

# Learning Audio Sequence Representations for Acoustic Event Classification

Zixing Zhang<sup>1</sup>, Ding Liu<sup>2</sup>, Jing Han<sup>3</sup>, Kun Qian<sup>4\*</sup>, and Björn W. Schuller<sup>1,3</sup>

1. GLAM – Group on Language, Audio & Music, Imperial College London, London SW7 2AZ, UK,  
E-mail: connectzzx@gmail.com, bjoern.schuller@imperial.ac.uk

2. Department of Informatics, Technische Universität München, 80333 Munich, Germany, E-mail: ding.liu@tum.de

3. ZD.B Chair of Embedded Intelligence for Health Care and Wellbeing, University of Augsburg, Augsburg 86159, Germany,  
E-mail: {jing.han, schuller}@informatik.uni-augsburg.de

4. Educational Physiology Laboratory, The University of Tokyo, Tokyo 113-0033, Japan, E-mail: qian@p.u-tokyo.ac.jp

---

## Abstract

Acoustic Event Classification (AEC) has become a significant task for machines to perceive the surrounding auditory scene. However, extracting effective representations that capture the underlying characteristics of the acoustic events is still challenging. Previous methods mainly focused on designing the audio features in a ‘hand-crafted’ manner. Interestingly, data-learned features have been recently reported to show better performance. Up to now, these were only considered on the frame-level. In this article, we propose an unsupervised learning framework to learn a vector representation of an audio sequence for AEC. This framework consists of a Recurrent Neural Network (RNN) encoder and a RNN decoder, which respectively transforms the variable-length audio sequence into a fixed-length vector and reconstructs the input sequence on the generated vector. After training the encoder-decoder, we feed the audio sequences to the encoder and then take the learnt vectors as the audio sequence representations. Compared with previous methods, the proposed method can not only deal with the problem of arbitrary-lengths of audio streams, but also learn the salient information of the sequence. Extensive evaluation on a large-size acoustic event database is performed, and the empirical results demonstrate that the learnt audio sequence representation yields a significant performance improvement by a large margin compared with other state-of-the-art hand-crafted sequence features for AEC.

**Keywords:** Audio sequence-to-vector, recurrent autoencoder, acoustic event classification, machine learning, deep learning, computer audition

---

## 1. Introduction

*Acoustic Event Classification* (AEC) plays an essential role in enabling the environmental awareness for intelligent machines, and has recently attracted considerable attention (Chu (2008); Stowell et al. (2015); Ye et al. (2015); Phan et al. (2016)). It can be referred to the field of *Computational Auditory Scene Analysis* (CASA) (Wang and Brown (2006)), which has an assumption that the audio recordings can be automatically categorised by the location where they were recorded. It has a close relationship to another well-documented task, *Acoustic Event Detection* (AED) (Phan et al. (2014); Xia et al. (2019)), where the latter focuses on automatically finding accurate onset and offset of an acoustic event (e. g., a car passing) in a given duration of audio recording. In many previous studies, the two tasks are jointly implemented in a system that, the au-

dio recordings were segmented by AED then classified into different groups in the scheme of the AEC task. The recent trend including this proposed work prefers a holistic AEC system which does not need a segmentation step in the AEC task.

One central goal of AEC is to extract discriminative representations that are robust enough to capture the acoustic event content. In the past decade, many efforts have been reported towards this direction. For example, following the success in speech recognition, Mel-Frequency Cepstral Coefficients (MFCC) have been applied as the most dominant feature type for AEC (Ma et al. (2006); Giannoulis et al. (2013)). However, unlike speech recognition, AEC highly relies on longer temporal information to make a decision (Chu et al. (2009)). In this regard, spectro-temporal features were introduced to capture the event modulation patterns in both time and frequency domains (Schröder

et al. (2015); Rakotomamonjy and Gasso (2015); Ren et al. (2017a)). For instance, a filterbank of two-dimensional Gabor functions was used to decompose the spectro-temporal power density into multiple components (Schröder et al. (2015)). Similar work was done in (Chu et al. (2009)), where a Gabor dictionary was implemented with atom parameters (i. e., scale, time, frequency, and phase) for the Matching Pursuit (MP) decomposition, and the generated MP features have shown their efficiency (Chu et al. (2009)). Furthermore, another trend aims to build higher-level features from the spectro-temporal features of acoustic event. In this context, Histogram of Gradients (HOG) representation was investigated to provide the information of the changing direction of spectral power (Rakotomamonjy and Gasso (2015)). Besides, a fusion of acoustic and visual features was proposed for the AEC task in (Xie and Zhu (2019)). Despite its efficiency, it is claimed that using a fusion of acoustic and visual features increases the computational cost. Before executing feature extraction and classification, it was also found that a segmentation step was needed and helps to improve the model robustness in a noisy scenario (Mulimani and Koolagudi (2019)). Although huge efforts have been made on designing optimal features, AEC still remains a challenging task since the audio contains high variety owing to the complex acoustic environment. For this reason, a solution is to combine a variety of features extracted in either time or frequency domain into a fused high-dimensional space (Zhang and Schuller (2012)). The primary assumption is that the classification model can automatically select important features for a specific class, which, however, can be quite challenging during model building.

Another ignored challenge is that most of the previous works were mainly based on Fourier transformation features, which cannot optimise the Heisenberg-like time–frequency trade-off (De Bruijn (1967)). Therefore, a multi-resolution analysis should be taken into account. Qian *et al.* used wavelet-based features in (Qian et al. (2017)) and found that it can improve the performance of the models built by only using Fourier transformation features. Different from these human hand-crafted features, some other models were built on higher representations learnt via *deep learning* (LeCun et al. (2015)). Ren *et al.* introduced scalogram-based representations to the AEC task in (Ren et al. (2017b, 2018b)). Furthermore, an attention-based convolutional neural network (CNN) model was proposed in (Ren et al. (2018a)). Zhang *et al.* additionally proposed a fine-resolution CNN (FRCNN), which can make the feature representations be easily customised in various

time–frequency resolutions (Zhang et al. (2020)). Abdoli *et al.* proposed a one-dimensional CNN based system which can learn representations directly from the audio signal, which outperformed the human hand-crafted features and the 2D CNN based representations (Abdoli et al. (2019)).

Apart from the above mentioned supervised learning scenarios, deep unsupervised representation learning techniques have achieved tremendous success in machine learning (Hinton et al. (2006); Bengio et al. (2006, 2013)). The key idea is to learn more complex abstractions as data representations in the higher layers of artificial deep neural networks from simple features in the lower layers in an unsupervised training fashion. The unsupervised representation learning has begun to be applied to AEC, and has shown its efficiency in state-of-the-art research. In (McLoughlin et al. (2015)), Deep Belief Networks (DBN) were employed for pre-training with unlabelled data. The extracted bottleneck features were then fed into a concatenated softmax layer for final classification. To capture the temporal information, sequential frames within a sliding window were batched as the network input. Similarly, a fully Deep Neural Network (DNN) structure was introduced in (Xu et al. (2016)), where the raw features in continuum were scaled into one super high-dimensional vector and then considered to be the input for a deep ‘pyramid’ structure. All these unsupervised representation learning researches have advanced the performance of AEC systems significantly.

However, all these works either attempt to learn high-level representations at the frame-level, as the studies did in the field of speech recognition (McLoughlin et al. (2015); Lee et al. (2009)), or assume that the analysed recordings share a fixed duration (Xu et al. (2016)). Indeed, many event sounds have a strong temporal domain signature as aforementioned. For instance, the chirping of insects is typically noise-like with a broad and flat spectrum, which makes it hard for a system to distinguish it as a noise or an insect sound within one or several audio frames. Moreover, the acoustic events are often presented in arbitrary lengths, rather than fixed lengths. This renders the work in (Xu et al. (2016)) infeasible in realistic applications. To overcome the raised problems for AEC, we propose an *unsupervised sequence representation learning* approach, which employs multilayer Grated Recurrent Unit Recurrent Neural Networks (GRU-RNN) to learn representations of audio sequences. The model consists of a RNN encoder to map an input sequence into a fixed-length vector, and a RNN decoder to reconstruct the input sequence from the generated vector into a sequence-to-sequence learn-

ing strategy. Our primary assumption is that the representation captures the sequence information as it integrates a ‘restoration ability’ with the help of the decoder.

The employed encoder-decoder architecture is inspired by the ones used in natural language processing (Sutskever et al. (2014)), where the architecture was used for, for example, translating sentences from one language to another (Sutskever et al. (2014); Bahdanau et al. (2014); Luong et al. (2015)), or predicting the next sentence from previous ones (Shang et al. (2015)). Significantly differing from these works, the essential idea of the proposed framework in this article aims to learn a *vector representation* of a sequence with an arbitrary length. The learnt representations can then be utilised for pattern recognition by any classification models.

The proposed approach is partially motivated by the work in (Srivastava et al. (2015)), where a Long Short-Term Memory (LSTM) encoder-decoder architecture was employed for video reconstruction and future prediction. In addition, it relates to (Dai and Le (2015)) as well, where the LSTM encoder-decoder was utilised for initialising the neural networks and further improving their generalisation capability. The proposed approach, however, is the attempt to obtain a vector representation in a purely unsupervised learning procedure.

The major contributions of this article mainly include: i) We propose an unsupervised learning framework to extract high-level audio sequence representations via a GRU-RNN encoder-decoder for AEC. Compared with previous works, this framework not only can deal with flexible-length audio recordings. More importantly, it also holds the potential to distil the inherent event characteristics embedded in the audio sequences through infinite unlabelled data, which are ubiquitous and cheap to obtain due to popularity of digital devices. ii) We evaluate the performance of the learnt sequence representations on a large-scale acoustic event database. The results demonstrate the high effectiveness and robustness of the learnt representations. iii) The proposed method cannot only benefit the future work in AEC system design, but also other relevant studies in audio or multi-modality related applications.

## 2. Related Work

There are two dominant methods to represent the audio sequence for AEC. The first method is likely inspired by speech recognition technology, in which the whole sequence is represented by sequential Low-Level Descriptors (LLDs) (e.g., MFCCs) frame by frame. Then, it uses generative models to estimate the joint probability distribution of features and labels to arrive at

a final judgment (Stowell et al. (2015)), or uses discriminative models like by a Support Vector Machine (SVM) to predict the frames successively then voting for a final decision (McLoughlin et al. (2015)). While the sequence temporal information is going to be utilised as mentioned above, they are still far from being well-explored. The second method intends to expand all descriptors and concatenates them into a long vector, and then feeds the vector into a model for discriminative training and evaluation (Xu et al. (2016)). This method simply assumes that all audio recordings have a fixed length. Also, this method possibly results in a curse of dimension issue when the recording duration increases. Rather than straightforwardly using the sequential frame-wise LLDs, recent promising methods are more in favour of the sequence-wise *statistic* features. These methods show the ability to handle the *arbitrary-length* recordings, and map them into fixed-dimensional vector representations.

One efficient method is the *Bag-of-Audio-Words* (BoAW) (Aucouturier et al. (2007); Lu et al. (2014)). It uses a codebook of acoustic words (i.e., frame-level LLDs) that are randomly selected or generated via a clustering method (i.e., *k*-means) on the training set, to quantise the frame-wise LLDs. Then, a histogram of the occurrences of each word in the dictionary is built over the whole sequence, and regarded as the sequence representation. Another popular method is based on *functionals* (e.g., mean, standard deviation, skewness, kurtosis, maximum, minimum), which are applied to each of the LLD contours to extract the statistic information over the whole sequence (Zhang and Schuller (2012)). However, all of these features for audio sequence are still hand-crafted.

Recently, some pre-trained models are proposed to extract to extract good embeddings from audio, e.g., the OpenL3<sup>1</sup>. These models were pre-trained by large amount of audio data and can be used to train shallow classifiers with limited size of data Cramer et al. (2019). We believe this contribution can significantly help overcome the (labelled) data scarcity issue for specific audio classification task. It is noted that building such efficient pre-trained models still need large amount of labelled data and huge computational resources. Additionally, transfer learning technologies are quite dependent on empirical knowledge that can make the pre-trained models work well. In this article, we propose to learn the audio sequence representation in an unsupervised way for the application of AEC. Although a

---

<sup>1</sup><https://openl3.readthedocs.io/en/latest/index.html>

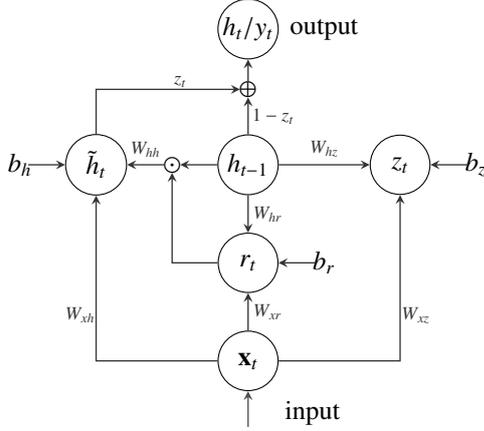


Figure 1: Illustration of Gated Recurrent Unit.

related work has been done in (Chung et al. (2016)), it is mainly focused on word-level audio for spoken term detection. To the best of our knowledge, this is the first effort in this direction towards modelling the long audio sequence for a classification task.

### 3. Unsupervised Learning of Audio Sequence Representations

In this article, we are interested in evaluating the performance of a RNN-based sequence-to-sequence encoder-decoder approach for AEC. Before an empirical evaluation, we first describe the proposed method in this section.

#### 3.1. Grated Recurrent Unit

To implement the RNN encoder-decoder, we select the Grated Recurrent Unit (GRU) as the recurrent hidden unit of our RNNs, which was initially proposed by Cho et al. (Cho et al. (2014)). Analogous to the LSTM unit, this recurrent unit can also capture the long-term dependencies in sequence-based tasks and can well address the vanishing gradient problem (Chung et al. (2014)). Hence, GRU is often regarded as an alternative to LSTM units. However, the GRU has fewer parameters since it does not have a separate memory cell nor an output gate, which results in a faster training process and less-data demand for generalisation. Besides, many experiments have shown that the GRU performs competitive to or slightly better than the LSTM unit in most tasks (Chung et al. (2014); Jozefowicz et al. (2015)).

The typical structure of a GRU is depicted in Fig. 1, which consists of a reset gate  $r$ , an update gate  $z$ , an

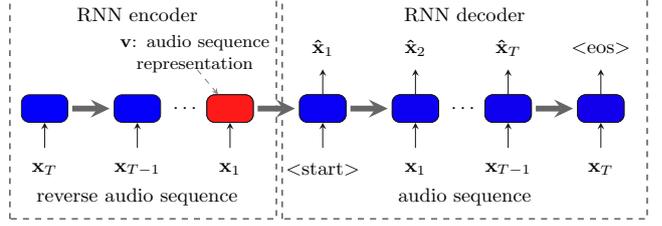


Figure 2: Unsupervised framework for learning of audio sequence representations with a sequence-to-sequence Recurrent Neural Network (RNN) encoder-decoder.

activation  $h$ , and a candidate activation  $\tilde{h}$ . Mathematically, let  $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T)$  be an input audio sequence, where  $\mathbf{x}_t \in \mathfrak{X}^d$  is in a  $d$  dimension feature space (e. g., MFCC). The activation  $h_t^j$  of the  $j$ -th GRU at time  $t$  is updated by the previous activation  $h_{t-1}^j$  and the candidate activation  $\tilde{h}_t^j$ , that is

$$h_t^j = (1 - z_t^j)h_{t-1}^j + z_t^j\tilde{h}_t^j. \quad (1)$$

The update gate  $z_t$  is calculated by

$$z_t^j = \text{sigm}(W_{xz}\mathbf{x}_t + W_{hz}\mathbf{h}_{t-1} + \mathbf{b}_z)^j, \quad (2)$$

where  $W$  denotes the weights matrix and  $\mathbf{b}$  stands for the bias vector. The update gate  $z_t^j$  is used for deciding how much the activation  $h_t^j$  is to be updated with a new activation  $\tilde{h}_t^j$ . Thus, when  $z_t^j$  is close to zero, the hidden state almost keeps unchanged in the next time-step. Opposed to this, when  $z_t^j$  is close to one, the hidden state will be overwritten by a new version. In this way, it is expected to maintain any important feature owing to the update gate of the GRU.

The candidate activation  $\tilde{h}_t^j$  is computed mainly by considering the input  $\mathbf{x}_t$ , the reset gate  $\mathbf{r}_t$ , and the previous time-step hidden activation  $\mathbf{h}_{t-1}$ , as follows

$$\tilde{h}_t^j = \text{tanh}(W_{xh}\mathbf{x}_t + W_{hh}(\mathbf{r}_t \odot \mathbf{h}_{t-1}) + \mathbf{b}_h)^j, \quad (3)$$

where  $\odot$  is an element-wise multiplication, and  $\mathbf{r}$  is a set of reset gates. Here, the reset gate  $r_t^j$  decides on how much the previous activation  $h_{t-1}^j$  impacts the candidate activation  $\tilde{h}_t^j$ . Only when  $r_t$  equal zero, the candidate activation will be overwritten by the current inputs. Similar to the update gate, the  $j$ -th reset gate is computed by

$$r_t^j = \text{sigm}(W_{xr}\mathbf{x}_t + W_{hr}\mathbf{h}_{t-1} + \mathbf{b}_r)^j. \quad (4)$$

#### 3.2. Audio Sequence Representation Learning

The proposed unsupervised representation learning framework of audio sequences is illustrated in Fig. 2,

which comprises a GRU-RNN *encoder* and a GRU-RNN *decoder*. The primary objective of this framework is to transform an arbitrary-length audio segment, given as a sequence of feature vectors  $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T)$ , into *one fixed-length* vector representation  $\mathbf{v}$ .

Specifically, the RNN encoder reads the acoustic feature  $\mathbf{x}_t$  sequentially and *reversely* as done in (Sutskever et al. (2014)), and the hidden state  $\mathbf{h}_t$  is updated accordingly by

$$\mathbf{h}_t = f(\mathbf{x}_t, \mathbf{h}_{t+1}, \mathbf{z}_t, \mathbf{r}_t) \quad (5)$$

where  $f$  denotes the GRU activation function as introduced in the above section. After the last acoustic feature  $\mathbf{x}_1$  has been read and processed, the hidden state  $\mathbf{h}_1$  of the RNN encoder can be viewed as the learnt vector representation  $\mathbf{v}$  of the whole input sequence.

The decoder aims to reconstruct the input sequence of the encoder in a ‘normal’ direction, as  $\hat{\mathbf{x}} = (\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \dots, \hat{\mathbf{x}}_T)$ . To do this, the last hidden state  $\mathbf{h}_1$  of the encoder is copied to the decoder as its initial hidden state, i. e.,  $\hat{\mathbf{h}}_1 = \mathbf{h}_1$ .

Then, the decoder predicts the feature vector  $\hat{\mathbf{x}}_t$  by given its hidden state  $\hat{\mathbf{h}}_t$ , update gate  $\hat{\mathbf{z}}_{t-1}$ , reset gate  $\hat{\mathbf{r}}_{t-1}$ , and its input  $\mathbf{x}_{t-1}$ , that is,

$$\hat{\mathbf{x}}_t = g(\mathbf{x}_{t-1}, \hat{\mathbf{h}}_t, \hat{\mathbf{z}}_{t-1}, \hat{\mathbf{r}}_{t-1}), \quad (6)$$

where  $g$  is the GRU activation function as well. Note that, rather than using the previously predicted feature sequence, we utilise the original feature sequence as the decoder input, which is motivated by the finding of the work (Bengio et al. (2015)). That is, the original feature sequence is helpful in improving the model robustness to its own errors when training.

The RNN encoder and decoder are jointly trained by minimising the reconstruction error, measured by the averaged Mean Square Error (MSE):

$$\frac{1}{T} \sum_{t=1}^T \|\mathbf{x}_t - \hat{\mathbf{x}}_t\|^2. \quad (7)$$

The whole training process is carried out in a fully *unsupervised* manner since label information is not required at all.

Finally, when the audio sequences are fed into the pre-trained encoder-decoder, and the last hidden state of the encoder for each audio sequence will be viewed as its fixed-dimensional vector representation  $\mathbf{v}$ . Since this vector is able to reconstruct itself by the RNN decoder, we believe that such a vector contains the whole sequence information in a compressed way.

Table 1: Quantitative description of Findsounds2016.

category	# class	# segment	duration
Animals	67	1 998	1h 53m
Birds	102	1 766	1h 53m
Household	53	2 097	1h 27m
Insects	7	235	16m
Mayhem	35	1 471	50m
Miscellaneous	70	2 628	1h 45m
Musical Instruments	57	4 112	3h 35m
Nature	18	754	1h 3m
Office	18	1 188	50m
People	45	2 165	1h 44m
Sports Recreation	22	266	9m
Tools	21	296	18m
TV Movies	22	645	24m
Vehicles	33	1 714	2h 9m
<b>Total</b>	<b>570</b>	<b>21 335</b>	<b>18h 23m</b>

## 4. Experiments and Results

In this section, we are devoted to estimating the effectiveness and the robustness of the proposed framework for learning audio sequence representations. Extensive experiments are conducted on a large-size acoustic event database, and the empirical results are compared with other state-of-the-art baselines.

### 4.1. Database Description

The database selected for our experiments – *Findsounds2016* – is supposed to be a large publicly available databases for the AEC research when conducting the experiments (Piczak (2015)). It was collected from the website of ‘www.findsounds.com’, which provides a comprehensive set of event-annotated audio recordings from real environments, reaching from nature (e. g., nature and animals) over human beings (e. g., people) to manufactured articles (e. g., musical instruments and vehicles). Specifically, we discarded two categories (i. e., Holidays and Noisemakers) from the original dataset due to the sample-overlapping with other categories, resulting in a final set of 14 common acoustic-event categories. Each category further includes a number of classes (subsets), giving rise to a total of 570 classes and 21 335 independent audio segments, with a total duration of more than 18 hours. More details on the number of segments and recording time per category are summarised in Table 1. The averaged duration over all audio segments is 3.1 s with a maximum and a minimum

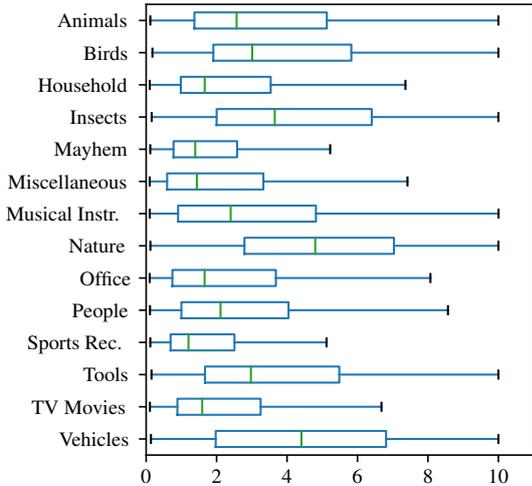


Figure 3: Duration distribution of audio segments for each acoustic-event category over the whole Findsounds2016 database.

of 10.0 s and 0.1 s, respectively. In detail, Fig. 3 illustrates the duration distribution for each acoustic-event category over the whole database. Obviously, these duration distributions are highly overlapped and mainly range from one to six seconds. Moreover, owing to the diversity of the audio formats in the original dataset retrieved from the web, we converted all audio files into a uniformed format with 16-bit encoding, mono-channel, and 16 kHz sampling rate. It is noted that the majority samples of this dataset are monophonic audio. Only a limited or negligible number of overlapped sounds exist.

For training the back-end classifier, each subset of the Findsounds2016 database was equally and sequentially partitioned into training set (7312 instances), test set (7106 instances), and validation set (6917 instances). In addition, we always upsampled the training set to alleviate the unbalanced class-distribution problem, by randomly repeating the samples in less dominated classes several times with the random seed zero.

#### 4.2. Experimental Setup

For training RNN encoder-decoders to learn the audio sequence representations, we theoretically could feed the raw signals into the network directly. However, the long sequence-length leads to a high requirement of computational resource. As MFCCs have been repeatedly verified to be the efficient features for most acoustic recognition tasks and we have limited computational resource, we extracted 13 MFCCs (including one logarithmic energy) per frame using a window size of 60 ms

at a step size of 60 ms. Compared with the conventional parameters for extracting MFCCs (i.e., window size: 25 ms, step size: 10 ms), the ones we selected due to i) the audio pattern normally changes slower than the speech pattern. In this case, within a longer frame window size, the audio signal is still considered to be stationary; ii) a longer frame window size is helpful to extract rich low fundamental frequency information, which fits for audio analysis; iii) a longer frame step size compared to one used for speech can reduce the extracted number of frames and consequently significantly speed up the network training process.

In this case, the longest sequence of Findsounds2016 has 167 MFCC feature vectors. Finally, all the extracted features were standardised by the means and variations of the training set.

To accelerate the RNN encoder-decoder training process, we used a mini batch of 64 sequences as network input. In this case, we padded zeros at the end of each sequence to force them equi-long. The padded zeros, however, are ignored when calculating the reconstruction errors (i.e., training loss) in the training process by setting their weights to zero. Further, to control the learning process, we checked the training loss after running every 500 batches. To update the network weights, we employed the classic Stochastic Gradient Decent (SGD) with an initial learning rate of 0.7. This value dynamically reduced with a decay factor of 0.99 when the training loss was not improved any more over the previous three checking points. Additionally, a gradient norm clipping operation was performed with a clipping ratio of 5 to handle the gradient blowup problem. The whole learning process was stopped once there was no training loss improvement over 20 successive checking points.

To assess the discrimination and the robustness of the learnt audio sequence representations via the pre-trained RNN encoder-decoder, we further adopted nowadays two of the most frequently used classification models. One of them is the SVM trained with the sequential minimal optimisation algorithm. The complexity value of  $C$  was optimised in  $\{0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5\}$  on the validation set. Another one is the GRU-RNN with one hidden layer, while the number of hidden units was optimised in  $\{128, 256, 512, 1024\}$  on the validation set. Additionally, the GRU-RNNs were trained with Adam SGD with an initial learning rate of  $10^{-4}$ , to which an exponential decay was applied at every  $10^4$  steps with a decay rate of 0.96. Further, the gradient norm clipping ratio was set to 1.2, and the batch size was set to 128. For equal comparison, the training processes of all networks were stopped at the 500th epoch.

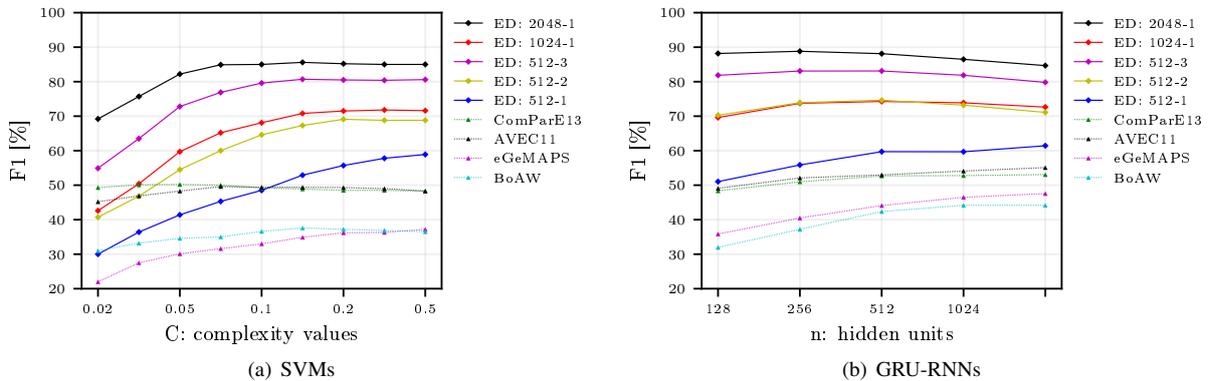


Figure 4: Performance comparison (F1-measure) between the *learnt audio sequence representations* via a variety of RNN Encoder-Decoders (ED) and four *hand-crafted features* on the *validation set* of Findsounds2016. Performance was evaluated by (a) the SVMs with various complexity values,  $C$ , or (b) the GRU-RNNs with various numbers of hidden units,  $n$ .

To measure the system performance, we utilised *F1-measure* (F1) as a primary metric, mainly due to the facts that i) F1 provides an overview performance in a multi-class setting as it is calculated by the harmonic mean of unweighted precision and recall; ii) F1 is among the most widely used evaluation metrics in AEC, for example, in a series of challenges of Detection and Classification of Acoustic Scenes and Events (DCASE) (Stowell et al. (2015); Mesaros et al. (2016)). Additionally, we took the *Unweighted Accuracy* (UA, or unweighted recall) as a complementary metric. It is obtained by the sum of the accuracies over all classes divided by the number of classes. Thus, UA also well indicates the system performance in a class-unbalanced task.

Further, unless stated otherwise, a one-side  $z$ -test was undertaken to evaluate the statistical significance of performance improvement.

#### 4.3. Compared Features of Audio Sequence

To verify the effectiveness of the learnt representation of audio sequence, we selected one BoAW feature set and three functional-based feature sets for comparison. All these feature sets are widely used for AEC or related acoustic tasks (e. g., emotion) nowadays. A brief description of the four feature sets is as follows:

- *BoAW* feature set: The codebook includes 2048 audio words. Each frame of the sequence is then assigned to the nearest 256 audio words. Afterwards, a normalised histogram is applied to convert the word occurrence accounts into a fixed-length vector (Lu et al. (2014)).

- the extended Geneva Minimalistic Acoustic Parameter Set (*eGeMAPS*): It consists of 88 important acoustic attributes, which were selected by extensive experiments on acoustic pattern classification tasks (Eyben (2016)).
- the 2011 Audio-Visual Emotion recognition Challenge (*AVEC11*) feature set: It contains 1941 attributes and was used in (Zhang and Schuller (2012)) for AEC.
- the INTERSPEECH 2013 Computational Paralinguistics Challenge (*ComParE13*) feature set: It includes a large-scale acoustic attributes up to 6373 (Eyben (2016)).

#### 4.4. Results

To evaluate the robustness of the proposed framework, we constructed the RNN encoder-decoders in several structures, mainly towards a *deep* or a *wide* direction. To assess the deeper networks, we fixed the number of hidden units per layer as 512, and then set the hidden layers to one, two, or three, resulting in three RNN encoder-decoders in different depths. To assess the wider networks, we fixed the depth of hidden layer as one, but set the number of hidden units to 512, 1024, or 2048, leading to additional two RNN encoder-decoders in different widths. Note that the RNN encoders and corresponding decoders always share the same structures.

Fig. 4 illustrates the performance of the learnt representations obtained by diverse RNN encoder-decoders, as well as four conventional feature sets based on BoAW

Table 2: Performance comparison (F1 and UA) between the *learnt audio sequence representations* via a variety of RNN Encoder-Decoders (ED) and four *hand-crafted features* on the *test* set of Findsounds2016. Performance was evaluated by both SVMs and GRU-RNNs.

[%] feature types	SVMs		GRU-RNNs	
	F1	UA	F1	UA
BoAW	41.9	35.3	44.4	39.5
eGeMAPS	36.4	34.9	47.6	41.4
AVEC11	50.4	42.8	54.0	48.7
ComParE13	49.7	43.6	53.2	46.2
ED: 512-1	58.1	52.9	61.1	54.9
ED: 512-2	68.4	63.4	71.8	67.4
ED: 512-3	80.6	76.6	80.5	78.4
ED: 1024-1	72.0	65.8	72.6	70.0
ED: 2048-1	<b>85.2</b>	<b>80.4</b>	<b>89.0</b>	<b>87.6</b>

or functionals (i. e., eGeMAPS, AVEC11, and ComParE13). The performance was estimated on the validation set of Findsounds2016 for 14 acoustic-event categories.

Specifically, Fig. 4 (a) depicts the feature performance when taking the SVMs as discriminative models. From this figure, one can obviously observe that the results delivered by the learnt representations are remarkably higher than the other four state-of-the-art baselines. The best result is achieved at 85.6 % of F1 by using the representations learnt by the RNN encoder-decoder with one hidden layer of 2 048 hidden units (ED: 2048-1). This result is almost double of the best baseline achieved by using ComParE13 or AVEC11 feature set (i. e., 50.2 % of F1).

Further, when increasing the depth of the neural networks from one to two and three, one can see a steady and significant performance improvement. Similarly, when extending the width of the neural networks from 512 to 1 024 and 2 048, again huge performance improvement is obtained. This indicates that appropriately increasing the complexity of the sequence-to-sequence model, either in a deep way or in a wide way, can notably improve the effectiveness of the learnt representations.

Similar observations can be found in Fig. 4 (b), where GRU-RNNs were employed as discriminative models. Generally speaking, however, GRU-RNNs yield better performance than SVM in all cases. The best result further rockets to 88.8 % of F1. Additionally, an interesting observation can be seen that the learnt representations performs better when using relatively simple networks for classification, yet the hand-crafted features incline

to choose the relative complex networks for classification in order to get better results. This indicates that the learnt representations is easier to be learnt by a simple machine learning model than the selected hand-crafted features.

We further evaluated the learnt representations on the test set by employing both SVMs and GRU-RNNs for classification with the best parameter settings optimised on the validation set. Table 2 displays the corresponding results in terms of F1 and UA. Consistently, the RNN encoder-decoder with 2048 hidden units offers the most efficient features, contributing to 85.2 % of F1 and 80.4 % of UA by means of SVMs, and 89.0 % of F1 and 87.6 % of UA by means of GRU-RNNs. Compared with the best baseline, they provide absolute gains as high as 35.0 % of F1 and 38.9 % of UA. To further investigate the effectiveness of the learnt representation, we randomly selected 20 samples from each categories and projected them into the leading two discriminant directions found by Linear Discriminant Analysis (LDA). The visualisation of the audio sequence representations is displayed in Fig. 5. Notably, the samples belong to different categories are strongly discriminative, which reasonably results in a high prediction accuracy.

To intuitively demonstrate the best performance we achieved by using the GRU-RNN based classifier with one hidden layer and 128 hidden units, Fig. 6 illustrates the prediction confusion matrix on the test set, which is obtained by using the vector representations learnt by the RNN encoder-decoder comprised of one hidden layer with 2 048 hidden units. Generally speaking, the acoustic segments represented by the proposed vectors can be well distinguished into corresponding categories. In more detail, one can notice that the category of ‘Miscellaneous’ (labelled as 5 in Fig. 6) is relatively easier to be misclassified into the others, which keeps in line with the fact that its contents include many quite similar acoustic events to the other categories.

In addition, we performed the same experiments on the acoustic-event classes. Rather than utilising the whole 570 classes, we discarded those classes having extremely sparse samples (fewer than 20). This leads to a subset of 229 selected classes, and a slightly smaller training set (6 277), test set (6 202), and validation set (6 122). Table 3 shows the corresponding results for various features or representations. Interestingly, the learnt representations consistently outperform the frequently used feature sets, and yield the highest F1 and UA of 47.7 % and 39.0 %, respectively, for 229 types of acoustic events.

In future work, one needs to consider implementing our method in an unsupervised learning paradigm,

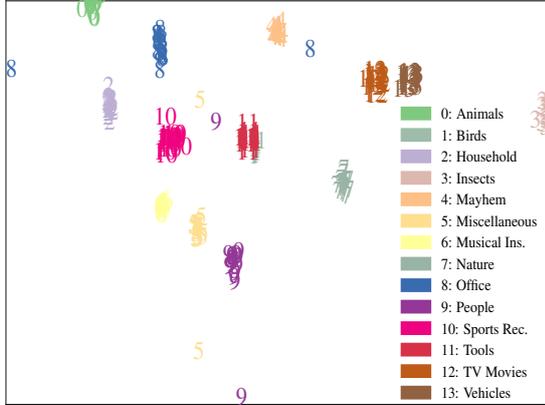


Figure 5: Visualisation of the audio sequence representations learnt from the RNN encoder-decoder. 20 samples per class are randomly selected through the whole dataset and projected into the leading two discriminant directions found by Linear Discriminant Analysis (LDA). Each sample is remarked by the category number (0~13) it belongs to.

0	835	11	39	0	3	6	0	1	10	15	0	13	3	0
1	10	761	8	3	3	3	1	10	23	1	0	1	0	0
2	20	11	910	6	13	4	1	0	0	3	7	0	3	3
3	0	3	0	94	10	1	0	0	1	0	0	0	0	0
4	0	4	0	11	653	4	3	0	1	0	10	0	1	3
5	27	31	13	7	17	957	31	24	30	10	24	24	21	18
6	4	4	8	1	15	55	1731	6	10	10	1	4	13	66
7	1	14	3	0	0	7	1	310	4	1	0	1	0	10
8	3	6	4	0	1	15	6	6	512	0	1	1	0	1
9	7	23	30	0	1	15	1	1	10	912	7	4	0	1
10	1	4	7	0	1	18	1	3	1	3	84	0	0	0
11	1	1	0	0	0	7	1	1	0	3	3	110	6	4
12	0	0	0	0	0	4	0	0	0	0	0	0	296	6
13	3	3	1	1	0	20	18	8	1	3	0	1	8	735
	0	1	2	3	4	5	6	7	8	9	10	11	12	13

Figure 6: Normalised confusion matrix ( $\times 10^{-4}$ ) of predictions on the test set obtained by the best GRU-RNN classification model (one hidden layer with 128 hidden units). The labels from 0 to 13 sequentially indicate the categories from ‘Animals’ to ‘Vehicles’ as listed in Table 1 within the same order.

Table 3: Performance comparison (F1 and UA) for classifying 229 classes of acoustic events between the *learnt audio sequence representations* via a variety of RNN Encoder-Decoders (ED) and four *hand-crafted features* on the *test* set of Findsounds2016. Performance was evaluated by both SVMs and GRU-RNNs.

feature types	SVMs		GRU-RNNs	
	F1	UA	F1	UA
BoAW	18.5	17.8	17.7	17.5
eGeMAPS	18.2	20.1	21.8	20.9
AVEC11	26.8	23.6	20.0	18.8
ComParE13	27.1	23.8	23.1	21.5
ED: 512-1	19.9	20.1	25.6	23.0
ED: 512-2	29.0	26.4	31.5	27.3
ED: 512-3	34.6	<b>32.0</b>	43.2	36.5
ED: 1024-1	24.5	23.8	32.6	28.5
ED: 2048-1	<b>35.1</b>	30.8	<b>47.7</b>	<b>39.0</b>

which has been demonstrated to be successful in extracting semantic representations from large size data set (Jansen et al. (2018)).

## 5. Conclusions

In this article, we proposed an unsupervised framework to learn the essential patterns of acoustic events that are embedded through the whole audio sequence. In this framework, a Recurrent Neural Networks (RNN) based sequence-to-sequence encoder-decoder is used, where the inputs are the sequential and reverse acoustic feature vectors and the targets are their counterparts in normal order. This encoder-decoder is trained without any category information such that it has the huge potential to explore big unlabelled data in the real world. We then extracted the bottleneck features as the audio sequence representations for acoustic event classification, and evaluated them through traditional machine learning algorithms.

This framework can address the audio sequences with arbitrary durations, and compress them into vector representations with a fixed dimension. Since the learnt representation can be well recovered to its original version by the decoder, it is thus supposed to contain the most important sequence information. The effectiveness and robustness of the proposed framework was extensively examined by the experiments on a large dataset, which have raised the state-of-the-art baselines into a significantly high level.

Encouraged by the achieved results, we will further evaluate our proposed method in a recently released weekly labelled dataset AudioSet (Gemmeke

et al. (2017)). We believe that the proposed learning representation approach is a major breakthrough in the development of the RNN based encoder-decoder models, which could potentially lead to a range of exciting applications way out of our chosen exemplary application. These applications, which highly characterised with sequential patterns via either audio or video signals, include activity detection, emotion recognition, polyphonic sound tagging, and the like.

## Acknowledgment

This work was partially supported by the Zhejiang Lab's International Talent Fund for Young Professionals (Project HANAMI), China, the JSPS Postdoctoral Fellowship for Research in Japan (ID No. P19081) from the Japan Society for the Promotion of Science (JSPS), Japan, the Grants-in-Aid for Scientific Research (No. 19F19081 and No. 20H00569) from the Ministry of Education, Culture, Sports, Science and Technology (MEXT), Japan, the European Union's Seventh Framework Programme through the ERC Starting Grant No. 338164 (iHEARu), and the Horizon 2020 Programme through the Research Innovation Action No. 645094 (SEWA).

## References

- Abdoli, S., Cardinal, P., Koerich, A.L., 2019. End-to-end environmental sound classification using a 1d convolutional neural network. *Expert Systems with Applications* 136, 252–263.
- Aucouturier, J.J., Defreville, B., Pachet, F., 2007. The bag-of-frames approach to audio pattern recognition: A sufficient model for urban soundscapes but not for polyphonic music. *The Journal of the Acoustical Society of America* 122, 881–891.
- Bahdanau, D., Cho, K., Bengio, Y., 2014. Neural machine translation by jointly learning to align and translate. in: *Proc. International Conference on Learning Representations (ICLR)*, San Diego, CA, USA. pp. 1–15.
- Bengio, S., Vinyals, O., Jaitly, N., Shazeer, N., 2015. Scheduled sampling for sequence prediction with recurrent neural networks. in: *Proc. Advances in Neural Information Processing Systems (NIPS)*, Montréal, Canada. pp. 1171–1179.
- Bengio, Y., Courville, A., Vincent, P., 2013. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 1798–1828.
- Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H., et al., 2006. Greedy layer-wise training of deep networks. in: *Proc. Advances in Neural Information Processing Systems (NIPS)*, Vancouver, Canada. pp. 153–160.
- Cho, K., Van Merriënboer, B., Bahdanau, D., Bengio, Y., 2014. On the properties of neural machine translation: Encoder-decoder approaches. in: *Proc. Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST)*, Doha, Qatar. pp. 103–111.
- Chu, S., 2008. Unstructured audio classification for environment recognition. in: *Proc. AAAI Conference on Artificial Intelligence*, Chicago, IL. pp. 1845–1846.
- Chu, S., Narayanan, S., Kuo, C.C.J., 2009. Environmental sound recognition with time–frequency audio features. *IEEE Transactions on Audio, Speech, and Language Processing* 17, 1142–1158.
- Chung, J., Gulcehre, C., Cho, K., Bengio, Y., 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555.
- Chung, Y., Wu, C., Shen, C., Lee, H., Lee, L., 2016. Audio Word2Vec: Unsupervised learning of audio segment representations using sequence-to-sequence autoencoder. in: *Proc. INTER-SPEECH*, San Francisco, CA, USA. pp. 765–769.
- Cramer, J., Wu, H.H., Salamon, J., Bello, J.P., 2019. Look, listen, and learn more: Design choices for deep audio embeddings. in: *Proc. ICASSP*, IEEE, Brighton, UK. pp. 3852–3856.
- Dai, A.M., Le, Q.V., 2015. Semi-supervised sequence learning. in: *Proc. Advances in Neural Information Processing Systems (NIPS)*, Montréal, Canada. pp. 3079–3087.
- De Bruijn, N., 1967. Uncertainty principles in Fourier analysis. in: *Inequalities (Proc. Sympos. Wright-Patterson Air Force Base, Ohio, 1965)*, Academic Press, New York, NY, USA. pp. 57–71.
- Eyben, F., 2016. Real-time speech and music classification by large audio feature space extraction. Springer, Berlin/Heidelberg, Germany.
- Gemmeke, J.F., Ellis, D.P.W., Freedman, D., Jansen, A., Lawrence, W., Moore, R.C., Plakal, M., Ritter, M., 2017. Audio set: An ontology and human-labeled dataset for audio events. in: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, New Orleans, LA. pp. 776–780.
- Giannoulis, D., Benetos, E., Stowell, D., Rossignol, M., Lagrange, M., Plumbley, M.D., 2013. Detection and classification of acoustic scenes and events: An IEEE AASP challenge. in: *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY. pp. 1–4.
- Hinton, G.E., Osindero, S., Teh, Y.W., 2006. A fast learning algorithm for deep belief nets. *Neural computation* 18, 1527–1554.
- Jansen, A., Plakal, M., Pandya, R., Ellis, D.P., Hershey, S., Liu, J., Moore, R.C., Saurous, R.A., 2018. Unsupervised learning of semantic audio representations. in: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, Calgary, AB, Canada. pp. 126–130.
- Jozefowicz, R., Zaremba, W., Sutskever, I., 2015. An empirical exploration of recurrent network architectures. in: *Proc. International Conference on Machine Learning (ICML)*, Lille, France. pp. 2342–2350.
- LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. *Nature* 521, 436–444.
- Lee, H., Pham, P., Largman, Y., Ng, A.Y., 2009. Unsupervised feature learning for audio classification using convolutional deep belief networks. in: *Proc. Advances in neural information processing systems (NIPS)*, Vancouver, Canada. pp. 1096–1104.
- Lu, X., Tsao, Y., Matsuda, S., Hori, C., 2014. Sparse representation based on a bag of spectral exemplars for acoustic event detection. in: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy. pp. 6255–6259.
- Luong, M.T., Pham, H., Manning, C.D., 2015. Effective approaches to attention-based neural machine translation. in: *Proc. Conference on Empirical Methods on Natural Language Processing (EMNLP)*, Lisbon, Portugal. pp. 1412–1421.
- Ma, L., Milner, B., Smith, D., 2006. Acoustic environment classification. *ACM Transactions on Speech and Language Processing* 3, 1–22.
- McLoughlin, I., Zhang, H., Xie, Z., Song, Y., Xiao, W., 2015. Robust sound event classification using deep neural networks. *IEEE/ACM Transactions on Audio Speech and Language Processing* 23, 540–552.
- Mesaros, A., Heittola, T., Virtanen, T., 2016. TUT database for acous-

- tic scene classification and sound event detection, in: Proc. European Signal Processing Conference (EUSIPCO), Budapest, Hungary. pp. 1128–1132.
- Mulimani, M., Koolagudi, S.G., 2019. Segmentation and characterization of acoustic event spectrograms using singular value decomposition. *Expert Systems with Applications* 120, 413–425.
- Phan, H., Hertel, L., Maass, M., Koch, P., Mertins, A., 2016. Label tree embeddings for acoustic scene classification, in: Proc. ACM International Conference on Multimedia (MM), Amsterdam, The Netherlands. pp. 486–490.
- Phan, H., Maaß, M., Mazur, R., Mertins, A., 2014. Random regression forests for acoustic event detection and classification. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23, 20–31.
- Piczak, K.J., 2015. ESC: Dataset for environmental sound classification, in: Proc. ACM International Conference on Multimedia (MM), Brisbane, Australia. pp. 1015–1018.
- Qian, K., Ren, Z., Pandit, V., Yang, Z., Zhang, Z., Schuller, B., 2017. Wavelets revisited for the classification of acoustic scenes, in: Proc. DCASE Workshop, Munich, Germany. pp. 108–112.
- Rakotomamonjy, A., Gasso, G., 2015. Histogram of gradients of time-frequency representations for audio scene classification. *IEEE/ACM Transactions on Audio Speech and Language Processing* 23, 142–153.
- Ren, J., Jiang, X., Yuan, J., Magnenat-Thalmann, N., 2017a. Sound-event classification using robust texture features for robot hearing. *IEEE Transactions on Multimedia* 19, 447–458.
- Ren, Z., Kong, Q., Qian, K., Plumbley, M., Schuller, B., 2018a. Attention-based convolutional neural networks for acoustic scene classification, in: Proc. DCASE Workshop, Surrey, UK. pp. 39–43.
- Ren, Z., Pandit, V., Qian, K., Yang, Z., Zhang, Z., Schuller, B., 2017b. Deep sequential image features for acoustic scene classification, in: Proc. DCASE Workshop, Munich, Germany. pp. 113–117.
- Ren, Z., Qian, K., Zhang, Z., Pandit, V., Baird, A., Schuller, B., 2018b. Deep scalogram representations for acoustic scene classification. *IEEE/CAA Journal of Automatica Sinica* 5, 662–669.
- Schröder, J., Goetze, S., Anemüller, J., 2015. Spectro-temporal gabor filterbank features for acoustic event detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23, 2198–2208.
- Shang, L., Lu, Z., Li, H., 2015. Neural responding machine for short-text conversation, in: Proc. Annual Meeting of the Association for Computational Linguistics (ACL), Beijing, China. pp. 1577–1586.
- Srivastava, N., Mansimov, E., Salakhutdinov, R., 2015. Unsupervised learning of video representations using LSTMs, in: Proc. International Conference on Machine Learning (ICML), Lille, France. pp. 843–852.
- Stowell, D., Giannoulis, D., Benetos, E., Lagrange, M., Plumbley, M.D., 2015. Detection and classification of acoustic scenes and events. *IEEE Transactions on Multimedia* 17, 1733–1746.
- Sutskever, I., Vinyals, O., Le, Q.V., 2014. Sequence to sequence learning with neural networks, in: Proc. Advances in neural information processing systems (NIPS), Montréal, Canada. pp. 3104–3112.
- Wang, D., Brown, G.J., 2006. Computational Auditory Scene Analysis: Principles, Algorithms, and Applications. Wiley-IEEE Press, Piscataway, NJ, USA.
- Xia, X., Togneri, R., Sohel, F., Zhao, Y., Huang, D., 2019. Multi-task learning for acoustic event detection using event and frame position information. *IEEE Transactions on Multimedia* 22, 569–578.
- Xie, J., Zhu, M., 2019. Investigation of acoustic and visual features for acoustic scene classification. *Expert Systems with Applications* 126, 20–29.
- Xu, Y., Huang, Q., Wang, W., Foster, P., Sigtia, S., Jackson, P.J., Plumbley, M.D., 2016. Fully deep neural networks incorporating unsupervised feature learning for audio tagging. arXiv preprint arXiv:1607.03681 .
- Ye, J., Kobayashi, T., Murakawa, M., Higuchi, T., 2015. Acoustic scene classification based on sound textures and events, in: Proc. ACM International Conference on Multimedia (MM), Brisbane, Australia. pp. 1291–1294.
- Zhang, T., Liang, J., Ding, B., 2020. Acoustic scene classification using deep cnn with fine-resolution feature. *Expert Systems with Applications* 143, 1–9. In press.
- Zhang, Z., Schuller, B., 2012. Semi-supervised learning helps in sound event classification, in: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan. pp. 333–336.