Deep Recurrent Neural Networks for Audio Classification in Construction Sites

Michele Scarpiniti, Danilo Comminiello, Aurelio Uncini
Department of Information Engineering, Electronics and Telecommunications (DIET)
Sapienza University of Rome
via Eudossiana 18, 00185 Rome, Italy
{michele.scarpiniti, danilo.comminiello, aurelio.uncini}@uniroma1.it

Yong-Cheol Lee
Department of Construction Management
Louisiana State University
Baton Rouge, USA
yclee@lsu.edu

***Abstract*—In this paper, we propose a Deep Recurrent Neural Network (DRNN) approach based on Long-Short Term Memory (LSTM) units for the classification of audio signals recorded in construction sites. Five classes of multiple vehicles and tools, normally used in construction sites, have been considered. The input provided to the DRNN consists in the concatenation of several spectral features, like MFCCs, mel-scaled spectrogram, chroma and spectral contrast. The proposed architecture and the feature extraction have been described. Some experimental results, obtained by using real-world recordings, demonstrate the effectiveness of the proposed idea. The final overall accuracy on the test set is up to 97% and overcomes other state-of-the-art approaches.**

***Index Terms*—Deep learning, Recurrent neural networks (RNNs), LSTM units, Audio processing, Environmental sound classification, Construction sites.**

I. Introduction

DUE to the flexibility and cheapness of acoustic sensors, many research efforts have been recently addressed towards the event classification of environmental audio data [1], [2]. When we consider generic outdoor scenarios, an automatic monitoring system based on microphones would be an invaluable tool in assessing and controlling any type of situation occurring in the environment [3]. This includes, but is not limited to, handling large civil and/or military events. The idea in these works is to use the Computational Auditory Scene Analysis (CASA) [4], which involves Computational Intelligence and Machine Learning techniques, to recognize the presence of specific objects into sound tracks.

Although until now the automated progress monitoring of construction sites is commonly performed by camera equipments [5], some recent works have been addressed to the use of audio classification [6]. One of the first attempts have been performed by Cheng et al. [7], who used Support Vector Machines (SVM) to analyze the activity of construction tools and equipments.

Other recent applications of automatic audio classification have also been addressed to audio-based construction sites monitoring [8], [9], in order to improve the construction process management of field activities. This approach is revealing itself as a promising method and a supportive resource for unmanned field monitoring and safety surveillance that leverages construction project management and decision making [9]. More recently, several studies extend these efforts to more complicated architectures exploiting Deep Learning techniques [10].

In the literature, it is possible to find several instances of successful applications in the field of environmental sound classification that make use of deep learning. For example, in the work of Piczak [11], the author exploits a 2-layered CNN working on the spectrogram of the data to perform environmental sound classification, reaching an average accuracy of 70% over different datasets. Other approaches, instead of using handcrafted features such as the spectrogram, perform an end-to-end environmental sound classification obtaining higher results with respect to the previous ones [12].

Motivated by the architecture described in [13], the recent work in [14] shows the remarkably effectiveness of deep learning in environmental sound classification. The aim of [14] was to develop an application able to recognize vehicles and tools used in construction sites, and classify them in terms of type and brand. This task will be tackled with a neural network approach, involving the use of a Deep Convolutional Neural Network (DCNN), which will be fed with the Mel spectrogram of the audio source as input, and the obtained results were very good achieving an average accuracy of 97% over a limited number of classes.

The aim of this paper is to extend the work in [14] by considering a Deep Recurrent Neural Network (DRNN) architecture, in order to exploit the typical time correlations of audio data. The considered DRNN is composed of two recurrent layers with LSTM cells plus a dense layer and a softmax decision. The proposed architecture is tested on five classes extracted from a real-world dataset, containing in situ recordings of multiple vehicles and tools. We demonstrate that the proposed approach for the classification of such vehicles and tools can obtain excellent results (average accuracy ranging in 95%–97%) related to the specific domain of construction sites.

This work has been supported by the project: "End-to-End Learning for 3D Acoustic Scene Analysis (ELeSA)" funded by Sapienza University of Rome Bando 2018 – grant MA21816436AA4280, and by the project: "SoFT: Fog of Social IoT", funded by Sapienza University of Rome Bando 2018 and 2019.

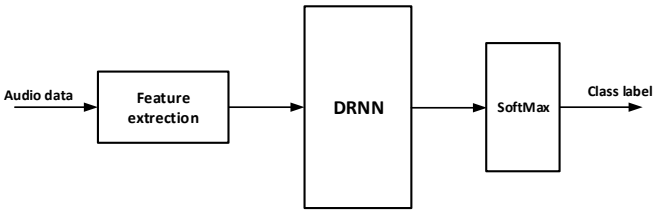


Fig. 1: A sketch of the general architecture.

The rest of this paper is organized as follows. Section II describes the proposed approach, in terms of architecture and extracted features, used to perform the sound classification. Section III introduces the experimental setup, while Section IV shows the obtained numerical results. Finally, Section V concludes the paper and outlines some future directions.

II. Proposed approach

The proposed approach is based on a Deep Recurrent Neural Network (DRNN) that consists in the cascade of several layers with feedback between input and output [10]. The input to this DRNN is constructed by concatenating many spectral features extracted from the audio signal [15]. A final softmax layer is used to implement the output classifier. A sketch of the general architecture is shown in Fig. 1.

A. Deep Recurrent Neural Networks (DRNNs)

The Recurrent Neural Network (RNN) is a kind of densely connected neural network which differs from the normal feed-forward networks for the introduction of "time" [10]. In particular, the output of the hidden layer in the recurrent neural network is fed back into its input, so that the input is a combination of the present and the recent past. RNNs exploit their peculiar structure to discover correlations between events separated by several temporal instants. These represent a sort of long-term dependencies, since a particular event is always a function of past events.

Like most neural networks, the recurrent one has a problem that obstacles its performance: during the learning the gradient tends to vanish. In a network the gradient expresses the change in all weights with regard to the change in error. Since the gradient computation passes through many stages of multiplication, then, if the quantity multiplied is slightly greater then one, the gradient become too large (exploding) else, if the quantity multiplied is less then one, gradient becomes too small (vanishing). Without an exact knowledge of the gradient, we can't adjust the weights and train the network.

One of the best solution to a vanishing gradient problem is a variant of the RNN that uses Long Short-Term Memory (LSTM) units. LSTM units help to preserve the error information that is back-propagated through time and layers. In fact, LSTM units are capable of learning long-term dependencies reducing the problems on the gradient. An LSTM introduces a new structure called *memory cell*, that is composed of four main elements: an input gate, a neuron with a self-recurrent connection (a connection to itself), a forget gate and an output

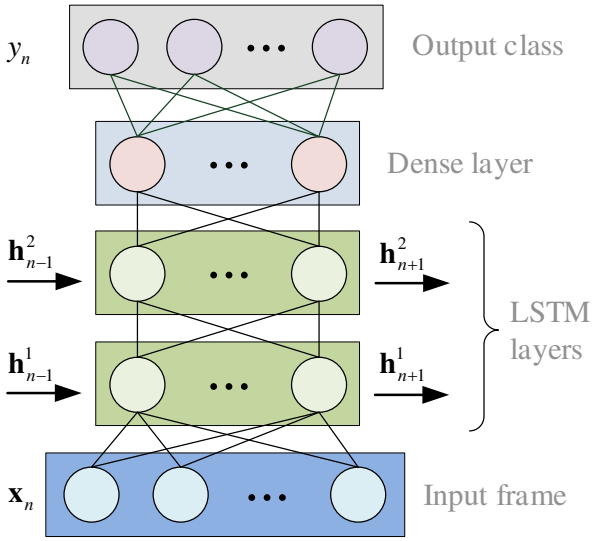


Fig. 2: The proposed DRNN layer organization.

gate. Compared with a standard recurrent neuron, the LSTM one contains three gates as novel elements. The basic idea is to adopt these gates to avoid the gradient to diverge: they can make the decision about what and when to store, read and write. For a complete description of the RNNs, LSTM units and their mathematical equations used for the output evaluation and training, we refer to [10].

The proposed architecture, shown in Fig. 2, is composed of two recurrent layers with LSTM cells plus a dense layer and a softmax activation function for the final classification. The whole network is trained by minimizing the categorical cross-entropy as loss function [10]:

$$\mathcal{L}\left(y, \widehat{y}\right) = -\sum_{i=1}^{N} y_i \log \widehat{y}_i, \tag{1}$$

where y and $\widehat{y}$ are the target class and the predicted class, respectively.

The chosen optimizer is the Adam algorithm, a gradient-based optimization algorithm that exploits the first and second order moments to obtain a smooth and fast convergence [16].

B. Features extraction

For training our DRNN, we extracted some important spectral features from each audio frame. In this work, we focus on four important feature families extracted from the spectrogram that is obtained using the short-time Fourier transform (STFT).

Mel spectrogram – It represents the spectrogram in the Mel scale [17]. The formula for converting from frequency to Mel scale is:

$$M(f) = 1125 \log\left(1 + \frac{f}{700}\right), \tag{2}$$

for f in the range 0 – 22050 Hz. A total of 128 frequency bands have been considered.

MFCC – The Mel-Frequency Cepstral Coefficients (MFCCs) are other features extracted by applying a Discrete

Cosine Transform (DCT) to the log-compressed Mel scale power spectrum [18]. A total of 40 MFCCs have been extracted.

Chroma – The chroma feature represents the tonal content of an audio signal in a compact form [19] and indicates how much energy of each pitch class is present in the audio signal. We consider 12 chroma bins at each frame. Although these features have been designed for a powerful representation of music, they can further improve classification in our case since are strongly related to the harmonic progression of vehicle engine signals [20].

Contrast – The spectral contrast [21] is evaluated by dividing the spectrogram into different sub-bands. For each sub-band, the energy contrast is estimated by a comparison of the mean energy in the top quantile to that of the bottom quantile. High contrast values generally correspond to clear, narrow-band signals, while low contrast values correspond to broad-band noise. We consider 7 coefficients for this feature.

The whole feature vector is then obtained by the concatenation of these four sets of coefficients, for a total of 187 elements.

III. Experimental setup

A. Dataset

Audio data of equipment operations has been collected in several construction sites consisting of diverse construction machines and equipments. The activities of these machines were observed during certain periods, and the audio signals generated were recorded accordingly. A Zoom H1 digital handy recorder has been used for data collection purposes. All files have been recorded by using a sample rate of 44,100 Hz and eight different files for each machine are available. About 15 minutes of sound data has been recorded for each class. The different files are referring to a single machine per class recorded during these are in use and working in similar environmental conditions.

Unlike artificially built datasets, when working with real data different problems arise, such as noise due to weather conditions and/or workers talking among themselves. Thus, we focused our work on the classification of a reduced number of classes, specifically: *Backhoe JD50D Compact*, *Compactor Ingersoll Rand*, *Concrete Mixer*, *Excavator Cat 320E*, *Excavator Hitachi 50U*. There are a couple of compactors and excavators, since machines of different brands sound in a different way. Classes which did not have enough usable audio (too short, excessive noise, low quality of the audio) were ignored for this work. For all of the five classes, approximately 75 minutes of audio are available and have been used to train and test the proposed architecture.

B. Data Preprocessing

All files have been preemptively pre-processed and the silence segments (frames where the Root Mean Square (RMS) is under the threshold of -30 dB) have been removed. In order to feed the network with enough and proper data, each audio file for each class is segmented into fixed length frames. In this work, two frame sizes have been considered: 30 ms and 50 ms, respectively. Experimental results will be performed on both these frame sizes. As first step, we split the original audio files into two parts: training samples (75% of the original dataset) and test samples (25% of the original dataset). This is equivalent to about 56 minutes of audio data for training the architecture and 18 minutes for testing it. We assured that instances of the training and test sets become from separate files.

After the partition in frames, we obtain 67.520 training and 22.514 testing audio segments of 50 ms or 112.412 training and 37.472 testing audio segments of 30 ms, all with 187 features per frame.

Using the Python library `librosa`[1] [22], we extract the waveform of the audio tracks from the audio samples and, using the same library, we generate the log-scaled Mel spectrogram of the signal and the other features.

IV. Experimental results

In order to validate the proposed architecture, we use the Python library `keras`[2] to build the DRNN with LSTM units as a Sequential model. The first two layers of this model are the keras CuDNNLSTM, classical LSTM layers implemented with CuDNN to run only on the GPU with TensorFlow backend. The NVIDIA CUDA® Deep Neural Network library (cuDNN) is a GPU-accelerated library of primitives and highly performance routines for deep learning. The last level of the model is a dense layer, a regular layer of neurons in a neural network. Each neuron receives input from all the neurons in the previous layer, thus densely connected. A final softmax layer implements the output classifier.

Simulations have been carried out by using a computer equipped with an Intel® i7-8700K CPU @ 3.70GHz, with an NVIDIA GeForce GTX 1080.

The hyper-parameters used in the experiments are the following. The spectral features have been extracted by implementing the FFT using a Hann window with a length of 2048 samples and an overlap of 512 samples. A total of 128 Mel bands, 40 MFCCs and 7 spectral contrast coefficients have been considered. A batch size of 128 and 512 samples is used for the frame size of 30 ms and 50 ms, respectively. The DRNN is composed of two LSTM layers, whose outputs have 128 and 32 units, respectively, a dense layer and an output softmax layer for the classification of the 5 classes. The categorical cross-entropy in (1) has been used as loss. The Adam optimizer has been employed with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a fixed learning rate of 0.001. The training run for a total of 100 epochs.

The proposed architecture has been compared to other two state-of-the-art approaches. Specifically, we compare it to the method exploiting the Deep Convolutional Neural Network (DCNN) already implemented in [14] for the same dataset

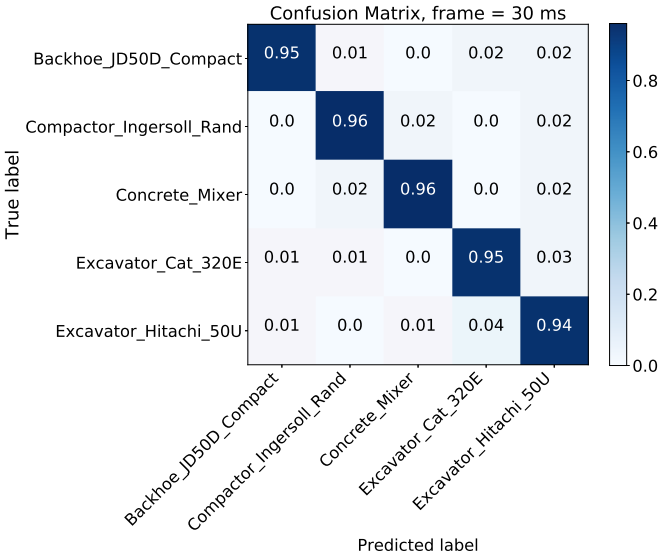
[1] Available at: https://librosa.github.io/librosa/

[2] Available at: https://keras.io/

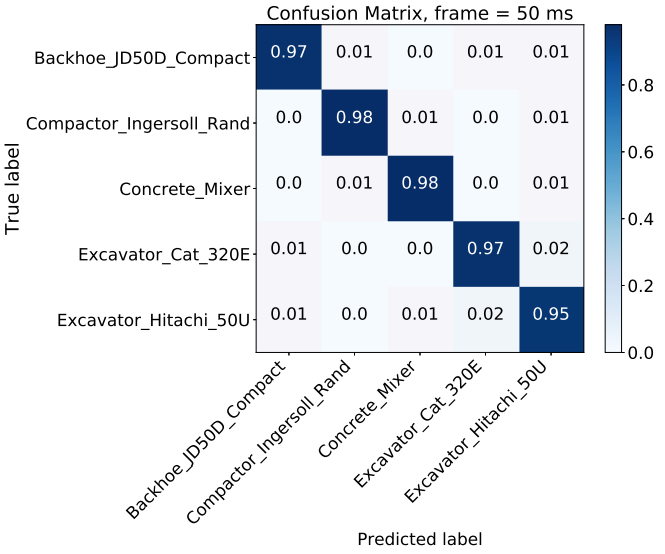
and one based on a four-layers Deep Belief Network (DBN) [10] that we implemented by using the same set of features from the same dataset. The main hyper-parameters have been set by a grid search and are summarized as follows. For the DCNN we have used 5 convolutional layers, followed by a dense layer and a soft-max layer as output. All layers use the ReLU activation function. A batch size of 64 has been used, while in the dense layer a dropout with rate 0.3 is used. The loss function is the cross-entropy and the Adam optimizer with a learning rate of 0.0005 has been chosen. A total of 100 training epoch has been run. However, a complete set of hyper-parameters can be found in [14]. For the DBN we have used 4 hidden layers plus an output soft-max layer. The hidden layer used a Gaussian activation function with a batch size of 32 samples. The training is based on the contrastive divergence algorithm [10] and the Adam optimizer has been used once again. A total of 240 training epochs has been run.

Numerical results have been evaluated in terms of accuracy, recall, precision and F_1 score [23]. Specifically, Table I shows these indicators for both the frames sizes. The table recaps the accuracy, recall, precision and F_1 score for each class, while the last line shows the overall index, obtained as a weighted average among all classes. Table I shows also that both the frame sizes are good choices for obtaining a high accuracy. However, a frame size of 50 ms is able to achieve the very high overall accuracy of 97%, while the accuracies for each class is never less than 95%. Moreover, Table I reveals that the class with the lower accuracy is the *Excavator Hitachi 50U*. This is due because some of the instances of this class are classified as the other excavator (Cat 320E) class present in the dataset. The normalized confusion matrices for both the frame sizes are shown in Fig. 3 that clearly shows the effectiveness of the proposed approach.

Table II compares the proposed approach with the other considered ones. Specifically, this table shows the results for both the frame sizes in terms of both training and testing overall accuracies. As it can be seen, the training accuracy is always better than the test one. Moreover, Table II shows that, differently from the proposed approach, the other considered methods achieve an higher accuracy by using a frame size of 30 ms instead of 50 ms. This is true for both the train and test accuracies. From this table it is quite clear that the proposed approach provides generally better numerical results, even if results of the DCNN are close enough providing a similar overall accuracy that exceed 97%. The DBN performs worse among the considered approaches, since its accuracy converges to 87.5% and the number of epochs needed are greater than the other approaches.



(a) Frame size: 30 ms



(b) Frame size: 50 ms

Fig. 3: Confusion matrices obtained by the proposed approach for frame sizes of (a) 30 ms and (b) 50 ms.

V. Conclusion

In this paper we have presented a Deep Recurrent Neural Network (DRNN) approach based on the LSTM units for the classification of audio data recorded in construction sites. Such architecture works with small audio frames and, for practical applications, the ability to perform a classification using very short samples can lead to the possibility to use such a network in time-critical applications in construction sites that require fast responses, such as hazard detection and activity monitoring. Up to now, the proposed architecture was tested on five classes related to vehicles and tools, obtaining an overall accuracy of 97%. The input to the DRNN consists in the concatenation of four sets of spectral features evaluated by the spectrogram of each audio frame.

Future works will be addressed to the increase of the number of classes to include more tools and vehicles employed in building sites, in order to lead to a more reliable and useful system. Moreover, the most interesting way to extend the work would be to test other deep learning architectures and try to combine different architectures in order to establish which kind

TABLE I: Performance metrics obtained by the proposed DRNN approach for the considered frame sizes.

	30 [ms]				50 [ms]			
Class	**Accuracy**	**Precision**	**Recall**	**F_1-score**	**Accuracy**	**Precision**	**Recall**	**F_1-score**
Backhoe JD50D Compact	0.95	0.96	0.95	0.96	0.97	0.98	0.97	0.97
Compactor Ingersoll Rand	0.96	0.95	0.96	0.96	0.98	0.97	0.98	0.98
Concrete Mixer	0.96	0.95	0.96	0.96	0.98	0.97	0.98	0.97
Excavator Cat 320E	0.95	0.94	0.95	0.95	0.97	0.96	0.97	0.97
Excavator Hitachi 50U	0.94	0.95	0.94	0.95	0.95	0.97	0.95	0.96
Overall	0.95	0.95	0.95	0.95	0.97	0.97	0.97	0.97

TABLE II: Train and test overall accuracy [%] for the proposed DRNN approach and compared ones.

	DRNN		**DCNN** [14]		**DBN**	
Frame Duration [ms]	Train	Test	Train	Test	Train	Test
30	97.10	95.32	98.23	**97.08**	90.49	**87.52**
50	98.91	**97.13**	96.95	95.74	90.02	87.40

of neural network approach can help the audio classification in construction sites.

REFERENCES

[1] S. Scardapane, M. Scarpiniti, M. Bucciarelli, F. Colone, M. V. Mansueto, and R. Parisi, "Microphone array based classification for security monitoring in unstructured environments," *AEÜ – International Journal of Electronics and Communications*, vol. 69, no. 11, pp. 1715–1723, November 2015.

[2] E. Weinstein, K. Steele, A. Agarwal, and J. Glass, "LOUD: a 1020-node modular micro-phone array and beamformer for intelligent computing spaces," MIT/LCS Technical Memo MIT-LCS-TM-642, Tech. Rep., 2004.

[3] B. Kaushik, D. Nance, and K. K. Ahuja, "A review of the role of acoustic sensors in the modern battlefield," in *Proc. of the 11-th AIAA/CEAS Aeroacoustics Conference*, Monterey, CA, USA, 23–25 May 2005, pp. 1–13.

[4] D. Wang and G. J. Brown, *Computational auditory scene analysis: principles, algorithms, and applications*. Wiley–IEEE Press, 2006.

[5] M. Golparvar-Fard, F. Peña-Mora, and S. Savarese, "Automated progress monitoring using unordered daily construction photographs and IFC-based building information models," *Journal of Computing in Civil Engineering*, vol. 29, no. 1, pp. 1–19, January 2015.

[6] B. Sherafat, A. Rashidi, Y.-C. Lee, and C. R. Ahn, "Hybrid kinematic-acoustic system for automated activity detection of construction equipment," *Sensors*, vol. 19, no. 4286, pp. 1–21, 2019.

[7] C.-F. Cheng, A. Rashidi, M. A. Davenport, and D. V. Anderson, "Activity analysis of construction equipment using audio signals and support vector machines," *Automation in Construction*, vol. 81, pp. 240–253, September 2017.

[8] T. Zhang, Y.-C. Lee, M. Scarpiniti, and A. Uncini, "A supervised machine learning-based sound identification for construction activity monitoring and performance evaluation," in *Proc. of 2018 Construction Research Congress (CRC 2018)*, New Orleans, Louisiana, USA, 2–4 April 2018, pp. 358–366.

[9] Y.-C. Lee, M. Scarpiniti, and A. Uncini, "Advanced sound identification classifiers using a grid search algorithm for accurate audio-based construction progress monitoring," *The ASCE Journal of Computing in Civil Engineering*, vol. 34, 2020.

[10] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. The MIT Press, 2016.

[11] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP 2015)*, Boston, MA, USA, 17–20 September 2015, pp. 1–6.

[12] Y. Tokozume and T. Harada, "Learning environmental sounds with end-to-end convolutional neural network," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017)*, New Orleans, LA, USA, 5–9 March 2017, pp. 2721–2725.

[13] S. Li, Y. Yao, J. Hu, G. Liu, X. Yao, and J. Hu, "An ensemble stacked convolutional neural network model for environmental event sound recognition," *Applied Sciences*, vol. 8, no. 7, 2018.

[14] A. Maccagno, A. Mastropietro, U. Mazziotta, M. Scarpiniti, Y.-C. Lee, and A. Uncini, "A CNN approach for audio classification in construction sites," in *Progresses in Artificial Intelligence and Neural Systems*, ser. Smart Innovation, Systems and Technologies, A. Esposito, M. Faundez-Zanuy, F. C. Morabito, and E. Pasero, Eds. Vietri sul Mare, Salerno, Italy: Springer, 12–14 June 2020, vol. 184.

[15] R. V. Sharan and T. J. Moir, "An overview of applications and advancements in automatic sound recognition," *Neurocomputing*, vol. 200, pp. 22–34, August 2016.

[16] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference for Learning Representations (ICLR 2015)*, San Diego, USA, 7–9 May 2015. [Online]. Available: https://arxiv.org/abs/1412.6980

[17] S. S. Stevens, V. John, and E. B. Newman, "A scale for the measurement of the psychological magnitude pitch," *The Journal of the Acoustical Society of America*, vol. 8, no. 1, pp. 185–190, January 1937.

[18] F. Zheng, G. Zhang, and Z. Song, "Comparison of different implementations of MFCC," *Journal of Computer Science and Technology*, vol. 16, no. 6, pp. 582–589, November 2001.

[19] M. Müller, *Fundamentals of Music Processing*. Springer, 2015.

[20] A. Valada, L. Spinello, and W. Burgard, "Deep feature lerning for acoustic-based terrain classification," in *Robotic Research*, A. Bicchi and W. Burgard, Eds. Springer, 2018, vol. 2, pp. 21–37.

[21] D.-N. Jiang, L. Lu, H.-J. Zhang, J.-H. Tao, and L.-H. Cai, "Music type classification by spectral contrast feature," in *2002 IEEE International Conference on Multimedia and Expo (ICME'02)*, Lausanne, Switzerland, 26–29 August 2002, pp. 113–116.

[22] B. McFee, V. Lostanlen, M. McVicar, A. Metsai, S. Balke, C. Thomé, C. Raffel, A. Malek, D. Lee, F. Zalkow, K. Lee, O. Nieto, J. Mason, D. Ellis, R. Yamamoto, S. Seyfarth, E. Battenberg, R. Bittner, K. Choi, J. Moore, Z. Wei, S. Hidaka, P. Friesch, F.-R. Stöter, D. Hereñú, T. Kim, M. Vollrath, and A. Weiss, "librosa/librosa: 0.7.2," January 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3606573

[23] D. M. W. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.