicles produce siren sounds with different specifications, the configuration in [9] could not be flexible to use in general scenarios. An alarm sound detection system based on SVM in combination with feature selection

of handcrafted features was proposed in [10]. It obtained an accuracy of more than 90% on evaluating the system's performance with a small dataset of 35 alarm sound samples and 35 background noise samples. The work was also lacking in evaluating the system's stability and the time cost of the feature engineering process on a massive dataset.

There are several works based on microcontrollers [11]–[13] and hardware design [14], [15] for siren sound detection. In [11], the ambulance's siren sound could be detected by employing two times Fast Fourier Transform (FFT) on a dsPIC microcontroller. Although this detection method could work even under the Doppler Effect, it was computationally expensive. Averagely, it needed 8 seconds to make a single prediction. Meucci *et al.* [12] developed another microcontroller-based system that employed the frequency content and the periodic repetition characteristics of siren sound to implement a pitch detection algorithm suited for EVD. The system was designed and optimized only for two-tone siren of 392 Hz and 660 Hz, and the authors did not evaluate its performance on other siren types or two-tone siren with other parameters yet. A simple algorithm to detect siren sound using the linear prediction model for hearing-impaired drivers was presented in [13], in which the Durbin's recursive algorithm made predictions if the coefficients maintained within a preselected tolerance for a preselected time. Although we can quickly implement this algorithm on Texas Instrument TMS DSPs, it was concluded to be not foolproof and may result in false detection. R. Dobre *et al.* [14], [15] proposed low-computational methods for siren detection based on analog electronics circuits. The authors used SPICE to simulate the initial design [14] and its improved version [15] of the circuit block used for siren detection. The circuits were tested with a siren signal using the SPICE simulator and showed success in detection. However, it lacked the tests on a larger dataset and the evaluation on the real printed circuit board (PCB).

In summary, the prior works on acoustic-based EVD and alarm sound recognition have both advantages and disadvantages. Their limitations include: (1) the limitation of experimental data, in which the authors only recorded a small number of recordings, used simulated data, or collected data in a single country only, this could lead to a

low level of generality for acoustic-based EVD problem; (2) the use of shallow learning algorithms or microcontroller-based approaches results in imperfect performances and the stability of the systems was not thoroughly evaluated yet; (3) modeling audio data only using handcrafted time-domain and/or frequency-domain features, and no work directly employing raw data for recognition; (4) without reporting the efficiency of systems according to different audio durations and the processing time. Recognizing this, we summed up and improved the above limitations by conducting comprehensive experiments on a larger dataset formed by the integration of our self-collected data and available standard datasets (UrbanSound8K [16] and ESC50 [17]). Moreover, inspired by the recent success of neural networks in audio recognition, we propose using convolutional neural networks to handle this task. Primarily, we propose to directly use raw data of audio clips to train the networks.

III. ACOUSTIC-BASED EVD USING CNN

A. CNN FOR AUDIO RECOGNITION

CNN has recently been employed for audio recognition problems successfully, such as in music tagging, environmental/urban sound classification [ESC] [18]–[21], and automatic speech recognition (ASR) [22], [23]. Boddapati *et al.* [18] explored the use of two well-known image recognition networks, Alexnet and GoogLeNet, for classifying environmental sounds. Those networks trained on audio's image-based representations, including spectrogram and MFCC, yielded classification accuracies up to 90%. Works of Salamon Bello [19] and Piczak [20] also showed the possibility of CNN-based ESC. Training with log-mel spectrogram input, models in [19], and [20] attained similar accuracies around 80%. In this work, we focus on using CNN for acoustic-based EVD problems, which employs the approach applied in image classification together with the idea of training CNN with raw audio waveforms.

CNN is partially similar to conventional deep neural networks, but it uses additional layers, namely convolutional layers and pooling layers, instead of only using a series of fully-connected layers. A CNN in classification task contains two major components: feature learning and classification. In the first component, a series of convolutional layers learn appropriate representation or useful features from the input; thus, we can consider this part as the feature extraction stage. In the later part, fully-connected layers play the role of a classifier, which processes the extracted features and assigns the probability to each class for making the prediction.

Generally, during the training phase, a CNN of L layers approximates the relationship between all input-output pairs (\mathbf{x}, \mathbf{y}) of the training dataset. The approximation can be described by Eq. (1), in which the operation of the l^{th} ($l = 1, 2, \dots, L$) convolutional layer is expressed by Eq. (2), where $\mathbf{W}^{(l)}$ is a set of kernels used for extracting useful features from the input, and \otimes indicates the convolutional operation. For the stacked fully connected layers

customarily added at the end of a CNN model, they can be described by Eq. (3), where $\mathbf{W}^{(l)}$ is the weight matrix. In both convolutional layer and fully connected layer, $\mathbf{b}^{(l)}$ and $f^{(l)}$ are respectively the bias vector and the activation function of in the l^{th} layer. At the input layer ($l = 0$): $\mathbf{a}^{(0)} = \mathbf{x}$. Lastly, placing at the end of the model is an output layer that has the number of neurons equal to the number of classes. The optimization problem of the CNN is to minimize the value of loss function computed from the difference between predicted output $\hat{\mathbf{y}}$ and ground-truth \mathbf{y} . Accurately, during the training process, the parameters of CNN are updated and optimized according to the back-propagated prediction error to reach the appropriate minima of the loss function.

$$\mathbf{y} \approx \hat{\mathbf{y}} = g^{(L)}(g^{(L-1)}(\dots(g^{(2)}(g^{(1)}(\mathbf{x})))) \quad (1)$$

$$\mathbf{a}^{(l)} \triangleq g^{(l)}(\mathbf{a}^{(l-1)}) = f^{(l)}(\mathbf{W}^{(l)} \otimes \mathbf{a}^{(l-1)} + \mathbf{b}^{(l)}) \quad (2)$$

$$\mathbf{a}^{(l)} \triangleq g^{(l)}(\mathbf{a}^{(l-1)}) = f^{(l)}(\mathbf{W}^{(l)} \mathbf{a}^{(l-1)} + \mathbf{b}^{(l)}) \quad (3)$$

Intuitively, there are two approaches to organizing audio input for CNN models. Firstly, since CNN originally works with image data, we can organize the audio data by 2D arrays to feed into the 2D-CNN model. With this consideration, we represent audio data by its spectrogram, a representation of the audio frequency spectrum over time, or by MFCC features extracted from sub-frames of an audio file. The second approach is in the case we employ 1-dimensional CNN, in which we should organize the input as 1D arrays. Thus, we can directly feed raw data of the audio signal to the 1D-CNN model, or we can extract handcrafted features from the audio signal and represent them as a 1D arrays before feeding them to the model. In this work, we propose to use raw data of audio waveform as the input of the 1D-CNN model rather than organizing handcrafted features in the 1D format.

B. SirenNet: THE PROPOSED CNN-BASED ENSEMBLE MODEL FOR ACOUSTIC-BASED EVD

With an assumption that the features directly learned from raw waveforms and handcrafted features like MFCCs and log-mel spectrogram may contain different patterns and information of a given sound, we explore the complementary relationship between these kinds of features by building a CNN-based ensemble model based on these two feature inputs and assessing the proposed model's performance in classifying siren sounds, vehicle horns, and noises. Such an ensemble model is called SirenNet, as presented in Fig. 2. The proposed architecture consists of two parts: a 2D-CNN stream and a 1D-CNN stream, which are referred to as MLNet and WaveNet, respectively. The WaveNet works directly with raw waveform, while the MLNet is trained with aggregated features formed by image-based features (MFCC, and log-mel spectrogram). Then, the prediction results (softmax values) derived from MLNet and WaveNet are combined using the averaging method to make the final predictions.

As a necessity for building the network, the length of the input samples must be equal. However, since the collected

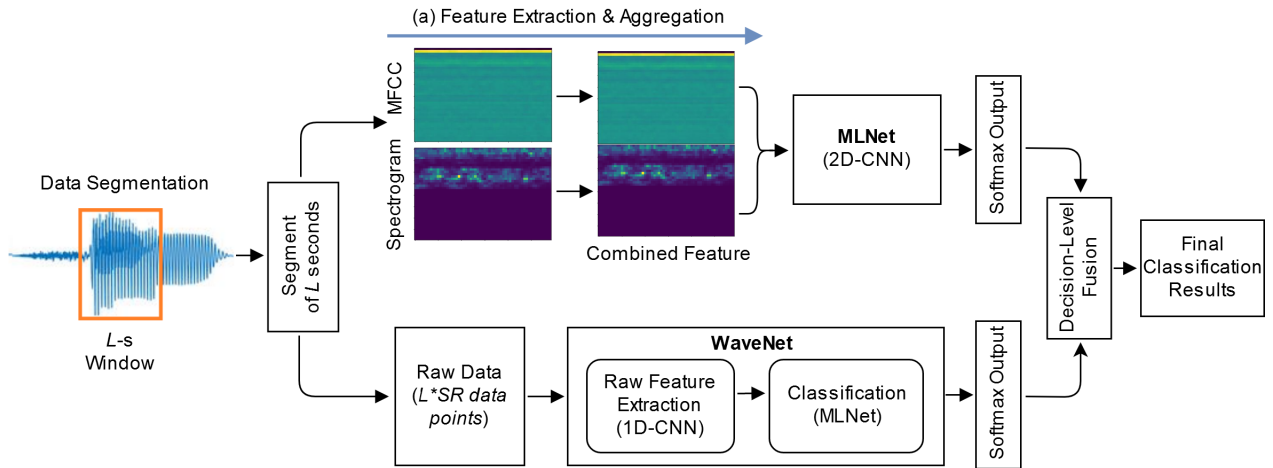


FIGURE 2. The framework of the CNN-based ensemble model (SirenNet) for acoustic-based EVD. SR is the sampling rate.

data have various lengths, we have to split the original audio signal to fixed-length samples before feeding them to the network, as shown in Fig. 2. During the training phase, using a sliding window of size L seconds and stride of $(L - s/2)$, we split the original raw waveform into segments of L seconds and use them as the fixed-length inputs. At each training epoch, we select different segments of the same recording, but the segments have the same label. This process naturally increases the number of training samples so that it also plays the role of sample-level data augmentation. We test the network using the majority voting approach, in which we also split a test sample to various fixed-length segments and then input them to the network and perform majority voting of the predictions corresponding to all input segments to obtain the final prediction result.

Finding a suitable length for the input samples is also a vital factor in training the acoustic-based EVD system because the input duration directly affects the network's computational demand and prediction performance. If the input has too small length, it may not contain sufficient information for classification, resulting in low classification accuracy. On the other hand, if the input length is overly long, it results in a high dimensionality of the input, leading to the higher computational cost for training the model. Thus, this work also investigates the suitable audio length for acoustic-based EVD problems, which partially contributes to building an efficient EVD system.

1) IMAGE RECOGNITION APPROACH WITH 2D-CNN (MLNet) AND AGGREGATED FEATURES

We take the advantages of CNN in image recognition for acoustic-based EVD by employing 2D-CNN to classify audio signals based on their 2D representations consisting of the log-mel spectrogram (log-mel feature) and the MFCC. Many auditory features have been introduced for audio recognition applications [24]. However, we only consider those two well-known features instead of diving into many handcrafted

features. MFCC feature takes into account the nonlinear frequency resolution, which can simulate the hearing characteristics of human ears, while the spectrogram shows the change in frequency components of an audio signal over time. Thus, these two auditory features may provide useful representation for siren recognition. Also, it has been proven by several studies that feature aggregation can help to improve the accuracy in audio recognition, such as in ESC [25], ASR [26]–[28], and breath-based person identification [29]. Inspired by the success of those works, we explore the efficiency of feature combinations in this work. We combine log-mel features and MFCCs of an audio signal into a single feature map before feeding into the 2D-CNN network, as shown in Fig. 2(a). The combination of the two features is conducted linearly.

We use Librosa [30], a python library for audio signal processing, to extract MFCCs and log-mel features. For each data sample, we select fixed-length segments of L seconds. Then, the corresponding MFCCs and log-mel features of those segments are computed. Each data segment is divided into 50% overlapping frames of 23 ms, resulting in 65 frames for a segment of 1.5 seconds and a sampling rate of 22.05 kHz. Next, 40 MFCCs and 64 log-mel features are extracted from each frame. Accordingly, we can represent MFCCs and log-mel features as matrixes of sizes $40 \times 65 \times 1$ and $64 \times 65 \times 1$ corresponding to frequency \times time \times channel, so that the combined feature is a $104 \times 65 \times 1$ matrix. The proposed MLNet is presented in Table 6 and Table 8, and we discuss detail about the design of this model in Section IV.

2) THE END-TO-END SYSTEM WITH 1D-CNN (WaveNet)

Alternatively, we propose to build an end-to-end model (WaveNet) for acoustic-based EVD. The advantage of this model is that it can automatically learn from the raw audio signal discriminative representations, bringing about a promising performance of the EVD system. EnvNet in [31] partially inspires the design of the proposed WaveNet, and Table 8 presents the architecture of WaveNet, indicating that

this model has two parts consisting of raw feature extraction with a series of 1-dimensional convolutional layers (1D Conv-layers), and the remaining part is for classification. For the 1D Conv-layers in the former part, $\mathbf{W}^{(l)}$ in Eq. (2) is a set of 1D kernels employed for transforming raw waveforms into useful features, which are also the input of the network's later part. Conceptually, the first 1D Conv-layer is responsible for catching the global view of the raw waveform and extract the local features, while the remaining 1D Conv-layers play the role of getting a more in-depth view of the data to find the useful discriminative features for the classification task.

For the classification part, our idea is to use the optimized MLNet as the classification part of WaveNet because we regard and process the features extracted by 1D Conv-layers as time-frequency representation such as log-mel spectrogram. In doing so, we set the number of filters of each 1D Conv-layer to 64, which are conceptually the same as the number of components of the log-mel spectrogram. More specifically, in our assumption, each filter is associated with a frequency characteristic. We process the output of the last 1D Conv-layer with a non-overlapping max-pooling layer of size 220, which is approximately equivalent to 10ms when the sampling rate is 22.05 kHz, resulting in an output matrix of size $64 \times 150 \times 1$ corresponding to frequency \times time \times channel if the input segment is 1.5 seconds. From this point forward, WaveNet works similarly to MLNet trained with MFCC and/or log-mel features. We provide the configuration of WaveNet in section IV.

IV. EXPERIMENTS AND RESULTS

A. EXPERIMENTAL DATA

Our experimental dataset contains three sound classes of the siren, vehicle horn, and noise. In collecting the dataset, we set several goals and requirements. Firstly, all data must be real-field recordings captured in road traffic or urban environments, rather than using simulated data. Secondly, the dataset must meet the need for quantity and diversity in building deep neural networks, so it should be adequately vast and varied in terms of sound's specification and recording conditions. To reach those two primary goals, we turn to four collection approaches: (1) we collect data from on-line resources which professionally provide audio/video clips of emergency vehicles, (2) integrating with published datasets which contain siren sounds and others related to the urban soundscape, (3) directly capturing data while driving in Taiwan, (4) reproducing the sound class which is hard to collect a large amount.

For the first approach, thanks to the availability of videos on two Youtube channels including "TGG - Global Emergency Responses" [32] and "[rescue911.de] - Worldwide Emergency Responses" [33], which provide a large number of videos about emergency responses recorded all over the world, we have access to a diverse database of siren sounds of police cars, fire engines, fire ladder cars, ambulance cars, and police motorcycles. Notably, the channels' owners have

TABLE 1. Summary of our experimental data.

Data Class	Data Sources			
	Our Collection	UrbanSound8K [16]	ESC-50 [17]	Total (no. samples)
Siren Sounds	7,773	929	40	8,742
Car Horns	7,083	429	40	7,552
Urban Noise	1,087	7,374	1,920	10,381
Total (no. sample)	15,943	8,732	2000	26,675
Total duration	17.7 hours	9.7 hours	2.8 hours	30.2 hours
Length of each clip	4 seconds	1-4 seconds	5 seconds	-
Sampling rate	44.1 kHz	8-192 kHz	44.1 kHz	-

captured videos from many areas, including in America (the USA, Canada, Cuba), in Europe (England, Germany, Scotland, Netherlands, Belgium), and Asia (Vietnam, Taiwan, Hong Kong, Singapore, Korea, Japan, and China). Besides, various siren types such as wail, yelp, and phaser horn are all included, and they can be operated solely or simultaneously. Also, they recorded data both inside and outside the cars traveling at various speeds so that the collected data also included Doppler Effect. For the second collection approach, we integrated our collected data with two datasets published in [16], [17], which respectively contain 8732 and 2000 environmental/urban sound clips. Next, we captured real siren, horns, and road noise in Taiwan's streets. Lastly, for vehicle horns, because the amount of this class is less than its two counterparts, we augmented the data by reproducing different recordings of horns and recording them at different scenarios, involving various distances and noise levels.

To organize the experimental data, we randomly stored original recordings to 5 folds, in which each fold had a relatively equal amount of audio length across all three classes. We carefully conducted this five-fold separation to avoid the problem of overfitting when building the system using k-fold cross-validation. This is because the recordings for training are entirely different from those for testing. For the data extracted from video clips [32], [33] and our real-field recordings, we split them into smaller non-overlapping 4-s clips, resulting in 15,943 data samples, as shown in Table 1. By integrating with data of [16], [17] we end up with a dataset of 26,675 samples. Although [16], [17] contain a small data of horns and siren sounds, they provide the diversity for the noise class, many useful subsets of urban noise such as drilling, engine idling, jackhammer, street music [16], natural soundscapes, human-non-speech sounds, and exterior/urban noises [17] are available. On the other hand, our collected data provides a large number of horns and siren clips. From Table 1, it is clear that our collected data complement the data from [16], [17] to create a relatively sizeable balanced dataset.

B. EXPERIMENT SETUP

The experimental data was integrated from different sources, resulting in different properties among data samples, such as

various sampling rates, one or two channels, and coding in different bit-depth. Thus, the preprocessing stage is required to standardize our data so that it benefits the experimental process. If an audio sample is stereophonic, we converted it into a monophonic signal by taking the average values of the signal amplitudes of each channel. Also, we resampled all recordings at a specific sampling rate. The resampling, monophonic conversion, and the extraction of spectrograms and MFCCs were conducted using Librosa [30].

In our experiments, we utilized a desktop PC built with 16 GB RAM, an Intel Core i7-9700K CPU (8 cores @3.60 GHz), and NVIDIA GeForce RTX 2080 Ti GPU. The PC was running Ubuntu 18.04.2 LTS, and we used the TensorFlow deep learning framework to implement the network designs. The baseline setup for feature extraction parameters and the network hyper-parameters are listed as follows: sampling rate is 22.05 kHz; frame length of 23ms; the percentage of frame overlap is 50%; 50 training epochs; the initial learning rate is 0.001; and we trained the models with Adam optimizer, a variant of Stochastic Gradient Descent [34]. We evaluated the proposed models using the k -fold cross-validation scheme and used the classification accuracy as the primary evaluation metric.

We conducted several experiments and reported the results in the next sections. We carried out an initial experiment to find appropriate parameters for later experiments. Then, we show the results of the proposed MLNet, WaveNet, and SirenNet. Also, we provide an analysis of experimental results and a comparison between this work and the prior works.

C. INITIAL EXPERIMENT

This initial experiment was a pilot investigation on the performance of the end-to-end architecture, the efficiency of features aggregation, and suitable parameters, including input duration and sampling rate for the acoustic-based EVD. Consequently, we used the best parameters obtained from this experiment as the standard setup for the later experiments. Respectively, we call the WaveNet and MLNet with initial configurations as init-WaveNet and init-MLNet. We compared the accuracies of those networks according to different input lengths, including 0.25s, 0.5s, 1s, 1.5s, 2s, and 3s, to find the suitable one for classification. We did not consider the longer durations such as 4s, 5s for three main reasons: (1) in terms of real-life EVD application it is not practical to use such a long input because it obviously leads to prolonged response; (2) almost all samples of our experimental data are shorter than 4s, especially data from [16], the use of padding technique when building the model may cause the decrease in the accuracy, which has been shown in [31]; (3) it is computationally expensive as it took too much time when we tried to train a WaveNet with input duration of 5s. In this experiment, we also considered different sampling rates (SR), in which the candidates for SR were 22,050 Hz, 16,000 Hz, and 8,000 Hz because these values of SR were commonly used in the field of audio recognition.

TABLE 2. The configuration of the init-WaveNet ($SR = 22.05$ kHz, and $L - s = 1.5s$).

Layer	Kernel size	Stride	Number of filters	Output shape (channel-last)
Input (<i>Raw data</i>)	-	-	-	(1, 33075, 1)
1D Conv1	(1, 8)	(1, 1)	64	(1, 33075, 64)
1D Conv2	(1, 8)	(1, 1)	64	(1, 33075, 64)
MaxPool1	(1, 220)	(1, 1)	-	(1, 150, 64)
Reshape	-	-	-	(1, 64, 150, 1)
2D Conv3	(4, 4)	(2, 2)	32	(1, 64, 150, 32)
2D Conv4	(4, 4)	(2, 2)	32	(1, 64, 150, 32)
MaxPool2	(2, 2)	(2, 2)	-	(1, 32, 75, 32)
Flatten	-	-	-	(1, 76800)
Fc1	-	-	512	(1, 512)
Fc2	-	-	64	(1, 64)
Output(#classes)	-	-	3	(1, 3)

TABLE 3. The configuration of the init-MLNet ($SR = 22.05$ kHz, $L - s = 1.5s$, and input is the combined feature).

Layer	Kernel size	Stride	Number of filters	Output shape (channel-last)
Input (<i>MFCC+log-mel</i>)	-	-	-	(1, 104, 65, 1)
2D Conv1	(4, 4)	(2, 2)	32	(1, 104, 65, 32)
2D Conv2	(4, 4)	(2, 2)	32	(1, 104, 65, 32)
MaxPool1	(2, 2)	(2, 2)	-	(1, 52, 32, 32)
Flatten	-	-	-	(1, 53248)
Fc1	-	-	512	(1, 512)
Fc2	-	-	64	(1, 64)
Output(#classes)	-	-	3	(1, 3)

For the initial configuration of WaveNet, we used two 1D Conv-layers with a filter size of 8 and stride of 1 for the raw feature extraction part, followed by two 2D Conv-layers with 4×4 receptive field and stride of 2×2 , and two fully-connected layers in the classification part. The detailed configuration of the init-WaveNet is presented in Table 2. As we analyzed in section III.B.2, we considered the features extracted by 1D Conv-layers as time-frequency representation, so we reshaped and presented the output of those layers as (frequency, time, channel). Note that the network configuration and data processing technique were fixed for the whole initial experiment regardless of the change in sampling rate and input duration. The configuration of the init-MLNet is shown in Table 3, in which we simplified the network with two 2D convolutional layers and two fully-connected layers. The output shape shown in Table 2 and Table 3 is in the case we apply the sampling rate of 22.05 kHz, and the input duration of 1.5 seconds.

The mean accuracies of the init-WaveNet with respect to different input durations and sampling rates are listed in Table 4. We performed 5-fold cross-validation in all experiments, so we provide the results with the average accuracy (%) and the standard deviation. From Table 4, we can see that with each input length of 1s, 1.5s, 2s, and 3s, the results of init-WaveNet for SR of 22.05 kHz are slightly higher than that of the two remaining sampling rates. However, for the shorter input durations of 0.25s and 0.5s, the init-WaveNet working

TABLE 4. Mean Accuracies (in %) of the init-WaveNet according to different sampling rates and input lengths (The values inside () are the standard deviations among the accuracies for 5-fold cross-validation).

Sampling rate	Input length (s)					
	0.25	0.5	1	1.5	2	3
22,050 Hz	91.76 (0.10)	92.19 (0.54)	92.89 (0.11)	93.99 (0.26)	94.70 (0.28)	94.79 (0.25)
16,000 Hz	88.95 (0.21)	90.84 (0.38)	92.31 (0.33)	93.89 (0.31)	94.41 (0.16)	94.69 (0.12)
8,000 Hz	89.20 (0.31)	90.71 (0.13)	92.87 (0.29)	93.91 (0.36)	94.73 (0.10)	94.28 (0.30)

TABLE 5. Mean Accuracies of the init-MLNet and the init-WaveNet according to different input lengths in case SR of 22.05 kHz.

Model (Feature)	Input length (s)					
	0.25	0.5	1	1.5	2	3
init-MLNet (log-mel)	78.58 (1.42)	80.54 (1.16)	85.24 (1.72)	91.18 (0.39)	92.15 (0.42)	92.56 (1.41)
init-MLNet (MFCC)	87.30 (0.54)	88.78 (0.63)	88.88 (0.77)	89.71 (0.67)	89.55 (0.98)	90.67 (0.61)
init-MLNet (MFCC+log-mel)	91.21 (0.55)	92.07 (0.53)	92.80 (0.49)	94.26 (0.51)	94.17 (0.36)	94.48 (0.96)
init-WaveNet (Raw data)	91.76 (0.10)	92.19 (0.54)	92.89 (0.11)	93.99 (0.26)	94.70 (0.28)	94.79 (0.25)

with the SR of 22.05 kHz yielded much better performance compared to the results of experiments on the SR of 16 kHz and 8 kHz, by approximately 2.5% and 1.5% respectively. Thus, we decided to choose 22.05 kHz as the default sampling rate for the later experiments. In the following experiments, we tested MLNet trained on MFCC and/or log-mel features with the SR of 22.05 kHz. The experiment results are summarized in Table 5.

For the input duration, as shown by statistics in Table 4 and Table 5, the accuracy generally tends to improve proportionally according to the increase in the length of the input waveform. This is true for all three values of sampling rates and both init-WaveNet and init-MLNet. We assume that the longer the original raw input, the more useful information provided to the network, resulting in better performance. However, the accuracy is at a high level when the input length ranges from 1.5s to 3s, and there is no significant gap in accuracy among this range. From this point of view, we choose the input length of 1.5s for system development due to the following reasons: (1) it still yields comparable accuracies compared with results of much longer input durations (2s and 3s); (2) we assume that such a duration is sufficient for representing the characteristics of siren sounds, especially two-tone siren and yelp siren which normally have the cycle of around 1s so that a sample of 1.5s can provide enough information for classifiers.

Results in Table 5 also show the efficiency of feature aggregation for init-MLNet; in other words, MFCC can complement the log-mel feature in acoustic-based EVD. Specifically, init-MLNet trained on the aggregated feature (MFCC+log-mel) yielded much better results than that of this model trained on a single feature, MFCC, or log-mel

feature. For example, at the input length of 1.5s, the init-MLNet (MFCC+log-mel) reached an accuracy of 94.26%, which is much higher than that of init-MLNet (log-mel) and init-MLNet (MFCC), by 3.08% and 4.55%, respectively. Last but not least, the initial experiment also presents the potential of WaveNet. Across all input lengths, the performance of init-WaveNet was better than that of init-MLNet trained on MFCC or log-mel features, in which init-WaveNet yielded higher average accuracies and smaller standard deviations compared to that of init-MLNet, as shown in Table 5. Considering the input length of 1.5s, init-WaveNet (raw data) yields an accuracy of 93.99%, which is respectively 2.81% and 4.28% higher than the results of MLNet (log-mel) and MLNet (MFCC).

D. RESULTS AND ANALYSIS

1) THE PROPOSED MLNet

It is essential to find a suitable number of convolutional layers and appropriate parameters for a network to maximize its performance. Thus, we conducted a series of experiments to investigate the influence of different numbers of 2D convolutional layers on the performance of the MLNet. As a result, we can choose a suitable architecture for the proposed MLNet applied to acoustic-based EVD. We consider the number of convolutional layers up to 6 because, with this configuration, the output of the last convolutional layer already reaches a small size. Also, since the amount of data for training is not large enough, the use of deeper architectures may lead to significant overfitting. Note that the standard configuration to all network versions is that we apply batch normalization to each layer to speed up the computational process, and we set a dropout of 0.5 for the fully-connected layers in order to avoid overfitting. In all convolutional layers, we use the receptive field of 4×4 , and we set the stride step to 2×2 . Table 6 shows the list of layers, memory cost, and the number of parameters of MLNet models with 2, 3, 4, 5, 6 convolutional layers, respectively.

Table 7 presents the experimental results of models with respect to different numbers of convolutional layers. It can be seen from Table 7 that MLNet working with aggregated feature (MFCC+log-mel) yields the highest average accuracy of 96.42% when it is configured with four convolutional layers, this result is higher than that of models with 2, 3, 5, and 6 convolutional layers by 2.16%, 1.73%, 1.33%, and 1.81%, respectively. It also indicates that using deeper architectures does not result in better performance. As a result of this investigation, we design our proposed MLNet with four convolutional layers and two fully-connected layers, as presented in Table 6. Moreover, in terms of memory cost, it requires a small amount of memory (653 K) to train the proposed MLNet.

2) THE PROPOSED WaveNet

Recall that the WaveNet has two parts, including a part for raw feature extraction using 1D-CNN and the other part for

TABLE 6. Parameters and memory cost of the MLNet with the different numbers of 2D convolutional layers.

Layer	2 Conv-layers	3 Conv-layers	4 Conv-layers	5 Conv-layers	6 Conv-layers
	#Parameters (memory)	#Parameters (memory)	#Parameters (memory)	#Parameters (memory)	#Parameters (memory)
Input (104, 65, 1)	0 (6.7 K)	0 (6.7 K)	0 (6.7 K)	0 (6.7 K)	0 (6.7 K)
Conv 4×4 – 32	544 (216.3 K)	544 (216.3 K)	544 (216.3 K)	544 (216.3 K)	544 (216.3 K)
Conv 4×4 – 32	16.4 K (216.3 K)	16.4 K (216.3 K)	16.4 K (216.3 K)	16.4 K (216.3 K)	16.4 K (216.3 K)
Conv 4×4 – 64	-	32.8K (106.4 K)	32.8K (106.4 K)	32.8K (106.4 K)	32.8K (106.4 K)
Conv 4×4 – 64	-	-	65.6 K (106.4 K)	65.6 K (106.4 K)	65.6 K (106.4 K)
Conv 4×4 – 128	-	-	-	131 K (53.2 K)	131 K (53.2 K)
Conv 4×4 – 128	-	-	-	-	262 K (53.2 K)
Fc-512	27.2 M (512)	13.6 M (512)	13.63 M (512)	6.8 M (512)	6.8 M (512)
Fc-64	32.8 K (64)	32.8 K (64)	32.8 K (64)	32.8 K (64)	32 K (64)
Output-3	195 (3)	195 (3)	195 (3)	195 (3)	195 (3)
Total parameters	27.3 M	13.7 M	13.8 M	7.1 M	7.3 M
Total memory	440 K	546 K	653 K	705 K	759 K

TABLE 7. The results of the MLNet configured with different numbers of convolutional layers. (SR=22.05 kHz, L – s = 1.5s).

Models (Aggregated feature)	Accuracy (std. deviation)
MLNet with 2 Conv-layers	94.26 (0.51)
MLNet with 3 Conv-layers	94.69 (0.48)
MLNet with 4 Conv-layers	96.42 (0.45)
MLNet with 5 Conv-layers	95.09 (0.32)
MLNet with 6 Conv-layers	94.61 (0.85)

TABLE 8. The configuration of the WaveNet, including MLNet used as the classification part. (SR = 22.05 kHz, and L – s = 1.5s).

	Layer	Kernel size	Stride	Number of filters	Output shape (channel-last)
Raw feature extraction	Input (Raw data)	-	-	-	(1, 33075, 1)
	1D Conv1	(1, 64)	(1, 1)	64	(1, 33075, 64)
	1D Conv2	(1, 64)	(1, 1)	64	(1, 33075, 64)
	MaxPool1	(1, 220)	(1, 1)	-	(1, 150, 64)
	Reshape	-	-	-	(1, 64, 150, 1)
Classification (MLNet)	2D Conv3	(4, 4)	(2, 2)	32	(1, 64, 150, 32)
	2D Conv4	(4, 4)	(2, 2)	32	(1, 64, 150, 32)
	MaxPool2	(2, 2)	(2, 2)	-	(1, 32, 75, 32)
	2D Conv5	(4, 4)	(2, 2)	64	(1, 32, 75, 64)
	2D Conv6	(4, 4)	(2, 2)	64	(1, 32, 75, 64)
	MaxPool3	(2, 2)	(2, 2)	-	(1, 16, 37, 64)
	Flatten	-	-	-	(1, 37888)
	Fc1	-	-	512	(1, 512)
	Fc2	-	-	64	(1, 64)
	Output(#classes)	-	-	3	(1, 3)

classification based on 2D-CNN, as shown in Fig. 2. Since we process the feature extracted from raw waveforms as time-frequency representation, we use the proposed MLNet as a classification part of WaveNet directly. Consequently, in this experiment, we aim to find a suitable configuration for the feature extraction part of WaveNet, involving finding a suitable number of 1D Conv-layers and the appropriate filter size. We conducted experiments with the data of 1.5s length sampled at 22.05 kHz, and the networks were configured with 1, 2, 3, and 4 1D Conv-layers in the raw feature extraction part, respectively. Besides, the filter sizes of 4, 6, 8, 16, 32,

TABLE 9. The Accuracies of the WaveNet with different numbers of 1D Conv-layers. (SR=22.05 kHz, L – s = 1.5s).

Filter size	Number of 1D convolutional layers			
	1 Conv-layer	2 Conv-layers	3 Conv-layers	4 Conv-layers
4	84.58 (4.17)	90.47 (2.46)	93.78 (0.45)	93.51 (0.68)
6	85.80 (2.30)	92.53 (1.87)	93.82 (0.74)	94.40 (0.37)
8	89.20 (0.97)	94.41 (0.68)	94.64 (0.78)	94.12 (1.05)
16	91.74 (0.65)	94.73 (0.54)	95.03 (0.26)	94.83(0.75)
32	92.20 (1.69)	95.37 (0.95)	95.95 (0.50)	95.48 (0.33)
64	94.88 (0.61)	96.51 (0.31)	95.47 (0.27)	95.38 (0.53)
128	95.35 (0.41)	95.50 (0.53)	94.57 (0.46)	93.81 (0.39)
256	94.49 (0.27)	94.67 (0.37)	90.96 (0.65)	90.59 (0.53)

64, 128, and 256 were separately applied to each model. The number of 1D Conv-layers was limited to 4 also to avoid overfitting when training models with our moderate dataset. At each layer, we used 64 filters and stride step of 1. We set the pooling size of the max-pooling layer used at the last 1D Conv-layer to 220, and hence a feature map of size 64×150×1 corresponding to time frequency ×time×channel is created.

We summarize the results of this investigation in Table 9. A common trend from Table 9 is that across different numbers of 1D Conv-layers, the classification accuracies are improved when the filter size is increased up to a specific value. However, when the number of convolutional layers becomes more extensive, the models tend to reach the highest accuracy with the smaller filter sizes, and the accuracies start to drop when the models have significantly large filter sizes, of 128, 64, 32, and 32 respectively for models with 1, 2, 3, and 4 convolutional layers.

Another vital observation is that WaveNet configured with more 1D Conv-layers performs better than the network with a single-convolutional layer. More specifically, in the case of one layer, we achieve the highest accuracy of 95.35% when the filter size is 128. For the case of two and three layers, the accuracy reaches the highest values when the filter size is 64

(96.51%) and 32 (95.95%), respectively. However, the deeper model with four Conv-layers yields lower accuracies compared to that of 2 and 3 Conv-layer models. This indicates that the convolutional operations of two and three Conv-layer models respectively with filter sizes of 64 and 32, can adequately extract the most useful features for classification, and when the model is deeper (four 1D Conv-layer), it starts to overfit.

Accordingly, we decided to configure the raw feature extraction part of the proposed WaveNet with two convolutional layers and the filter size of 64, as shown in Table 8, and the remaining part of WaveNet is the proposed MLNet. Notably, with the same configuration of two 1D Conv-layers and the filter size of 8, the model using MLNet for classification yields a higher accuracy of 94.41% by 0.42% compared to the result (93.99%) of the model with the initial configuration (init-WaveNet). This fact indicates that the idea of processing features extracted by 1D-CNN as time-frequency format and classify them by MLNet is practically efficient.

3) RESULTS OF THE PROPOSED ENSEMBLE MODEL (SirenNet) AND ANALYSIS

Next, we evaluated the performance of the proposed SirenNet (Fig. 2) to prove the complementary relationship between the raw features and handcrafted features, including MFCC and log-mel spectrogram. For every sample, we evaluated the predictions of each network stream (MLNet and WaveNet). After that, the softmax outputs from these two streams are averaged, and the final classification was performed according to the maximum value of the averaged softmax output. We summarize the results of SirenNet in Table 10 and a confusion matrix in Table 11, in which Table 10 also provides information about the model's loading time and inference time. From the statistics of Table 10, it is clear that SirenNet achieved a promising accuracy of 98.24%, which is higher than the results of WaveNet and MLNet by 1.75% and 1.85%, separately. This result indicates that raw features learned by WaveNet have the capability of complementing MFCC and log-mel features in acoustic-based emergency vehicle detection. Also, with such a high accuracy, the SirenNet is auspiciously applied to real-world applications, which is one of the significant contributions of this work. For the utilization time, we can see that the time cost of MLNet (11 ms) and WaveNet (14 ms) are almost comparable and acceptably low. In the case of ensemble architecture, although the time cost of this model (27 ms) almost doubles that of the single networks, the inference time of SirenNet is still short enough for real-time operation of the EVD system.

From the confusion matrix in Table 11, we can see the detail about the rates of correct prediction and misclassification across three sound classes. The misclassification rates between siren sounds and vehicle horns are meager, which is 0.44%, and 0.11%, respectively. This result shows the advantage of SirenNet since there is little probability of predicting the sound of ordinary vehicles as siren sounds of emergency vehicles and vice versa. Meanwhile, the significant

TABLE 10. Results of the SirenNet and comparison to single networks.

Model	Accuracy (%)	Model loading time (s)	Inference time (s)
SirenNet (Raw data, MFCC+log-mel)	98.24 (0.36)	0.805	0.027
WaveNet (Raw data)	96.51(0.31)	0.389	0.011
MLNet (MFCC+log-mel)	96.42 (0.45)	0.415	0.014

TABLE 11. Normalized confusion matrix obtained by testing the SirenNet on the dataset of 1.5s length.

True Class	Predicted Class		
	Sirens	Horns	Noise
Sirens	0.9854	0.0011	0.0135
Horns	0.0044	0.9789	0.0167
Noise	0.0104	0.0074	0.9822

TABLE 12. Results of the proposed models on different input durations.

Model	Input Length (s)			
	1.5	1	0.5	0.25
SirenNet (Raw data, MFCC+log-mel)	98.24 (0.36)	97.74 (0.79)	97.42 (0.39)	96.89 (0.57)
WaveNet (Raw data)	96.51 (0.31)	95.59 (0.32)	94.43 (0.51)	92.20 (0.65)
MLNet (MFCC+log-mel)	96.42 (0.45)	95.49 (0.16)	95.37 (0.46)	94.47 (0.59)

misclassifications were made between sirens or horns with noise, in which 1.35% of sirens and 1.67% of horns are predicted as noise, this may be due to the heavy noise in the recordings of sirens and horns.

We also evaluated the efficiency of the proposed SirenNet, WaveNet, and MLNet when they worked with data samples shorter than 1.5s. The results of this evaluation are listed in Table 12. Notably, the proposed models show excellent performances, even working with short input durations of 1s, 0.5s, and 0.25s. Across those three cases of input length, the optimized WaveNet and MLNet yielded much higher accuracies compared to the results of networks with *initial* configurations and trained with longer input length of 1.5s; this also results in high accuracies of the ensemble model (SirenNet). The results of SirenNet are 97.74%, 97.42%, and 96.89% respectively for input lengths of 1s, 0.5s, and 0.25s. It is clear that even with concise recordings such as 0.25 second where the performances of single networks start to degrade significantly, 92.20 % for WaveNet and 94.47% for MLNet (MFCC+log-mel), the SirenNet still yields a high accuracy of 96.98%, only 1.35% lower than the result of the experiment on the data of six times longer (1.5s). This result further confirms the efficiency of the proposed SirenNet.

TABLE 13. Comparisons of the classification accuracies obtained with other systems.

Work	Feature(s)	Model/Method	Accuracy (%)
L. Marchegiani et al. [8], 2017	Empirical Binary Masks (EBMs), Gammatonegrams (GTG), MFCC	k-NN	83.00 ± 1.30 (EBMs), 75.60 ± 1.14 (GTG), 62.20 ± 2.77 (MFCC)
J. Schroder et al. [7], 2013	MFCC, Spectrogram	Part-based Models (PBM), HMM	86.00 (PBMs), 80.00 (HMM+MFCC), 74.00 (HMM+Spectrogram)
J.J. Liaw et al. [4], 2013	Longest Common Subsequence (LCS)	LCS Comparison	85.00
This work	Raw data	1D-CNN (WaveNet)	96.51
This work	MFCC+Spectrogram	2D-CNN (MLNet)	96.42
This work	Aggregated features: Raw data, MFCC, Spectrogram	CNN (SirenNet)	98.24

At last, we compared the results of our work with that of several existing related works listed in the literature review section. Some works based on microcontrollers [11]–[13] and circuit design [14], [15] only reported the possibilities of siren detection systems, and they did not evaluate the system's accuracy on an extensive dataset. Thus, we only focused on the comparison with works based on machine learning or deep learning approaches. Table 13 shows the comparison between our works and prior works [4], [7], [8] based on the mean accuracies of sound source classification across different classifiers and features. In terms of feature for classification, to the best of our knowledge, this work is the first one investigating the use of raw waveform for acoustic-based EVD. Equally important, the proposed WaveNet yielded promising classification accuracy of 96.51% which is better than the results of related works including 83%, 86%, and 85% respectively in [4], [7], and [8]. For the use of handcrafted features (MFCC, and log-mel spectrogram), we can see that our proposal of aggregating MFCC and the log-mel feature is also useful, as the result of MLNet (96.42%) is much higher than the result of [4], [7], and [8]. Specifically, k-NN(MFCC) in [8] yielded an average accuracy of 62.20%, while the results of HMM(MFCC) and HMM(Spectrogram) in [7] were 80% and 74%, respectively. Finally, the proposed ensemble model (SirenNet) achieved the highest accuracy (98.24%) among all models.

V. CONCLUSION AND FEATURE WORK

In this work, we introduced a deep-learning model (SirenNet) based on convolutional neural networks for siren-sound-based emergency vehicle detection. The proposed SirenNet is composed of two single CNN-based networks,

including an end-to-end network (the WaveNet), which works with raw waveform input, and the MLNet trained on well-known handcrafted features (MFCC, and log-mel spectrogram). We conducted all experiments on an extensive data set, including our collection of 17.7 hours of recordings and 12.5 hours of recordings from Urbansound8k and ESC-50 datasets. The use of an end-to-end architecture and the combination of MFCC and log-mel features for acoustic-based EVD are first investigated in this work, and those schemes brought about promising results, in which the WaveNet and MLNet respectively yielded accuracies of 96.51% and 96.42%. The ensemble architecture (SirenNet) further boosted the classification accuracy to reach the highest value of 98.24%. Those experimental results showed the efficiency of our proposed models and proved the complementary relationship between features automatically extracted from raw waveforms and handcrafted features. Also, the SirenNet requires a short inference time of 27 ms, which is well acceptable for real-time detection. The results of this work lay a good foundation for future development and applications.

Although we have achieved promising results in this work, future work is still needed to improve the detection performance and to meet the need for reliable and convenient emergency vehicle detection systems. For example, the primary focus in our future work could be the localization of siren sources so that the detection system could also provide information about the direction of the emergency vehicles to drivers.

REFERENCES

- [1] *Ergonomics-Danger Signals are Further Subdivided and Work Areas-Auditory Danger Signals*, Standard 7731, International Organization for Standardization, 2013.
- [2] *Taiwan National Fire Agency, Ministry of the Interior*. Accessed: Dec. 2019. [Online]. Available: <http://www.nfa.gov.tw>
- [3] *Overview of the Electronic Siren Prepare for Ambulances*, Standard 337, Fire and Disaster Management Agency, 1970.
- [4] J.-J. Liaw, W.-S. Wang, H.-C. Chu, M.-S. Huang, and C.-P. Lu, "Recognition of the ambulance siren sound in taiwan by the longest common subsequence," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2013, pp. 3825–3828.
- [5] F. Beritelli, S. Casale, A. Russo, and S. Serrano, "An automatic emergency signal recognition system for the hearing impaired," in *Proc. IEEE 12th Digit. Signal Process. Workshop, 4th IEEE Signal Process. Educ. Workshop*, Sep. 2006, pp. 179–182.
- [6] P. W. Daniel Ellis, "Detecting alarm sounds," in *Proc. Recognit. Real-World Sounds, Workshop Consistent Reliable Acoustic Cues*, Aalborg, Denmark, 2001, pp. 59–62.
- [7] J. Schroder, S. Goetze, V. Grutzmacher, and J. Anemuller, "Automatic acoustic siren detection in traffic noise by part-based models," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2013, pp. 493–497.
- [8] L. Marchegiani and I. Posner, "Leveraging the urban soundscape: Auditory perception for smart vehicles," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Singapore, May 2017, pp. 6547–6554.
- [9] O. Karpis, "System for vehicles classification and emergency vehicles detection," *IFAC Proc. Volumes*, vol. 45, no. 7, pp. 186–190, 2012.
- [10] D. Carmel, A. Yeshurun, and Y. Moshe, "Detection of alarm sounds in noisy environments," in *Proc. 25th Eur. Signal Process. Conf. (EUSIPCO)*, Kos, Greece, Aug. 2017, pp. 1839–1843.
- [11] T. Miyazaki, Y. Kitazono, and M. Shimakawa, "Ambulance siren detector using FFT on dsPIC," in *Proc. 1st IEEE/IIAE Int. Conf. Intell. Syst. Image Process.*, Sep. 2013, pp. 266–269.

- [12] F. Meucci, L. Pierucci, E. Del Re, L. Lastrucci, and P. Desii, "A real-time siren detector to improve safety of guide in traffic environment," in *Proc. 16th Eur. Signal Process. Conf. (EUSIPCO)*, 2008, pp. 25–29.
- [13] S. W. Park and J. Trevino, "Automatic detection of emergency vehicles for hearing impaired drivers," M.S. thesis, Dept. Elect. Eng. Comput. Sci., Texas A&M Univ.-Kingsville, Kingsville, TX, USA, 2013, vol. 192, pp. 637–665.
- [14] R. A. Dobre, V. A. Nita, A. Ciobanu, C. Negrescu, and D. Stanomir, "Low computational method for siren detection," in *Proc. IEEE 21st Int. Symp. Design Technol. Electron. Packag. (SIITME)*, Braşov, Romania, Oct. 2015, pp. 291–295.
- [15] R. A. Dobre, C. Negrescu, and D. Stanomir, "Improved low computational method for siren detection," in *Proc. IEEE 23rd Int. Symp. Design Technol. Electron. Packag. (SIITME)*, Oct. 2017, pp. 318–323.
- [16] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 1041–1044.
- [17] K. J. Piczak, "ESC: Dataset for environmental sound classification," in *Proc. 23rd ACM Int. Conf. Multimedia*, 2015, pp. 1015–1018.
- [18] V. Boddapati, A. Petef, J. Rasmusson, and L. Lundberg, "Classifying environmental sounds using image recognition networks," *Procedia Comput. Sci.*, vol. 112, pp. 2048–2056, Jan. 2017.
- [19] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 279–283, Mar. 2017.
- [20] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *Proc. IEEE 25th Int. Workshop Mach. Learn. Signal Process.*, Boston, MA, USA, Sep. 2015, pp. 17–20.
- [21] J. Lee, T. Kim, J. Park, and J. Nam, "Raw waveform-based audio classification using sample-level CNN architectures," 2017, *arXiv:1712.00866*. [Online]. Available: <http://arxiv.org/abs/1712.00866>
- [22] S. Thomas, S. Ganapathy, G. Saon, and H. Soltan, "Analyzing convolutional neural networks for speech activity detection in mismatched acoustic conditions," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Florence, Italy, May 2014, pp. 2519–2523.
- [23] O. Abdel-Hamid, A. R. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 10, pp. 1533–1545, Oct. 2014.
- [24] G. Peeters, "A large set of audio features for sound description (similarity and classification) in the CUIDADO project," IRCAM, Paris, France, Tech. Rep. 54, 2004.
- [25] Y. Su, K. Zhang, J. Wang, and K. Madani, "Environment sound classification using a two-stream CNN based on decision-level fusion," *Sensors*, vol. 19, no. 7, p. 1733, Apr. 2019, doi: [10.3390/s19071733](https://doi.org/10.3390/s19071733).
- [26] S. Karlos, N. Fazakis, K. Karanikola, S. Kotsiantis, and K. Sgarbas, "Speech recognition combining MFCCs and image features," in *Speech and Computer (Lecture Notes in Computer Science)*, vol. 9811, A. Ronzhin, R. Potapova, and G. Németh, Eds. Cham, Switzerland: Springer, 2016.
- [27] A. Zolnay, R. Schlueter, and H. Ney, "Acoustic feature combination for robust speech recognition," in *Proc. ICASSP*, vol. 1, 2005, pp. I-457–I-460.
- [28] Y. Shao and D. Wang, "Robust speaker identification using auditory features and computational auditory scene analysis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Las Vegas, NV, USA, Mar. 2008, pp. 1589–1592.
- [29] V.-T. Tran and W.-H. Tsai, "Stethoscope-sensed speech and breath-sounds for person identification with sparse training data," *IEEE Sensors J.*, vol. 20, no. 2, pp. 848–859, Jan. 2020.
- [30] B. McFee, C. Raffel, D. Liang, D. P. W. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "Librosa: Audio and music signal analysis in Python," in *Proc. 14th Python Sci. Conf.*, 2015, pp. 18–25.
- [31] Y. Tokozume and T. Harada, "Learning environmental sounds with end-to-end convolutional neural network," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 2721–2725.
- [32] Youtube Channel. (2010). *TGG—Global Emergency Responses*. Accessed: 2019. [Online]. Available: <https://www.youtube.com/c/TGGGlobalEmergencyResponses>
- [33] D. Steinhardt. (2010). *Worldwide Emergency Responses*. Youtube Channel. Accessed: 2019. [Online]. Available: <http://www.rescue911.eu/> and <https://www.youtube.com/user/wwwrescue911de/search?query=canada>
- [34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: <http://arxiv.org/abs/1412.6980>



VAN-THUAN TRAN received the M.S. degree in electrical engineering and computer science from the National Taipei University of Technology, Taipei, Taiwan, in 2018, where he is currently pursuing the Ph.D. degree in electronic engineering. From 2015 to 2016, he worked as an Engineer at JGCS Consortium, NSRP Project, Thanh Hoa, Vietnam. His research interests include multimedia signal processing and applied artificial intelligence.



WEI-HO TSAI (Member, IEEE) received the Ph.D. degree in communication engineering from National Chiao-Tung University, Hsinchu, Taiwan, in 2001. From 2001 to 2003, he was with Philips Research East Asia, Taipei, Taiwan, where he worked on speech processing problems in embedded systems. From 2003 to 2005, he served as a Postdoctoral Fellow at the Institute of Information Science, Academia Sinica, Taipei. He is currently a Professor with the Department of

Electronic Engineering, National Taipei University of Technology, Taiwan. His research interests include spoken language processing and music information retrieval.

• • •