ON-LINE FRAME-SYNCHRONOUS COMPENSATION OF NON-STATIONARY NOISE

V. Barreaud, I. Illina, D. Fohr

LORIA/INRIA
54602 Villers-lès-Nancy FRANCE
e-Mail: {barreaud,illina,fohr}@loria.fr

ABSTRACT

We present a frame-synchronous noise compensation algorithm that uses Stochastic Matching approach to cope with time-varying unknown noise. This method proposes to estimate simple mapping function in parallel with Viterbi alignment. The technique is entirely general since no assumption is made on the nature, level and variation of noise. Our algorithm is evaluated on the VODIS database recorded in a moving car. For various tasks, our technique outperforms significantly classical methods. For instance, using the affine transformation the proposed algorithm gives an error rate improvement of 13.3 % compared to Parallel Model Combination (PMC), 15.5 % on Spectral Subtraction (SS) and 27.8 % on frame-synchronous Mean Cepstre Removal (MCR) for the numbers recognition task in real noise.

1. INTRODUCTION

An automatic speech recognition (ASR) system gives a significant degradation in performances when used in an environment that does not match its training environment. This mismatch is due mostly to additional noise sources and discrepancies in channels and speakers. Those mismatch sources may be non-stationary and little a priori information about them is available.

Several techniques have been proposed to enhance speech in a robust manner. Two possible approaches can be explored. First, the parameters of the HMMs can be modified to make the transformed stochastic models better characterize the distorted features. This approach, called adaptation, gathers several techniques such as PMC [1], MAP [2] and MLLR [3]. Second, the corrupted features can be adjusted thanks to a transformation that is estimated from the noise characteristics. This set of methods, called compensation, gathers techniques such as MCR and Stochastic Matching [4]. The method developed here belongs to this category.

When acoustic environments are known to be non-stationary, three types of methods can be used. First, noise and channel can be modeled by HMMs trained by prior measurement of the environment [5]. Second, a bank of Kalman filters can be used to compensate the effect of time-varying noise [6]. Finally, sequential EM algorithms track additive noise parameters in cepstral domain [7].

Our work is based on [8] where an approximation of the mismatch function was performed in order to reduce the Kullback-Leibler information. Those derivations led to a recursively updated bias which expression was close to the one obtained in [4] with a Maximum-Likelihood approach. Compared to [4], where the batch estimation of mismatch function is derived, we use a frame per frame approach. Frame synchronous algorithms are naturally appealing to cope with non-stationary noise sources even if they often face convergence problems linked to the scarcity of data. One of the most popular frame synchronous technique is MCR: the mean of the incoming sequence of cepstra is computed and subtracted to the next observation.

We believe that this method can be enhanced by taking into account statistics of the HMMs derived during the recognition. In the frame synchronous compensation mode, complete statistics (*forward-backward* probabilities) are difficult to obtain because the end of sentence is not available. One solution is to calculate these statistics on short windows as in [8]. Another solution, proposed here, is to approximate these statistics by *forward probabilities*. The basic idea of our method is as follows. First, the hypothesis is made that during the Viterbi alignment, the states linked to the highest *forward* probabilities give a good modelisation of the speech observations. Then, the parameters of the mismatch function are estimated in order to enhance the likelihood of the observation given those states. Consequently, this on-line algorithm performs compensation in parallel with recognition and does not need any *a priori* information on the nature of the noise. Compensation transform is estimated frame per frame and confidence in its parameters is gained as forward probabilities computation goes along.

In this paper, we first present the theoretical framework beneath this method. Then two linear mapping functions are developed and a discussion on a forgetting process is open. In section 3, we present experimental results to compare our algorithm with classical techniques in time varying artificial

noise and in real-life car noise. Finally, in section 4, we draw conclusion and describe future work.

2. FRAME-SYNCHRONOUS COMPENSATION

2.1. Description of the algorithm

In the following, the one dimension case is treated, all the derivation being easily extended to multidimensional case if diagonal covariance matrices are used. We assume that the clean signal spectrum x_s is distorted by noise and gives y_s. It can be modeled as follows:

$$y_s = h_s \otimes x_s + n_s \tag{1}$$

where $\otimes$ is the convolution operator, h_s is the channel noise, n_s is the additive noise and the s subscript denotes the spectral domain. In the cepstral domain, (1) becomes:

$$y = x - g(x_s, n_s, h_s) \approx x - g(y)$$

where $g(x_s, n_s, h_s)$ is a non-linear function without any regular expression. Usually, the exact values of x_s, n_s and h_s are unknown. In practice, this function is approximated by $g(y)$. The goal of compensation is to find a transformation f such that $f(y)$ approaches x: $f(y) \approx x$.

Let us consider a Hidden Markov Model recognition system of M N-states models. Each state n is characterized by mixture of K gaussian probability functions of mean $\mu_{(n,k)}$ and variance $\sigma_{(n,k)}$ with $k \in \{1, \ldots, K\}$. In the following, the pair (n, k) represent the k-th Gaussian component of the n-th state.

Let $S_t = \{s_0, \ldots, s_t\}$ denotes a partial state sequence and $Y_T = \{y_0, \ldots, y_T\}$ be a sequence of noisy observations in the cepstral domain, corresponding to the sequence of clean features $X_T = \{x_0, \ldots, x_T\}$.

Consider θ as the set of parameters of a transformation $f_\theta(y)$ from the testing observation space to the training space. It has been shown in [8] that the set θ maximizing the Kullback-Leibler information $J(\theta) = E\{log(p(Y_t|\theta)\}$ can be approximated by a sequence $\{\theta_i\}$ that maximizes the auxiliary function Q:

$$\begin{aligned} \theta_{t+1} &= \underset{\theta}{argmax} Q_{t+1}(\Theta_t, \theta) \\ Q_{t+1}(\Theta_t, \theta) &= \sum_{\tau=1}^{t+1} L_{\tau|t+1}(\Theta_{\tau-1}) \end{aligned}$$

with $\Theta_t = (\theta_0, \ldots, \theta_t)$ and the auxiliary function defined thanks to the following expression of likelihood:

$$L_{\tau|t+1}(\Theta_{\tau-1}) = \log(|f'_\theta(y_t)|) - \frac{1}{2} \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k) \frac{(f_\theta(y_t) - \mu_{(n,k)})^2}{\sigma^2_{(n,k)}}$$

In which $f'_\theta(y_t)$ is the partial derivative of the compensation function with respect to the observation y for the time frame t and $\gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k)$ is the probability of the τ-th emitting state s_τ being n and its principal Gaussian component g_τ being k knowing the sequence of observations Y_{t+1} and $\Theta_{\tau-1}$ (*forward-backward* probability).

Let a simple transformation $f_{\theta_t}(y_{t+1}) = y_{t+1} + b_t$. Then the bias parameters $B_{\tau-1} = \{b_0, \ldots, b_{\tau-1}\}$ can thus be estimated over the optimum Viterbi path:

$$b_{t+1} = b_t - \frac{\sum_{n=1}^{N} \sum_{k=1}^{K} \gamma_{t+1|t+1,B_t}(n,k) \frac{y_{t+1} + b_t - \mu_{(n,k)}}{\sigma^2_{(n,k)}}}{\sum_{\tau=1}^{t+1} \sum_{n=1}^{N} \sum_{k=1}^{K} \frac{\gamma_{\tau|t+1,B_{\tau-1}}(n,k)}{\sigma^2_{(n,k)}}} \tag{2}$$

where

$$\gamma_{\tau|t+1,B_{\tau-1}}(n,k) = p(s_\tau = n, g_\tau = k | Y_{t+1}, B_{\tau-1})$$

(derived in [9] thanks to *forward - backward* probabilities). (2) converges toward an optimum bias that maximizes the overall likelihood of a state sequence.

The $\gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k)$ probability is unavailable during alignment. In our algorithm, we make the hypothesis that the *forward probability*

$$\alpha_{\tau|\Theta_{\tau-1}}(n,k) = p(Y_\tau, s_\tau = n, g_\tau = k | \Theta_{\tau-1})$$

could be used as a weighting factor in equation (2) and leads to the following expression:

$$b_{t+1} = b_t - \frac{\sum_{n=1}^{N} \sum_{k=1}^{K} \alpha_{t+1|B_t}(n,k) \frac{y_{t+1} + b_t - \mu_{(n,k)}}{\sigma^2_{(n,k)}}}{\sum_{\tau=1}^{t+1} \sum_{n=1}^{N} \sum_{k=1}^{K} \frac{\alpha_{\tau|B_{\tau-1}}(n,k)}{\sigma^2_{(n,k)}}} \tag{3}$$

Equation (3) can be simplified: we assume that the sums over all possible states and Gaussian components at time τ can be fairly approximated by the contribution of the pair (n,k) that maximizes $\alpha_{\tau|B_{\tau-1}}(n,k)$ alone. Let $(n,k)_\tau$ be that pair. Then (3) becomes:

$$b_{t+1} = b_t - \frac{\frac{y_{t+1} + b_t - \mu_{(n,k)_{t+1}}}{\sigma^2_{(n,k)_{t+1}}}}{\sum_{\tau=1}^{t+1} \frac{1}{\sigma^2_{(n,k)_\tau}}} \tag{4}$$

Thus, in our method, computation of the bias at time t does not require backtracking along a path. On the contrary, at each time frame *t*, the most probable state in the *forward probability* sense is used to re-estimate the transformation parameters that would maximize this probability. Hence, the compensation function is recursively computed at each time frame by a simple equation.

Similarly, an affine transform $f_t(y_{t+1}) = a_t * y_{t+1} + b_t$ can be estimated as follow:

$$
\begin{aligned}
b_{t+1} &= b_t - \frac{(m_{t+1} + E_{t+1} + \frac{t+1}{a_t^2} - (y_{t+1} - \frac{1}{a_t})D_{t+1})}{\delta_{t+1}} \\
a_{t+1} &= a_t - \frac{1}{\delta_{t+1}} \left(m_{t+1}(y_{t+1}C_{t+1} - D_{t+1}) - \frac{1}{a_t}C_{t+1} \right) \\
\delta_{t+1} &= \frac{1}{C_{t+1}\left(E_{t+1} + \frac{t+1}{a_t^2}\right) - (D_{t+1})^2} \\
E_{t+1} &= \sum_{\tau=1}^{t+1} \frac{y_\tau^2}{\sigma^2_{(n,k)_\tau}}, \quad D_{t+1} = \sum_{\tau=1}^{t+1} \frac{y_\tau}{\sigma^2_{(n,k)_\tau}} \\
C_{t+1} &= \sum_{\tau=1}^{t+1} \frac{1}{\sigma^2_{(n,k)_\tau}}, \quad m_{t+1} = \frac{a_t y_{t+1} + b_t - \mu_{(n,k)_{t+1}}}{\sigma^2_{(n,k)_{t+1}}}
\end{aligned}
$$

2.2. Forgetting Factor

This algorithm estimates a transformation thanks to a sequence of states that are not guaranteed to be part of the optimal Viterbi path. Thus, if the first states of the sequence effectively model the observations sequence, the matching transformation is efficient. Hence the following observations are correctly mapped onto the training space. On the other hand, if the state sequence does not belong to this optimal Viterbi path, then our algorithm should reduce the influence of those states on the estimation process. Thus, we propose to integrate a forgetting process in our algorithm to reduce the influence of the past events in the case of non-stationary noise. As shown in equation (5), we choose to weight the re-estimation term by a fixed coefficient ff:

$$
b_{t+1} = b_t - ff * \frac{\frac{y_{t+1} + b_t - \mu_{(n,k)_{t+1}}}{\sigma^2_{(n,k)_{t+1}}}}{\sum_{\tau=1}^{t+1} \frac{1}{\sigma^2_{(n,k)_\tau}}} \qquad (5)
$$

In the above equation, ff is a forgetting factor with $0 \leq ff \leq 1$. The value of this factor is currently fixed experimentally. A similar forgetting factor is used for the computation of the affine function's parameters.

3. EXPERIMENTAL FRAMEWORK

3.1. VODIS Database

All the experiments have been conducted on the Voice-Operated Driver Information Systems (VODIS) Database. This corpus gathers 200 french speakers. The speakers were divided into two sets: the training set (*Training*, 159 speakers) and the test set (*Test*, 41 speakers). The recordings were made in french, in a moving car with various driving situations (opened window, traffic/highway, radio). Speakers were asked to utter phone numbers (*phone numbers* task, 95% confidence interval is ±1%) and numbers up to 12000 (*numbers* task, 95% confidence interval is ±1%). Notice that french phone numbers are composed of numbers ranking from 0 to 99. The speech sequences have been collected by two microphones, synchronously. The first microphone (*close talk*) was placed close to the mouth of the speaker and collected "clean" speech with an average Signal to Noise Ratio (SNR) of 20.7 dB. The second one was placed on the rear-view mirror and collected distorted speech with an average SNR of 10.8 dB (*far-talk*). The signal was sampled at 11025 Hz, and encoded in 36 dimensions cepstra sequence composed by 12 MFCC, 12 Δ and 12 $\Delta\Delta$. We used 3-states phoneme models, each state composed of a mixture of 8 Gaussian probability density functions. The models were trained over all the *close-talk* utterances of *Training*.

3.2. Non Stationary Added Noise

The *close-talk* of the *Test* set for the *numbers* task have been artificially corrupted by additive noise in the time domain (buccaneer2.wav of the NOISEX database) in three different manners. In *linear* experiment, noise is linearly increasing throughout each sentence. In *invLin* experiment, noise is linearly decreasing throughout. And finally, in *triangle* experiment, noise linearly increases during the first half of each sentence and then linearly decreases.

Table 1 presents the word accuracy (in percent) of our algorithms on clean data (*clean*) and on artificially corrupted data (*linear*, *triangle* and *invLin*). Two transformations have been tested: a simple bias (*Bias*) and an affine transform (*Affine*). They are compared to recognition without compensation (Baseline) and with MCR. Our algorithm significantly outperforms MCR for the *triangle* and *invLin*. This difference shows that the use of Viterbi statistics gives a more precise estimation of bias right from the first frames, contrary to MCR.

	Baseline	MCR	Bias	Affine
clean (SNR: 20.3 dB)	89.9	90.2	91.9	90.9
linear (SNR: 16.8 dB)	67.3	83.6	82.3	83.4
triangle (SNR: 17.7 dB)	75.6	81.3	82.3	83.8
invLin (SNR: 16 dB)	50.5	52.4	76.9	79.4

Table 1. Word accuracy (in %) on added noise numbers recognition task (*ff* = 0.8).

Figure 1 represents the evolution of *Bias* on the first cepstral coefficient (c_0) of one *numbers* sentence corrupted by a linearly decreasing additive noise and two values of ff: 0.8 (no errors) and 1.0 (no forgetting process, sentence was not recognized). This figure shows that a lower *ff* enables to reduce the influence of an incorrect state.

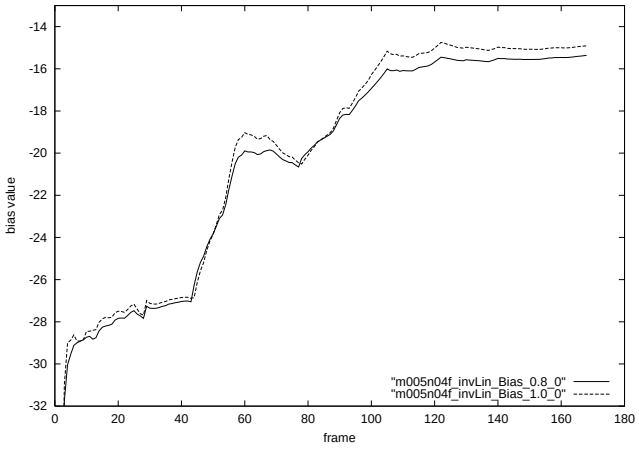

Fig. 1. Evolution of the Bias's value on c_0 for one number utterance, corrupted by *invLin* noise.

3.3. Real Car Noise

Experiments was then conducted on naturally noisy utterances (*Test* collected on *far-talk*). Table 2 represents the results of classical compensation methods (Baseline, MCR, SS, PMC) and transforms given by our algorithm (*Bias* and *Affine* with a forgetting factor of 0.8). It shows that *Affine* method outperforms significantly all classical methods for the *numbers* task and that both *Bias* and *Affine* methods are significantly more efficient for the *phone numbers* task. Table 3 represents the influence of the forgetting factor (ff)

	Baseline	MCR	SS	PMC	Bias	Affine
numbers	63.5	67.3	72.1	72.8	72.9	**76.4**
phone numbers	78.6	80.8	79.3	81.6	**83.5**	**86.3**

Table 2. Word accuracy on far-talk test set (SNR: 10.8 dB, ff=0.8 for affine transformation).

on word accuracy for both the *phone numbers* and the *numbers* tasks realized on the *far-talk* set. It shows that, in real life noise, *Affine* is more easily influenced by ff than *Bias*.

ff	0.4	0.5	0.6	0.7	0.8	0.9	1.0
numbers							
Bias	**72.9**	**72.9**	72.8	72.5	72.5	72.7	72.9
Affine	74.5	75.2	75.6	75.1	**76.4**	75.6	74.6
phone numbers							
Bias	**84.2**	84.1	84.0	83.9	83.8	84.0	83.5
Affine	86.2	86.1	86.0	**86.4**	86.3	86.1	85.0

Table 3. Word accuracy on far-talk test set with respect to the forgetting factor ff.

4. CONCLUSION AND FUTURE WORK

This article presents an on-line frame-synchronous noise compensation algorithm using the theoretical framework of Stochastic Matching. This algorithm does not need any *a priori* information on the environment and compensates non-stationary noise. The basic idea is to use *forward probabilities* in the estimation of the simple affine transformation's parameters. Moreover, a simple yet efficient forgetting process allows our algorithm to cope with non stationary environment. To evaluate the algorithm we have chosen to recognize *numbers* and *phone numbers* pronounced in a moving car. For both these tasks, our method significantly outperforms the frame-synchronous MCR and spectral subtraction techniques and PMC. Moreover, contrary to MCR and PMC, our technique does not require any specific models training and, thus, can be used along with other compensation techniques. Future work will involve studies on class-specific transforms based on a tree structure and a self-evoluting forgetting factor.

5. REFERENCES

[1] M.J.F. Gales, *Model-Based Techniques for Noise Robust Speech Recognition*, Ph.D. thesis, Gonville and Caius College, September 1995.

[2] J.-L. Gauvain and C.-H. Lee, "Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains," *IEEE Transaction on Speech and Audio Processing*, vol. 2, no. 2, pp. 291–298, 1994.

[3] 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, 1995.

[4] A. Sankar and C.H. Lee, "A Maximum Likelihood Approach to Stochastic Matching for Robust Speech Recognition," *IEEE Transaction on Speech and Audio Processing*, pp. 190–202, 1996.

[5] A. Varga and R.K. Moore, "Hidden Markov Model Decomposition of Speech and Noise," in *ICASSP*, 1990, pp. 845–848.

[6] N.S. Kim, "Time-Varying Noise Compensation Using Multiple Kalman Filters," in *ICASSP*, 1999, pp. 1540–1543.

[7] N.S. Kim, D.K. Kim, and S.R. Kim, "Application of Sequential Estimation to Time Varying Environment Compensation," in *IEEE Workshop on Speech Recognition and Understanding*, 1997, pp. 389–395.

[8] L. Delphin-Poulat, C. Mokbel, and J. Idier, "Frame Synchronous Stochastic Matching Based on the Kullback-Leibler Information," in *ICASSP*, 1998, pp. 89–92.

[9] V. Krishnamurthy and J.B. Moore, "On-line Estimation of HMM Parameters Based on the Kullback-Leibler Information Measure," *IEEE Transaction on Signal Processing*, vol. 41, no. 8, pp. 2557–2572, 1993.