Robust Acoustic Scene Classification in the Presence of Active Foreground Speech

Siyuan Song, Brecht Desplanques, Celest De Moor, Kris Demuynck, Nilesh Madhu
IDLab, Department of Electronics and Information Systems, Ghent University - imec, Belgium
Email: {Siyuan.Song, Brecht.Desplanques, Celest.Demoor, Kris.Demuynck, Nilesh.Madhu}@ugent.be

***Abstract*—We present an iVector based Acoustic Scene Classification (ASC) system suited for real life settings where active foreground speech can be present. In the proposed system, each recording is represented by a fixed-length iVector that models the recording's important properties. A regularized Gaussian backend classifier with class-specific covariance models is used to extract the relevant acoustic scene information from these iVectors. To alleviate the large performance degradation when a foreground speaker dominates the captured signal, we investigate the use of the iVector framework on Mel-Frequency Cepstral Coefficients (MFCCs) that are derived from an estimate of the noise power spectral density. This noise-floor can be extracted in a statistical manner for single channel recordings. We show that the use of noise-floor features is complementary to multi-condition training in which foreground speech is added to training signal to reduce the mismatch between training and testing conditions. Experimental results on the DCASE 2016 Task 1 dataset show that the noise-floor based features and multi-condition training realize significant classification accuracy gains of up to more than 25 percentage points (absolute) in the most adverse conditions. These promising results can further facilitate the integration of ASC in resource-constrained devices such as hearables.**

***Index Terms*—Acoustic scene classification, factor analysis, iVector, Gaussian backend, noise-floor estimation**

I. Introduction

Enhancement of audio and speech is typically a first stage in many audio applications, with a view to improving the user experience. For optimal results, the parameters of the enhancement approaches should be adaptive to the environment. To enable such advanced techniques, the nature of the background environment should be determined first. The classification of the background audio into acoustic *scenes* is termed as Acoustic Scene Classification (ASC) [1]. ASC is an emerging technology with potential applications in many fields. For example, smartphones could sense their environment in order to adapt their settings, e.g. using silent mode when situated in a concert hall [2]. Hearing aids could adapt their filtering characteristics in changing scenarios, e.g. using beamforming when the user is in a conversation with one or multiple persons or giving an experience of surround sound during a music concert or when the user is on a street.

DCASE is a yearly international challenge where the detection or classification of acoustic scenes and events is an important re-occurring task [3], [4]. Within the last decade, plenty of methods have emerged in this field [5]–[8] and the classification accuracy on evaluation data that purely consists of recordings of acoustic scenes is continually improving. However, in many real life applications, there is no guarantee that the characterizing sound of an acoustic scene will be the dominant signal. Foreground speech might be captured as well, which is not always relevant towards the detection of the acoustic scene. For example, in telecommunications, or for audio captured by hearables, it is very likely that the captured audio is dominated by the speech of the user. In this case, existing ASC methods degrade significantly. Practical integration of ASC into the audio processing chain of such devices, therefore, requires robustness to foreground speech.

Whereas research along this direction has not been actively pursued in the previous years, there exists a plethora of scientific work on the *denoising* of captured audio for the enhancement of foreground speech. For single-microphone captures (the focus in this paper), such denoising approaches typically take the form of a time-frequency gain function, which suppresses regions dominated by the background noise while preserving the speech-dominant regions. An overview of *classical* speech enhancement algorithms, based on statistical signal models, may be found in [9]. These have been surpassed in the recent past by model-based approaches and approaches based on deep-learning (e.g., [10]–[15] to cite but a few). Consequently, it seems that a logical approach to making ASC robust to foreground speech would be to apply a gain function which preserves the background and attenuates the *foreground speech*. Such a gain function maybe derived from the speech-enhancement gain. Other alternatives that suggest themselves are: the use of a Voice-Activity Detector (VAD) to extract features only in speech-absent frames or, as recently proposed in [16], an explicit first stage to *remove foreground speech*.

However each alternative brings attendant disadvantages: constructing a gain function to remove speech while preserving the background leads to (non-linear) artefacts (e.g., due to estimation errors) in the resulting signal spectrum, which harms rather than improves the ASC backend. Using a VAD flag to estimate features only on speech-absent frames would lead to a long processing delay if foreground speech remains continually active. And, methods to explicitly remove foreground speech may be quite complex and may, again, introduce artefacts in the resulting signal. We propose, therefore, to investigate the feasibility of the noise-floor *estimate*, as an input stream for the ASC system. Such an estimate can be obtained by statistical methods with low complexity and in a real-time manner - thereby lending itself to practical applications.

Keeping complexity and low resource training in mind, we consider an ASC-backend based on a statistical framework. The winning contribution of the DCASE 2016 [17] challenge was the *binaural* iVector-based approach [18], described in [8]. The system has a classification accuracy of 88.7%, (an improvement of 11.5% over the GMM-based baseline approach). In contrast, a deep convolutional neural network reported in [8] achieves 83.3% accuracy, indicating that the iVector approach is a robust choice in this relatively low-resource setting. Thus, our baseline shall be a competitive (in-house) iVector solution, on which we first investigate the effect of foreground speech.

Next, we extract the noise-floor from the input signal and derive the acoustic features for ASC from this. We consider statistical methods for estimating the noise-floor, and we refer the reader to [19]–[21] for several well-known approaches. We finally contrast the robustness of the noise-floor based ASC system against our baseline. We also investigate the benefit of multi-condition training, in which foreground speech is added to the signals during the training process.

The remainder of this paper is organized as follows. In Section II we describe our iVector baseline system where a modified Gaussian backend classifier with class-specific covariance models is adopted. In Section III, we (briefly) discuss the noise-floor estimation (based on [21]) and its integration in the acoustic feature extraction of the iVector system. Section IV describes the multi-condition training. In Section V, the dataset and experimental setup is described. In Section VI, we evaluate the performance in different settings with varying levels of foreground speech. Section VII concludes the paper.

II. ASC iVECTOR FRAMEWORK

We rely on the iVector framework to enable the processing of audio recordings by relatively simple backend classifiers. Through factor analysis the information in the variable length recording described by a sequence of MFCC feature vectors is compressed to a fixed length representation called the iVector. A Gaussian backend classifier, which fits a simple Gaussian model to the iVectors of each class, is then used to extract the most likely acoustic scene.

A. Feature Extraction

The MFCC features are constructed from the signal amplitude spectrum extracted with a frame length of 40 ms and 50% overlap between successive frames. We extract 40 mel filter bank energies. 21 cepstral coefficients are kept, including the zeroth cepstral coefficient c_0. Normalization of this c_0 spectrum mean energy component did not have a significant performance impact during experiments on the DCASE 2016 Task 1 dataset.

Instead of adding first and second order derivatives to the MFCC features, we use Shifted Delta Cepstral (SDC) coefficients to obtain dynamic features with a larger temporal context [22]. At frame index n, a total of $2K+1$ Δ-feature vectors (first order time differences of the first N static features) for the current frame and K frames before and after the current frame are concatenated to the static MFCC coefficients. The considered $2K+1$ frames are separated by shifts of P frames. Thus, a vector of length $(2K+1) \times N$ is added to each feature vector.

B. Factor Analysis

The variability in the observed MFCC features is caused by multiple underlying factors. To capture the most significant hidden properties of a recording such as the acoustic scene, we extract a fixed-length iVector [18] for each recording. The iVector framework uses an Universal Background Model (UBM) as a reference model and expresses for each recording how it deviates from this UBM. The UBM is a Gaussian Mixture Model (GMM) that is trained on a wide variety of data and deviations from this model are analyzed in the supervector domain. This domain is obtained by concatenating the GMM mean vectors of each component into a large supervector. The supervector $\boldsymbol{s}$ of a recording is approximated by adding a supervector that lays in the total variability subspace defined by a low-rank matrix $\boldsymbol{T}$ to the UBM supervector $\boldsymbol{m}$:

$$\boldsymbol{s} = \boldsymbol{m} + \boldsymbol{T}\boldsymbol{w}. \tag{1}$$

$\boldsymbol{T}$ is called the iVector extractor, and the corresponding supervector subspace should cover most of the important variability observed in the data. The maximum a posteriori estimate $\boldsymbol{w}$ of the coordinates when enforcing a standard normal prior is called the iVector. $\boldsymbol{w}$ contains all relevant information of a recording, described by a fixed-length vector. Technical details of the iVector extraction process can be found in [23]. The unsupervised training process of the total-variability extractor $\boldsymbol{T}$ is initiated by Principal Component Analysis (PCA), followed by an Expectation-Maximization (EM) algorithm [23].

C. Gaussian Backend Classifier

The estimated iVectors capture all kinds of variability, including within-class variability in the context of ASC. A simple Gaussian Backend (GB) suppresses this variability and focuses on inter-scene variability to perform the classification.

The standard GB approach fits a multivariate normal distribution to each class. For regularization purposes the common full covariance matrix $\boldsymbol{\Sigma}_s$ is shared between all classes and it is estimated by taking the unweighted average across all within-class covariance matrices $\boldsymbol{\Sigma}_c$. The class model that produces the maximum log-likelihood corresponds with the predicted acoustic scene. The simplified decision function $d(\boldsymbol{w})$ is

$$d(\boldsymbol{w}) = \arg\max_c \left\{ -\frac{1}{2}(\boldsymbol{w}-\boldsymbol{\mu}_c)^T \boldsymbol{\Sigma}_s^{-1} (\boldsymbol{w}-\boldsymbol{\mu}_c) \right\} \tag{2}$$

with $\boldsymbol{\mu}_c$ the estimated mean iVector of acoustic scene class c.

The hypothesis that the covariance matrix can be shared across all classes is reasonable in fields where the intra-class variability should be very similar, e.g. speaker variability in language or emotion recognition in speech. However, in the case of ASC the intra-class variability might be more class-specific and we propose to use class-dependent covariance

matrices instead. The risk of overfitting is increased due to the larger number of model parameters and we decide to keep the regularization effect by sharing knowledge between classes during the covariance estimation. The estimated covariance matrix for class c is defined by: $\widetilde{\boldsymbol{\Sigma}_c} = \alpha\boldsymbol{\Sigma}_s + (1-\alpha)\boldsymbol{\Sigma}_c$, in which $\alpha \in [0,1]$ controls the regularization effect. In this case, the decision function $d(\boldsymbol{w})$ becomes

$$\arg\max_c \left\{ -\frac{1}{2}\log|\widetilde{\boldsymbol{\Sigma}_c}| - \frac{1}{2}(\boldsymbol{w}-\boldsymbol{\mu}_c)^T \widetilde{\boldsymbol{\Sigma}_c}^{-1}(\boldsymbol{w}-\boldsymbol{\mu}_c) \right\} \quad (3)$$

III. Noise-floor based features for ASC

The signal model for single-channel speech enhancement is typically formulated in the Short-Time Fourier Transform (STFT) domain as:

$$X(\ell,k) = S(\ell,k) + N(\ell,k)\,, \quad (4)$$

where $S(\ell,k)$ is the STFT representation of the foreground speech, $N(\ell,k)$ is the background and $X(\ell,k)$ is the captured (microphone) signal. In this representation ℓ is the discrete-frequency index and k is the frame index. In the following discussion, the time- and frequency-indices will be omitted for brevity.

Statistical methods for noise-floor estimation are based on the following assumptions: (a) the foreground speech and background noise may be modelled as independent random variables and (b) the second-order statistics of the background noise is stationary over a longer period than the foreground speech. Essentially, the noise-floor estimate is obtained by recursively smoothing an estimate of the noise periodogram $\mathbb{E}\big(|N|^2\big|X\big)$, thereby removing the short variations attributable to speech and preserving the long-term characteristics of the background. In order to efficiently track a varying noise-floor, [21] proposed the use of an *adaptive* approach to the estimation of $\mathbb{E}\big(|N|^2\big|X\big)$, driven by the conditional probability of speech presence given the observed signal. Denoting the conditional probability of the hypothesis that speech is present by $P\big(\mathcal{H}_1\big|X\big)$ and the probability of the alternative hypothesis by $P\big(\mathcal{H}_0\big|X\big) = 1 - P\big(\mathcal{H}_1\big|X\big)$, we may obtain the following estimate of the noise periodogram:

$$\mathbb{E}\big(|N|^2\big|X\big) = (1 - P\big(\mathcal{H}_1\big|X\big))|X|^2 + P\big(\mathcal{H}_1\big|X\big)\widehat{\sigma_N^2}\,, \quad (5)$$

where $\widehat{\sigma_N^2}$ represents the noise-floor estimate at the previous time-frame ($\widehat{\sigma_N^2} = \widehat{\sigma_N^2}(k-1)$). The estimated noise periodogram from (5) is subsequently recursively smoothed with a fixed smoothing factor to obtain the updated noise-floor estimate $\widehat{\sigma_N^2}$ for the current frame.

In the case of noise-floor based MFCC feature extraction, the acoustic features are now derived from the noise-floor estimate $\widehat{\sigma_N^2}$. The rest of the feature extraction process remains the same compared to section II-A. These MFCCs are used during training and evaluation of the proposed system.

IV. Multi-condition training

For both the baseline and the proposed system using the noise-floor features, we also explore multi-condition training to further reduce the mismatch between training and testing conditions. This should result in better robustness of the classification system. A common way to measure the relation between the speech power and the noise power in a recording is via the Signal-to-Noise Ratio (SNR). In this typical context, we refer to 'noise' as the unwanted signal. In the case of ASC, the definition of 'noise' is ambiguous, as the noise in the recording is useful for classification. Instead, we use the Speech-to-Background Ratio (SBR). To calculate the SBR, we need to estimate the speech and background power level. For speech, we use the active speech level of the speech fragment, for the background we use the Root Mean Square (RMS) value. In the proposed multi-condition training, mixed training data of different SBRs is used as training corpus compared to default of only using the original DCASE background noise.

V. Experimental setup

A. Dataset

The proposed ASC system and foreground speech compensation techniques are evaluated on the DCASE 2016 Task 1 dataset [17]. This dataset contains 15 different acoustic scenes. The development dataset consists of almost 10 hours of data. For each acoustic scene, 39 minutes of data is provided, which is divided into 78 segments with a duration of 30 seconds. The development data is split into four folds to enable cross-validation for hyperparameter tuning. The evaluation dataset contains approximately 3 hours of data. Each acoustic scene has 13 minutes of audio, divided into 26 segments of 30 seconds. Recordings were made using a binaural microphone and the sampling rate is 44.1 kHz. Our motivation in selecting this dataset is to validate our approaches for a situation where only limited *labelled* training data is available, which is commonly the case in bespoke industry scenarios.

To simulate the presence of foreground speech we mix the recordings with speech from the pitch-tracking database from Graz University of Technology [24] and the Multilingual Speech Database from NTT-AT[1]. We downsampled the complete DCASE dataset to 16 kHz mono to allow the use of these speech datasets, and to reduce the computational load during simulations. We make sure that there is no overlap between the speaker set used for mixing the training data and the speaker set used on the evaluation data.

B. Setup

All systems operate on MFCC features extracted by the procedure described in Section II-A. The temporal context is increased in the form of SDCs with parameters $M = 2$, $K = 2$, $N = 11$ and $P = 3$. The number of components in the UBM is 256 and the rank of the iVector extractor T is set to 150. A weight $\alpha = 0.7$ is used to regularize the Gaussian backend with class-dependent covariance matrices. When the noise-floor estimation is applied, it is adopted for both the development and the evaluation dataset. All hyper-parameters are determined by four fold cross-validation on the development set.

[1] https://www.ntt-at.com/product/speech2002/

TABLE I
IMPACT OF THE NOISE-FLOOR FEATURES AND MULTI-CONDITION TRAINING ON DCASE 2016 ASC ACCURACY FOR DIFFERENT SBRS.

Accuracy(%) / Development dataset	Evaluation dataset	No speech	−5 dB	0 dB	5 dB	10 dB	15 dB	20 dB
{No speech}	Without noise-floor	**81.3**	62.3	59.2	55.1	50.5	44.9	40.0
	With noise-floor	76.2	74.4	70.8	67.4	63.3	57.4	52.8
{No speech, −5 dB}	Without noise-floor	79.2	**77.2**	**77.7**	**76.7**	70.3	61.5	47.7
	With noise-floor	76.2	75.1	74.6	71.0	**71.0**	**70.3**	**66.2**

TABLE II
IMPACT OF CLASS-DEPENDENT GAUSSIAN BACKEND COVARIANCE MODELS ON DCASE 2016 ASC ACCURACY FOR DIFFERENT SBRS.

Accuracy(%) / Development dataset	Evaluation dataset	No speech	−5 dB	0 dB	5 dB	10 dB	15 dB	20 dB
Standard Gaussian Backend (GB)		74.9	70.3	68.7	66.4	62.1	**57.7**	48.2
Class-dependent GB covariance models		**76.2**	**74.4**	**70.8**	**67.4**	**63.3**	57.4	**52.8**

The performance of all systems is expressed by the classification accuracy on the evaluation set, which is sufficient as a global performance measure as the DCASE 2016 dataset is balanced.

VI. RESULTS AND DISCUSSION

In this section, we evaluate the impact of the noise-floor based MFCC features, the multi-condition training and the introduction of class-dependent covariance matrices in the Gaussian backend.

A. Baseline System in Clean Conditions

For a fair comparison we need to start from a competitive baseline system. The best reported iVector based result in the DCASE 2016 Task 1 challenge is 88.7% [8] on the 44.1 kHz binaural evaluation data. Our proposed baseline system with class-dependent covariance matrices in the GB achieves 88.2% on the same data. Similar to [8], the binaural data is processed by considering four possible input combinations of the audio channels (L, R, L+R and L-R) and applying score fusion by averaging the four different log-likelihood outputs. The achieved performance is close enough to the results reported in [8] to do a fair study of the robustness of the system against active foreground speech. Adding foreground speech to the evaluation data is the most straightforward on mono 16 kHz audio. If we consider a single audio channel, the performance of our baseline system drops to 85.4%. Reducing the sample rate to 16 kHz further reduces the accuracy to 81.3%.

B. Speech Robust ASC

The performance of the proposed techniques to increase the robustness against foreground speech is shown in Table I. The first two rows represent the results when the original DCASE development data is used for training and no speech is added. Different speech levels are added on top of the DCASE evaluation dataset with different SBRs of {-5, 0, 5, 10, 15, 20} dB. For each testing condition the SBR level is kept constant. In the first row, the reported accuracy of the iVector system on the standard MFCC features without noise-floor clearly deteriorates for increasing SBR levels when the foreground speech becomes increasingly prominent. The original accuracy of 81.3% halves to 40% in the 20 dB SBR condition. As shown in the second row, this degradation is *reduced significantly* when using the noise-floor based MFCC features. The performance is more consistent and, in the most adverse 20 dB SBR condition, the system now obtains 52.8% accuracy. However, the performance in clean conditions slightly degraded to 76.2%. This small degradation is to be expected as the noise-floor estimation will unavoidably remove some information that is useful for ASC. In certain real life scenarios, where the recordings can be strongly dominated by foreground speech rather than noise, the use of noise-floor based MFCC features should be preferred. The final two rows in Table I show the impact of Multi-Condition Training (MCT) for both standard and noise-floor based MFCC feature extraction. The training corpus now consists of the original DCASE development data and the development data augmented with speech with an SBR of −5 dB. Similarly to the use of noise-floor based features, the use of multi-condition training reduces the negative impact of foreground speech. Using the default MFCC features with MCT outperforms the standard training protocol with noise-floor based MFCCs, but its advantage tapers off for increasing SBR levels. For 20 dB SBR it is still outperformed by the noise-floor technique. We can increase the strength of the augmentation in MCT as shown in Fig. 1, which presents the results for {no speech, −5 dB}, {no speech, −5 dB, 5 dB}, and {no speech, 5 dB, −5 dB, 10 dB} MCT training corpuses. But, it may be seen that the increased performance for high SBR levels again comes at the cost of decreased performance on the original clean DCASE data. The final row of Table I and Fig. 1 show that improvements of noise-floor based MFCCs with MCT are complementary for higher SBR levels. The combination of both techniques (noise-floor based features and MCT) levels (flattens) the performance across a very wide range of SBR levels and is recommended if the encountered SBR levels in the real life applications are unpredictable.

C. Gaussian Backend with Class-dependent Covariances

The impact of introducing class-dependent covariance models in the GB is now evaluated on a wide range of SBR levels, including the clean evaluation data. The classification accuracies of the iVector system trained on noise-floor based MFCCs extracted from the original DCASE development data are shown in Table II. The GB classifier with class-dependent covariance models consistently outperforms the standard GB on the whole range of SBR levels (except for SBR=15 dB where both systems have similar performance). This indicates one has to be careful with the hypothesis that the intra-class variabilities are similar, especially in the context of ASC.

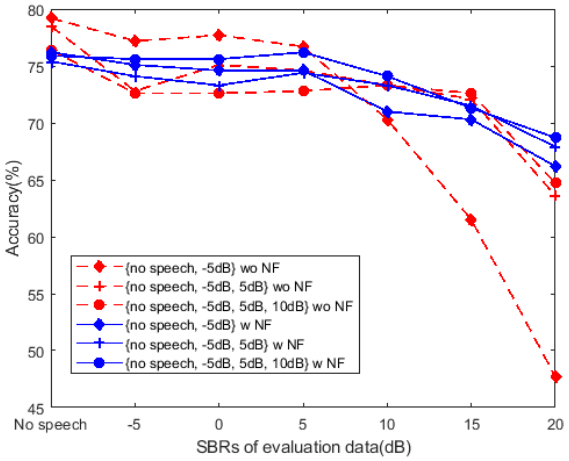

Fig. 1. Impact of multi-condition training on the proposed systems. NF stands for the use of noise-floor based MFCC features.

VII. Conclusions

This paper presents two complementary techniques to increase the robustness of an ASC system against foreground speech: the incorporation of noise-floor based features and Multi-Condition Training (MCT). The noise-floor based features can successfully extract the acoustic scene in the presence of foreground speech, but MCT still helps in eliminating any mismatch between training and testing conditions. The combination of both techniques achieves very consistent DCASE 2016 classification performance that is *almost independent* of the energy level of the encountered nuisance speech. An absolute improvement of up to 25% is seen in the most adverse conditions. Future work will verify if the noise-floor can be used for feature extraction and/or for data augmentation in neural network based ASC solutions.

References

[1] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Transactions on Multimedia*, vol. 17, no. 10, pp. 1733–1746, 2015.

[2] D. Barchiesi, D. Giannoulis, D. Stowell, and M. D. Plumbley, "Acoustic scene classification: Classifying environments from the sounds they produce," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 16–34, 2015.

[3] A. Mesaros, T. Heittola, E. Benetos, P. Foster, M. Lagrange, T. Virtanen, and M. D. Plumbley, "Detection and classification of acoustic scenes and events: Outcome of the DCASE 2016 challenge," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 2, pp. 379–393, 2017.

[4] A. Mesaros, T. Heittola, and T. Virtanen, "Acoustic scene classification: an overview of DCASE 2017 challenge entries," in *16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE, 2018, pp. 411–415.

[5] S. Suh, S. Park, Y. Jeong, and T. Lee, "Designing acoustic scene classification models with cnn variants," DCASE 2020 Challenge, Tech. Rep, Tech. Rep., 2020.

[6] H. Hu, C.-H. H. Yang, X. Xia, X. Bai, X. Tang, Y. Wang, S. Niu, L. Chai, J. Li, H. Zhu *et al.*, "Device-robust acoustic scene classification based on two-stage categorization and data augmentation," *arXiv preprint arXiv:2007.08389*, 2020.

[7] Y. Sakashita and M. Aono, "Acoustic scene classification by ensemble of spectrograms based on adaptive temporal divisions," *Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge*, 2018.

[8] H. Eghbal-Zadeh, B. Lehner, M. Dorfer, and G. Widmer, "CP-JKU submissions for DCASE-2016: A hybrid approach using binaural i-vectors and deep convolutional neural networks," *IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE)*, vol. 6, pp. 5024–5028, 2016.

[9] R. C. Hendriks, T. Gerkmann, and J. Jensen, *DFT-Domain based single-microphone noise reduction for speech enhancement*, ser. Synthesis Lect. Speech and Audio Proc. Morgan & Claypool, 2013.

[10] P. Mowlaee, R. Saeidi, and R. Martin, "Model-driven speech enhancement for multisource reverberant environment," in *Proceedings of the International Conference on Latent Variable Analysis and Signal Separation*, 2011, pp. 454–461.

[11] C. Joder, F. Weninger, and D. Virette, "Integrating noise estimation and factorization-based speech separation: A novel hybrid approach," in *Proc. IEEE International Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 131–135.

[12] S. Mirsamadi and I. Tashev, "Causal speech enhancement combining data-driven learning and suppression rule estimation," in *Proceedings of the International Conference on Speech Communication and Technology (INTERSPEECH)*, 2016, pp. 2870–2874.

[13] Y. Yang and C. Bao, "DNN-based AR-Wiener filtering for speech enhancement," in *Proc. IEEE International Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2901–2905.

[14] S. Elshamy and T. Fingscheidt, "DNN-based cepstral excitation manipulation for speech enhancement," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 11, pp. 1803–1814, 2019.

[15] M. Strake, B. Defraene, K. Fluyt, W. Tirry, and T. Fingscheidt, "Speech enhancement by LSTM-based noise suppression followed by CNN-based speech restoration," *EURASIP Journal on Advances in Signal Processing*, vol. 2020, no. 1, pp. 1–26, 2020.

[16] S. Liu, A. Triantafyllopoulos, Z. Ren, and B. W. Schuller, "Towards speech robustness for acoustic scene classification," *Proc. Interspeech 2020*, pp. 3087–3091, 2020.

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

[18] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010.

[19] R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statistics," *IEEE Transactions on speech and audio processing*, vol. 9, no. 5, pp. 504–512, 2001.

[20] S. Rangachari and P. C. Loizou, "A noise-estimation algorithm for highly non-stationary environments," *Speech Communication*, vol. 48, no. 2, pp. 220–231, 2006.

[21] T. Gerkmann and R. C. Hendriks, "Unbiased MMSE-based noise power estimation with low complexity and low tracking delay," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 4, pp. 1383–1393, 2012.

[22] P. A. Torres-Carrasquillo, E. Singer, M. A. Kohler, R. J. Greene, D. A. Reynolds, and J. R. Deller Jr, "Approaches to language identification using Gaussian mixture models and shifted delta cepstral features," in *7th International Conference on Spoken Language Processing*, 2002.

[23] O. Glembek, L. Burget, P. Matějka, M. Karafiát, and P. Kenny, "Simplification and optimization of i-vector extraction," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2011, pp. 4516–4519.

[24] G. Pirker, M. Wohlmayr, S. Petrik, and F. Pernkopf, "A pitch tracking corpus with evaluation on multipitch tracking scenario," in *Proc. INTERSPEECH*, 2011, pp. 1509–1512.