SUPERVISED MODEL TRAINING FOR OVERLAPPING SOUND EVENTS BASED ON UNSUPERVISED SOURCE SEPARATION

Toni Heittola[⋆] *Annamaria Mesaros*[†] *Tuomas Virtanen*[⋆] *Moncef Gabbouj*[⋆]

[⋆] Department of Signal Processing, Tampere University of Technology
[†] Department of Information and Computer Science, Aalto University

ABSTRACT

Sound event detection is addressed in the presence of overlapping sounds. Unsupervised sound source separation into streams is used as a preprocessing step to minimize the interference of overlapping events. This poses a problem in supervised model training, since there is no knowledge about which separated stream contains the targeted sound source. We propose two iterative approaches based on EM algorithm to select the most likely stream to contain the target sound: one by selecting always the most likely stream and another one by gradually eliminating the most unlikely streams from the training. The approaches were evaluated with a database containing recordings from various contexts, against the baseline system trained without applying stream selection. Both proposed approaches were found to give a reasonable increase of 8 percentage units in the detection accuracy.

Index Terms— acoustic event detection, sound source separation, supervised model training, acoustic pattern recognition

1. INTRODUCTION

A *sound event* is a segment of audio which can be characterized and identified by a textual label. Sound events can be used to describe and understand the human and social activities. Automatic sound event detection aims at processing a continuous acoustic signal and converting it into a sequence of event labels with associated start times and end times. The sound event detection can be utilized in a variety of application areas, including context-based indexing and retrieval in multimedia databases [1, 2], unobtrusive monitoring in health care [3], and audio-based surveillance [4]. Furthermore, the detected events can be used as mid-level-representation in other research areas, e.g. audio context recognition [5, 6], automatic tagging [7], and audio segmentation [8].

Early research on sound event detection concentrated on detecting only one sound event at a time, considerably simplifying the detection problem [9, 10, 11]. Everyday auditory scenes are usually complex in sound events, having multiple overlapping sound events active at the same time. If an algorithm that detects only a single event at a time is applied to material consisting of overlapping events, the majority of detection errors will be caused by temporally overlapping sound events. In order to detect all sound events, a way to deal with overlapping events is needed. Recently, the problem of overlapping events has been addressed at various levels of the detection process. At the signal level, unsupervised sound source separation can be used to minimize the acoustical interference of overlapping sound sources [12]. In the acoustic model training, the overlapping events can be taken into account by modeling all possible event combinations as new intermediate classes [13, 14]. In the event detection stage, overlapping events can be detected with multiple iterative detection passes and by excluding already detected events from the following detection iterations until the desired amount of overlapping events have been reached [15]. In addition to these approaches, multiple audio signals and sound source localization methods along with video based methods can be used to better handle overlapping event in the detection [16].

In this paper, we tackle the problem of overlapping events by applying unsupervised sound source separation as a preprocessing stage for the event detection. In the source separation stage, the mixture signal is split into *streams* containing roughly homogeneous spectral content, each differing significantly from the other streams. Following the concept of noise adaptive training used in robust speech recognition [17], the same signal enhancement method should be applied both before model training and detection stages. Due to the unsupervised nature of the separation, there is no knowledge about which sound source is separated into which stream, making it challenging to take full advantage of the separated audio as such in the supervised model training.

We propose a method to train reliable acoustic event models by iteratively selecting the most appropriate training material from audio separated in an unsupervised manner. Prior knowledge about the temporal location of events given by annotations is used to get initial models for event classes. Two alternative approaches using expectation maximization (EM) algorithm to select the stream that contains the target sound are proposed: one selecting always the most likely stream and another gradually eliminating the most unlikely streams from the training. The proposed method is evaluated with a database recorded in realistic environments with a high degree of overlapping sound events. The method is compared to the baseline system trained without the stream selection. At the general level, this work extends our context-dependent sound event detection system presented in [12] with event priors and proposed model training approach.

The rest of this paper is organized as follows. Section 2 presents the sound event detection system for overlapping events, and Section 3 explains the model training using recordings with overlapping events. Section 4 presents the experimental results, and Section 5 discusses them. Section 6 provides conclusions and future work.

2. SOUND EVENT DETECTION

The overview of the sound event detection system is presented in Figure 1. Sound source separation is applied on the mixture signal to produce the streams $(S1, S2, S3, S4)$. In this study, the number of streams is fixed to four. Feature extraction and sound event

This work was financially supported by Academy of Finland under the grants 258708 and 265024

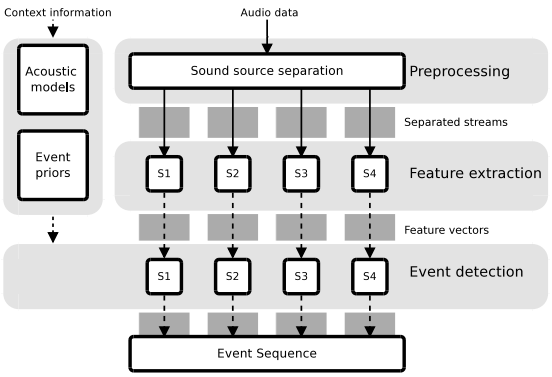

Fig. 1. Overview of sound event detection system.

detection are performed on each of these streams separately and the resulting event sequences are combined into a multi-source symbolic representation of the original signal.

In the event detection stage, a given context is used to select a context-specific set of events with context-specific acoustic event models and prior probabilities. This provides more accurate modeling, since many sound events are acoustically dissimilar across contexts [15]. Furthermore, some sound events are more likely than others, and the differences in occurrence rates are even more obvious between contexts.

2.1. Source separation

In the source separation stage, a given input audio signal that consists of multiple overlapping sounds (mixture signal) is decomposed into its sound sources (ideally). The proposed system utilizes an unsupervised sound source separation method based on non-negative matrix factorization (NMF) of the magnitude spectrogram of the mixture signal [18]. The method models the mixture signal as a sum of components, each having a fixed magnitude spectrum and a time-varying gain. Due to the unsupervised nature of the method, the outcome of the factorization cannot be strictly controlled. Sound sources in the mixture signal may get represented as the sum of one or more components, and at the same time each component can contain parts from one or more sound sources. However, typically the factorization achieves good separation of sound sources. A more detailed description of the separation algorithm is presented in [12].

Most of the sound events have diverse characteristics and they cannot be accurately modeled with fixed spectrums and time-varying gains. However, the function of the algorithm is better explained by reconstructing the streams with Wiener filtering: a time-varying Wiener filter of each component separates a stream which contains roughly homogeneous spectral content that differs significantly from the other streams. The resulting streams represent a combination of the physical sources present in the mixture signal, rather than exact physical sound sources. In this paper, the original multisource spectrum is split into four streams (number of components) limiting the the event detection to finding a maximum of four simultaneous sound events. This is in agreement with the average amount of overlapping events in our evaluation database.

2.2. Event models

The coarse shape of the power spectrum of the input signal is represented with 16 mel-frequency cepstral coefficients (MFCCs). In order to describe the dynamic properties of the cepstrum, first and second time derivatives of the static coefficients are also utilized. Features are calculated in 20 ms frames with 50 % overlap.

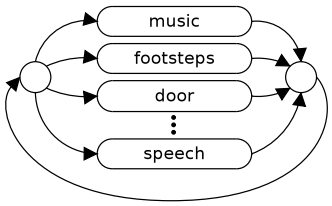

Fig. 2. Fully-connected sound event model network.

Continuous-density hidden Markov models (HMMs) with three state left-to-right topology are used to model sound-event-conditional feature distributions. The probability density functions of observations in each state are modeled with a mixture of multivariate Gaussian density functions (16 Gaussians). The model training process is described in detail in Section 3. In the testing stage, the trained sound event models are connected into a network with transitions from each model to any other. A model network is shown in Figure 2.

Manually annotated training recordings are used to estimate the event priors, i.e., transition probabilities in the network. Annotated events are regarded as a separate entities, and their event-lengths are accumulated (in precision of seconds). Normalized lengths of each event class are used as event priors.

2.3. Detection

Sound event detection is applied separately for each stream. This is obtained by applying Viterbi decoding inside the network of sound event models. The alignment of states and observations given by the Viterbi algorithm produces estimates of event segment boundaries and event labels. Detection results from each stream are merged into a single set of events as in [12].

When calculating the path cost through the model network, the balance between likelihoods provided by the event priors and the acoustic models is adjusted using a weight parameter. The number of events in the resulting event sequence is controlled by using a cost for inter-event transitions. Both these parameters are experimentally chosen using a development set, and are tuned so that the output has approximately equal amount of events as the manually annotated ground truth. A more detailed description of the detection stage is presented in [15].

3. MODEL TRAINING

In the the source separation stage, each original recording is split into four audio streams. The training material for an event class is selected based on annotated time-segments. Since the source separation is done in an unsupervised manner, there is no exact knowledge about which stream contains most suitable training material for the target sound event class. The problem is to select which of the four streams contains the target event class. In this work, we assume that there is always one single stream containing the target sound, and other three streams are regarded to contain overlapping events. The stream selection for training is illustrated in Figure 3.

Regardless of the stream selection, the overlapping events might still cause some interference and variability to the training material. However, this is assumed to be averaged out in the model training

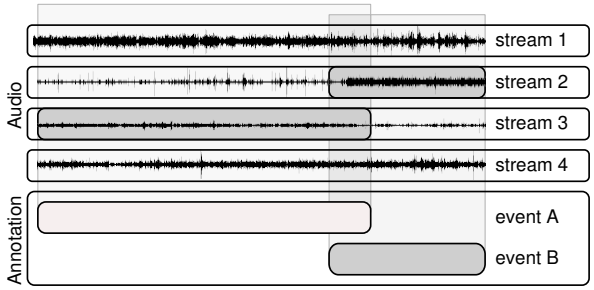

Fig. 3. Separated audio streams and material selection for model training. Annotated events A and B are separated into distinct streams 3 and 2.

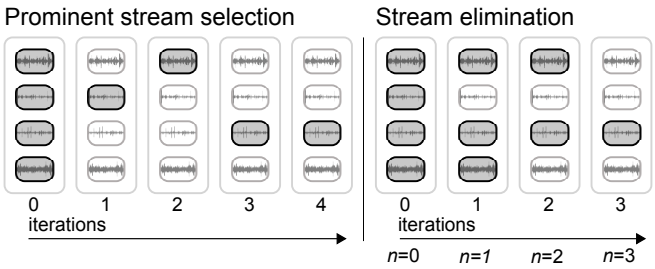

Fig. 4. Example of proposed stream selection approaches. Prominent stream selection: in each iteration only one $a_{s,m}$ is set to one, rest are zero. Stream elimination: in each iteration one more $a_{s,m}$ is set to zero.

due to the large training set, and the models will learn a reliable representation of the target sound events.

3.1. Expectation maximization algorithm

The iterative stream selection is based on expectation-maximization (EM) algorithm [19]. In order to simplify the notation, we present training of a single event class model λ. The described procedure is identical for each of the classes. An audio segment extracted from an annotated time-segment s in a stream with index m is denoted as $x_{s,m}$. Let us denote a set of events that are annotated to content target class by set $\mathcal{C}$.

The EM algorithm is used to iteratively associate subset of the $x_{s,m}$ for training acoustic model λ for a sound class. Acoustic model λ is initialized by training a model using all annotated time-segments S from all four separated streams, $x_{\mathcal{C},1:4}$. Notation $x_{\mathcal{C},1:4}$ denotes all the x indexed by event set $s \in \mathcal{C}$ and $m \in [1, 4]$. After this the EM algorithm operates iteratively repeating the E step and M step while the value of the likelihood function $P(\lambda \mid x_{\mathcal{C},1:4})$ is maximized at each iteration. Using Bayesian expansion, expression to be maximized is $P(x_{\mathcal{C},1:4} \mid \lambda)$, which is defined as

$$P(x_{\mathcal{C},1:4} \mid \lambda) \equiv \sum_{s \in \mathcal{C}} \sum_{m} P(x_{s,m}, a_s = m \mid \lambda), \tag{1}$$

where latent variable a_s denotes the index of the stream that contains the target event. This can be further expanded into

$$P(x_{\mathcal{C},1:4} \mid \lambda) = \sum_{s \in \mathcal{C}} \sum_{m} P(x_{s,m} \mid \lambda) P(a_s = m \mid x_{s,m}, \lambda). \tag{2}$$

Above, $P(x_{s,m} \mid \lambda)$ is the likelihood of $x_{s,m}$ for the HMM event model. Let us denote the posterior probability $P(a_s = m \mid x_{s,m}, \lambda)$ by $a_{s,m}$. The EM algorithm iterates over expectation – calculating $a_{s,m}$ and maximization – recalculating model λ:

$$\text{(E):} \qquad a_{s,m} = P(a_s = m \mid x_{s,m}, \lambda) \tag{3}$$

$$\text{(M):} \qquad \lambda \leftarrow \arg\max_{\lambda} \sum_{s \in \mathcal{C}} \sum_{m} P(x_{s,m} \mid \lambda) a_{s,m}. \tag{4}$$

The expectation step represents the stream selection, and is given as

$$a_{s,m} = \frac{P(x_{s,m} \mid \lambda)}{\sum_{m'} P(x_{s,m'} \mid \lambda)}. \tag{5}$$

The maximization step in Eq. 4 represents the training of the new models and is solved by conventional Baum-Welch algorithm used to train HMMs.

In order to simplify the maximization step, a is made binary as described in the next section. As a result of this, only those $x_{s,m}$ for which $a_{s,m} = 1$ are used in the maximization. This avoids using weighted observations so that standard HMM training algorithms can be used.

3.2. Stream selection

We propose two approaches to make a binary. In the first one, only the most likely stream is selected, i.e., $a_{s,m}$ having the highest likelihood among $a_{s,1:4}$ is set to one and $a_{s,m'}$ for other m is set to zero. This approach is later denoted as *prominent stream selection*.

In the second one, the n smallest $a_{s,m}$ among $a_{s,1:4}$ are set to zero, i.e. eliminated. We set n equal to the iteration count. This approach is later denoted as *stream elimination*. The illustration of how the stream selection approaches are applied to one training instance is shown in Figure 4.

Prominent stream selection is repeated until convergence, i.e. the stream indexes do not change. The stream elimination is repeated until only one stream is left.

4. SYSTEM EVALUATION

The sound event detection system is trained and tested using an audio database collected from real-life contexts. The training and testing are done in a context-dependent manner, using context-dependent count-based priors and acoustical models.

4.1. Database

The database consists of 103 recordings ranging from 10 to 30 minutes resulting in total 1133 minutes of audio. The recordings were collected from ten audio contexts: basketball game, beach, inside a bus, inside a car, hallways, inside an office facility, restaurant, grocery shop, street, and stadium with track and field events. There were 8-14 recordings made in each context using binaural microphones placed inside the ears of the person recording. In this study we are using monophonic versions of the recordings, i.e., the two channels are averaged to one channel.

All clearly audible sound events in the recordings were manually annotated by indicating the start and end times of the sound events. Total of 61 distinct event classes are used in the study. The event classes include e.g. speech, laughter, applause, car door, road, dishes, door, chair, music, and footsteps. The number of events that can be active at the same time was not limited. In this sense, the

	A1	pre / rec	A30	pre / rec
Baseline	**36.7±2.4**	33.1 / 41.2	**57.2±2.2**	53.8 / 61.2
Prominent stream selection				
Iteration 1	42.8±5.2	38.9 / 47.6	60.6±3.6	58.1 / 63.4
Iteration 2	43.8±4.4	39.4 / 49.3	60.6±2.3	57.7 / 63.9
Iteration 3	**44.5±5.9**	40.0 / 50.2	**60.9±2.9**	58.1 / 64.1
Iteration 4	44.1±5.8	39.7 / 49.8	60.5±2.3	57.8 / 63.6
Stream elimination				
Iteration 1, n=1	37.9±2.3	34.3 / 42.4	58.4±0.7	55.2 / 62.0
Iteration 2, n=2	40.4±4.0	36.3 / 45.6	60.2±1.7	57.0 / 63.9
Iteration 3, n=3	**44.9±4.7**	40.2 / 51.1	**60.8±2.8**	58.0 / 64.0

Table 1. Sound event detection accuracy, calculated based on precision (pre) and recall (rec), for the baseline and systems using proposed stream selection approaches.

recordings can be regarded as polyphonic. Usually in a natural auditory scene the event classes are not equally represented. While many event classes are very common and shared between multiple contexts (e.g. speech), some event classes can be quite rare or they are highly context-specific (e.g. referee whistle in basketball game or pressure release noise inside the bus). A more detailed description of the database and event class statistics can be found in [11].

4.2. Performance evaluation

For evaluating the system output, we will use the block-wise detection accuracy metric proposed in [12]. This metric evaluates how well the events detected in non-overlapping time blocks coincide with the annotations. The detected events are regarded only at the block level, and in this study we are using two block lengths: one second (denoted by A1) and 30 seconds (denoted by A30).

Inside a block, precision and recall are calculated, and block-wise detection accuracy is represented by the F-score. An event is regarded as correctly detected if it has been detected and annotated somewhere within the considered block.

4.3. Results

The detection accuracy of the models produced by the proposed stream selection approaches was evaluated and compared against a baseline system which is using event models trained without stream selection. The event models used in the baseline are also used as initial models for the stream selection process.

The evaluation database was split randomly into five equal-sized sets, with one set being used as test data and other four for training the system. The split was done five times for a five-fold cross-validation setup. One fold was used in the development stage for determining parameters for the event sequence decoding. The evaluation results are presented as the average of the other four folds.

The event detection results for the baseline system and the proposed stream selection approaches are presented in Table 1 (best performance highlighted). The results show average detection accuracy along with 95 % confidence interval. The number of iteration steps for the prominent stream selection approach was four, since only minimal changes (0.1 % change) were noticed after the fourth iteration. In the stream elimination approach, the elimination parameter n was increased with one in each iteration. After three iterations only the most likely stream was left and the iteration was ended.

Detection accuracy increases steadily with both of the selection approaches throughout the iterations. In the end, both approaches provide similar level of increase in the block-wise accuracy compared to the baseline system. In the one second block-level, the improvement in detection accuracy is over 8 percentage units for both selection approaches after three iterations. In the 30 second block-level, the improvement is more modest being only 3 percentage units. The increased accuracy of the detection is especially noticeable in the recall of the detection for both block-levels, i.e. bigger portion of annotated events are correctly detected.

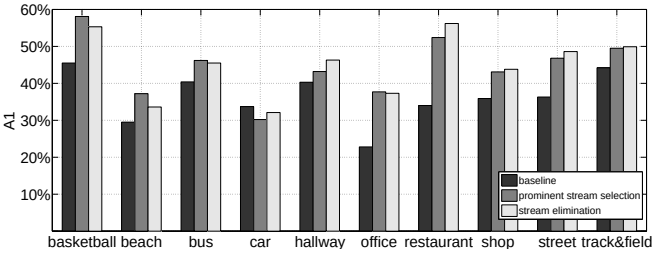

Fig. 5. Context-wise detection accuracy (A1) after three stream selection iterations, along with the baseline accuracy.

The context-wise results are shown in Figure 5. For restaurant and office, the proposed stream selection approach will give significant improvement, whereas for recording made inside a car, the detection accuracy even drops a bit. This may be due to the fact that car environment is very noisy and the degree of overlapping between events is low.

5. DISCUSSIONS

The main difficulty when using the prominent stream selection approach is to know how many iterations are needed. In this study we stopped the number of iterations at four, but in fact the maximum detection accuracy was obtained after three iterations. Results in Table 1 show that accuracy does not change significantly after the first iteration. This means that the first iteration already selects most of the correct streams for each target class.

The stream elimination approach is more straightforward, as one needs to perform iterations until only one stream is left. In this approach, the detection accuracy increases gradually, reaching maximum at the end of the process.

Compared to previous work using sound source separation [12], the presented work increases significantly (52.6 % to 60.9 % in A30) the performance through using event priors and the proposed stream selection method in training the models. Compared to detection on polyphonic audio, that does not use any source separation, the performance is more than doubled [15].

6. CONCLUSIONS

A method for training acoustic event models from acoustic material containing high degree of overlapping events was proposed. In the preprocessing stage, the unsupervised sound source separation was applied to the audio signal in order to minimize the interference of overlapping events. The most appropriate training material for the target sound class was selected iteratively from the separated audio streams using an EM algorithm. The approaches for selecting streams work by selecting the most likely or eliminating the most unlikely streams. Both approaches were found to give reasonable increase in the detection accuracy compared to the baseline system. This highlights the benefits of carefully selecting training material.

7. REFERENCES

[1] R. Cai, L. Lu, A. Hanjalic, H. Zhang, and L-H. Cai, "A flexible framework for key audio effects detection and auditory context inference," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 3, pp. 1026–1039, 2006.

[2] 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*, vol. 4, no. 2, pp. 1–23, 2008.

[3] 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 *Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on*, 2009, pp. 1218 –1221.

[4] A. Härmä, M. F. McKinney, and J. Skowronek, "Automatic surveillance of the acoustic activity in our living environment," *IEEE International Conference on Multimedia and Expo*, pp. 634–637, 2005.

[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 *18th European Signal Processing Conference*, Aalborg, Denmark, 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 *Emerging Signal Processing Applications (ESPA)*, 2012, pp. 99 –102.

[8] G. Wichern, Jiachen 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] X. Zhou, X. Zhuang, M. Liu, H. Tang, M. Hasegawa-Johnson, and T. Huang, "HMM-based acoustic event detection with AdaBoost feature selection," in *Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*, Berlin, Heidelberg, 2008, pp. 345–353, Springer-Verlag.

[10] X. Zhuang, X. Zhou, M. A. Hasegawa-Johnson, and T. S. Huang, "Real-world acoustic event detection," *Pattern Recognition Letters*, vol. 31, no. 12, pp. 1543 – 1551, 2010, Pattern Recognition of Non-Speech Audio.

[11] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, "Acoustic event detection in real-life recordings," in *18th European Signal Processing Conference*, Aalborg, Denmark, 2010, pp. 1267–1271.

[12] T. Heittola, A. Mesaros, T. Virtanen, and A. Eronen, "Sound event detection in multisource environments using source separation," in *Workshop on Machine Listening in Multisource Environments, CHiME2011*, Florence, Italy, 2011.

[13] T. Butko, González P., Segura C., Nadeu C., and Hernando J., "Two-source acoustic event detection and localization: Online implementation in a smart-room," in *19th European Signal Processing Conference*, Barcelona, Spain, 2011, pp. 1317–1321.

[14] A. Temko and C. Nadeu, "Acoustic event detection in a meeting-room environment," *Pattern Recognition Letters*, vol. 30, no. 14, pp. 1281–1288, 2009.

[15] T. Heittola, A. Mesaros, T. Virtanen, and A. Eronen, "Context-dependent sound event detection," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2013, no. 1, 2013.

[16] Taras Butko, Cristian Canton-Ferrer, Carlos Segura, Xavier Giró, Climent Nadeu, Javier Hernando, and Josep R. Casas, "Acoustic event detection based on feature-level fusion of audio and video modalities," *EURASIP Journal on Advanced Signal Processing*, vol. 2011, no. 1, 2011.

[17] L. Deng, A. Acero, M. Plumpe, and X. Huang, "Large-vocabulary speech recognition under adverse acoustic environments," in *International Conference on Spoken Language Processing (ICSLP)*, 2000, pp. 806–809.

[18] T. Virtanen, "Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[19] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.