Conversational Speech Recognition in Non-stationary Reverberated Environments

Rudy Rotili[1], Emanuele Principi[1], Martin Wöllmer[2], Stefano Squartini[1], and Björn Schuller[2]

[1] Dipartimento di Ingegneria dell'Informazione
Università Politecnica delle Marche, Ancona, Italy
{r.rotili,e.principi,s.squartini}@univpm.it
[2] Institute for Human-Machine Communication
Technische Universität München, Germany
{woellmer,schuller}@tum.de

Abstract. This paper presents a conversational speech recognition system able to operate in non-stationary reverberated environments. The system is composed of a dereverberation front-end exploiting multiple distant microphones, and a speech recognition engine. The dereverberation front-end identifies a room impulse response by means of a blind channel identification stage based on the Unconstrained Normalized Multi-Channel Frequency Domain Least Mean Square algorithm. The dereverberation stage is based on the adaptive inverse filter theory and uses the identified responses to obtain a set of inverse filters which are then exploited to estimate the clean speech. The speech recognizer is based on tied-state cross-word triphone models and decodes features computed from the dereverberated speech signal. Experiments conducted on the Buckeye corpus of conversational speech report a relative word accuracy improvement of 17.48% in the stationary case and of 11.16% in the non-stationary one.

1 Introduction

In the recent years, several research efforts have been devoted to distant speech recognition (DSR) systems [14]. The motivation behind this is that DSR systems are perceived as more user-friendly, comfortable and intuitive than solutions using head-set microphones. The task still represents a great research challenge, as the acquired speech signal is more affected by distortions, such as noise and reverberation. In addition, if multiple speakers are present (e.g. in meetings), the presence of overlapping speech makes the task even more challenging.

In this paper, the focus is on DSR in reverberated environments, thus other causes of degradation will not be considered. According to [13], dereverberation techniques can be classified depending on the component of the DSR in which they operate. Signal-based approaches, in particular, dereverberate the microphone signals before the feature extraction stage. Here the attention is focused on these techniques, more specifically on the inverse filtering methods [6].

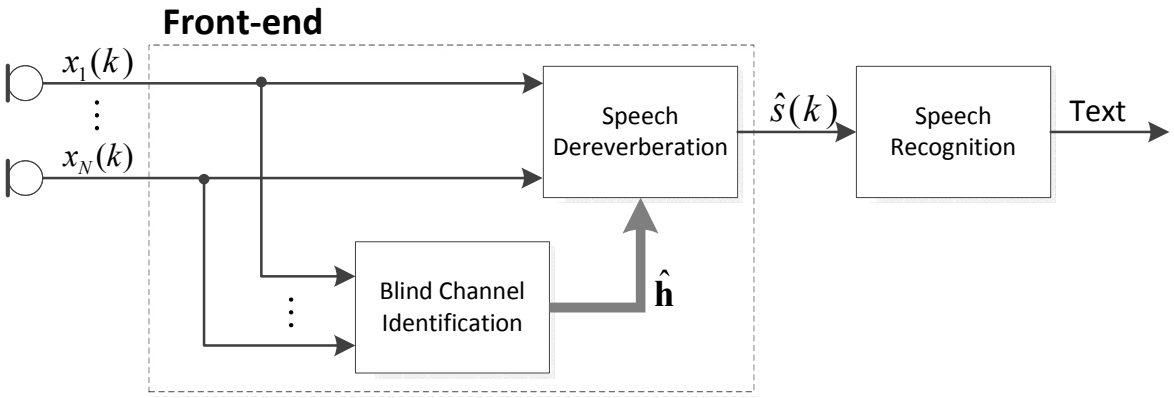

Fig. 1. System architecture

Assuming that the room impulse responses (RIRs) are available or estimated, inverse filtering methods aim at calculating a set of inverse filters for the RIRs and using them to dereverberate the microphone signals. Other signal-based approaches proposed in the literature cope with the reverberation problem by using beamforming techniques [14], spectral enhancement [6], non-negative matrix factorization [4], or linear-prediction residual enhancement [6].

This work proposes a system able to recognize conversational speech in a non-stationary reverberated acoustic environment (Fig. 1). The system is composed of a dereverberation front-end and a speech recognition engine which is based on the Hidden Markov Model toolkit (HTK) [17] and has been used as a baseline system in [15]. The front-end operates before the feature extraction stage and is based on the dereverberation algorithm proposed in [10] by some of the authors and on the identification algorithm proposed in [3]. It blindly identifies RIRs by means of the Unconstrained Normalized Multi-Channel Frequency Domain Least Mean Square (UNMCFLMS) algorithm, which are then equalized to recover the clean speech source. The recognizer processes cepstral mean normalized Mel-Frequency Cepstral Coefficient (MFCC) features and consists of context-dependent tied state cross-word triphone Hidden Markov Models (HMM) trained on conversational speech. A set of experiments have been conducted in stationary and non-stationary reverberated scenarios. The front-end capabilities of operating in non-stationary conditions have been assessed evaluating the *Normalized Projection Misalignment* (NPM) curves. The entire system has been evaluated in terms of word recognition accuracy on the artificially reverberated Buckeye corpus of conversational speech: The relative improvement over reverberated signals is 17.48% in the stationary case and 11.16% in the non-stationary one.

The outline of the paper is the following: Section 2 illustrates the blind dereverberation algorithm; Section 3 presents the conversational speech recognition system; Section 4 details the performed experiments and shows the obtained results; finally, Section 5 concludes the paper and presents some future developments.

2 Blind Dereverberation

2.1 Problem Statement

Let us consider a reverberant room with a single speech source and an array of N microphones, i.e. single-input multiple-output (SIMO) system. The observed signal at each sensor is then given by

$$x_n(k) = \mathbf{h}_n^T \mathbf{s}(k) \qquad n = 1, 2, \ldots, N \tag{1}$$

where $\mathbf{h}_n = [h_{n,0}\ h_{n,1} \cdots h_{n,L_h-1}]^T$ is the L_h-tap room impulse response between the source and n-th sensor, $\mathbf{s}(k) = [s(k)\ s(k-1) \cdots s(k-L_h+1)]^T$ is the input vector, and $(\cdot)^T$ denotes the transpose operator. Applying the z transform, equation (1) can be rewritten as:

$$X_n(z) = H_n(z)S(z), \qquad n = 1, 2, \ldots, N. \tag{2}$$

The objective is to obtain an estimate $\hat{s}(k)$ of the clean speech source by using only the microphone signals.

2.2 Blind Channel Identification

Considering the previously described SIMO system a Blind Channel Identification (BCI) algorithm aims to find the RIRs vector $\mathbf{h}_n$ by using only the microphone signals $x_n(k)$. Here, BCI is performed through the Unconstrained Normalized Multi-Channel Frequency-Domain Least Mean Square (UNMCFLMS) algorithm [3], an adaptive technique that offers a good compromise among fast convergence, adaptivity, and low computational complexity.

A brief review of UNMCFLMS now follows, please refer to [3] for details. The derivation of UNMCFLMS is based on cross relation criteria using the overlap and save technique. The frequency-domain cost function for the q-th frame is defined as

$$J_f = \sum_{n=1}^{N-1} \sum_{i=i+1}^{N} \underline{\mathbf{e}}_{ni}^H(q) \underline{\mathbf{e}}_{ni}(q) \tag{3}$$

where $\underline{\mathbf{e}}_{ni}(q)$ is the frequency-domain block error signal between the n-th and i-th channels and $(\cdot)^H$ denotes the Hermitian transpose operator. Defining $\mathbf{h}_{nm^*} = [\mathbf{h}_{1m^*}^T\ \mathbf{h}_{2m^*}^T\ \cdots \mathbf{h}_{Nm^*}^T\ ]^T$, the update equation of the UNMCFLMS is

$$\begin{aligned} \widehat{\underline{\mathbf{h}}}_{nm^*}(q+1) = & \widehat{\underline{\mathbf{h}}}_{nm^*}(q) - \rho[\mathbf{P}_{nm^*}(q) + \delta \mathbf{I}_{2L_h \times L_h}]^{-1} \\ & \times \sum_{n=1}^{N} \mathbf{D}_{x_n}^H(q) \underline{\mathbf{e}}_{ni}(q), \quad i = 1, 2, \ldots, N \end{aligned} \tag{4}$$

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

$$\widehat{\underline{\mathbf{h}}}_{nm^*}(q) = \mathbf{F}_{2L_h \times 2L_h} \left[\widehat{\mathbf{h}}_{nm^*}(q)\ \mathbf{0}_{1 \times L_h} \right]^T \tag{5}$$

$$\underline{\mathbf{e}}_{ni}(q) = \mathbf{F}_{2L_h \times 2L_h} \left[\mathbf{0}_{1\times L_h} \ \left\{ \mathbf{F}^{-1}_{L_h \times L_h} \underline{\mathbf{e}}_{ni}(q) \right\}^T \right]^T \tag{6}$$

$$\mathbf{P}_{nm^*}(q) = \sum_{n=1, n\neq i}^{N} \mathbf{D}^H_{x_n}(q) \mathbf{D}_{x_n}(q). \tag{7}$$

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

$$\underline{\mathbf{e}}_{ni}(q) = \mathbf{D}_{x_n}(q) \widehat{\underline{\mathbf{h}}}_{nm^*}(q) - \mathbf{D}_{x_i}(q) \widehat{\underline{\mathbf{h}}}_{im^*}(q) \tag{8}$$

where the diagonal matrix

$$\mathbf{D}_{x_n}(q) = \mathrm{diag}\left(\mathbf{F} \left\{ [x_n(qL_h - L_h) \ x_n(qL_h - L_h + 1) \ \cdots \ x_n(qL_h + L_h - 1)]^T \right\} \right) \tag{9}$$

is the DFT of the q-th frame input signal block for the n-th channel. In order to guarantee proper convergence and a non-zero error signal, the algorithm is initialized in the time domain to satisfy the unit-norm constraint:

$$\widehat{\mathbf{h}}_n(0) = \left[1/\sqrt{N} \ 0 \ \cdots \ 0 \right]^T, \qquad n = 1, 2, \ldots, N. \tag{10}$$

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 for 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, therefore the UNMCFLMS is an appropriate choice, but different solutions have been proposed in literature in order to alleviate the problem [6].

2.3 Adaptive Inverse Filtering

Given the N room transfer functions (RTFs) $H_n(z)$, a set of inverse filters $G_n(z)$ can be found by using the Multiple-Input/Output Inverse Theorem (MINT) [5] such that

$$\sum_{n=1}^{N} H_n(z) G_n(z) = 1, \tag{11}$$

assuming that the RTFs do not have any common zeros. In the time-domain, the inverse filter vector denoted as $\mathbf{g}$, is calculated by minimizing the following cost function:

$$C = \left\| \mathbf{H}\mathbf{g} - \mathbf{v} \right\|^2, \tag{12}$$

where $\| \cdot \|$ denote the l_2-norm operator and $\mathbf{g} = [\mathbf{g}_1^T \ \mathbf{g}_2^T \cdots \mathbf{g}_N^T]^T$, with $\mathbf{g}_n = [\mathrm{g}_{n,0} \ \mathrm{g}_{n,1} \cdots \mathrm{g}_{n,L_i-1}]^T$.

The vector $\mathbf{v}$ is the target vector, i.e. the Kronecker delta shifted by an appropriate modeling delay ($0 \leq d \leq NL_i$), while $\mathbf{H} = [\mathbf{H}_1\mathbf{H}_2 \cdots \mathbf{H}_N]$ and $\mathbf{H}_n$ is the convolution matrix of the RIR between the source and n-th microphone. When the matrix $\mathbf{H}$ is given or estimated through a system identification algorithm, the inverse filter set can be calculated as

$$\mathbf{g} = \mathbf{H}^{\dagger}\mathbf{v} \tag{13}$$

where $(\cdot)^{\dagger}$ denotes the Moore-Penrose pseudoinverse.

Considering the presence of disturbances, i.e. additive noise or RTFs fluctuations, the cost function (12) is modified as follows [2]:

$$C = \|\mathbf{Hg} - \mathbf{v}\|^2 + \gamma \|\mathbf{g}\|^2, \tag{14}$$

where the regularization parameter $\gamma \geq 0$ is a scalar coefficient representing the weight assigned to the disturbance term.

In [2] a general cost function, embedding noise and fluctuations, is derived together with the inverse filter. However, the inverse filter computation requires a matrix inversion that, in the case of long RIRs, can result in a high computational burden. Instead, an adaptive algorithm [10], based on the steepest-descent technique, has been here adopted to satisfy the real-time constraints:

$$\mathbf{g}_n(q+1) = \mathbf{g}_n(q) + \mu(q)[\mathbf{H}^T(\mathbf{v} - \mathbf{H}\mathbf{g}_n(q)) - \gamma\mathbf{g}_n(q)], \tag{15}$$

where $\mu(q)$ is the step-size and q is the time frame index. The convergence of the algorithm to the optimal solution is guaranteed if the usual conditions for the step-size in terms of autocorrelation matrix $\mathbf{H}^T\mathbf{H}$ 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 choosing a step-size that minimizes the cost function at the next iteration:

$$\begin{aligned} \mu(q) &= \frac{\mathbf{e}^T(q)\mathbf{e}(q)}{\mathbf{e}^T(q)\left(\mathcal{H}^T\mathcal{H} + \gamma I\right)\mathbf{e}(q)}, \\ \mathbf{e}(q) &= \mathcal{H}^T\left[\mathbf{v} - \mathcal{H}\mathbf{g}_{m^*}(q)\right] - \gamma\mathbf{g}_{m^*}(q). \end{aligned} \tag{16}$$

The illustrated algorithm presents two advantages: First, the regularization parameter γ makes the dereverberation process more robust to estimation errors due to the BCI algorithm [2]. Second, the complexity of the algorithm is decreased since no matrix inversion is required and operations can be performed in the frequency-domain through FFTs.

3 Automatic Speech Recognition System

The HMM system applied for processing features computed from the dereverberated speech signal was identical to the back-end used in [15]. 39 cepstral mean normalized MFCC features (including deltas and double deltas) are extracted from the speech signal every 10 ms using a window size of 25 ms. Each phoneme

is represented by three emitting states (left-to-right HMMs) with 16 Gaussian mixtures. The initial monophone HMMs were mapped to tied-state cross-word triphone models with shared state transition probabilities. Two Baum-Welch iterations were performed for re-estimation of the triphone models. Finally, the number of mixture components of the triphone models was increased to 16 in four successive rounds of mixture doubling and re-estimation (four iterations in every round). Both, acoustic models and a back-off bigram language model were trained on the non-reverberated version of the Buckeye training set.

4 Experiments

4.1 Data

Experiments have been conducted on the Buckeye corpus of conversational speech [7]. The corpus consists of interviews of forty native American English speakers speaking in conversational style. Signals have been recorded with close-talking microphones in quiet conditions with a sample rate of 16 kHz. The 255 recording sessions, each of which is approximately 10 min long, were subdivided into turns by cutting whenever the subject's speech was interrupted by the interviewer, or once a silence segment of more than 0.5 s length occurred. We used the same speaker independent training and test sets as in [15]. The lengths of the sets are 23.1 h and 2.6 h, respectively, and the vocabulary size is 9.1 k.

4.2 Experimental Setup

The experimental setup consists of a speaker located in the meeting room shown in Fig. 2. Inside, a table is present and an array of three omnidirectional microphones is located on its centre. Two reverberated conditions have been considered: *Stationary*, where the speaker talks at the seat denoted as "START" for the all duration of the utterance, and *non-stationary*, where the speaker talks at seat "START" for the first 60 s, and at seat "END" for the remaining time. The difference between the two conditions is that in the first the impulse response does not change, while in the second it changes instantaneously.

Three reverberation times (T_{60}) have been considered: 240 ms, 360 ms, and 480 ms. The reverberated test sets have been created concatenating the clean utterances in order to obtain segments with a minimum length of 120 s, and convolving them with the appropriate impulse responses. All the impulse responses are 1024 taps long, and have been generated by means of Habets' RIR Generator tool[1].

Experiments have been conducted on a Intel® Core™i7 machine running at 3 GHz with 4 GB of RAM. In this machine, the C++ implementation of dereverberation front-end achieves real-time execution with a real-time factor of 0.04.

[1] `http://home.tiscali.nl/ehabets/rirgenerator.html`

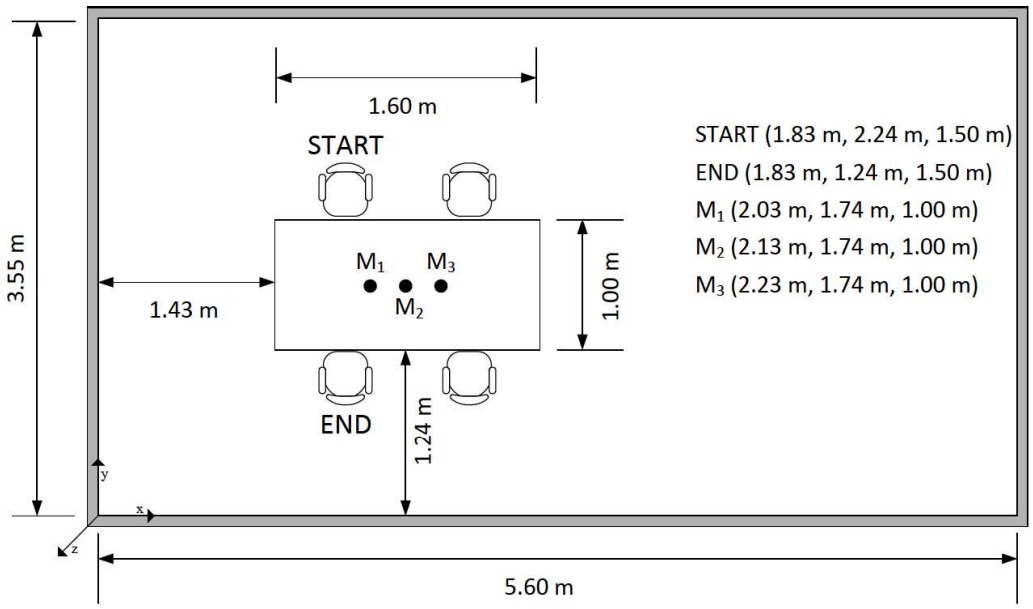

Fig. 2. Room setup: Microphones and speaker's positions coordinates are shown in brackets

4.3 Blind Impulse Response Estimation Performance

The performance of the BCI stage has been evaluated separately to highlight its behaviour in non-stationary conditions. The performance metric used to this end is the NPM [3], defined as:

$$\mathrm{NPM}\,(q) = 20\log_{10}\left(\frac{\|\epsilon(q)\|}{\|\mathbf{h}\|}\right), \tag{17}$$

where

$$\epsilon(q) = \mathbf{h} - \frac{\mathbf{h}^T\widehat{\mathbf{h}}(q)}{\widehat{\mathbf{h}}^T(q)\widehat{\mathbf{h}}(q)}\widehat{\mathbf{h}}(q) \tag{18}$$

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 frame.

Fig. 3 shows the NPM curves obtained in the stationary and non-stationary conditions for a Buckeye utterance of length 120 s and reverberated with $T_{60} = 480$ ms. In stationary conditions, the algorithm reaches an NPM value below -8 dB after about 25 s. In non-stationary conditions, the curve exhibits a peak when the impulse response changes, then starts lowering again reaching a value below -9 dB at the end of the utterance. This shows that the algorithm is able to track the abrupt change of RIRs and it does not suffer from misconvergence. However, a difference of about 2 dB between the stationary and non-stationary NPM curves can be noticed after 30 s from the RIRs change. The behaviour can be explained considering that the BCI algorithm convergence rate depends on the initialization strategy. In this situation, the identification of the "END" impulse response is initialized with the last estimation of the "START" one and not as in equation (10) as indicated in [3].

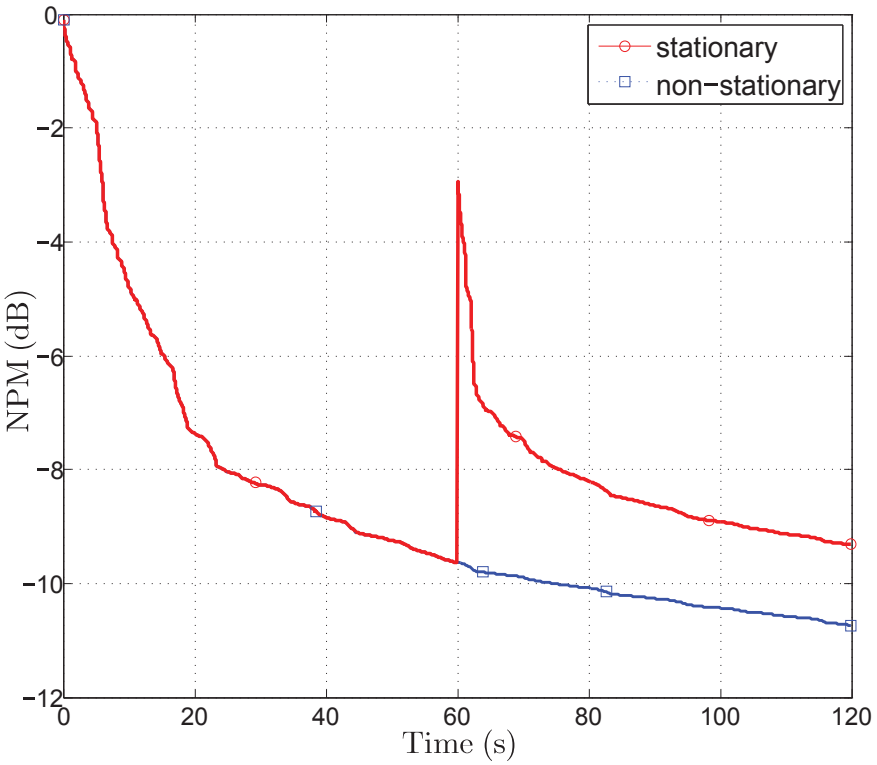

Fig. 3. NPM curves for the stationary and non-stationary conditions

The experiment suggests that a proper re-initialization of the algorithm, e.g. properly dealing with the behaviour of the cost function of equation (14), could be beneficial to the improvement of the tracking capabilities. This idea will be verified in the next section in terms of word recognition accuracy.

4.4 Speech Recognition Results

The word recognition accuracy obtained on clean data without the speech dereverberation front-end is 50.97% (see also [15]). Results obtained on the stationary and non-stationary reverberated conditions are shown in Table 1 for each T_{60}. The column "Average" contains the word accuracy average over the three T_{60}s.

Results show that, without processing, the word accuracy degrades by 12.92% in the stationary case and by 11.91% in the non-stationary one, and that as expected the difference increases with T_{60}. In the stationary case, the dereverberation front-end improves the word accuracy by on average 6.65%, i.e. 17.48% relative. It is worth highlighting that the differences across the three T_{60} are less pronounced: The motivation is that the dereveberation process strictly depends on the quality of the RIRs estimates, thus when a good match between them and the real filter is obtained, the equalization process is effective regardless the reverberation time. In the non-stationary case, the change tracking capability showed in the previous section in terms of NPM are confirmed: The average word accuracy degrades only by 1.28% w.r.t. the dereverberated stationary case, giving a relative improvement of 11.16%.

Table 1. Word accuracy (%) for the addressed conditions

		240 ms	360 ms	480 ms	Average
no processing	stationary	42.71	37.06	34.38	38.05
	non-stationary	43.83	38.30	35.04	39.06
dereverberated	stationary	46.36	44.79	42.95	44.70
	non-stationary	44.84	43.47	41.94	43.42

Re-initializing the BCI algorithm to equation (10) as suggested previously results in a average word recognition accuracy of 44.91%. The re-initialization is performed in an oracle style after the first 60 s of speech, i.e. when the impulse response changes. The word accuracy improves by 1.49% on average w.r.t. the non re-initialized solution and is similar to the dereveberated stationary result. This demonstrates that a proper re-initialization strategy of the BCI algorithm indeed improves the overall performance.

5 Conclusions

In this paper, a speech recognition system able to operate in non-stationary reverberated environments has been presented. The system is composed of a dereverberation front-end and a speech recognition engine able to recognize spontaneous speech. The performance of the front-end has been evaluated in terms of Normalized Projection Misalignment: Results showed that the blind channel identification stage is able to track an abrupt change of RIRs and it does not suffer from misconvergence. The entire system has been evaluated using the Buckeye corpus of conversational speech in stationary and non-stationary environments. In the stationary case, the front-end provides a 17.48% relative word accuracy improvement and the performance is less dependent on the value of T_{60}. In the non-stationary case, the RIRs tracking capabilities are confirmed: The average word accuracy degrades only of 1.28% w.r.t. the stationary scenario. Re-initializing the channel identification algorithm in an oracle style resulted in a average word accuracy improvement of 1.49% demonstrating the effectiveness of the idea.

In future works, the idea of re-initializing the blind channel identification algorithm will be exploited by suitably managing the cost function when the impulse response changes. In addition, the entire system performance will be assessed in different non-stationary conditions, e.g. in a moving-talker scenario. Noise will be addressed modifying the channel identification algorithm [1] and introducing suitable techniques in the speech recognizer feature extraction stage [9,12]. Finally the proposed front-end will be applied in other relevant human-computer interaction scenarios, such as keyword spotting [8,16] and emotion recognition [11].

References

1. Haque, M., Hasan, M.: Noise robust multichannel frequency-domain LMS algorithms for blind channel identification. IEEE Signal Process. Lett. 15, 305–308 (2008)
2. Hikichi, T., Delcroix, M., Miyoshi, M.: Inverse filtering for speech dereverberation less sensitive to noise and room transfer function fluctuations. EURASIP Journal on Advances in Signal Process. 2007(1) (2007)
3. Huang, Y., Benesty, J.: A class of frequency domain adaptive approaches to blind multichannel identification. IEEE Trans. Speech Audio Process. 51(1), 11–24 (2003)
4. Kumar, K., Singh, R., Raj, B., Stern, R.: Gammatone sub-band magnitude-domain dereverberation for ASR. In: Proc. of ICASSP, pp. 4604–4607 (May 2011)
5. Miyoshi, M., Kaneda, Y.: Inverse filtering of room acoustics. IEEE Trans. Signal Process. 36(2), 145–152 (1988)
6. Naylor, P., Gaubitch, N.: Speech Dereverberation. Signals and Communication Technology. Springer (2010)
7. Pitt, M., Dilley, L., Johnson, K., Kiesling, S., Raymond, W., Hume, E., Fosler-Lussier, E.: Buckeye corpus of conversational speech, 2nd release (2007), `http://www.buckeyecorpus.osu.edu`, Columbus, OH: Department of Psychology, Ohio State University (Distributor)
8. Principi, E., Cifani, S., Rocchi, C., Squartini, S., Piazza, F.: Keyword spotting based system for conversation fostering in tabletop scenarios: Preliminary evaluation. In: Proc. of 2nd Int. Conf. on Human System Interaction, Catania, pp. 216–219 (2009)
9. Principi, E., Cifani, S., Rotili, R., Squartini, S., Piazza, F.: Comparative evaluation of single-channel MMSE-based noise reduction schemes for speech recognition. Journal of Electrical and Computer Engineering 2010, 6 (2010)
10. Rotili, R., Cifani, S., Principi, E., Squartini, S., Piazza, F.: A robust iterative inverse filtering approach for speech dereverberation in presence of disturbances. In: Proc. of IEEE APCCAS, pp. 434–437 (December 2008)
11. Schuller, B., Batliner, A., Steidl, S., Seppi, D.: Recognising realistic emotions and affect in speech: state of the art and lessons learnt from the first challenge. Speech Communication, 1062–1087 (February 2011)
12. Schuller, B., Wöllmer, M., Moosmayr, T., Rigoll, G.: Recognition of noisy speech: A comparative survey of robust model architecture and feature enhancement. EURASIP Journal on Audio, Speech, and Music Processing 2009, 17 (2009)
13. Sehr, A., Maas, R., Kellermann, W.: Reverberation model-based decoding in the logmelspec domain for robust distant-talking speech recognition. IEEE Trans. on Audio, Speech, and Lang. Process. 18(7), 1676–1691 (2010)
14. Wölfel, M., McDonough, J.: Distant Speech Recognition, 1st edn. Wiley, New York (2009)
15. Wöllmer, M., Schuller, B., Rigoll, G.: A novel Bottleneck-BLSTM front-end for feature-level context modeling in conversational speech recognition. In: Proc. of ASRU, Waikoloa, Big Island, Hawaii, pp. 36–41 (December 2011)
16. Wöllmer, M., Marchi, E., Squartini, S., Schuller, B.: Multi-stream LSTM-HMM decoding and histogram equalization for noise robust keyword spotting. Cognitive Neurodynamics 5(3), 253–264 (2011)
17. Young, S., Everman, G., Kershaw, D., Moore, G., Odell, J.: The HTK Book. Cambridge University Engineering (2006)