A Neural Network Approach for Sound Event Detection in Real Life Audio

Michele Valenti, Dario Tonelli, Fabio Vesperini, Emanuele Principi, Stefano Squartini
Università Politecnica delle Marche, Department of Information Engineering, Ancona, Italy
Email: m.valenti@staff.univpm.it

***Abstract*—This paper presents and compares two algorithms based on artificial neural networks (ANNs) for sound event detection in real life audio. Both systems have been developed and evaluated with the material provided for the third task of the Detection and Classification of Acoustic Scenes and Events (DCASE) 2016 challenge. For the first algorithm, we make use of an ANN trained on different features extracted from the down-mixed mono channel audio. Secondly, we analyse a binaural algorithm where the same feature extraction is performed on four different channels: the two binaural channels, the averaged monaural signal and the difference between the binaural channels. The proposed feature set comprehends, along with mel-frequency cepstral coefficients and log-mel energies, also activity information extracted with two different voice activity detection (VAD) algorithms. Moreover, we will present results obtained with two different neural architectures, namely multilayer perceptrons (MLPs) and recurrent neural networks. The highest scores obtained on the DCASE 2016 evaluation dataset are achieved by a MLP trained on binaural features and adaptive energy VAD; they consist of an averaged error rate of 0.79 and an averaged F1 score of 48.1%, thus marking an improvement over the best score registered in the DCASE 2016 challenge.**

I. Introduction

Automatic sound event detection (SED), also known as acoustic event detection, is nowadays considered as one of the most important topics in the field of computational auditory scene analysis (CASA). Thanks to works like Bregman's "Auditory Scene Analysis: The Perceptual Organization of Sound" [1] we can trace back the birth of CASA to 1994, when the field of auditory scene analysis (ASA) was firstly introduced in order to model humans' sound perception. Following this work, many other contributions were written to describe how artificial systems can be designed to mimic human perception; most of these works will be later collected in Divenyi's book [2] in 2004.

SED is defined as the task of analysing a continuous audio signal in order to extract a description of the sound events occurring in the audio stream. This description is commonly expressed as a label that marks the start, the ending, and the nature (*e.g.,* children crying, cutlery, glass jingling) of the occurred sound. In particular, in multi-label SED it is assumed that more than one event can be active (and should be detected) at a time, therefore foreseeing the overlapping of two or more of these labels. This problem is addressable as a "mixture problem" and it is usually not trivial to solve mainly due to the superimposition (in the audio spectral domain) of energies belonging to the different events and to the presence of acoustic non-idealities such as noise and reverberation [3], [4].

Labels extracted with a SED system usually allow us to achieve a better insight of the considered acoustic scenario, for instance they can be used as mid-level representation useful in other CASA research areas. In [5], [6] authors make use of SED for designing audio context recognition systems, while in [7] and [8] SED is exploited for automatic tagging and audio segmentation respectively. Moreover, SED also found many direct applications in a variety of scenarios, some examples being context-based indexing and retrieval in multimedia databases [9], unobtrusive health monitoring [10], and audio-based surveillance [11]–[13].

As we can notice from [6], [10], hidden Markov models (HMMs) have been widely used in the literature with the purpose of modelling acoustic events in a SED system. In recent years, new approaches to SED have been proposed, marking a distinct trend towards the use of artificial neural networks (ANNs). An interesting comparison between computational costs of different systems is carried out in [14] highlighting that ANNs are able to achieve top performance at the cost of being the most computationally expensive approach. A brilliant example of such performance is given in [15], where different ANNs are trained on a big video dataset and then used for different scopes, among which we can also find SED. For a wider overview of the most recent SED techniques the reader can refer to the comprehensive analysis carried out by Sharan *et al.* in [16].

In occasion of the Detection and Classification of Acoustic Scenes and Events (DCASE) 2016 challenge, many novel systems featuring recurrent neural networks (RNNs) and multilayer perceptrons (MLPs) have been proposed, even though only one of them [17] managed to outperform the baseline system (based on a Gaussian mixture model (GMM)). Due to this fact, it is the authors' opinion that there is still a lot of space for research in approaching SED with ANNs.

In this paper we propose a system which, for the first time (up to the authors' knowledge), relies on a voice activity detection (VAD) algorithm for the detection of acoustic events; events which, after being detected, are then classified by an ANN. During our experiments we compare different well-established audio features, *i.e.,* log-mel energies and mel-frequency cepstral coefficients (MFCCs), extracted in both monaural and binaural configuration. Moreover, we will evaluate two VAD algorithms (*i.e.,* adaptive energy (AE) and Sohn's

ISBN 978-0-9928626-7-1

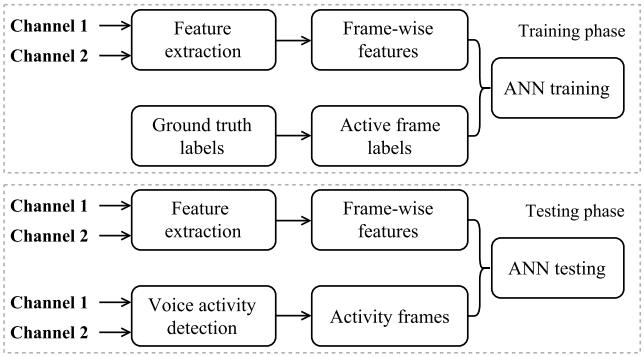

Fig. 1. Block diagram of training and testing phases. At test time a VAD algorithm is used to determine onset and offset instants of the detected events.

VAD), as well as two different ANN architectures, *i.e.,* MLPs and RNNs. Our aim is therefore to give a novel contribution by presenting a robust system capable to improve the results obtained by participants to the DCASE 2016 challenge.

Here is the outline of the paper. In Section II we introduce the method proposed for the SED task, we therefore describe the feature extraction processes, the proposed neural architectures and the VAD algorithms we tested. In Section III we describe the dataset and the metrics used to evaluate our system, and, together with our experimental setup, we report our main results. Finally, in Section IV we draw our conclusions and highlight some possibilities for future development of new SED systems.

II. Proposed method

As we can see from Figure 1 it is possible to divide the system functioning into two phases: training and testing. During training we do not need to use any algorithm for event detection, since onset and offset instants are already provided in the ground truth, therefore we can simply train an ANN to recognise the different events. At test time, on the other hand, onset and offset instants are not given, therefore we firstly make use of a VAD algorithm for determining them, and secondly we feed the corresponding audio sequences to the ANN classifier.

A. Feature representations

In order to perform the SED task with ANNs, a set of one or more audio representations is typically extracted from the raw audio signal, that is the *feature set*. Aiming to evaluate the impact of binaural information on the classification performance, we will, in the first instance, distinguish the proposed sets between monaural and binaural feature sets. We highlight that, for all the extracted feature sets, a frame-wise short-time Fourier transform (STFT) is firstly applied to the audio signal on frame windows of 40 ms with 50% overlap. Moreover, all feature extraction processes described hereafter have been performed with openSMILE [18], a license-free software package developed by the Technical University of Munich.

The first monaural set is known as log-mel spectrogram. In this case a down-mixing of the two audio channels is required before calculating the STFT coefficients. After the STFT coefficients are extracted, we apply a mel conversion of the frequency scale with a 26-bands mel-scale filter bank and compute the logarithm of all the energies so obtained. To complete the set, we also extract the first order delta coefficients operating on a context window of 10 frames. Given the log-mel coefficients and their respective deltas, this first set is composed of 52 coefficients for each frame.

The second monaural set is composed of another set of widely used features, that is MFCCs. Starting from the same STFT coefficients previously obtained, we now compute the log-mel spectrogram with a 40-bands mel-scale filter bank. Then, we apply a discrete cosine transform (DCT) to each energy vector and, after excluding the 0^{th} order coefficient, we obtain a feature vector of 20 MFCCs. In order to complete the set, we also calculate the first and second order delta coefficients, therefore obtaining a longer feature vector of 60 coefficients.

For the two binaural sets we extract the log-mel (or the MFCC) features not only from the average of the two channels, but also from their difference and the two channels separately; this gives us a total of four channels. We decided to do so because, for example, if an important event is predominant in only one of the two channels, averaging them could lower the signal-to-noise ratio, thus increasing the system failure probability. For the first binaural set we extract the log-mel coefficients as for the first monaural set, but, given the presence of four channels, we now have a total of 108 coefficients for each frame. In order to avoid an excessive feature vector dimension, in this set we avoid using the first and second order delta coefficients. Similarly, the second binaural set is obtained by extracting 20 MFCCs for each channel; since again we avoid using the delta coefficients, this leads us to a total of 80 coefficients for each frame.

B. Neural networks

In this work two different ANNs architectures are tested for the SED problem, *i.e.,* MLPs and RNNs. The first layer of all the examined models consists of a set of nodes to which the audio representation (taken on a frame scale) is applied, with the number of nodes varying from 52 to 108, depending on the chosen feature representation.

The input is then propagated to the following three hidden layers, composed of 512 tanh neurons (each) for MLPs and 54 rectifier neurons for RNNs. Finally, the last layer of our networks is designed to output the class associated by the network with the given input. To do so, this layer is composed of a number of softmax neurons equal to the number of possible classes, *i.e.,* 11 if we are dealing with a "home" scenario, and 7 in case of a "residential area" (see Table I).

C. Sound event detection and classification

At test time we want the system to detect and correctly classify as many acoustic events as possible which occur in

ISBN 978-0-9928626-7-1

raw audio files of 30 seconds. Since no onset nor offset instants are given, we decide to use VAD algorithms in order to detect these instants. With this intent, two different VAD algorithms have been tested, *i.e.,* AE, and Sohn's VAD.

The AE approach makes use of two energy thresholds in order to determine the starting and ending point of an event-active audio sequence, these thresholds being the "mean plus variance" (MPV) and the "mean minus variance" (MMV). These numbers are firstly calculated over all the training dataset and then used to extract information about events activity: whenever a frame's energy exceeds the MPV threshold an onset event is triggered. Then, the event detection remains positive until the energy content drops below the MMV threshold.

Sohn's VAD [19], on the other hand, is a method based on a statistical modelling of the audio in the time-frequency domain, with the model parameters being estimated with a maximum likelihood (ML) method. With this technique, the decision regarding the event's activity is devolved to a comparison between the averaged log-likelihood ratio (containing the a-priori and a-posteriori signal-to-noise ratios) and a fixed threshold η, with $\eta \in (0, 1)$.

Whenever an audio file is processed by one of the two VAD algorithms we are able to extract the starting and ending instants between which an audio event has (supposedly) occurred. Hence, we can feed the network with the feature representation of the corresponding frames and finally obtain the event classification. We remind that, in case of MLPs trained on a frame base, we can only obtain one label for each frame, whereas RNNs are also able to output one label for an entire batch of frames. Due to this, we need to average all MLP outputs so to obtain the event's acoustic label, whereas we decide to let RNNs output only one label for each batch of frames corresponding to a detected event.

III. Experiments

A. Datasets and metrics

The data we used during our experiments consists of the two datasets provided for the third task (SED in real life audio) of the DCASE 2016 challenge [20], both of them containing recordings of 3-5 minutes divided into two different acoustic scenarios: "home" and "residential area". For each scenario different classes were defined, and we report them in Table I.

The first dataset, called *development dataset*, was at first provided in order to make all challengers able to compare their development results. The second (*evaluation*) dataset, on the other hand, is used for the final evaluation of the submitted systems. We highlight that the ground truth for the evaluation dataset is still not public at present time, therefore the scores presented in this paper were calculated by the challenge organizers on the results we submitted to them.

The development dataset consists of 10 recordings for the "home" scenario, and 12 for the "residential area", and for both a four-folds cross-validation data splitting is provided by the organizers of the challenge. While creating the cross-validation folds, the challenge organizers requested that the test subset does not contain classes unavailable in training subsets, therefore the class distribution between the test subsets is not assumed to be uniform.

TABLE I
Classes and their occurrences for the "home" and "residential area" scenarios for the SED in real life audio task of the DCASE 2016 challenge.

Home	*Occurrences*	*Residential area*	*Occurrences*
rustling	60	banging	23
snapping	57	bird singing	271
cupboard	40	car passing by	108
cutlery	76	children shouting	31
dishes	151	people walking	52
drawer	51	people speaking	44
glass jingling	36	wind blowing	30
object impact	250		
people walking	54		
washing dishes	84		
water tap running	47		

The evaluation dataset contains 5 recordings for both the "home" and the "residential area" scenarios each. For this dataset no cross-validation is performed, so it is possible to train only one system with all the development dataset (including files previously meant for testing purpose) and then test it with the evaluation files.

Scores used to evaluate all systems are the F1 and error rate (ER) scores, which are used to evaluate the system over segments of one second. Following the notation introduced in [20], for the evaluation purpose an event can be a: true positive (TP), if both the system and the ground truth indicate it as active; a false positive (FP), if the system indicates it as active, but it is not present in the ground truth; a false negative (FN), if the system does not detect it, but it is active in the ground truth. With this notation it is possible to define the precision (P), the recall (R), and the F1 score of the system as:

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \quad \mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}, \quad \mathrm{F1} = \frac{2\cdot\mathrm{P}\cdot\mathrm{R}}{\mathrm{TP}+\mathrm{FN}}. \tag{1}$$

Concerning the ER, we must divide all possible errors into three categories: substitutions (S), insertions (I), and deletions (D). A substitution occurs when the system correctly detects an event but gives it the wrong label; moreover, we consider insertions all those FPs which are not substitutions, whereas we call deletions all those FNs which are not substitutions. According to this notation we define the ER as:

$$\mathrm{ER} = \frac{\sum_{k=1}^{K}\mathrm{S}(k) + \sum_{k=1}^{K}\mathrm{D}(k) + \sum_{k=1}^{K}\mathrm{I}(k)}{\sum_{k=1}^{K}\mathrm{N}(k)}, \tag{2}$$

where $\mathrm{N}(k)$ is the number of active events in the ground truth, and k is the segment's index. Finally we highlight that, in obtaining the final score for the development dataset, we average the four per-fold scores as described in [20].

B. Experimental setup

Concerning the network training, we initialize all weights according to a normal distribution with zero mean and 0.1

ISBN 978-0-9928626-7-1

variance. We then train the networks following the Adam [21] method for stochastic optimization, for which we keep the default hyper-parameter configuration, as specified in [21] and implemented in the Lasagne [22] Python library.

In order to prevent overfitting, for each fold we check the network performance on the respective fold's test set after each training epoch. If no improvement on this set is encountered for 60 consecutive epochs, the training is forced to an early stop. By doing so we are able to fine-tune the network hyper-parameters and obtain the architectures proposed in Section II.

After this phase we perform experiments on the evaluation data, for which we use the whole development training and test sets as training and validation data respectively. Then, at test time, we evaluate the system on the secret challenge data.

C. Development results

As introduced in Section II, during our experiments we tested and compared different neural architectures, VAD algorithms, and feature representations. In Table II we report the results obtained with Sohn's VAD for 16 different system configurations, whereas in Table III the same classifier and feature configurations are analysed in conjunction with AE VAD.

Table II highlights that the use of binaural audio features always enhances the system's performance in terms of both F1 and ER scores. Moreover, we can also notice that MLPs generally perform better than RNNs, in particular according to F1 scores, where no RNN manages to achieve more than 34.4% F1 score. Finally, we report that the best system's configuration featuring Sohn's VAD is a MLP trained with binaural MFCC features, with a VAD threshold equal to 0.70. This system manages to reach 0.88 ER and 39.8% F1 score, both averaged on the four folds.

Table III mostly confirms what emerged from the analysis of the previous table. Also with adaptive evergy VAD, the use of binaural features always improves the classification accuracy, even if differences are now less marked, with the highest improvement in F1 scores being +2%. Moreover, it is interesting to notice that the difference between MLPs and RNNs accuracies is now reduced, maybe highlighting that the difference between their classification power thins if a better VAD algorithm leads to a better event detection. The best performing system featuring AE VAD is again a MLP which, with binaural log-mel features, manages to reach 0.78 ER and 43.1% F1 scores, averaged on the four folds as for the previous results.

D. Evaluation results

In Table IV we report the main results for the most promising system configurations tested on the evaluation dataset. As we can see, scores tend to be higher than the ones obtained on the development dataset, especially for MLPs, highlighting the benefit introduced by the addition in the training set of those files previously used for testing. The expansion of the training set can be viewed as the expansion of the "knowledge" from which the network can learn at training time, therefore, when

TABLE II
COMPARISON OF SCORES OBTAINED ON THE DEVELOPMENT DATASET USING DIFFERENT FEATURES, CLASSIFIERS AND SOHN'S VAD THRESHOLDS (η). SCORES ARE AVERAGED AMONG THE FOUR CROSS-VALIDATION FOLDS.

Features	η	*Classifier*	*ER*	*F1* (%)
Monaural log-mel	0.98	MLP	0.93	34.6
Binaural log-mel	0.98	MLP	0.89	38.6
Monaural log-mel	0.70	MLP	0.90	35.4
Binaural log-mel	0.70	MLP	0.89	39.4
Monaural MFCC	0.98	MLP	0.92	35.7
Binaural MFCC	0.98	MLP	0.88	39.6
Monaural MFCC	0.70	MLP	0.91	36.2
Binaural MFCC	**0.70**	**MLP**	**0.88**	**39.8**
Monaural log-mel	0.98	RNN	0.91	29.6
Binaural log-mel	0.98	RNN	0.88	35.6
Monaural log-mel	0.70	RNN	0.95	28.2
Binaural log-mel	0.70	RNN	0.88	34.4
Monaural MFCC	0.98	RNN	0.98	30.5
Binaural MFCC	0.98	RNN	0.88	34.1
Monaural MFCC	0.70	RNN	0.91	31.2
Binaural MFCC	0.70	RNN	0.88	31.0

TABLE III
COMPARISON OF SCORES OBTAINED ON THE DEVELOPMENT DATASET USING DIFFERENT FEATURES, CLASSIFIERS AND AE VAD. SCORES ARE AVERAGED AMONG THE FOUR CROSS-VALIDATION FOLDS.

Features	*Classifier*	*ER*	*F1* (%)
Monaural log-mel	MLP	0.78	41.2
Binaural log-mel	**MLP**	**0.78**	**43.1**
Monaural MFCC	MLP	0.81	40.1
Binaural MFCC	MLP	0.82	42.1
Monaural log-mel	RNN	0.85	41.2
Binaural log-mel	RNN	0.82	43.1
Monaural MFCC	RNN	0.92	40.7
Binaural MFCC	RNN	0.89	41.0

TABLE IV
COMPARISON OF SCORES OBTAINED ON THE EVALUATION DATASET USING DIFFERENT FEATURES, CLASSIFIERS AND VAD ALGORITHMS.

Features	*VAD*	*Classifier*	*ER*	*F1* (%)
Monaural log-mel	Sohn ($\eta = 0.70$)	MLP	0.80	40.2
Binaural log-mel	Sohn ($\eta = 0.70$)	MLP	0.78	46.5
Monaural MFCC	AE	MLP	0.79	45.1
Binaural MFCC	**AE**	**MLP**	**0.78**	**48.1**
Monaural MFCC	AE	RNN	0.82	41.0

ISBN 978-0-9928626-7-1 © EURASIP 2017

TABLE V
COMPARISON BETWEEN THE PROPOSED SYSTEM AND THE THREE (OUT OF 17) BEST PERFORMING DCASE 2016 SYSTEMS PROPOSED FOR SED IN REAL LIFE AUDIO.

Features	*VAD*	*Classifier*	*ER*	*F1 (%)*
Binaural log-mel	**AE**	**MLP**	**0.79**	**48.1**
Binaural mel energy	-	RNN [17]	0.81	47.8
Binaural mel energy	-	GMM [20]	0.88	23.7
Binaural mel energy + TDOA	-	RNN [17]	0.89	34.3

this happens, it is expectable to reach a better generalization performance. This behaviour is confirmed, the best performing configuration manages to achieve a 0.79 ER and 48.1% F1 scores, and it consists of a MLP classifier trained on binaural MFCC features.

Table V compares our best system to the three best performing ones proposed for the third task of the DCASE 2016 challenge. The first and the third ranks were achieved by Adavanne *et al.*, which made use of RNN-LSTM architectures trained on spatial and harmonic features [17] extracted from the two binaural channels. On the other hand, the second best system is the baseline proposed in [20], based on a GMM modelling of each acoustic event, plus one for the absence of sound events, which was trained with the non-labelled frame's features (MFCCs and their delta/delta-deltas were used). As we can see from the table, the proposed system manages to improve the F1 score by 0.3% while reducing the error rate by 0.02.

IV. CONCLUSIONS AND FUTURE WORK

In this paper we proposed and evaluated a system for SED in real life audio. We compared different audio features, extracted in both monaural and binaural configurations, with which we trained different neural network classifiers. Moreover, we tested two different VAD algorithms for detecting sound activities to be classified by the proposed networks at test time. The proposed best performing system achieves an improvement on the winner of the third task in the DCASE 2016 challenge, thus highlighting the competitiveness of the proposed approach. Finally, given the improvement carried by the use of binaural features, we believe that future work should address the development of a binaural algorithm using one or more networks for each channel and a decision function for a decision fusion stage.

REFERENCES

[1] A. S. Bregman, *Auditory Scene Analysis: The Perceptual Organization of Sound*. MIT press, 1994.

[2] P. Divenyi, *Speech Separation by Humans and Machines*. Springer Science & Business Media, 2004.

[3] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, "Acoustic event detection in real life recordings," in *Proc. of the European Signal Processing Conference*, Aalborg, Denmark, 2010, pp. 1267–1271.

[4] R. Rotili, E. Principi, S. Squartini, and B. Schuller, "A real-time speech enhancement framework in noisy and reverberated acoustic scenarios," *Cognitive Computation*, vol. 5, no. 4, pp. 504–516, Aug. 2013.

[5] S. Chu, S. Narayanan, and C.-C. J. Kuo, "Environmental sound recognition with time-frequency audio features," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 6, pp. 1142–1158, 2009.

[6] T. Heittola, A. Mesaros, A. Eronen, and T. Virtanen, "Audio context recognition using audio event histograms," in *Proc. of IEEE 18th European Signal Processing Conference*, 2010, pp. 1272–1276.

[7] M. Shah, B. Mears, C. Chakrabarti, and A. Spanias, "Lifelogging: Archival and retrieval of continuously recorded audio using wearable devices," in *Proc. of IEEE International Conference on Emerging Signal Processing Applications (ESPA)*, 2012, pp. 99–102.

[8] G. Wichern, J. Xue, H. Thornburg, B. Mechtley, and A. Spanias, "Segmentation, indexing, and retrieval for environmental and natural sounds," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 688–707, 2010.

[9] M. Xu, C. Xu, L. Duan, J. S. Jin, and S. Luo, "Audio keywords generation for sports video analysis," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 4, no. 2, p. 11, 2008.

[10] Y.-T. Peng, C.-Y. Lin, M.-T. Sun, and K.-C. Tsai, "Healthcare audio event classification using hidden Markov models and hierarchical hidden Markov models," in *Proc. of IEEE International Conference on Multimedia and Expo (ICME)*, 2009.

[11] A. Harma, M. F. McKinney, and J. Skowronek, "Automatic surveillance of the acoustic activity in our living environment," in *Proc. of IEEE International Conference on Multimedia and Expo (ICME)*, 2005.

[12] M. Crocco, M. Cristani, A. Trucco, and V. Murino, "Audio surveillance: a systematic review," *ACM Computing Surveys (CSUR)*, vol. 48, no. 4, p. 52, 2016.

[13] E. Principi, D. Droghini, S. Squartini, P. Olivetti, and F. Piazza, "Acoustic cues from the floor: a new approach for fall classification," *Expert Systems with Applications*, vol. 60, pp. 51–61, 2016.

[14] S. Sigtia, A. M. Stark, S. Krstulović, and M. D. Plumbley, "Automatic environmental sound recognition: Performance versus computational cost," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 2096–2107, 2016.

[15] S. Hershey, S. Chaudhuri *et al.*, "CNN architectures for large-scale audio classification," *Accepted for Publication at ICASSP 2017*, 2016.

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

[17] S. Adavanne, G. Parascandolo, T. Heittola, and T. Virtanen, "Sound event detection in multichannel audio using spatial and harmonic features," in *Proc. of Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE)*, 2016.

[18] F. Eyben, F. Weninger, F. Gross, and B. Schuller, "Recent developments in opensmile, the munich open-source multimedia feature extractor," in *Proc. of the 21st ACM international conference on Multimedia*, 2013, pp. 835–838.

[19] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Processing Letters*, vol. 6, no. 1, pp. 1–3, 1999.

[20] A. Mesaros, T. Heittola, and T. Virtanen, "TUT database for acoustic scene classification and sound event detection," in *Proc. of IEEE 24th European Signal Processing Conference (EUSIPCO)*, 2016, pp. 1128–1132.

[21] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[22] S. Dieleman, J. Schlter *et al.*, "Lasagne: First release." 2015. [Online]. Available: http://dx.doi.org/10.5281/zenodo.27878

ISBN 978-0-9928626-7-1 © EURASIP 2017