ROBUST ASR IN REVERBERANT ENVIRONMENTS USING TEMPORAL CEPSTRUM SMOOTHING FOR SPEECH ENHANCEMENT AND AN AMPLITUDE MODULATION FILTERBANK FOR FEATURE EXTRACTION

Feifei Xiong[†], *Niko Moritz*[†], *Robert Rehr*[‡], *Jörn Anemüller*[‡],
Bernd T. Meyer[‡], *Timo Gerkmann*[‡], *Simon Doclo*[†‡], *Stefan Goetze*[†]

[†]Fraunhofer Institute for Digital Media Technology IDMT,
Project Group Hearing-, Speech- and Audio-Technology (HSA), Oldenburg, Germany
{feifei.xiong,niko.moritz,s.goetze}@idmt.fraunhofer.de
[‡]University of Oldenburg, Department of Medical Physics and Acoustics
and Cluster of Excellence Hearing4All, Oldenburg, Germany
{robert.rehr,joern.anemueller,bernd.meyer,timo.gerkmann,simon.doclo}@uni-oldenburg.de

ABSTRACT

This paper presents techniques aiming at improving automatic speech recognition (ASR) in single channel scenarios in the context of the REVERB (REverberant Voice Enhancement and Recognition Benchmark) challenge. System improvements range from speech enhancement over robust feature extraction to model adaptation and word-based integration of multiple classifiers. The selective temporal cepstrum smoothing (TCS) technique is applied to enhance the reverberant speech signal at moderate noise levels, based on a statistical model of room impulse responses (RIRs) and minimum statistics (MS), considering estimates of late reverberations and the noise power spectrum densities (PSDs). Robust feature extraction is performed by amplitude modulation filtering of the cepstrogram to extract its temporal modulation information. As an alternative classifier, the acoustic models have been adopted using different RIRs and a RIR selection scheme based on a multi-layer perceptron (MLP) system that uses spectro-temporal features as the input. In the final stage, a system combination approach achieved by recognizer output voting error reduction (ROVER) is employed to obtain a jointly optimal recognized transcription. The proposed system has been evaluated in two different processing modes, i.e. *utterance-based batch processing* and *full batch processing*, which results in an overall average absolute improvement of 11% under variant reverberant conditions compared to the baseline system.

Index Terms— Automatic speech recognition, speech enhancement, feature extraction, amplitude modulation filterbank, acoustic model adaptation, reverberation, REVERB challenge

1. INTRODUCTION

Improving the performance of automatic speech recognition (ASR) in reverberant environments is still a major challenge in the signal enhancement and machine learning community [1, 2]. Strategies that aim to alleviate the influence of the reverberation effect range from dereverberation techniques in audio processing [3, 4] over robust features extraction methods to reverberant signal modeling in ASR [5, 6]. The REVERB (REverberant Voice Enhancement and Recognition Benchmark) challenge [7] provides a common evaluation framework for developing and combining a variety of algorithms in a multidisciplinary manner to attest robustness against reverberation. It contains common datasets, tasks and evaluation metrics for both speech enhancement and ASR in 1-channel (*1ch*), 2-channel (*2ch*) and 8-channel (*8ch*) microphone-arrays scenarios.

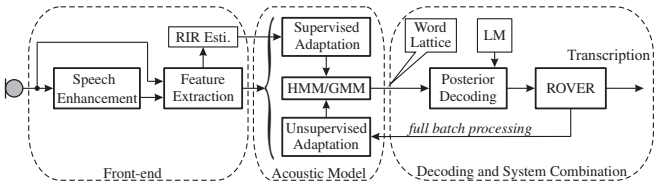

Fig. 1. Diagram of the proposed system structure including speech enhancement, robust feature extraction, acoustic model adaptation, posterior decoding and ROVER-based system combination.

This contribution proposes a system including speech enhancement, robust feature extraction, acoustic model adaptation, posterior decoding and word hypothesis fusion of multiple ASR systems, as depicted in Fig. 1. The evaluation of performance focuses on the ASR task in the *1ch* scenario and covers both, the *full batch processing* using all utterances from a single test condition and the *utterance-based batch processing* where each utterance is separately processed. In the following, the key components of the system are briefly described. The speech enhancement component is based on a hybrid system of dereverberation and noise reduction in the short-time Fourier transform (STFT) domain. Due to the stationary of the low-level background noise, the minimum statistics (MS) method [8] is employed to estimate the noise power spectral density (PSD). After estimating the reverberant speech PSD by the temporal cepstrum smoothing (TCS) technique [9], the approach described in [10] is used to determine the late reverberation PSD from the reverberant speech variance. With this quantity, the clean speech power is estimated, where TCS is employed again. Finally, the estimated PSDs are used to compute the minimum mean-square error (MMSE) estimate of the clean speech spectrum using the method described in [11]. A robust feature extraction method [5, 12] is adopted here to cope with reverberation by using an amplitude modulation filter set to extract the temporal modulation components spanning the frequency modulation bands of the cepstrogram.

To alleviate the mismatch between acoustic models and acoustic features that are derived from different acoustic test conditions, a RIR selection scheme is applied to select the best matching pre-

trained acoustic model. Therefore, the acoustic models are adopted to different RIRs using the maximum likelihood linear regression (MLLR) [13] technique. The RIR selection stage [14, 15] is based on a multi-layer perceptron (MLP) classifier that uses 2D Gabor features [16] as the input. To obtain the optimal hypotheses from competing word alternatives, a rescoring of word lattices is performed, which is based on estimates of posterior probability distributions of word sequences [17]. Finally, a system combination is performed that is based on recognizer output voting error reduction (ROVER) [18], in order to obtain a jointly optimal transcription for recognition from differently trained hidden Markov models (HMMs).

Notation: Matrices are written in bold uppercase letters and vectors are printed in boldface while scalars are printed in italic. Time-domain variables are written *italic* while STFT-domain variables are written in $\mathsf{sans-serif}$ letters. k, m and ℓ are the time, spectral and temporal frame index, respectively. The superscript ^{c} denotes the cepstral domain and the symbols $*$, $\odot$, $\oslash$ represent the convolution, element-by-element multiplication and element-by-element division of two vectors, respectively. $\mathrm{E}\{\cdot\}$ is the expectation value calculation and f_s is the sampling frequency. $\mathcal{F}\{\cdot, L\}$ and $\mathcal{F}^{-1}\{\cdot, L\}$ are the discrete Fourier transform (DFT) and the inverse DFT (IDFT) of size L, respectively.

The remainder of this paper is organized as follows: Section 2 describes the speech enhancement algorithm and the acoustic features are introduced in Section 3. Acoustic model adaptation as well as the selection scheme is addressed in Section 4, and Section 5 shows the decoding strategy and system combination. The experimental procedure and final results of the proposed system are presented in Section 6. Concluding remarks are given in Section 7.

2. SPEECH ENHANCEMENT

The single channel speech enhancement algorithm is used to dereverberate the speech signal. For this, the MMSE estimator of the clean speech magnitude described in [11] is used, which, however, requires the PSDs of the speech signal and the interference to be known. Consequently, we estimate these quantities from the corrupted input signal. It has been shown, e.g. in [3], that attenuating late reverberation is crucial for ASR since early reflections can be mitigated well by e.g. cepstral mean subtraction (CMS) [19]. The interference signal is considered to contain late reverberation and noise, while the desired signal is formed by the direct path and some early reflections. A hybrid speech enhancement system to deal with the late reverberation and noise is proposed to obtain the enhanced speech that improves the ASR performance.

2.1. System Structure

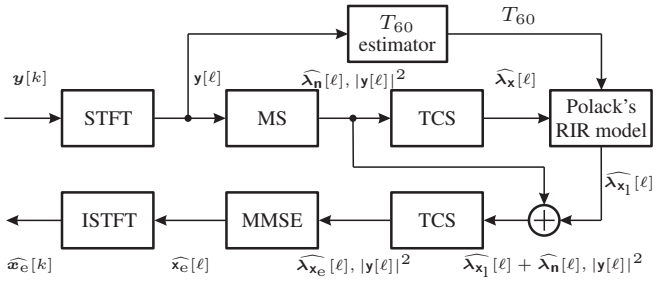

Fig. 2. Overview of the hybrid speech enhancement algorithm.

As depicted in Fig. 2, the recorded speech $\boldsymbol{y}[k]$ consists of the reverberant speech signal $\boldsymbol{x}[k]$ and the noise $\boldsymbol{n}[k]$. In other words, it also can be split into the desired signal $\boldsymbol{x}_\mathrm{e}[k]$ and interference signal $\boldsymbol{y}_\mathrm{i}[k]$, represented as follows,

$$\begin{aligned} \boldsymbol{y}[k] &= \boldsymbol{x}[k] + \boldsymbol{n}[k]\,, && (1)\\ &= \boldsymbol{s}[k] * \boldsymbol{h}[k] + \boldsymbol{n}[k]\,, && (2)\\ &= \boldsymbol{x}_\mathrm{e}[k] + \boldsymbol{x}_\mathrm{l}[k] + \boldsymbol{n}[k]\,, && (3)\\ &= \boldsymbol{x}_\mathrm{e}[k] + \boldsymbol{y}_\mathrm{i}[k]\,, && (4) \end{aligned}$$

where the reverberant speech signal $\boldsymbol{x}[k]$ can be modeled as the convolution of the clean signal $\boldsymbol{s}[k]$ and the room impulse response (RIR) $\boldsymbol{h}[k]$. Also, the reverberant speech $\boldsymbol{x}[k]$ can be rewritten as early part $\boldsymbol{x}_\mathrm{e}[k]$ and late reverberation $\boldsymbol{x}_\mathrm{l}[k]$. Usually the early part, considered as the desired signal, includes the direct path and early reflections up to about 50 ms of $\boldsymbol{h}[k]$ [20]. Although the REVERB challenge focuses on robustness against reverberation, a certain amount of stationary ambient noise is also added to the reverberant speech signal with a desired-signal-to-noise-ratio (DSNR) of approx. 20 dB [7].

The noise PSD $\boldsymbol{\lambda}_{\mathbf{n}}[\ell]$ is estimated using the minimum statistics (MS) method [8]. MS offers quite accurate noise PSD estimation if the noise signal is more or less stationary, i.e. varies only slowly. For this, first, an estimate of the input signal PSD is obtained by smoothing the noisy periodograms recursively. It is assumed that the minima in this PSD originate from time-frequency bins that do not contain speech. Accordingly, these minima are tracked using a search window spanning over the estimated input PSDs computed for the last 1.5 s. To ensure that the noise PSD estimate is not influenced by reverberation, we enlarged this window to 3 s. After obtaining $\widehat{\boldsymbol{\lambda}_{\mathbf{n}}}[\ell]$ in the STFT domain (for frames of length 32 ms with 16 ms frame shift), the PSD of the reverberant speech $\boldsymbol{\lambda}_{\mathbf{x}}[\ell]$ is estimated based on a statistical room acoustics model and temporal cepstrum smoothing.

2.2. Temporal Cepstrum Smoothing

We employ temporal cepstrum smoothing (TCS) [21, 9] for estimating the reverberant and the clean speech power. This approach smoothes the maximal likelihood (ML) estimate of the clean or reverberated speech PSD over time in the cepstral domain. Due to the compact representation of speech in the cepstral domain, speech related and non-speech related coefficients can be selectively smoothed. Note that here we take the estimate of the reverberant speech $\widehat{\boldsymbol{\lambda}_{\mathbf{x}}}[\ell]$ as an example. The same procedure is repeated to estimate $\widehat{\boldsymbol{\lambda}_{\mathbf{x}_\mathrm{e}}}[\ell]$ as shown in Fig. 2. The reverberant speech PSD is obtained by the ML estimate [22],

$$\widehat{\boldsymbol{\lambda}_{\mathbf{x}}^{\mathrm{ml}}}[\ell] = |\mathbf{y}[\ell]|^2 - \widehat{\boldsymbol{\lambda}_{\mathbf{n}}}[\ell]\,. \quad (5)$$

Then, the cepstral representation of the above ML estimate is calculated as

$$\widehat{\boldsymbol{\lambda}_{\mathbf{x}}^{c,\mathrm{ml}}}[\ell_c] = \mathcal{F}^{-1}\left\{\ln\left(\max\left(\widehat{\boldsymbol{\lambda}_{\mathbf{x}}^{\mathrm{ml}}},\, \xi_{\min}\cdot\widehat{\boldsymbol{\lambda}_{\mathbf{n}}}\right)\right),\, L_c\right\}, \quad (6)$$

where L_c is the IDFT length and ℓ_c is the cepstral or quefrency index. $\xi_{\min}$ is the lower bound of the *a priori* reverberant-signal-to-noise-ratio and is chosen as -30 dB. After that, smoothing is applied to (6), i.e.,

$$\widehat{\boldsymbol{\lambda}_{\mathbf{x}}^{c}}[\ell_c] = \boldsymbol{\alpha}^{c}[\ell_c] \odot \widehat{\boldsymbol{\lambda}_{\mathbf{x}}^{c}}[\ell_c - 1] + (\mathbf{1} - \boldsymbol{\alpha}^{c}[\ell_c]) \odot \widehat{\boldsymbol{\lambda}_{\mathbf{x}}^{c,\mathrm{ml}}}[\ell_c]\,, \quad (7)$$

where $\boldsymbol{\alpha}^c[\ell_c]$ represents a quefrency dependent smoothing coefficient vector, which should be chosen such that the coefficients relevant for speech production are maintained while the remaining coefficients are strongly smoothed. Thus, in a speech enhancement framework, usually $\boldsymbol{\alpha}^c[\ell_c]$ is chosen small for the speech spectral envelope represented by the low quefrencies and the fundamental period peak in the cepstrum [21]. In contrast to speech enhancement, preserving the fundamental frequency is not crucial for ASR systems, and $\boldsymbol{\alpha}^c[\ell_c]$ is chosen as [23],

$$\boldsymbol{\alpha}^c[\ell_c] = \begin{cases} 0.0 & \ell_c = 0, \ldots, \lceil f_s \cdot 0.5\,\mathrm{ms} \rceil - 1 \\ 0.5 & \ell_c = \lceil f_s \cdot 0.5\,\mathrm{ms} \rceil, \ldots, \lceil f_s \cdot 1\,\mathrm{ms} \rceil - 1 \\ 0.9 & \text{otherwise}\,. \end{cases} \tag{8}$$

Note that the application range of ℓ_c above is $0, \ldots, L_c/2$, which will be applied accordingly to the symmetric counterpart $L_c/2 + 1, \ldots, L_c - 1$. Finally, the reverberant speech PSD estimate $\widehat{\boldsymbol{\lambda}}_{\mathbf{x}}[\ell]$ is achieved after transforming the cepstral domain version back to the frequency domain,

$$\widehat{\boldsymbol{\lambda}}_{\mathbf{x}}[\ell] = \mathbf{b} \odot \exp\left(\mathcal{F}\left\{\widehat{\boldsymbol{\lambda}}^c_{\mathbf{x}},\, L_c\right\}\right)\,, \tag{9}$$

where the factor $\mathbf{b}$ compensates for the bias that occurs because of the averaging in the cepstral domain. According to [9], $\mathbf{b}$ is a function of the smoothing factor $\boldsymbol{\alpha}^c[\ell_c]$ in (8).

2.3. Late Reverberation PSD Estimation

According to Equations (2) and (3), the reverberant speech signal $\boldsymbol{x}[k]$ in the STFT domain can be expressed as follows,

$$\begin{aligned} \mathbf{x}[\ell] &= \mathbf{s}[\ell] \odot \mathbf{h}[\ell]\,, && (10)\\ &= \mathbf{x}_{\mathrm{e}}[\ell] + \mathbf{x}_{\mathrm{l}}[\ell]\,, && (11)\\ \mathbf{x}_{\mathrm{e}}[\ell] &= \sum_{\ell_{\mathrm{e}}=0}^{L_{\mathrm{e}}-1} \mathbf{s}[\ell - \ell_{\mathrm{e}}] \odot \mathbf{h}[\ell_{\mathrm{e}}]\,, && (12)\\ \mathbf{x}_{\mathrm{l}}[\ell] &= \sum_{\ell_{\mathrm{e}}=L_{\mathrm{e}}}^{\infty} \mathbf{s}[\ell - \ell_{\mathrm{e}}] \odot \mathbf{h}[\ell_{\mathrm{e}}]\,, && (13) \end{aligned}$$

where the parameter L_{e} is a number of frames which corresponds to the duration of early part of the RIR. Consequently, $L_{\mathrm{e}} L_{\mathrm{f}}/f_s$ is the start time of the late reverberation, while L_{f} denotes the number of samples between two successive analysis frames (hop size or window shift).

Reverberation suppression is achieved by an estimate of late reverberation PSD $\boldsymbol{\lambda}_{\mathbf{x}_{\mathrm{l}}}[\ell] = \mathrm{E}\{|\mathbf{x}_{\mathrm{l}}[\ell]|^2\}$ based on Polack's statistical RIR model [3]. By using the reverberant speech PSD estimation $\widehat{\boldsymbol{\lambda}}_{\mathbf{x}}[\ell]$ obtained in (9), the late reverberation PSD estimate can be calculated as [10]

$$\widehat{\boldsymbol{\lambda}}_{\mathbf{x}_{\mathrm{l}}}[\ell] = \exp(-2\rho L_{\mathrm{f}} L_{\mathrm{e}}) \cdot \widehat{\boldsymbol{\lambda}}_{\mathbf{x}}[\ell - L_{\mathrm{e}}]\,, \tag{14}$$

where ρ is the decay rate related to the reverberation time T_{60} as $\rho = 3\ln(10)/(T_{60} f_s)$. Blind reverberation time estimation as shown in Fig. 2 is achieved by the method proposed in [24].

2.4. MMSE Estimator

After the estimate of noise and late reverberation PSDs, the interference PSD from (4) can be obtained in a straightforward way as $\widehat{\boldsymbol{\lambda}}_{\mathbf{y}_{\mathrm{i}}}[\ell] = \widehat{\boldsymbol{\lambda}}_{\mathbf{x}_{\mathrm{l}}}[\ell] + \widehat{\boldsymbol{\lambda}}_{\mathbf{n}}[\ell]$, which will be used as the input of another TCS procedure (cf. Section 2.2) as drawn in Fig.2, in order to estimate the PSD $\widehat{\boldsymbol{\lambda}}_{\mathbf{x}_{\mathrm{e}}}[\ell]$ of the desired signal $\mathbf{x}_{\mathrm{e}}[\ell]$.

In the final step, a parameterized MMSE spectral magnitude estimator [11] is used to determine the weighting function $\mathbf{g}[\ell]$ to obtain the enhanced speech signal $\widehat{\mathbf{x}}_{\mathrm{e}}[\ell]$. A reduced computationally complex version [25] is used, which is defined as

$$\begin{aligned} \mathbf{g}[\ell] &= \Big(\mathbf{1} \oslash (\mathbf{1} + \boldsymbol{\nu}[\ell])\Big)^{\mathsf{p}_0} \odot \mathbf{g}_0[\ell] + \\ &\quad \Big(\boldsymbol{\nu}[\ell] \oslash (\mathbf{1} + \boldsymbol{\nu}[\ell])\Big)^{\mathsf{p}_\infty} \odot \widehat{\boldsymbol{\xi}}[\ell] \oslash (\mu \cdot \mathbf{1} + \widehat{\boldsymbol{\xi}}[\ell]), && (15)\\ \mathbf{g}_0[\ell] &= \Big(\widehat{\boldsymbol{\xi}}[\ell] \oslash (\mu \cdot \mathbf{1} + \widehat{\boldsymbol{\xi}}[\ell]) \oslash \widehat{\boldsymbol{\zeta}}[\ell]\Big)^{1/2} \cdot \\ &\quad \Big(\Gamma(\mu + \gamma/2)/\Gamma(\mu)\Big)^{1/\gamma}, && (16)\\ \boldsymbol{\nu}[\ell] &= \widehat{\boldsymbol{\zeta}}[\ell] \odot \widehat{\boldsymbol{\xi}}[\ell] \oslash (\mu \cdot \mathbf{1} + \widehat{\boldsymbol{\xi}}[\ell])\,, && (17) \end{aligned}$$

where $\Gamma(\cdot)$ is the complete gamma function and $\mathbf{1}$ is the vector of all ones with the same length of $\mathbf{g}[\ell]$. The estimates of the *a priori* and *a posteriori* desired-signal-to-interference-ratios are denoted as $\widehat{\boldsymbol{\xi}}[\ell] = \widehat{\boldsymbol{\lambda}}_{\mathbf{x}_{\mathrm{e}}}[\ell] \oslash \widehat{\boldsymbol{\lambda}}_{\mathbf{y}_{\mathrm{i}}}[\ell]$, and $\widehat{\boldsymbol{\zeta}}[\ell] = |\mathbf{y}[\ell]|^2 \oslash \widehat{\boldsymbol{\lambda}}_{\mathbf{y}_{\mathrm{i}}}[\ell]$, respectively. The constant parameters (μ, γ) can be fine-tuned to yield several types of estimators. In [11], $\mu = 0.5$ and $\gamma = 0.5$ have been identified as a good compromise between the amount of musical noise and the clarity of speech and are therefore also applied here. For obtaining the correct approximation for the selected values of μ and γ, p_0 and p_∞ have to be set to 0.5 and 1.0, respectively [25]. Subsequently, the estimated desired signal $\widehat{\mathbf{x}}_{\mathrm{e}}[\ell]$ is calculated by

$$\widehat{\mathbf{x}}_{\mathrm{e}}[\ell] = \max(\mathbf{g}[\ell], \mathrm{g}_{\min}) \odot \mathbf{y}[\ell]\,, \tag{18}$$

where $\mathrm{g}_{\min}$ is a lower bound for the weighting function $\mathbf{g}[\ell]$ which alleviates speech distortions, however, also limits the amount of interference suppression. In conformance with [26], -10 dB is chosen as a good value to improve the ASR performance in reverberant environments. Then, an inverse STFT (ISTFT) is conducted to reconstruct the output speech signal in the time domain $\widehat{\boldsymbol{x}_{\mathrm{e}}}[k]$ used for the subsequent ASR experiments.

3. FEATURE EXTRACTION

The baseline features for the challenge are mel-frequency cepstral coefficients (MFCCs) [27] plus their delta and double-delta coefficients combined with cepstral mean subtraction (CMS). To improve robustness towards channel mismatch and quasi-stationary noise, we investigate the effects of mean variance normalization (MVN) [19] applied to MFCCs and other feature types. Amplitude modulation features are employed as an alternative for MFCCs, and 2D Gabor features are used to select the appropriate HMM model adapted to a specific acoustic setup.

3.1. Amplitude Modulation Filterbank (AMFB) Features

The AMFB is employed as an ASR feature extraction method that analyzes the temporal dynamics of the speech [5, 12]. Here, the computation of AMFB features is based on the log-mel-spectrogram (for frames of length 25 ms with 10 ms frame shift), and a subsequent discrete cosine transform along the spectral axis, i.e. the cepstrogram. The amplitude modulation filters $\mathbf{q}$ are complex exponential functions that are modulated by a Hann-envelope, as described in (19) by the Hadamard product of $\mathbf{s}_{\mathrm{carr}}$ (cf. (20)) and $\mathbf{h}_{\mathrm{env}}$ (cf. (21)), in which i is the imaginary unit, ℓ_c denotes the frame

index of the cepstrogram, and W_{ℓ_c} is the Hann-envelope window length with the center index ℓ_{c_0}. Note that beyond the length W_{ℓ_c}, the coefficients of Hann-envelope are set to zeros in Equations (21) and (24).

$$\mathbf{q}[\ell_c] = \mathbf{s}_{\text{carr}}[\ell_c] \odot \mathbf{h}_{\text{env}}[\ell_c]\,, \quad (19)$$

$$\mathbf{s}_{\text{carr}}[\ell_c] = \exp\left(i\omega(\ell_c - \ell_{c_0})\right), \quad (20)$$

$$\mathbf{h}_{\text{env}}[\ell_c] = 0.5 - 0.5 \cdot \cos\left(\frac{2\pi(\ell_c - \ell_{c_0})}{W_{\ell_c} + 1}\right). \quad (21)$$

The periodicity of the sinusoidal-carrier function is defined by the radian frequency ω. By varying ω and W_{ℓ_c}, the AMFB can be tuned to cover different temporal amplitude modulation frequencies with different bandwidths. For the AMFB feature extraction we selected five AM filters, whose center frequencies and bandwidth settings are chosen according to the psycho-physically motivated AM filterbank proposed in [28], which has a constant bandwidth of 5 Hz for AM frequencies up to 10 Hz and a constant-Q relationship with value of 2 for higher modulation frequencies.

3.2. 2D Gabor Filterbank (GBFB) Features

2D Gabor features have been shown in earlier research to outperform traditional features in high-noise conditions for ASR tasks [16, 29]. In low-noise conditions with a high amount of reverberation, we found AMFB features to perform better than Gabor features calculated with a filterbank (GBFB features). However, it was shown that this feature type is suitable to estimate important room parameters when combined with a non-linear classifier [14]; in this study, GBFB features are hence used in a similar fashion, i.e. to generate estimates of the specific acoustic scene, which is subsequently applied to model adaptation [15].

2D Gabor filters $\mathbf{G}$ are localized spectro-temporal patterns that are used to filter log-mel-spectrograms, with a high sensitivity towards AM-modulations. They are defined by

$$\mathbf{G}[m, \ell] = \mathbf{S}_{\text{carr}}[m, \ell] \odot \mathbf{H}_{\text{env}}[m, \ell]\,, \quad (22)$$

$$\mathbf{S}_{\text{carr}}[m, \ell] = \exp\left(i\omega_m(m - m_0) + i\omega_\ell(\ell - \ell_0)\right), \quad (23)$$

$$\mathbf{H}_{\text{env}}[m, \ell] = 0.5 - 0.5 \cdot \cos\left(\frac{2\pi(m - m_0)}{W_m + 1}\right) \cdot \cos\left(\frac{2\pi(\ell - \ell_0)}{W_\ell + 1}\right), \quad (24)$$

where m and ℓ denote the (mel-)spectral and temporal frame indices, and W_m, W_ℓ are the Hann-envelope window lengths with the center indices m_0, ℓ_0, respectively. The periodicity of the sinusoidal-carrier function is defined by the radian frequencies ω_m, ω_ℓ, which allow the Gabor filters to be tuned to particular directions of spectro-temporal modulation. The inclusion of special cases of Gabor filters enables purely spectral and temporal filters. To construct the GBFB features, each log-mel-spectrogram is filtered with 48 filters in filterbank that cover temporal modulations from 2 to 25 Hz and spectral modulations from -0.25 to 0.25 cycle/channel, respectively. Filtering results in 600-dimensional feature vectors per time step, which are fed to a neural net (cf. Section 4.2). For details about the parameters of the filterbank, the reader is referred to [14].

4. ACOUSTIC MODEL ADAPTATION

The aim of acoustic model adaptation is to improve ASR performance either by feature normalization or by adapting the HMMs towards a particular acoustic test condition. Since there are various modes and different schemes for adaptation/adaptive training [30], cluster-based *supervised* adaptation and an *unsupervised* adaptation in the *full batch processing* mode are adopted in this contribution.

4.1. Cluster-based Supervised Adaptation

ASR performance degrades dramatically when the training and test conditions mismatch which can, e.g., be caused by differing room characteristics/RIRs. In order to alleviate the mismatch of reverberant conditions between training and test data, cluster-based adaptation is adopted here which is based on a series of sets of HMMs that is determined by the different kinds of measured RIRs from the REVERB challenge (24 *1ch* RIRs available).

Maximum likelihood linear regression (MLLR) [13] is employed, which uses the ML criterion to estimate a linear transform to adapt Gaussian mean and variance parameters of HMMs. Note that we only consider *mean* adaptation (MLLRMEAN) since mean values of Gaussian mixture models (GMMs) affect strongly the ASR performance in reverberant environments [6]. Several RIR-dependent models are estimated according to variant acoustic test conditions, which are determined by the available measured RIRs representing different rooms and different speaker-to-microphone positions. It is worthwhile noting that the acoustic condition of the test data is not available, which is detected blindly to select an appropriate adapted model for recognition. The selection scheme investigated in [14] is used here, which is based on a MLP classifier briefly described in the following.

4.2. Multi-layer Perceptron (MLP) Classifier

GBFB features (cf. Section 3.2) combined with a multi-layer perceptron (MLP) classifier are suitable to estimate acoustic room parameters [14]. The current study exploits the sensitivity of GBFBs and MLPs to estimate the current acoustic scene associated with specific RIRs (which also convey information about reverberation parameters) [15]. The MLP was trained to select the optimal RIR-dependent model based on all the 24 training datasets as described in Section 4.1. The MLP had 600 input neurons (dimensionality of GBFB features without temporal context), 400 hidden units and 24 output neurons (corresponding to 24 RIRs in the training data). In the MLP forward run, the frame-wise probabilities for each class (corresponding to the 24 acoustic scenarios) are merged to a single decision by temporal integration and application of a *winner-takes-all* decision rule (cf. [14]). The model with the highest scores is subsequently selected.

4.3. Unsupervised Adaptation

In the *full batch processing* mode in which all the test data in each test set is available before recognition, an *unsupervised* adaptation scheme is explored. For this, the labels of a previous recognition step are used to adapt the HMMs to the test data. This method is sometimes also referred to as *self* adaptation [31]. The constrained MLLR (CMLLR) technique [32], which simultaneously updates the mean and variance parameters of the HMM-GMMs, is used in the baseline system of the REVERB challenge for unsupervised adaptation step. However, we use MLLRMEAN instead to adapt just the means of the acoustic models.

Furthermore, an enhanced version of the aforementioned unsupervised adaptation is proposed in order to further reduce the recognized WERs. Before processing the unsupervised adaptation with its own recognized transcription of each test set from each individual model, the recognized transcription can be further improved by

system combination (cf. Section 5.2). Thus, the improved WERs obtained from the system combination can be used as better transcriptions for the unsupervised adaptation (cf. Fig. 1).

5. DECODING AND SYSTEM COMBINATION

The final stage of ASR systems is to find the best sequence of words given a sequence of acoustic observations (i.e. the enhanced features, cf. Fig. 1), by taken into account a statistical language model and a pronunciation lexicon (supplied by the REVERB challenge).

5.1. Posterior Decoding

For ASR systems based on HMMs with underlying GMMs, the Viterbi algorithm is the most popular decoding method to find the 1-best hypotheses with the maximal log-probability, which usually minimizes the sentence error rate. However, speech recognizers are usually evaluated primarily for their WERs. In order to minimize the WERs, the decoding scheme from [17] is adopted, which is a lattice-based rescoring algorithm that explicitly estimates and minimizes word errors. The posterior distribution over all hypotheses will be firstly approximated, and the expected WERs for lattice-based hypotheses are computed, in which the word-lattice is generated by the *token-passing* decoder [33], as an alternative formulation of the Viterbi algorithm. The final result picks the one with the lowest expected WER, as implemented in the SRI language modeling (SRILM) toolkit [34]. In addition, this posterior decoding also brings another advantage, that is, the posterior probability of the recognized word can be considered as a confidence score measure [35] for system combination as discussed below.

5.2. ROVER for System Combination

National Institute of Standards and Technology (NIST)'s ROVER [18] has been shown to be effective to further reduce the WER by combining the output transcriptions of multiple recognizers. A voting scheme is applied to select the best scoring word based on the new transcription, which depends on two factors. One is the word frequency f_{rover} and the other is confidence score c_{rover} of each word w in the corresponding set j, represented as a scoring formula,

$$\boldsymbol{s}_{\mathrm{rover}}[\mathsf{w},\mathsf{j}] = \eta_{\mathrm{rover}} \cdot \boldsymbol{f}_{\mathrm{rover}}[\mathsf{w},\mathsf{j}] + (1-\eta_{\mathrm{rover}}) \cdot \boldsymbol{c}_{\mathrm{rover}}[\mathsf{w},\mathsf{j}]\,, \quad (25)$$

where η_{rover} denotes the weight parameter to be the tradeoff between using $\boldsymbol{f}_{\mathrm{rover}}$ and $\boldsymbol{c}_{\mathrm{rover}}$. In addition, confidence scores for NULL or silence transition arcs will be determined separately by another trained parameter η_{sil} due to its non-associated c_{rover}. In general, these two parameters need to be fine-tuned to obtain the best output word transcription based on the development test set. Herein, the confidence scores are measured by the word lattice-based posterior decoding (cf. Section 5.1).

6. EXPERIMENTS AND RESULTS

Experiments and results shown in the following are all carried out according to the instructions of the REVERB challenge [7].

6.1. Database

The database provided by the REVERB challenge consists of simulated data (SimData) and real recordings (RealData) for different room sizes and speaker-to-microphone distances. Based on the WSJCAM0 corpus [36], SimData is artificially generated by convolving clean WSJCAM0 signals with the measured RIRs, as well as adding additional measured noises with a DSNR of approx. 20 dB. 6 different acoustic conditions are simulated in the SimData, considering 3 different room sizes with 2 different speaker-to-microphone distances [7]. The RealData utterances of the MC-WSJ-AV corpus [37] were recorded in a room with 2 different speaker-to-microphone distances. The sampling frequency of the REVERB data is 16 kHz.

To evaluate the ASR performance, a training set, a development test set (Dev.) and a final evaluation test set (Eval.) are provided. For a multi-condition training set, 24 RIRs and several types of stationary noise signals are measured according to the above 6 reverberant conditions. The text prompts of the utterances are based on WSJ 5K corpus [38], and a language model is generated as bigram.

6.2. ASR System

A baseline speech recognition framework is provided by the REVERB challenge that is based on the HTK toolkit [39], where MFCCs including delta and double-delta coefficients (dimension of 39) are employed together with CMS, referred as to MFCC-CMS. A multi-condition acoustic model is trained according to the ML criterion, where context-dependent triphone HMMs with 3 states per model are applied together with 12 GMMs per state. The baseline bigram language model is used and the language model scaling factor is adjusted to 14.0 for MFCCs and 25.0 for AMFB features (dimension of 117). The MLLRMEAN and CMLLR adaptation schemes involve two passes [39], i.e. the first pass is a global adaptation that builds a global transform used in the second pass, for which a regression class tree with up to 256 leaf nodes is generated. During the word lattice generation considering of time and memory consumption, the number of tokens is chosen to 8.

If the speech enhancement module (cf. Section 2) is applied, it is used for the training data as well as the test data and henceforth indicated by the acronym SE. MVN is applied to MFCCs and AMFB features, indicated by MFCC-MVN and AMFB-MVN, respectively. For the MLP-MFCC-MVN classifier, 24 acoustic models are generated by adapting the *mean* of the multi-condition trained models w.r.t. 24 measured RIRs. In the *utterance-based batch processing* mode, the GBFB features and the MLP classifier are used to determine which model (among the 24 RIR-dependent models) fits best to a given test utterance. However, in the *full batch processing* mode the MLP decisions of all utterances from a single test condition are accumulated, i.e. a single model is selected that matches the majority of the test utterances. This is done in order to eliminate individual estimation errors of the MLP.

6.3. Results

Table 1 shows the ASR results of both the Dev. and the Eval. test set in the *utterance-based batch processing* mode. Firstly, considering the Dev. part, it can be seen from MFCC-MVN that posterior decoding performs slightly better than 1-best (Viterbi) decoding, e.g. approx. 0.3% absolute improvement in average WERs for the SimData. In general, MVN provides more robust feature extraction than the baseline CMS, especially in rooms with high reverberation, e.g. 5% average absolute improvement for the RealData. A constant 1 to 1.5% absolute WER reduction can be observed by the proposed speech enhancement algorithm for both MFCCs and AMFB features. With the RIR-dependent adapted models and MLP selector, an absolute improvement of about 1.5% is obtained if MLP-MFCC-MVN is compared to MFCC-MVN, e.g. average WER of the SimData. The AMFB features achieve an average absolute WER reduction of more than 4% compared to MFCCs, and this difference becomes particularly evident under far-field conditions in large

test set	utterance-based batch processing + multi-condition training	SimData							RealData		
		Room1		Room2		Room3		Ave.	Room1		Ave.
		Near	Far	Near	Far	Near	Far	-	Near	Far	-
Dev.	MFCC-CMS$_{\text{baseline}}$	15.49	18.90	23.51	42.40	27.25	46.07	28.92	52.96	51.61	52.28
	MFCC-MVN$_{\text{Viterbi}}$	14.33	18.29	22.48	38.35	25.82	42.28	26.91	47.16	46.82	46.99
	MFCC-MVN[1]	14.36	18.02	22.38	37.76	25.62	41.74	26.63	46.79	47.16	46.97
	SE-MFCC-MVN[2]	15.49	18.76	22.87	34.56	24.21	35.51	25.22	45.98	45.93	45.95
	MLP-MFCC-MVN[3]	13.40	17.48	19.37	35.08	23.24	41.32	24.96	45.04	46.21	45.62
	AMFB-MVN[4]	11.90	15.86	18.39	32.14	20.72	35.53	22.41	42.17	45.11	43.63
	SE-AMFB-MVN[5]	13.25	16.08	18.68	29.09	18.62	**30.09**	20.95	41.55	43.68	42.60
	ROVER[4,5]	11.92	14.38	16.79	28.89	**18.32**	30.59	20.13	40.42	41.08	40.74
	ROVER[2,4,5]	11.63	14.04	16.64	**28.07**	19.93	32.74	20.49	**40.05**	41.70	40.87
	ROVER[1,2,3,4,5]	**11.04**	**13.74**	**16.54**	28.49	18.35	31.70	**19.96**	40.92	**39.71**	**40.31**
Eval.	MFCC-CMS$_{\text{baseline}}$	20.84	21.72	23.43	38.59	28.43	44.79	29.62	59.09	55.81	57.45
	MFCC-MVN[1]	16.81	19.33	22.05	34.54	27.18	39.82	26.61	51.17	50.03	50.60
	SE-MFCC-MVN[2]	17.50	19.45	21.96	31.60	25.22	34.94	25.10	48.77	48.51	48.64
	MLP-MFCC-MVN[3]	15.77	18.23	19.86	32.77	24.65	40.56	25.30	52.09	48.68	50.38
	AMFB-MVN[4]	16.01	17.47	18.43	29.25	21.72	34.33	22.86	47.56	46.02	46.79
	SE-AMFB-MVN[5]	16.37	17.28	18.14	26.16	20.68	30.61	21.53	43.47	44.43	43.95
	ROVER[4,5]	14.59	15.99	16.43	25.87	19.33	**30.27**	20.40	42.64	43.15	42.89
	ROVER[2,4,5]	13.86	**14.88**	16.23	**25.52**	**19.31**	30.85	20.10	**41.92**	42.37	42.14
	ROVER[1,2,3,4,5]	**13.64**	14.93	**16.11**	25.65	19.77	30.51	**20.09**	41.97	**42.27**	**42.12**

Table 1. ASR results for both the Dev. and the Eval. test set in the *utterance-based batch processing* mode. Results are given in WERs [%] to evaluate the ASR performance with 5 strategies proposed in this work labeled by the superscript number from 1 to 5, besides the baseline approach. The results with 3 different combinations for ROVER are enumerated, e.g. ROVER^{2,4,5} means that ROVER combines the output transcriptions from strategy 2, 4, and 5.

rooms. If the speech enhancement is added before the AMFB features, further absolute improvement in WERs of approx. 1.5% is obtained. By combining the recognizer outputs of all tested systems using ROVER, an additional average absolute WER reduction of 1 to 2% is achieved. The similar trends of the WER reduction can be observed for the Eval. test set in Table 1. Arguably, the improvement by ROVER also depends on how much complementary information the selected recognizers can offer. From the Dev. test set, we found that the best average results come from the case that all 5 available recognizer transcriptions are combined by ROVER with fine-tuned parameters $\eta_{\text{rover}} = 0.08$ and $\eta_{\text{sil}} = 0.98$, even though it did not give the best at some specific test conditions, e.g. at Near position of RealData. Finally, compared to the baseline results of the Eval. test set, 9.53% and 15.33% absolute reduction of the average WERs are yielded by our best recognizer transcription for SimData and RealData, respectively.

The results of the *full batch processing* mode are given in Table 2. Unsupervised adaptation for each individual test set is launched after the *utterance-based batch processing* mode. Compared to MFCC-MVN$_{\text{CMLLR,self}}$ using its own recognition transcription, lower WERs achieved by MFCC-MVN$_{\text{CMLLR,rover}}$ with the recognition transcription from ROVER for the Dev. test set indicate that a better recognition transcription assists CMLLR to better adapt the model to the direction that matches more to the test set condition. Therefore, the better recognition results obtained by ROVER from Table 1 are employed to improve the unsupervised adaptation performance in each test set condition. Furthermore, it is seen from the Dev. test set that *variance* adaptation does not help to increase ASR performance when CMLLR is compared with MLLRMEAN scheme. The similar trend of the WERs reduction as in the *utterance-based batch processing* mode in Table 1 can be observed in Table 2 achieved by the 5 proposed strategies for both the Dev. and the Eval. test set. After several combination trials with different recognizer outputs for ROVER for the Dev. test set, it can be found that the combination of SE-MFCC-MVN, AMFB-MVN and SE-AMFB-MVN (ROVER^{2,4,5}) gives the best average scores, while more recognizer transcriptions even decrease ROVER performance, which may be due to the fact that too many erroneously recognized recognition words may cover the good ones during the voting in ROVER. Compared to the baseline results of the Eval. test set, 7.3% and 11.53% absolute reduction of the average WERs are obtained for SimData and RealData, respectively.

7. CONCLUSION

This contribution proposed a single channel combined system consisting of speech enhancement, robust feature extraction, acoustic model adaptation, posterior decoding and ROVER-based system combination, which could substantially improve the ASR performance on the *1ch* task in both the *utterance-based batch processing* and *full batch processing* modes of the REVERB challenge. Compared to the baseline results of the final evaluation test set, we obtained an average WER absolute improvement of 12.43% in the *utterance-based batch processing* mode and 9.42% in the *full batch processing* mode, respectively. Each component of our proposed system could itself and/or in combination contribute to the mentioned improvements. Results also show that speech enhancement based on temporal cepstrum smoothing is proven to be advantageous to cope with the reverberation effect to ASR systems. Another major improvement comes from AMFB features, which may indicate that capturing the temporal modulation information is crucial for feature extraction when facing the reverberant speech for ASR systems.

test set	full batch processing + multi-condition training + unsupervised adaptation	SimData							RealData		
		Room1		Room2		Room3		Ave.	Room1		Ave.
		Near	Far	Near	Far	Near	Far	-	Near	Far	-
Dev.	MFCC-CMS$_{\mathrm{CMLLR,baseline}}$	13.27	17.08	20.80	36.83	23.54	39.44	25.14	47.91	46.55	47.23
	MFCC-MVN$_{\mathrm{CMLLR,self}}$	14.04	17.80	20.93	33.05	22.03	37.46	24.20	45.35	43.34	44.35
	MFCC-MVN$_{\mathrm{CMLLR,rover}}$	13.52	16.91	20.51	32.41	21.46	36.15	23.48	44.48	41.22	42.85
	MFCC-MVN$_{\mathrm{MLLRMEAN}}{}^{1}$	12.54	16.94	19.08	31.43	21.17	35.51	22.76	42.36	41.76	42.06
	SE-MFCC-MVN$_{\mathrm{MLLRMEAN}}{}^{2}$	13.82	17.23	19.40	28.42	19.81	29.97	21.43	40.61	41.15	40.87
	MLP-MFCC-MVN$_{\mathrm{MLLRMEAN}}{}^{3}$	12.02	16.10	18.09	30.76	21.17	35.56	22.27	41.73	41.15	41.44
	AMFB-MVN$_{\mathrm{MLLRMEAN}}{}^{4}$	10.45	13.62	16.05	27.04	16.96	29.75	18.96	38.37	38.96	38.66
	SE-AMFB-MVN$_{\mathrm{MLLRMEAN}}{}^{5}$	10.57	14.01	15.87	25.07	15.36	27.20	18.00	38.30	**36.84**	37.57
	ROVER^{4,5}	10.23	13.20	15.38	25.59	15.60	27.82	17.96	**38.05**	37.66	37.85
	ROVER^{2,4,5}	**9.93**	**12.98**	**15.26**	**24.75**	**15.18**	**26.76**	**17.46**	38.19	36.91	**37.55**
	ROVER^{1,2,3,4,5}	10.25	13.35	15.48	25.39	16.27	28.46	18.19	**38.05**	37.18	37.61
Eval.	MFCC-CMS$_{\mathrm{CMLLR,baseline}}$	16.38	18.89	20.37	33.00	24.96	39.26	25.47	50.85	48.11	49.48
	MFCC-MVN$_{\mathrm{MLLRMEAN}}{}^{1}$	15.45	17.84	18.88	28.56	22.64	35.42	23.12	42.48	42.84	42.66
	SE-MFCC-MVN$_{\mathrm{MLLRMEAN}}{}^{2}$	16.16	18.13	18.73	26.13	21.89	31.36	22.06	40.59	41.22	40.90
	MLP-MFCC-MVN$_{\mathrm{MLLRMEAN}}{}^{3}$	14.74	16.76	18.56	28.32	22.45	35.06	22.64	43.63	42.98	43.30
	AMFB-MVN$_{\mathrm{MLLRMEAN}}{}^{4}$	13.64	15.13	15.37	24.54	18.48	30.05	19.53	39.73	40.82	40.27
	SE-AMFB-MVN$_{\mathrm{MLLRMEAN}}{}^{5}$	13.96	14.83	15.45	23.19	18.00	28.41	18.96	37.59	39.23	38.41
	ROVER^{4,5}	13.20	14.22	**14.63**	22.95	17.45	28.38	18.46	**37.16**	39.43	38.29
	ROVER^{2,4,5}	**12.72**	**13.86**	14.68	**22.77**	**17.16**	**27.90**	**18.17**	37.24	**38.69**	**37.96**
	ROVER^{1,2,3,4,5}	12.96	14.30	14.81	23.38	18.44	29.01	18.81	37.24	38.83	38.03

Table 2. ASR results for both the Dev. and the Eval. test set in the *full batch processing* mode. Results are given in WERs [%] to evaluate the ASR performance with 5 strategies proposed in this work labeled by the superscript number from 1 to 5 (refer to Table 1).

8. ACKNOWLEDGMENTS

The research leading to these results has received funding from the EU Seventh Framework Programme project Dereverberation and Reverberation of Audio, Music, and Speech (DREAMS) under grant agreement ITN-GA-2012-316969, from the DFG-Cluster of Excellence EXC 1077/1 "Hearing4all", and from the MWK PhD Program "Signals and Cognition".

9. REFERENCES

[1] M. Wölfel and J. McDonough, *Distant Speech Recognition*, John Wiley & Sons Ltd, 2009.

[2] T. Yoshioka, A. Sehr, M. Delcroix, K. Kinoshita, R. Maas, T. Nakatani, and W. Kellermann, "Making Machines Understand Us in Reverberant Rooms: Robustness against Reverberation for Automatic Speech Recognition," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 114–126, Nov. 2012.

[3] E.A.P. Habets, *Single- and Multi-Microphone Speech Dereverberation using Spectral Enhancement*, Ph.D. thesis, University of Eindhoven, Eindhoven, The Netherlands, Jun. 2007.

[4] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, "Speech Dereverberation based on Variance-Normalized Delayed Linear Prediction," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1717–1731, Sep. 2010.

[5] N. Moritz, J. Anemüller, and B. Kollmeier, "Amplitude Modulation Spectrogram based Features for Robust Speech Recognition in Noisy and Reverberant Environments," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Prague, Czech Republic, May 2011, pp. 5492–5495.

[6] A. Sehr, R. Maas, and W. Kellermann, "Reverberation Model-based Decoding in the Logmelspec Domain for Robust Distant-Talking Speech Recognition," *IEEE Trans. Audio, Speech and Language Processing*, vol. 18, no. 7, pp. 1676–1691, Sep. 2010.

[7] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, E. Habets, R. Haeb-Umbach, V. Leutnant, A. Sehr, W. Kellermann, R. Maas, S. Gannot, and B. Raj, "The REVERB Challenge: A Common Evaluation Framework for Dereverberation and Recognition of Reverberant Speech," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, Oct. 2013.

[8] R. Martin, "Noise Power Spectral Density Estimation based on Optimal Smoothing and Minimum Statistics," *IEEE Trans. Speech and Audio Processing*, vol. 9, no. 5, pp. 504–512, Jul. 2001.

[9] T. Gerkmann and R. Martin, "On the Statistics of Spectral Amplitudes after Variance Reduction by Temporal Cepstrum Smoothing and Cepstral Nulling," *IEEE Trans. Signal Processing*, vol. 57, no. 11, pp. 4165–4174, Nov. 2009.

[10] K. Lebart, J.M. Boucher, and P.N. Denbigh, "A New Method based on Spectral Subtraction for Speech Dereverberation," *Acta Acustica united with Acustica*, vol. 87, no. 3, pp. 359–366, May 2001.

[11] C. Breithaupt, M. Krawczyk, and R. Martin, "Parameterized MMSE Spectral Magnitude Estimation for the Enhancement of Noisy Speech," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Las Vegas, NV, USA, Apr. 2008, pp. 4037–4040.

[12] N. Moritz, J. Anemüller, and B. Kollmeier, "Amplitude Modulation Filters as Feature Sets for Robust ASR: Constant Abso-

lute or Relative Bandwidth?," in *Proc. Interspeech*, Portland, USA, Sep. 2012, pp. 1230–1233.

[13] C.J. Leggetter and P.C. Woodland, "Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density Hidden Markov Models," *Computer Speech and Language*, vol. 9, pp. 171–185, Feb. 1995.

[14] F. Xiong, S. Goetze, and B.T. Meyer, "Blind Estimation of Reverberation Time based on Spectro-Temporal Modulation Filtering," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, May 2013, pp. 443–447.

[15] F. Xiong, S. Goetze, and B.T. Meyer, "Estimating Room Acoustic Parameters for Speech Recognizer Adaptation and Combination in Reverberant Environments," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Florence, Italy, May 2014.

[16] M.R. Schädler, B.T. Meyer, and B. Kollmeier, "Spectro-Temporal Modulation Subspace-Spanning Filter Bank Features for Robust Automatic Speech Recognition," *J. Acoust. Soc. Am.*, vol. 131, no. 5, pp. 4134–4151, 2012.

[17] A. Stolcke, Y. König, and M. Weintraub, "Explicit Word Error Minimization in N-Best List Rescoring," in *Proc. 5th European Conference on Speech Communication and Technology*, Rhodes, Greece, Sep. 1997, vol. 1, pp. 163–166.

[18] J.G. Fiscus, "A Post-Processing System to Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction (ROVER)," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, Santa Barbara, CA, USA, Dec. 1997, pp. 347–354.

[19] B. Atal, "Effectiveness of Linear Prediction Characteristics of the Speech Wave for Automatic Speaker Identification and Verification," *J. Acoust. Soc. Am.*, vol. 55, no. 6, pp. 1304–1322, Jun. 1974.

[20] H. Kuttruff, *Room Acoustics*, Spon Press, London, 4th edition, 2000.

[21] C. Breithaupt, T. Gerkmann, and R. Martin, "A Novel A Priori SNR Estimation Approach based on Selective Cepstro-Temporal Smoothing," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Las Vegas, NV, USA, Apr. 2008, pp. 4897–4900.

[22] Y. Ephraim and D. Malah, "Speech Enhancement using a Minimum Mean-Square Error Short-Time Spectral Amplitude Estimator," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. 32, no. 6, pp. 1109–1121, Dec. 1984.

[23] H. Meutzner, A. Schlesinger, S. Zeiler, and D. Kolossa, "Binaural Signal Processing for Enhanced Speech Recognition Robustness in Complex Listening Environments," in *Proc. 2nd CHiME Workshop on Machine Listening in Multisource Environments*, Vancouver, Canada, Jun. 2013, pp. 7–12.

[24] J. Eaton, N.D. Gaubitchy, and P.A. Naylor, "Noise-Robust Reverberation Time Estimation using Spectral Decay Distributions with Reduced Computational Cost," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, May 2013, pp. 161–165.

[25] C. Breithaupt, *Noise Reduction Algorithms for Speech Communications - Statistical Analysis and Improved Estimation Procedures*, Ph.D. thesis, Ruhr-Universität Bochum, Bochum, Germany, 2008.

[26] R. Maas, E.A.P. Habets, A. Sehr, and W. Kellermann, "On the Application of Reverberation Suppression to Robust Speech Recognition," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Kyoto, Japan, Mar. 2012, pp. 297–300.

[27] S.B. David and P. Mermelstein, "Comparison of Parametric Representation for Monosyllabic Word Recognition in Continuously Spoken Sentences," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 28, no. 4, pp. 357–366, Aug. 1980.

[28] T. Dau, B. Kollmeier, and A. Kohlrausch, "Modeling Auditory Processing of Amplitude Modulation. I. Detection and Masking with Narrow-Band Carriers," *J. Acoust. Soc. Am.*, vol. 102, no. 5, pp. 2892–2905, Nov. 1997.

[29] N. Moritz, M.R. Schädler, B.T. Meyer K. Adiloglu, T. Jürgens, T. Gerkmann, B. Kollmeier, S. Doclo, and S. Goetze, "Noise Robust Distant Automatic Speech Recognition Utilizing NMF based Source Separation and Auditory Feature Extraction," in *Proc. 2nd CHiME Workshop on Machine Listening in Multisource Environments*, Vancouver, Canada, Jun. 2013, pp. 1–6.

[30] K. Yu, *Adaptive Training for Large Vocabulary Continuous Speech Recognition*, Ph.D. thesis, Hughes Hall College and Cambridge University Engineering Department, University of Cambridge, UK, Jul. 2006.

[31] M.J.F. Gales, "Adaptive Training for Robust ASR," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, Madonna di Campiglio, Italy, Dec. 2001, pp. 15–20.

[32] V.V. Digalakis, D. Rtischev, and L.G. Neumeyer, "Speaker Adaptation using Constrained Estimation of Gaussian Mixtures," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 3, no. 5, pp. 357–366, Sep. 1995.

[33] S.J. Young, N.H. Russell, and J.H.S. Thornton, "Token Passing: A Simple Conceptual Model for Connected Speech Recognition Systems," Tech. Rep., Cambridge University Engineering Department, 1989.

[34] A. Stolcke, "SRILM - An Extensible Language Modeling Toolkit," in *International Conference on Spoken Language Processing*, Denver, Colorado, USA, Sep. 2002, pp. 901–904.

[35] F. Wessel, K. Macherey, and R. Schlüter, "Using Word Probabilities as Confidence Measures," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Seattle, WA, USA, May 1998, vol. 1, pp. 225–228.

[36] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals, "WSJCAM0: A British English Speech Corpus for Large Vocabulary Continuous Speech Recognition," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Detroit, Michigan, USA, May 1995, pp. 81–84.

[37] M. Lincoln, I. McCowan, J. Vepa, and H.K. Maganti, "The Multi-Channel Wall Street Journal Audio Visual Corpus (MC-WSJ-AV): Specification and Initial Experiments," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, San Juan, Puerto Rico, Nov. 2005, pp. 357–362.

[38] J. Garofalo, D. Graff, D. Paul, and D. Pallett, "CSR-I (WSJ0) Complete," in *Linguistic Data Lconsortium (LDC)*, Philadelphia, USA, 2007.

[39] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X.A. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book (for HTK Version 3.4)*, Cambridge University Engineering Department, Cambridge, 2009, http://htk.eng.cam.ac.uk/.