We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



1

A Real-Time Speech Enhancement Front-End for Multi-Talker Reverberated Scenarios

Rudy Rotili, Emanuele Principi, Stefano Squartini and Francesco Piazza
Università Politecnica delle Marche
Italy

1. Introduction

In the direct human interaction, the verbal and nonverbal communication modes play a fundamental role by jointly cooperating in assigning semantic and pragmatic contents to the conveyed message and by manipulating and interpreting the participants' cognitive and emotional states from the interactional contextual instance. In order to understand, model, analyse, and automatize such behaviours, converging competences from social and cognitive psychology, linguistic, philosophy, and computer science are needed.

The exchange of information (more or less conscious) that take place during interactions build up a new knowledge that often needs to be recalled, in order to be re-used, but sometime it also needs to be appropriately supported as it occurs. Currently, the international scientific research is strongly committed towards the realization of intelligent instruments able to recognize, process and store relevant interactional signals: The goal is not only to allow efficient use of the data retrospectively but also to assist and dynamically optimize the experience of interaction itself while it is being held. To this end, both verbal and nonverbal (gestures, facial expressions, gaze, etc.) communication modes can be exploited. Nevertheless, voice is still a popular choice due to informative content it carries: Words, emotions, dominance can all be detected by means of different kinds of speech processing techniques. Examples of projects exploiting this idea are CHIL (Waibel et al. (2004)), AMI-AMIDA (Renals (2005)) and CALO (Tur et al. (2010)).

The applicative scenario taken here as reference is a professional meeting, where the system can readily assists the participants and where the participants themselves do not have particular expectations on the forms of supports provided by the system. In this scenario, it is assumed that people are sitting around a table, and the system supports and enrich the conversation experience by projecting graphical information and keywords on a screen.

A complete architecture of such a system has been proposed and validated in (Principi et al. (2009); Rocchi et al. (2009)). It consists of three logical layers: Perception, Interpretation and Presentation. The Perception layer aims to achieve situational awareness in the workplace and is composed of two essential elements: Presence Detector and Speech Processing Unit. The first determines the operating states of the system: Presence (the system checks if there are people around the table); conversation (the system senses that a conversation is ongoing). The Speech Processing Unit processes the captured audio signals and identifies the keywords that are exploited by the system in order to decide which stimuli to project. It consists of

two main components: The multi-channel front-end (speech enhancement) and the automatic speech recognizer (ASR).

The Interpretation module is responsible of the recognition of the ongoing conversation. At this level, semantic representation techniques are adopted in order to structure both the content of the conversation and how the discussion is linked to the speakers present around the table. Closely related to this module is the Presentation one that, based on conversational analysis just made, dynamically decides which stimuli have to be proposed and sent. The stimuli are classified in terms of conversation topics and on the basis of their recognition, they are selected and projected on the table.

The focus of this chapter is on the speech enhancement stage of the Speech Processing Unit and in particular on the set of algorithms constituting the front-end of the ASR. In a typical meeting scenario, participants' voices can be acquired through different type of microphones. Depending on the choice made, the microphone signals are more or less susceptible to the presence of noise, the interference from other co-existing sources and reverberation produced by multiple acoustic paths. The usage of close-talking microphones can mitigate the aforementioned problems but they are invasive and the meeting participants can feel uncomfortable in such situation. A less invasive and more flexible solution is the choice of far-field microphone arrays. In this situation, the extraction of a desired speech signal can be a difficult task since noise, interference and reverberation are more relevant.

In the literature, several solutions have been proposed in order to alleviate the problems (Naylor & Gaubitch (2010); Woelfel & McDonough (2009)): Here, the attention is on two popular techniques among them, namely blind source separation (BSS) and speech dereverberation. In (Huang et al. (2005)), a two stage approach leading to sequential source separation and speech dereverberation based on blind channel identification (BCI) is proposed. This can be accomplished by converting the multiple-input multiple-output (MIMO) system into several single-input multiple-output (SIMO) systems free of any interference from the other sources. Since each SIMO system is blindly identified at different time, the BSS algorithm does not suffer of the annoying permutation ambiguity problem. Finally, if the obtained SIMO systems room impulse responses (RIRs) do not share common zeros, dereverberation can be performed by using the Multiple-Input/Output Inverse Theorem (MINT) (Miyoshi & Kaneda (1988)).

A real-time implementation of this approach has been presented in (Rotili et al. (2010)), where the optimum inverse filtering approach is substituted by an iterative technique, which is computationally more efficient and allows the inversion of long RIRs in real-time applications (Rotili et al. (2008)). Iterative inversion is based on the well known steepest-descent algorithm, where a regularization parameter taking into account the presence of disturbances, makes the dereverberation more robust to RIRs fluctuations or estimation errors due to the BCI algorithm (Hikichi et al. (2007)).

The major drawback of such implementation is that the BCI stage need to know "who speaks when" in order to estimate the RIRs related to the right speaker. To overcome the problem, in this chapter a solution which exploits a speaker diarization system is proposed. Speaker diarization steers the BCI and the ASR, thus allowing the identification task to be accomplished directly on the microphone mixture.

The proposed framework, is developed on the NU-Tech platform (Squartini et al. (2005)), a freeware software which allows the efficient management of the audio stream by means of the ASIO interface. NU-Tech provides a useful plug-in architecture which has been exploited for the C++ implementation. Experiments performed over synthetic conditions at 16 kHz sampling rate confirm the real-time capabilities of the implemented architecture and its effectiveness as multi-channel front-end for the subsequent speech recognition engine. The chapter outline is the following. In Sec. 2 the speech enhancement front-end, aimed at separating and dereverberating the speech sources is described, whereas Sec. 3 details the ASR engine and its parametrization. Sec. 4 is targeted to discuss the simulations setup and performed experiments. Conclusions are drawn in Sec. 5.

2. Speech enhancement front-end

Let M be the number of independent speech sources and N the number of microphones. The relationship between them is described by an $M \times N$ MIMO FIR (finite impulse response) system. According to such a model, the n-th microphone signal at k-th sample time is:

$$x_n(k) = \sum_{m=1}^{M} \mathbf{h}_{nm}^T \mathbf{s}_m(k, L_h), \qquad k = 1, 2, ..., K, \quad n = 1, 2, ..., N \tag{1}$$

where $(\cdot)^T$ denotes the transpose operator and

$$\mathbf{s}_m(k, L_h) = [s_m(k)\ s_m(k-1)\ \cdots\ s_m(k-L_h+1)]^T. \tag{2}$$

is the m-th source. The term

$$\mathbf{h}_{nm} = [h_{nm,0}\ h_{nm,1}\ \cdots\ h_{nm,L_h-1}]^T, \quad n = 1, 2, ..., N, \quad m = 1, 2, ..., M \tag{3}$$

is the L_h-taps RIR between the n-th microphone and the m-th source. Applying the z transform, Eq. 1 can be rewritten as:

$$X_n(z) = \sum_{m=1}^{M} H_{nm}(z) S_m(z), \qquad n = 1, 2, ..., N \tag{4}$$

where

$$H_{nm}(z) = \sum_{l=0}^{L_h-1} h_{nm,l} z^{-1}. \tag{5}$$

The objective is recovering the original clean speech sources s_m by means of a speech dereverberation approach: Indeed, it is necessary to automatically identify who is speaking, accordingly estimating the unknown RIRs and then apply a seperation and dereverberation process to restore the original speech quality.

The reference framework proposed in (Huang et al. (2005); Rotili et al. (2010)) consists of three main stages: source separation, speech dereverberation and BCI. Firstly source separation is accomplished by transforming the original MIMO system in a certain number of SIMO systems and secondly the separated sources (but still reverberated) pass through the dereverberation process yielding the final cleaned-up speech signals. In order to make the two procedures properly working, it is necessary to estimate the MIMO RIRs of the audio

channels between the speech sources and the microphones by the usage of the BCI stage. As mentioned in the introductory section, this approach suffers from the BCI stage inability of estimating the RIRs without the knowledge of the speakers' activities. To overcome this disadvantage a speaker diarization system can be introduced to steer the BCI stage. The block diagram of the proposed framework is shown in Fig. 1 where $N = 3$ and $M = 2$ have been considered. Speaker Diarization takes as input the central microphone mixture and for each

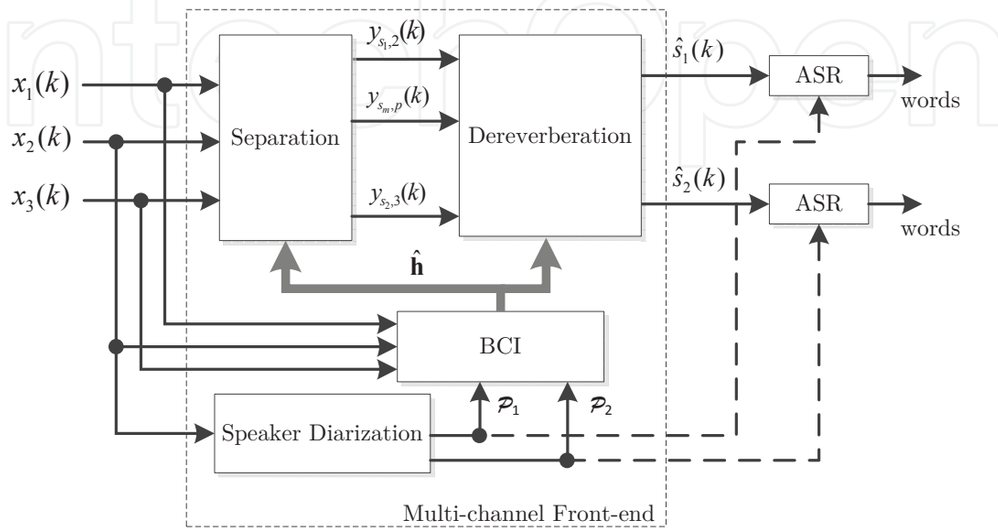


Fig. 1. Block diagram of the proposed framework.

frame, the output $\mathcal{P}_m$ is "1" if the m-th source is the only active, and "0" otherwise. In such a way, the front-end is able to detect when to perform or not to perform the required operation. Using the information carried out by the Speaker Diarization stage, the BCI will estimate the RIRs and the speech recognition engine will perform recognition if the corresponding source is the only active.

2.1 Blind channel identification

Considering a SIMO system for a specific source s_{m^*}, a BCI algorithm aims to find the RIRs vector $\mathbf{h}_{nm^*} = [\mathbf{h}_{1m^*}^T \ \mathbf{h}_{2m^*}^T \ \cdots \mathbf{h}_{Nm^*}^T \ ]^T$ by using only the microphone signals $x_n(k)$. In order to ensure this, two identifiability condition are assumed satisfied (Xu et al. (1995)):

1. The polynomial formed from $\mathbf{h}_{nm^*}$ are co-prime, i.e. the room transfer functions (RTFs) $H_{nm^*}(z)$ do not share any common zeros (channel diversity);
2. $\mathcal{C}\{s(k)\} \geq 2L_h + 1$, where $\mathcal{C}\{s(k)\}$ denotes the linear complexity of the sequence $s(k)$.

This stage performs the BCI through the unconstrained normalized multi-channel frequency-domain least mean square (UNMCFLMS) algorithm (Huang & Benesty (2003)). It is an adaptive technique well suited to satisfy the real-time constraints imposed by the case study since it offers a good compromise among fast convergence, adaptivity, and low computational complexity.

Here, we briefly review the UNMCFLMS in order to understand the motivation of its choice in the proposed front-end. Refer to (Huang & Benesty (2003)) for details. The derivation

of UNMCFLMS is based on cross relation criteria (Xu et al. (1995)) using the overlap-save technique (Oppenheim et al. (1999)).

The frequency-domain cost function for the q-th frame is defined as

$$J_f = \sum_{n=1}^{N-1} \sum_{i=i+1}^{N} \mathbf{e}_{ni}^H(q)\mathbf{e}_{ni}(q) \tag{6}$$

where $\mathbf{e}_{ni}(q)$ is the frequency-domain block error signal between the n-th and i-th channels and $(\cdot)^H$ denotes the Hermitian transpose operator. The update equation of the UNMCFLMS is expressed as

$$\begin{aligned} \widehat{\mathbf{h}}_{nm^*}(q+1) = \widehat{\mathbf{h}}_{nm^*}(q) - \rho[\mathbf{P}_{nm^*}(q) + \delta\mathbf{I}_{2L_h\times L_h}]^{-1} \\ \times \sum_{n=1}^{N} \mathbf{D}_{x_n}^H(q)\mathbf{e}_{ni}(q), \quad i = 1,\ldots,N \end{aligned} \tag{7}$$

where $0 < \rho < 2$ is the step-size, δ is a small positive number and

$$\begin{aligned} \widehat{\mathbf{h}}_{nm^*}(q) &= \mathbf{F}_{2L_h\times 2L_h}\left[\widehat{\mathbf{h}}_{nm^*}(q)\ \mathbf{0}_{1\times L_h}\right]^T, \\ \mathbf{e}_{ni}(q) &= \mathbf{F}_{2L_h\times 2L_h}\left[\mathbf{0}_{1\times L_h}\ \left\{\mathbf{F}_{L_h\times L_h}^{-1}\mathbf{e}_{ni}(q)\right\}^T\right]^T, \\ \mathbf{P}_{nm^*}(q) &= \sum_{n=1,n\neq i}^{N} \mathbf{D}_{x_n}^H(q)\mathbf{D}_{x_n}(q) \end{aligned} \tag{8}$$

while $\mathbf{F}$ denotes the discrete Fourier transform (DFT) matrix. The frequency-domain error function $\mathbf{e}_{ni}(q)$ is given by

$$\mathbf{e}_{ni}(q) = \mathbf{D}_{x_n}(q)\widehat{\mathbf{h}}_{nm^*}(q) - \mathbf{D}_{x_i}(q)\widehat{\mathbf{h}}_{im^*}(q) \tag{9}$$

where the diagonal matrix

$$\mathbf{D}_{x_n}(q) = \mathrm{diag}\left(\mathbf{F}\left\{[x_n(qL_h - L_h)\ x_n(qL_h - L_h + 1)\cdots x_n(qL_h + L_h - 1)]^T\right\}\right) \tag{10}$$

is the DFT of the q-th frame input signal block for the n-th channel. From a computational point of view, the UNMCFLMS algorithm ensures an efficient execution of the circular convolution by means of the fast Fourier transform (FFT). In addition, it can be easily implemented in a real-time application since the normalization matrix $\mathbf{P}_{nm^*}(q) + \delta\mathbf{I}_{2L_h\times L_h}$ is diagonal, and it is straightforward to compute its inverse.

Though UNMCFLMS allows the estimation of long RIRs, it requires a high input signal-to-noise ratio. In this paper, the presence of noise has not been taken into account and therefore the UNMCFLMS still remain an appropriate choice. Different solutions have been proposed in literature in order to alleviate the misconvergence problem of the UNMCFLMS in presence of noise. Among them, the algorithms presented in (Haque et al. (2007); Haque & Hasan (2008); Yu & Er (2004)) guarantee a significant robustness against noise and they could be used to improve our front-end.

2.2 Source separation

Here we briefly review the procedure already described in (Huang et al. (2005)) according to which it is possible to transform an $M \times N$ MIMO system (with $M < N$) in M $1 \times N$ SIMO systems free of interferences, as described by the following relation:

$$Y_{s_m,p}(z) = F_{s_m,p}(z)S_m(z) + B_{s_m,p}(z), \quad m = 1,2,\ldots,M, \quad p = 1,2,\ldots,P \tag{11}$$

where $P = C_N^M$ is the number of combinations. It must be noted that the SIMO systems outputs are reverberated, likely more than the microphone signals due to the long impulse response of equivalent channels $F_{s_m,p}(z)$. Related formula and the detailed description of the algorithm can be found in (Huang et al. (2005)). Different choices can be made in order

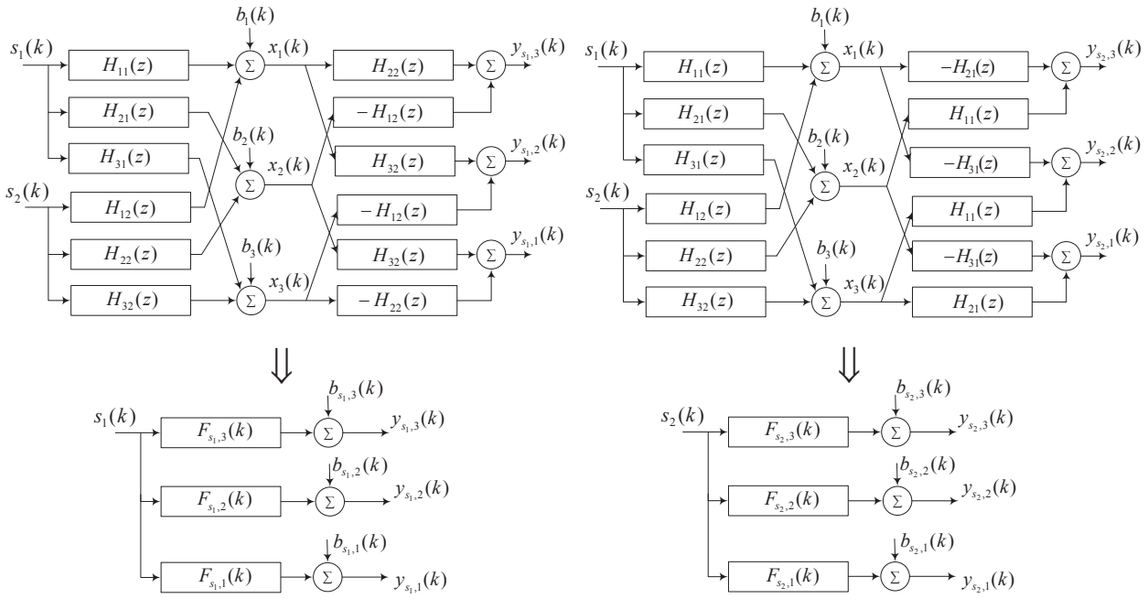


Fig. 2. Conversion of a 2×3 MIMO system in two 1×3 SIMO systems.

to calculate the equivalent SIMO system. In the block scheme of Fig. 2, representing the MIMO-SIMO conversion, is depicted a possible solution when $M = 2$ and $N = 3$. With this choice the first SIMO systems corresponding to the source s_1 is

$$\begin{aligned} F_{s_1,1}(z) &= H_{32}(z)H_{21}(z) - H_{22}(z)H_{31}(z), \\ F_{s_1,2}(z) &= H_{32}(z)H_{11}(z) - H_{12}(z)H_{31}(z), \\ F_{s_1,3}(z) &= H_{22}(z)H_{11}(z) - H_{12}(z)H_{21}(z). \end{aligned} \tag{12}$$

The second SIMO system corresponding to the source s_2 can be found in a similar way, thus results, $F_{s_1,p}(z) = F_{s_2,p}(z)$ with $p = 1,2,3$. As stated in the previous section the presence of additive noise is not taken into account in this contribution and than all the terms $B_{s_m,p}(z)$ of Eq. 11 are equal to zero. Finally it is important to highlight that in using this separation algorithm a lower computation complexity w.r.t. traditional independent component analysis technique is achieved and since the MIMO system is decomposed into a number of SIMO systems which are be blindly identified at different time the permutation ambiguity problem is avoided.

2.3 Speech dereverberation

Given the equivalent SIMO system $F_{s_{m^*},p}(z)$ related to the specific source s_{m^*}, a set of inverse filters $G_{s_{m^*},p}(z)$ can be found by using the MINT theorem such that

$$\sum_{p=1}^{P} F_{s_{m^*},p}(z) G_{s_{m^*},p}(z) = 1, \tag{13}$$

assuming that the polynomials $F_{s_{m^*},p}(z)$ have no common zeros. In the time-domain, the inverse filter vector denoted as $\mathbf{g}_{s_{m^*}}$, is calculated by minimizing the following cost function:

$$C = \left\| \mathbf{F}_{s_{m^*}} \mathbf{g}_{s_{m^*}} - \mathbf{v} \right\|^2, \tag{14}$$

where $\| \cdot \|$ denote the l_2-norm operator and

$$\mathbf{g}_{s_{m^*}} = \left[\mathbf{g}_{s_{m^*},1}^T \; \mathbf{g}_{s_{m^*},2}^T \; \cdots \; \mathbf{g}_{s_{m^*},P}^T \right]^T, \tag{15}$$

$$\mathbf{g}_{s_{m^*},p} = \left[g_{s_{m^*},p}(1) \; g_{s_{m^*},p}(2) \; \cdots \; g_{s_{m^*},p}(L_g) \right]^T, \tag{16}$$

$$\mathbf{v} = [\underbrace{0, \cdots, 0}_{d}, 1, \cdots, 0]^T, \tag{17}$$

with $p = 1, 2, \cdots, P$. The vector $\mathbf{v}$ is the target vector, i.e. the Kronecker delta shifted by an appropriate modeling delay ($0 \leq d \leq PL_g$) while $\mathbf{F}_{s_{m^*}} = \left[\mathbf{F}_{s_{m^*},1} \; \mathbf{F}_{s_{m^*},2} \; \cdots \; \mathbf{F}_{s_{m^*},P} \right]$ where $\mathbf{F}_{s_{m^*},p}$ is the convolution matrix of the equivalent FIR filter $\mathbf{f}_{s_{m^*},p} = \left[f_{s_{m^*},p}(1) \; f_{s_{m^*},p}(1) \; \cdots \; f_{s_{m^*},p}(L_f) \right]$ of length L_f. When the matrix $\mathbf{F}_{s_{m^*}}$ is obtained as shown in the previous section, the inverse filter set can be calculated as

$$\mathbf{g}_{s_{m^*}} = \mathbf{F}_{s_{m^*}}^{\dagger} \mathbf{v} \tag{18}$$

where $(\cdot)^{\dagger}$ denotes the Moore-Penrose pseudoinverse. In order to have a unique solution L_g must be chosen in such a way that $\mathbf{F}_{s_{m^*}}$ is square i.e.

$$L_g = \frac{L_f - 1}{P - 1}. \tag{19}$$

Considering the presence of disturbances, i.e. additive noise or RTFs fluctuations, the cost function Eq. 14 is modified as follows (Hikichi et al. (2007)):

$$C = \left\| \mathbf{F}_{s_{m^*}} \mathbf{g}_{s_{m^*}} - \mathbf{v} \right\|^2 + \gamma \left\| \mathbf{g}_{s_{m^*}} \right\|^2, \tag{20}$$

where the parameter $\gamma (\geq 0)$, called regularization parameter, is a scalar coefficient representing the weight assigned to the disturbance term. It should be noticed that Eq. 20 has the same form to that of Tikhonov regularization for ill-posed problems (Egger & Engl (2005)).

Let the RTF for the fluctuation case be given by the sum of two terms, the mean RTF ($\overline{\mathbf{F}}_{s_{m^*}}$) and the fluctuation from the mean RTF ($\widetilde{\mathbf{F}}_{s_{m^*}}$) and let $E\langle \widetilde{\mathbf{F}}_{s_{m^*}}^T \widetilde{\mathbf{F}}_{s_{m^*}} \rangle = \gamma \mathbf{I}$. In this case a general

cost function, embedding noise and fluctuation case, can be derived:

$$C = \mathbf{g}_{s_{m^*}}^T \mathcal{F}^T \mathcal{F} \mathbf{g}_{s_{m^*}} - \mathbf{g}_{s_{m^*}}^T \mathcal{F}^T \mathbf{v} - \mathbf{v}^T \mathcal{F} \mathbf{g}_{s_{m^*}} + \mathbf{v}^T \mathbf{v} + \gamma \mathbf{g}_{s_{m^*}}^T \mathbf{g}_{s_{m^*}} \tag{21}$$

where

$$\mathcal{F} = \begin{cases} \mathbf{F}_{s_{m^*}} & \text{(noise case)} \\ \overline{\mathbf{F}}_{s_{m^*}} & \text{(fluctuation case).} \end{cases} \tag{22}$$

The filter that minimizes the cost function in Eq. 21 is obtained by taking derivatives with respect to $\mathbf{g}_{s_{m^*}}$ and setting them equal to zero. The required solution is

$$\mathbf{g}_{s_{m^*}} = \left(\mathcal{F}^T \mathcal{F} + \gamma \mathbf{I} \right)^{-1} \mathcal{F}^T \mathbf{v}. \tag{23}$$

The usage of Eq. 23 to calculate the inverse filters requires a matrix inversion that, in the case of long RIRs, can result in a high computational burden. Instead, an adaptive algorithm (Rotili et al. (2008)) has been here adopted to satisfy the real-time constraint. It is based on the steepest-descent technique, whose recursive estimator has the form

$$\mathbf{g}_{s_{m^*}}(q+1) = \mathbf{g}_{s_{m^*}}(q) - \frac{\mu(q)}{2} \nabla C. \tag{24}$$

Moving from Eq. 21 through simple algebraic calculations, the following expression is obtained:

$$\nabla C = -2[\mathcal{F}^T(\mathbf{v} - \mathcal{F}\mathbf{g}_{s_{m^*}}(q)) - \gamma \mathbf{g}_{s_{m^*}}(q)]. \tag{25}$$

Substituting Eq. 25 into Eq. 24 is

$$\mathbf{g}_{s_{m^*}}(q+1) = \mathbf{g}_{s_{m^*}}(q) + \mu(q)[\mathcal{F}^T(\mathbf{v} - \mathcal{F}\mathbf{g}_{s_{m^*}}(q)) - \gamma \mathbf{g}_{s_{m^*}}(q)], \tag{26}$$

where $\mu(q)$ is the step-size. The convergence of the algorithm to the optimal solution is guaranteed if the usual conditions for the step-size in terms of autocorrelation matrix $\mathcal{F}^T\mathcal{F}$ eigenvalues hold. However, the achievement of the optimum can be slow if a fixed step-size value is chosen. The algorithm convergence speed can be increased following the approach in (Guillaume et al. (2005)), where the step-size is chosen in order to minimize the cost function at the next iteration. The analytical expression obtained for the step-size is the following:

$$\mu(q) = \frac{\mathbf{e}^T(q)\mathbf{e}(q)}{\mathbf{e}^T(q)\left(\mathcal{F}^T\mathcal{F} + \gamma I\right)\mathbf{e}(q)} \tag{27}$$

where

$$\mathbf{e}(q) = \mathcal{F}^T\left[\mathbf{v} - \mathcal{F}\mathbf{g}_{s_{m^*}}(q)\right] - \gamma \mathbf{g}_{s_{m^*}}(q).$$

In using the previously illustrated algorithm, different advantages are obtained: The regularization parameter which takes into account the presence of disturbances, makes the dereverberation process more robust to estimation errors due to the BCI algorithm (Hikichi et al. (2007)); the real-time constraint can be met also in the case of long RIRs since no matrix inversion is required. Finally, the complexity of the algorithm has been decreased computing the required operation in the frequency-domain by using FFTs.

2.4 Speaker diarization

The speaker diarization stage drives the BCI and the ASRs so that they can operate into speaker-homogeneous regions. Current state-of-the-art speaker diarization systems are based on clustering approaches, usually combining hidden Markov models (HMMs) and the bayesian information criterion metric (Fredouille et al. (2009); Wooters & Huijbregts (2008)). Despite their state-of-art performance, such systems have the drawback of operating on the entire signals, making them unsuitable to work online as required by the proposed framework.

The approach taken here as reference has been proposed in (Vinyals & Friedland (2008)), and its block scheme for $M = 2$ and $N = 3$, is shown in Fig. 3. The algorithm operation is divided in two phases, training and recognition. In the first, the acquired signals, after a manual removal of silence periods, are transformed in feature vectors composed of 19 mel-frequency cepstral coefficients (MFCC) plus their first and second derivatives. Cepstral mean normalization is applied to deal with stationary channel effects. Speaker models are represented by mixture of Gaussians trained by means of the expectation maximization algorithm. The number of Gaussians and the end accuracy at convergence have been empirically determined, and set to 100 and 10^{-4} respectively. In this phase the voice activity detector (VAD) is also trained. The adopted VAD is based on bi-gaussian model of the log-energy frame. During the training a two gaussian model is estimated using the input sequence: The gaussian with the smallest mean will model the silence frames whereas the other gaussian corresponds to frames of speech activity.

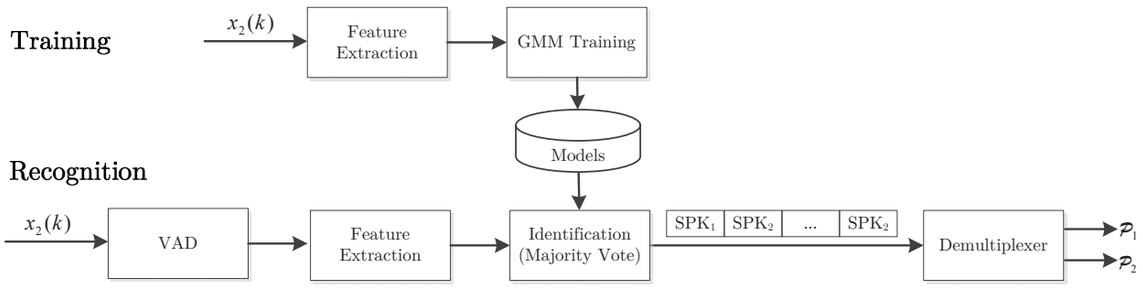


Fig. 3. The speaker diarization block scheme: "SPK_1" and "SPK_2" are the speaker identities labels assigned to each chunk.

In the recognition phase, the first operation consists in a voice activity detection in order to remove the silence periods: frames are tagged as silence or not based on the bi-gaussian model, using a maximum likelihood criterion.

After the voice activity detection, the signals are divided into non overlapping chunks, and the same feature extraction pipeline of the training phase extracts feature vectors. The decision is then taken using majority vote on the likelihoods: every feature vector in the current segment is assigned to one of the known speaker's model based on the maximum likelihood criterion. The model which has the majority of vectors assigned determines the speaker identity on the current segment. The Demultiplexer block associates each speaker label to a distinct output and sets it to "1" if the speaker is the only active, and "0" otherwise.

It is worth pointing out that the speaker diarization algorithm is not able to detect overlapped speech, and an oracle overlap detector is used to overcome this lack.

2.5 Speech enhancement front-end operation

The proposed front-end requires an initial training phase where each speaker is asked to talk for 60 s. During this period, the speaker diarization stage trains the both the VAD and speakers' models.

In the testing phase, the input signal is divided into non overlapping chunks of 2 s, the speaker diarization stage provides as output the speakers' activity $\mathcal{P}_m$. This information is employed both in the BCI stage and ASR engines: only when the m-th source is the only active the related RIRs are updated and the dereverberated speech recognized. In all the other situations the BCI stage provide as output the RIRs estimated at the previous step while the ASRs are idle.

The Separation stage takes as input the microphone signals and outputs the interference free signals that are subsequently processed by Dereverberation stage. Both stages perform theirs operations using the RIRs vector provided by the BCI stage.

The front-end performances are strictly related to the speaker diarization errors. In particular, the BCI stage is sensitive to false alarms (speaker in hypothesis but not in reference) and speaker errors (mapped reference is not the same as hypothesis speaker). If one of these occurs, the BCI performs the adaptation of the RIRs using an inappropriate input frame providing as output an incorrect estimation. An additional error which produces the previously highlighted behaviour is the miss speaker overlap detection.

The sensitivity to false alarms and speaker errors could be reduced imposing a constraint in the estimation procedure and updating the RIR only when a decrease in the cost function occurs. A solution to miss overlap error would be to add an overlap detector and not to perform the estimation if more than one speaker is simultaneously active. On the other hand, missed speaker errors (speaker in reference but not in hypothesis) does not negatively affect the RIRs estimation procedure, since the BCI stage does not perform the adaptation in such frames. Only a reduced convergence rate can be noticed in this case.

The real-time capabilities of the proposed front-end have been evaluated calculating the real-time factor on a Intel® Core™i7 machine running at 3 GHz with 4 GB of RAM. The obtained value for the speaker diarization stage is 0.03, meaning that a new result is output every 2.06 s. The real-time factor for the others stage is 0.04 resulting in a total value of 0.07 for the entire front-end.

3. ASR engine

Automatic speech recognition has been performed by means of the Hidden Markov Model Toolkit (HTK) (Young et al. (2006)) using HDecode, which has been specifically designed for large vocabulary speech recognition tasks. Features have been extracted through the HCopy tool, and are composed of 13 MFCC, deltas and double deltas, resulting in a 39 dimensional feature vector. Cepstral mean normalization is included in the feature extraction pipeline. Recognition has been performed based on the acoustic models available in (Vertanen (2006)).

The models differ with respect to the amount of training data, the use of word-internal or cross-word triphones, the number of tied states, the number of Gaussians per state, and the initialization strategy. The main focus of this work is to achieve real-time execution of the complete framework, thus an acoustic model able to obtain adequate accuracies and

real-time ability was required. The computational cost strongly depends on the number of Gaussians per state, and in (Vertanen (2006)) it has been shown that real-time execution can be obtained using 16 Gaussians per state. The main parameters of the selected acoustic model are summarized in Table 1.

Training data	WSJ0 & WSJ1
Initialization strategy	TIMIT bootstrap
Triphone model	cross-word
# of tied states (approx.)	8000
# of Gaussians per state	16
# of silence Gaussians	32

Table 1. Characteristics of the selected acoustic model.

The language model consists of the 5k words bi-gram model included in the Wall Street Journal (WSJ) corpus. Recognizer parameters are the same as in (Vertanen (2006)): using such values, the word accuracy obtained on the November '92 test set is 94.30% with a real-time factor of 0.33 on the same hardware platform mentioned above. It is worth pointing out that the ASR engine and the front-end can jointly operate in real-time.

4. Experiments

4.1 Corpus description

The acoustic scenario under study is made of an array of three microphones and two speech sources located in a small office. The room arrangement is depicted in Fig. 4. The data set

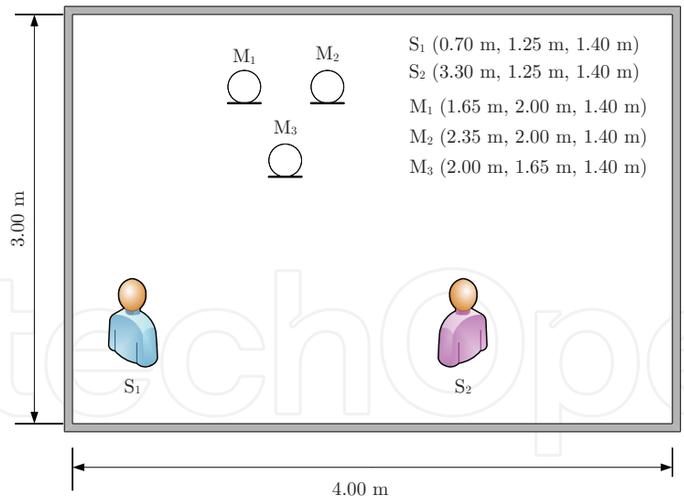


Fig. 4. Room setup.

used for the speech recognition experiments has been constructed from the WSJ November '92 speech recognition evaluation set. It consists of 330 sentences (about 40 minutes of speech), uttered by eight different speakers, both male and female. The data set is recorded at 16 kHz and does not contain any additive noise or reverberation.

A suitable database representing the described scenario has been artificially created using the following procedure: The 330 clean sentences are firstly reduced to 320 in order to have the

same number of sentences for each speaker. These are then convolved with RIRs generated using the RIR Generator tool (Habets (2008)). No background noise has been added. Two different reverberation conditions have been taken into account: the low and the and high reverberant ones, corresponding to $T_{60} = 120$ ms and $T_{60} = 240$ ms respectively (with RIRs 1024 taps long).

For each channel, the final overlapped and reverberated sentences have been obtained by coupling the sentences of two speakers. Following the WSJ November '92 notation, speaker 440 has been paired with 441, 442 with 443, etc. This choice makes possible to cover all the combinations of male and female speakers, resulting in 40 sentences per couple of speakers. The mean value of overlap has been fixed to 15% of the speech frames for the overall dataset. For each sentence the amount of overlap is obtained as a random value drown from the uniform distribution on the interval [12, 18]. This assumption allows the artificial database to reflect the frequency of overlapped speech in real-life scenarios such as two-party telephone conversation or meeting (Shriberg et al. (2000)).

4.2 Front-end evaluation

As stated in Sec. 2 the proposed speech enhancement front-end consists in four different stages. Here we focus the attention on the evaluation of the Speaker Diarization and BCI stages which represent the most crucial parts of the entire system. An extensive evaluation of the Separation and Dereverberation stages can be found in (Huang et al. (2005)) and (Rotili et al. (2008)) respectively.

The performance of the speaker diarization algorithms are measured by the diarization error rate[1] (DER). DER is defined by the following expression:

$$\mathrm{DER} = \frac{\sum_{s=1}^{S} \mathrm{dur}(s)(\max(N_{\mathrm{ref}}(s), N_{\mathrm{hyp}}(s)) - N_{\mathrm{correct}}(s))}{\sum_{s=1}^{S} \mathrm{dur}(s) N_{\mathrm{ref}}(s)} \tag{28}$$

where dur is the duration of the segment, S is the total number of segments in which no speaker change occurs, $N_{\mathrm{ref}}(s)$ and $N_{\mathrm{hyp}}(s)$ indicate respectively the number of speakers in the reference and in the hypothesis, and $N_{\mathrm{correct}}(s)$ indicates the number of speakers that speak in the segment s and have been correctly matched between the reference and the hypothesis. As recommended by the National Institute for Standards and Technology (NIST), evaluation has been performed by means of the "md-eval" tool with a collar of 0.25 s around each segment to take into account timing errors in the reference. The same metric and tool are used to evaluate the VAD performance[2].

Performance for the sole VAD are reported in table Table 2. Table 3 shows the results obtained testing the speaker diarization algorithm on the clean signals, as well as on the two reverberated scenarios in the previous illustrated configurations. For the seek of comparison two different configurations have been considered:

- REAL SD w/ ORACAL-VAD: The speaker diarization system uses an "Oracle" VAD;

[1] http://www.itl.nist.gov/iad/mig/tests/rt/2004-fall/

[2] Details can be found in "*Spring 2005 (RT-05S) Rich Transcription Meeting Recognition Evaluation Plan*". The "md-eval" tool is available at `http://www.itl.nist.gov/iad/mig//tools/`

- REAL SD w/ REAL-VAD: The system described in Sec. 2.4.

The performance across the three scenarios are similar due to the matching of the training and testing conditions, and are consistent with (Vinyals & Friedland (2008)).

	Clean	$T_{60} = 120\,\text{ms}$	$T_{60} = 240\,\text{ms}$
REAL-VAD	1.85	1.96	1.68

Table 2. VAD error rate (%).

	Clean	$T_{60} = 120\,\text{ms}$	$T_{60} = 240\,\text{ms}$
REAL-SD w/ ORACLE-VAD	13.57	13.30	13.24
REAL-SD w/ REAL-VAD	15.20	15.20	14.73

Table 3. Speaker diarization error rate (%).

The BCI stage performance are evaluated by means of a channel-based measure called Normalized Projection Misalignment (NPM) (Morgan et al. (1998)) defined as

$$\text{NPM}\,(q) = 20\log_{10}\left(\frac{\|\epsilon(q)\|}{\|\mathbf{h}\|}\right), \tag{29}$$

where

$$\epsilon(q) = \mathbf{h} - \frac{\mathbf{h}^T\widehat{\mathbf{h}}(q)}{\widehat{\mathbf{h}}^T(q)\widehat{\mathbf{h}}(q)}\widehat{\mathbf{h}}(q) \tag{30}$$

is the projection misalignment vector, $\mathbf{h}$ is the real RIR vector whereas $\widehat{\mathbf{h}}(q)$ is the estimated one at the q-th iteration, i.e. the frame index.

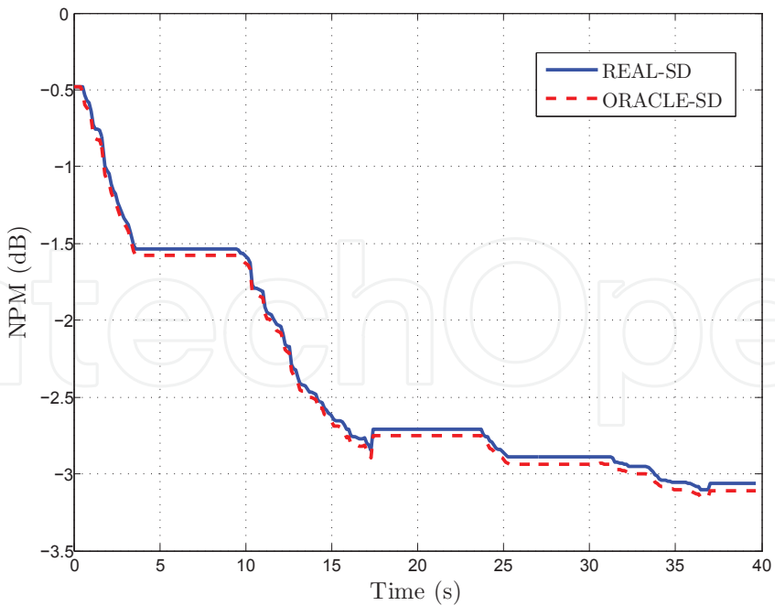


Fig. 5. NPM curves for the "Real" and "Oracle" speaker diarization system.

Fig. 5 shows the NPM curve for the identification of the RIRs relative to source s_1 at $T_{60} = 240\,\text{ms}$ for an input signal of 40 s. In order to understand how the performance of

the Speaker Diarization stage affect the RIRs identification we compare the curves obtained for ORACLE-SD where the speaker diariazion operates in an "Oracle" fashion, i.e. it operates at 100% of its possibilities, and REAL-SD case. As expected the REAL-SD NPM is always above the ORACLE-SD NPM. Parts where the curves are flat indicate speech segment in which source s_1 is the not only active source i.e. it is overlapped to s_2 or we have silence.

4.3 Full system evaluation

In this section the objective is to evaluate the recognition capabilities of the ASR engine fed by speech signals coming from the multichannel DSP front-end, therefore the performance metric employed is the word recognition accuracy.

The word recognition accuracy obtained assuming ideal source separation and dereverberation is 93.60%. This situation will be denoted as "Reference" in the remainder of the section.

Four different setups have been addressed:

- Unprocessed: The recognition is performed on the reverberant speech mixture acquired from Mic_2 (see Fig. 4);
- ASR w/o SD: The ASRs do not exploit the speaker diarization output;
- ASR w/ ORACLE-SD: The ASRs exploit the "Oracle" speaker diarization output;
- ASR w/ REAL-SD: The ASRs exploit the "Real" speaker diarization output.

Fig. 6 reports the word accuracy for both the low and high reverberant conditions when the complete test file is processed by the multi-channel DSP front-end and recognition is performed on the separated and dereverberated streams (*Overall*) for all the three setup. Fig. 7 shows the word accuracy values attained where the recognition is performed starting from the first silence frame after the BCI and Dereverberation stages converge[3] (*Convergence*).

Observing the results of Fig. 6, it can be immediately stated that feeding the ASR engine with unprocessed audio files leads to very poor performances. The missing source separation and the related wrong matching between the speaker and the corresponding word transcriptions result in a significant amount of insertions which justify the occurrence of negative word accuracy values.

Conversely, when the audio streams are processed, the ASRs are able to recognize most of the spoken words, specially once the front-end algorithms have reached the convergence. The usage of speaker diarization information to drive the ASRs activity significantly increases the performance. As expected the usage of the "Real" speaker diarization instead of an "Oracle" one lead to a decrease in performance of about 15% for the low reverberant condition and of a 10% for the high reverberant condition. Despite this, the word accuracy is still higher then the one obtained without speaker diarization, providing an average increase of about 20% for both the reverberation time.

In the *Convergence* evaluation case study, when $T_{60} = 120\,\text{ms}$ and the "Oracle" speaker diarization is employed, a word accuracy of 86.49% is obtained, which is about 7% less than the result attainable in the "Reference" conditions. In this case, the usage of the "Real"

[3] Additional experiments have demonstrated that this is reached after $20 - 25$ s of speech activity.

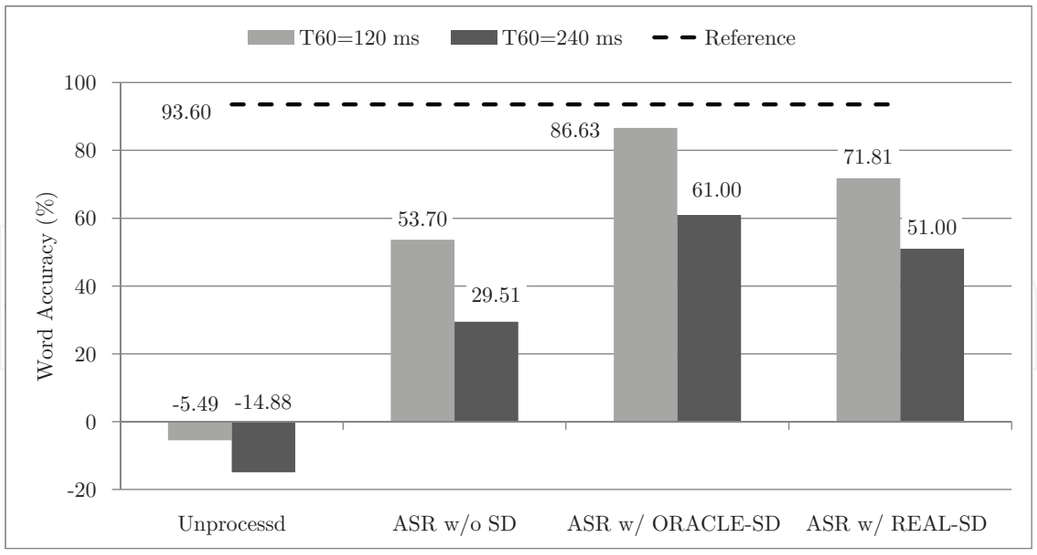


Fig. 6. Word accuracy for the *Overall* case.

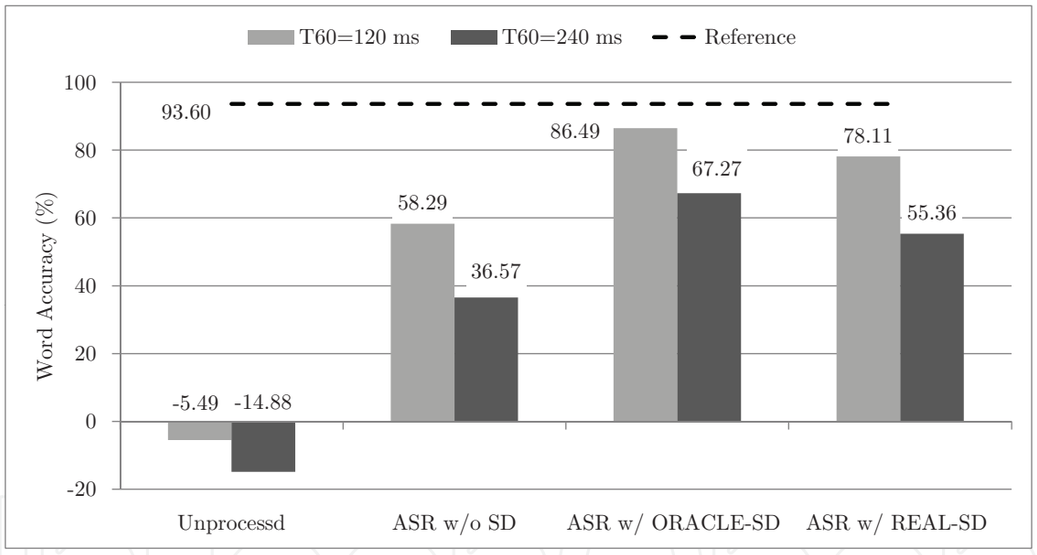


Fig. 7. Word accuracy for the *Convergence* case.

speaker diarization lead to decrease of only 8%. As expected, the reverberation effect has a negative impact on the recognition performances especially in presence of high reverberation, i.e. $T_{60} = 240\,\text{ms}$. However, it must be observed that the convergence margin is even more significant w.r.t. the low reverberant scenario, further highlighting the effectiveness of the proposed algorithmic framework as multichannel front-end.

5. Conclusion

In this paper, an ASR system was successfully enhanced by an advanced multi-channel front-end to recognize the speech content coming from multiple speakers in reverberated acoustic conditions. The overall architecture is able to blindly identify the impulse responses,

to separate the existing multiple overlapping sources, to dereverberate them and to recognize the information contained within the original utterances. A speaker diarization system able to steer the BCI stage and the ASRs has been also included in the overall framework. All the algorithms work in real-time and a PC-based implementation of them has been discussed in this contribution. Performed simulations, based on a existing large vocabulary database (WSJ) and suitably addressing the acoustic scenario under test, have shown the effectiveness of the developed system, making it appealing in real-life human-machine interaction scenarios. As future works, an overlap detector will be integrated in the speaker diarization system and its impact in terms of final recognition accuracy will be evaluated. In addition other applications different form ASR such as emotion recognition (Schuller et al. (2011)), dominance detection (Hung et al. (2011)) or keyword spotting (Wöllmer et al. (2011)) will be considered in order to assess the effectiveness of the front-end in other recognition tasks.

6. References

Egger, H. & Engl, H. (2005). Tikhonov regularization applied to the inverse problem of option pricing: convergence analysis and rates, *Inverse Problems* 21(3): 1027–1045.

Fredouille, C., Bozonnet, S. & Evans, N. (2009). The LIA-EURECOM RT'09 Speaker Diarization System, *RT'09, NIST Rich Transcription Workshop*, Melbourne, Florida, USA.

Guillaume, M., Grenier, Y. & Richard, G. (2005). Iterative algorithms for multichannel equalization in sound reproduction systems, *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 3, pp. iii/269–iii/272.

Habets, E. (2008). Room impulse response (RIR) generator.
URL: *http://home.tiscali.nl/ehabets/rirgenerator.html*

Haque, M., Bashar, M. S., Naylor, P., Hirose, K. & Hasan, M. K. (2007). Energy constrained frequency-domain normalized LMS algorithm for blind channel identification, *Signal, Image and Video Processing* 1(3): 203–213.

Haque, M. & Hasan, M. K. (2008). Noise robust multichannel frequency-domain LMS algorithms for blind channel identification, *IEEE Signal Processing Letters* 15: 305–308.

Hikichi, T., Delcroix, M. & Miyoshi, M. (2007). Inverse filtering for speech dereverberation less sensitive to noise and room transfer function fluctuations, *EURASIP Journal on Advances in Signal Processing* 2007(1).

Huang, Y. & Benesty, J. (2003). A class of frequency-domain adaptive approaches to blind multichannel identification, *IEEE Transactions on Speech and Audio Processing* 51(1): 11–24.

Huang, Y., Benesty, J. & Chen, J. (2005). A Blind Channel Identification-Based Two-Stage Approach to Separation and Dereverberation of Speech Signals in a Reverberant Environment, *IEEE Transactions on Speech and Audio Processing* 13(5): 882–895.

Hung, H., Huang, Y., Friedland, G. & Gatica-Perez, D. (2011). Estimating dominance in multi-party meetings using speaker diarization, *IEEE Transactions on Audio, Speech, and Language Processing* 19(4): 847–860.

Miyoshi, M. & Kaneda, Y. (1988). Inverse filtering of room acoustics, *IEEE Transactions on Signal Processing* 36(2): 145–152.

Morgan, D., Benesty, J. & Sondhi, M. (1998). On the evaluation of estimated impulse responses, *IEEE Signal Processing Letters* 5(7): 174–176.

Naylor, P. & Gaubitch, N. (2010). *Speech Dereverberation*, Signals and Communication Technology, Springer.

Oppenheim, A. V., Schafer, R. W. & Buck, J. R. (1999). *Discrete-Time Signal Processing*, 2 edn, Prentice Hall, Upper Saddle River, NJ.

Principi, E., Cifani, S., Rocchi, C., Squartini, S. & Piazza, F. (2009). Keyword spotting based system for conversation fostering in tabletop scenarios: Preliminary evaluation, *Proc. of 2nd Conference on Human System Interactions*, pp. 216–219.

Renals, S. (2005). AMI: Augmented Multiparty Interaction, *Proc. NIST Meeting Transcription Workshop*.

Rocchi, C., Principi, E., Cifani, S., Rotili, R., Squartini, S. & Piazza, F. (2009). A real-time speech-interfaced system for group conversation modeling, *19th Italian Workshop on Neural Networks*, pp. 70–80.

Rotili, R., Cifani, S., Principi, E., Squartini, S. & Piazza, F. (2008). A robust iterative inverse filtering approach for speech dereverberation in presence of disturbances, *Proceedings of IEEE Asia Pacific Conference on Circuits and Systems*, pp. 434–437.

Rotili, R., De Simone, C., Perelli, A., Cifani, A. & Squartini, S. (2010). Joint multichannel blind speech separation and dereverberation: A real-time algorithmic implementation, *Proceedings of 6th International Conference on Intelligent Computing*, pp. 85–93.

Schuller, B., Batliner, A., Steidl, S. & Seppi, D. (2011). Recognising realistic emotions and affect in speech: state of the art and lessons learnt from the first challenge, *Speech Communication* .

Shriberg, E., Stolcke, A. & Baron, D. (2000). Observations on Overlap : Findings and Implications for Automatic Processing of Multi-Party Conversation, *Word Journal Of The International Linguistic Association* pp. 1–4.

Squartini, S., Ciavattini, E., Lattanzi, A., Zallocco, D., Bettarelli, F. & Piazza, F. (2005). NU-Tech: implementing DSP algorithms in a plug-in based software platform for real time audio applications, *Proceedings of 118th Convention of the Audio Engineering Society*.

Tur, G., Stolcke, A., Voss, L., Peters, S., Hakkani-Tur, D., Dowding, J., Favre, B., Fernandez, R., Frampton, M., Frandsen, M., Frederickson, C., Graciarena, M., Kintzing, D., Leveque, K., Mason, S., Niekrasz, J., Purver, M., Riedhammer, K., Shriberg, E., Tien, J., Vergyri, D. & Yang, F. (2010). The CALO meeting assistant system, *IEEE Trans. on Audio, Speech, and Lang. Process.*, 18(6): 1601 –1611.

Vertanen, K. (2006). Baseline WSJ acoustic models for HTK and Sphinx: Training recipes and recognition experiments, *Technical report*, Cavendish Laboratory, University of Cambridge.
URL: *http://www.keithv.com/software/htk/us/*

Vinyals, O. & Friedland, G. (2008). Towards semantic analysis of conversations: A system for the live identification of speakers in meetings, *Proceedings of IEEE International Conference on Semantic Computing*, pp. 426 –431.

Waibel, A., Steusloff, H., Stiefelhagen, R. & the CHIL Project Consortium (2004). CHIL: Computers in the Human Interaction Loop, *International Workshop on Image Analysis for Multimedia Interactive Services*.

Woelfel, M. & McDonough, J. (2009). *Distant Speech Recognition*, 1st edn, Wiley, New York.

Wöllmer, M., Marchi, E., Squartini, S. & Schuller, B. (2011). Multi-stream lstm-hmm decoding and histogram equalization for noise robust keyword spotting, *Cognitive Neurodynamics* 5: 253–264.

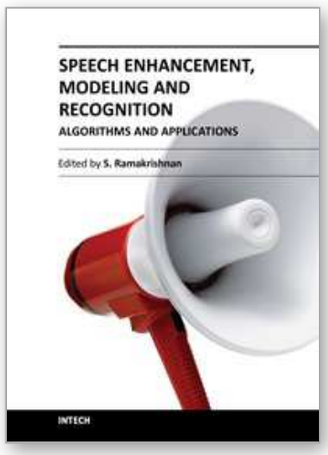
Wooters, C. & Huijbregts, M. (2008). The ICSI RT07s Speaker Diarization System, *in* R. Stiefelhagen, R. Bowers & J. Fiscus (eds), *Multimodal Technologies for Perception of Humans, Lecture Notes in Computer Science*, Springer-Verlag, Berlin, Heidelberg, pp. 509–519.

Xu, G., Liu, H., Tong, L. & Kailath, T. (1995). A Least-Squares Approach to Blind Channel Identification, *IEEE Transactions On Signal Processing* 43(12): 2982–2993.

Young, S., Everman, G., Kershaw, D., Moore, G. & Odell, J. (2006). *The HTK Book*, Cambridge University Engineering.

Yu, Z. & Er, M. (2004). A robust adaptive blind multichannel identification algorithm for acoustic applications, *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 2, pp. ii/25–ii/28.

IntechOpen



Speech Enhancement, Modeling and Recognition- Algorithms and Applications

Edited by Dr. S Ramakrishnan

ISBN 978-953-51-0291-5
Hard cover, 138 pages
Publisher InTech
Published online 14, March, 2012
Published in print edition March, 2012

This book on Speech Processing consists of seven chapters written by eminent researchers from Italy, Canada, India, Tunisia, Finland and The Netherlands. The chapters covers important fields in speech processing such as speech enhancement, noise cancellation, multi resolution spectral analysis, voice conversion, speech recognition and emotion recognition from speech. The chapters contain both survey and original research materials in addition to applications. This book will be useful to graduate students, researchers and practicing engineers working in speech processing.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Rudy Rotili, Emanuele Principi, Stefano Squartini and Francesco Piazza (2012). A Real-Time Speech Enhancement Front-End for Multi-Talker Reverberated Scenarios, Speech Enhancement, Modeling and Recognition- Algorithms and Applications, Dr. S Ramakrishnan (Ed.), ISBN: 978-953-51-0291-5, InTech, Available from: http://www.intechopen.com/books/speech-enhancement-modeling-and-recognition-algorithms-and-applications/a-real-time-speech-enhancement-front-end-for-multi-talker-reverberated-scenarios

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2012 The Author(s). Licensee IntechOpen. This is an open access article distributed under the terms of the Creative Commons Attribution 3.0 License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

IntechOpen

IntechOpen