RECENT ADVANCES IN PD-MEMLIN FOR SPEECH RECOGNITION IN CAR CONDITIONS

Luis Buera, Eduardo Lleida, Antonio Miguel, and Alfonso Ortega

Communication Technologies Group (GTC)
Aragon Institute of Engineering Research (I3A) University of Zaragoza, Spain
{lbuera,lleida,amiguel,ortega}@unizar.es

ABSTRACT

In a previous work, Phoneme-Dependent Multi-Environment Models based LInear Normalization, PD-MEMLIN, was presented and it was proved to be effective to compensate environment mismatch. Since PD-MEMLIN transformations have to be estimated from stereo data corpora, and the computational cost is high, two approaches are proposed: Coefficient Progressive PD-MEMLIN, CPPD-MEMLIN, and blind PD-MEMLIN. The first one consists on a partial normalization of the feature vector, reducing the computational cost, while blind PD-MEMLIN can be applied over any non stereo data corpora, thus the estimation of the transformation is based on an iterative technique from noisy data and a target clean speech model. Some experiments with SpeechDat Car database were carried out in order to study the behavior of the proposed techniques in a real acoustic environment. In the previous work, PD-MEMLIN with stereo data and normalizing 13 MFCC coefficients reached 77.67% of improvement. In this paper, CPPD-MEMLIN with only 4 coefficients obtains an average improvement of 72.40%, and blind PD-MEMLIN obtains an average improvement of 73.96%.

1. INTRODUCTION

When testing and training acoustic conditions are different, the accuracy of speech recognition systems rapidly degrades. In order to compensate this mismatch, several techniques have been developed. They can be grouped into two important categories: acoustic models adaptation, and feature compensation, or normalization. The first one, which only modifies the acoustic models, can be more specific, whereas, feature compensation [1], which modifies the feature vectors, needs less data and computation time. Hybrid techniques also exist and they have proved to be effective [2]. The use of one or other kind of algorithms depends on the application.

There are several feature compensation families [3, 4], but one of the most promised research line is based on Minimum Mean Squared Error, MMSE, estimation. Techniques like Codeword-Dependent Cepstral Normalization, CDCN [5], Stereo based Piecewise LInear Compensation for Environments, SPLICE [6], Multi-Environment Models based LInear Normalization, MEMLIN [7], or Phoneme Dependent MEMLIN, PD-MEMLIN, which is the phoneme dependent version of MEMLIN, [8] are some examples of MMSE based feature compensation.

A previous work [8] shows that PD-MEMLIN is effective in order to compensate the effects of dynamic and adverse car conditions. Although, this technique has two limitations. Firstly, the computation cost, and secondly, stereo data is needed to estimate the linear transformations in a previous training process, and they may not be always available. In this paper, two approaches are presented in order to compensate these problems: Coefficient Progressive PD-MEMLIN, CPPD-MEMLIN, and blind PD-MEMLIN. In CPPD-MEMLIN, only a subset of parameters of the feature vectors are normalized, instead of using all of them, obtaining a lighter computational cost technique; while, blind PD-MEMLIN training process is only based on noisy signal to estimate the transformations and there is no need of stereo data. These two algorithms will be compared with classic PD-MEMLIN, using stereo data and normalizing all the coefficients of the feature vectors.

This paper is organized as follows: in Section 2, an overview of PD-MEMLIN is presented. The blind version of PD-MEMLIN is introduced in Section 3. CPPD-MEMLIN is explained in Section 4. The results for CPPD-MEMLIN and blind PD-MEMLIN with SpeechDat Car database [9] are presented and discussed in Section 5. Finally, the conclusions are included in Section 6.

2. PD-MEMLIN

Phoneme Dependent Multi-Environment Models based LInear Normalization is an empirical feature vector normalization technique which uses stereo data in order to estimate the different compensation linear transformations in the previous training process. The clean feature space is modelled as a mixture of Gaussians for each phoneme. The

This work has been supported by the national project TIC2002-04103-C03-01

noisy space is split in several basic acoustic environments and each environment is modelled as a mixture of Gaussians for each phoneme. The transformations are estimated for all basic environments between a clean phoneme Gaussian and a noisy Gaussian of the same phoneme. This can be shown in Fig. 1 for one environment.

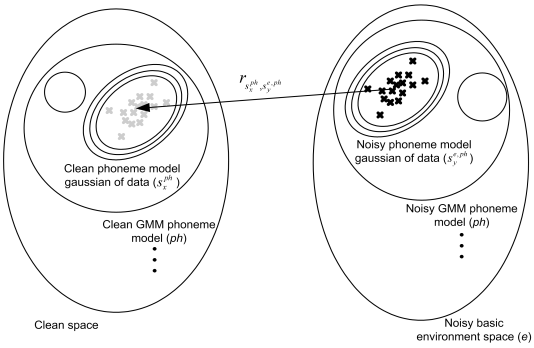

Fig. 1. Scheme of PD-MEMLIN transformations for one environment.

2.1. Approximations

Three approximations are assumed: firstly, some basic environments are defined in the noisy space, and noisy feature vectors, y, follow the distribution of Gaussian mixture for each basic environment and phoneme

$$p_{e,ph}(y) = \sum_{s_y^{e,ph}} p(y|s_y^{e,ph})p(s_y^{e,ph}), \tag{1}$$

$$p(y|s_y^{e,ph}) = N(y; \mu_{s_y^{e,ph}}, \Sigma_{s_y^{e,ph}}), \tag{2}$$

where $s_y^{e,ph}$ denotes the correspondent Gaussian of the noisy model for the e environment and ph phoneme; $\mu_{s_y^{e,ph}}$, $\Sigma_{s_y^{e,ph}}$, and $p(s_y^{e,ph})$ are the mean vector, the diagonal covariance matrix, and the weight associated to $s_y^{e,ph}$.

Second, clean feature vectors, x, are modelled following the distribution of Gaussian mixture

$$p_{ph}(x) = \sum_{s_x^{ph}} p(x|s_x^{ph})p(s_x^{ph}), \tag{3}$$

$$p(x|s_x^{ph}) = N(x; \mu_{s_x^{ph}}, \Sigma_{s_x^{ph}}), \tag{4}$$

where s_x^{ph} denotes the correspondent Gaussian of the clean model and phoneme; $\mu_{s_x^{ph}}$, $\Sigma_{s_x^{ph}}$, and $p(y|s_x^{ph})$ are the mean, diagonal covariance matrix, and the weight associated to s_x^{ph}.

Third, for each time frame, t, x is approached as a function, Ψ, of the noisy feature vector, y_t, clean model Gaussians, s_x^{ph}, and noisy environment model Gaussians, $s_y^{e,ph}$

$$x \simeq \Psi(y_t, s_x^{ph}, s_y^{e,ph}) = y_t - r_{s_x^{ph},s_y^{e,ph}}, \tag{5}$$

where $r_{s_x^{ph},s_y^{e,ph}}$ is the independent term of the linear transformation, and it depends on each pair of Gaussians, s_x^{ph} and $s_y^{e,ph}$.

2.2. Cepstral enhancement

Given the noisy vector, y_t, the clean one is estimated by MMSE criterion

$$\hat{x}_t = E[x|y_t] = \int_x xp(x|y_t)dx, \tag{6}$$

where $p(x|y_t)$ is the Probability Density Function (PDF) of x given y_t. With the three previous approximations, (6), can be approximated as expression (7).

In (7), $p(e|y_t)$ is the environment weight, $p(ph|y_t, e)$ is the probability of the phoneme ph, given the noisy feature vector and the environment, $p(s_y^{e,ph}|y_t, e, ph)$ is the probability of the noisy Gaussian given y_t, the environment, and the phoneme, and finally $p(s_x^{ph}|y_t, e, ph, s_y^{e,ph})$ is the probability of the clean Gaussian given y_t, e, ph and $s_y^{e,ph}$.

$r_{s_x,s_y^{e,ph}}$ and $p(s_x^{ph}|y_t, e, ph, s_y^{e,ph})$ are computed through a previous training process. The other probabilities are estimated on line for each time frame in the recognition phase.

The probability of the environment, $p(e|y_t)$, is estimated using a recursive solution as

$$p(e|y_t) = \beta \cdot p(e|y_{t-1}) + (1-\beta)\frac{\sum_{ph} p_{e,ph}(y_t)}{\sum_e \sum_{ph} p_{e,ph}(y_t)}, \tag{8}$$

where β is the memory constant, close to 1 (0.98 in this paper), and $p(e|y_0)$ is considered uniform for all environments. Also, $p(ph|y_t, e)$ and $p(s_y^{e,ph}|y_t, e, ph)$, are estimated as

$$p(ph|y_t, e) = \frac{p_{e,ph}(y_t)}{\sum_{ph} p_{e,ph}(y_t)}. \tag{9}$$

$$p(s_y^{e,ph}|y_t, e, ph) = \frac{p(y_t|s_y^{e,ph})p(s_y^{e,ph})}{\sum_{s_y^{e,ph}} p(y_t|s_y^{e,ph})p(s_y^{e,ph})}. \tag{10}$$

In order to compute $p(s_x^{ph}|y_t, e, ph, s_y^{e,ph})$, and $r_{s_x^{ph},s_y^{e,ph}}$, a previous training process with available stereo data for each environment and phoneme is needed: $X_{e,ph} = \{x_1^{e,ph}, ..., x_{t_{e,ph}}^{e,ph}, ..., x_{T_{e,ph}}^{e,ph}\}$, for clean feature vectors and $Y_{e,ph} = \{y_1^{e,ph}, ..., y_{t_{e,ph}}^{e,ph}, ..., y_{T_{e,ph}}^{e,ph}\}$ for noisy ones, with $t_{e,ph} \in [1, T_{e,ph}]$.

The conditional probability, $p(s_x^{ph}|y_t, e, ph, s_y^{e,ph})$, can be considered time independent, and it may be estimated using (1), (2), (3), and (4): expression (11).

Finally, $r_{s_x^{ph},s_y^{e,ph}}$ can be obtained by minimizing the weighted square error, $E_{s_x^{ph},s_y^{e,ph}}$ (expressions (12) and (13)).

$$\hat{x}_t \simeq y_t - \sum_{e} \sum_{ph} \sum_{s_x^{ph}} \sum_{s_y^{e,ph}} r_{s_x^{ph}, s_y^{e,ph}} p(e|y_t) p(ph|y_t, e) p(s_y^{e,ph}|y_t, e, ph) p(s_x^{ph}|y_t, e, ph, s_y^{e,ph}). \quad (7)$$

$$p(s_x^{ph}|y_t, e, ph, s_y^{e,ph}) \simeq p(s_x^{ph}|s_y^{e,ph}) = \frac{\sum_{t_{e,ph}} p(x_{t_{e,ph}}^{e,ph}|s_x^{ph}) p(y_{t_{e,ph}}^{e,ph}|s_y^{e,ph}) p(s_x^{ph}) p(s_y^{e,ph})}{\sum_{t_{e,ph}} \sum_{s_x^{ph}} p(x_{t_{e,ph}}^{e,ph}|s_x^{ph}) p(y_{t_{e,ph}}^{e,ph}|s_y^{e,ph}) p(s_x^{ph}) p(s_y^{e,ph})}. \quad (11)$$

$$E_{s_x^{ph}, s_y^{e,ph}} = \sum_{t_{e,ph}} p(s_x^{ph}|x_{t_{e,ph}}^{e,ph}, e, ph) p(s_y^{e,ph}|y_{t_{e,ph}}^{e,ph}, e, ph) (x_{t_{e,ph}}^{e,ph} - y_{t_{e,ph}}^{e,ph} + r_{s_x^{ph}, s_y^{e,ph}})^2. \quad (12)$$

$$r_{s_x^{ph}, s_y^{e,ph}} = arg \min_{r_{s_x^{ph}, s_y^{e,ph}}} (E_{s_x^{ph}, s_y^{e,ph}}) = \frac{\sum_{t_{e,ph}} p(s_x^{ph}|x_{t_{e,ph}}^{e,ph}, e, ph) p(s_y^{e,ph}|y_{t_{e,ph}}^{e,ph}, e, ph) (y_{t_{e,ph}}^{e,ph} - x_{t_{e,ph}}^{e,ph})}{\sum_{t_{e,ph}} p(s_x^{ph}|x_{t_{e,ph}}^{e,ph}, e, ph) p(s_y^{e,ph}|y_{t_{e,ph}}^{e,ph}, e, ph)}. \quad (13)$$

In (12) and (13), $p(s_y^{e,ph}|y_{t_{e,ph}}^{e,ph}, e, ph)$ is the probability of $s_y^{e,ph}$, given the noisy feature vector, $y_{t_{e,ph}}^{e,ph}$, the environment and the phoneme, and it can be obtained as (10). Also, in the same expressions, $p(s_x^{ph}|x_{t_{e,ph}}^{e,ph}, e, ph)$ is the probability of s_x^{ph} given the clean feature vector, e and ph, and it is estimated as

$$p(s_x^{ph}|x_{t_{e,ph}}^{e,ph}, e, ph) = \frac{p(x_{t_{e,ph}}^{e,ph}|s_x^{ph}) p(s_x^{ph})}{\sum_{s_x^{ph}} p(x_{t_{e,ph}}^{e,ph}|s_x^{ph}) p(s_x^{ph})}. \quad (14)$$

3. BLIND PD-MEMLIN

Since in many cases stereo data may not be available, an iterative blind training version of PD-MEMLIN has been developed. As it can be observed in Section 2, only two terms need stereo data in order to be estimated: $p(s_x^{ph}|s_y^{e,ph})$ and $r_{s_x^{ph}, s_y^{e,ph}}$. Blind PD-MEMLIN training process has two phases: the initialization, and the adjust iterative phase.

3.1. Initialization

Initialization is based on two steps. The first one estimates a rough approximation ($p_0(s_x^{ph}|s_y^{e,ph})$, and $r_{0,s_x^{ph}, s_y^{e,ph}}$) for the two variables, and the second one obtains the final initialization terms, $p_{ini}(s_x^{ph}|s_y^{e,ph})$, and $r_{ini,s_x^{ph}, s_y^{e,ph}}$.

$p_0(s_x^{ph}|s_y^{e,ph})$ is estimated by a modified Kullback Liebner distance. Given a noisy Gaussian of the e environment and ph phoneme, $s_y^{e,ph}$, and a clean one of the ph phoneme, s_x^{ph}, the modified Kullback Liebner distance, $d_{KL}(s_y^{e,ph}, s_x^{ph})$, can be obtained as expression (15), where $\Sigma_{s_x^{ph}}(i,i)$ is the i^{th} term of the diagonal covariance matrix of the s_x^{ph} Gaussian, and $\Sigma_{s_y^{e,ph}}(i,i)$ is the i^{th} term of the diagonal covariance matrix of the noisy Gaussian, $s_y^{e,ph}$. Note that the use of this expression is based on the assumption that noise modifies the mean vectors of the Gaussians in a more important way than covariance matrices.

As modified Kullback Liebner distance between two Gaussians is not symmetric, and it is not proportional to the likelihood of the two Gaussians, a pseudo-likelihood, $pl_{KL}(s_y^{e,ph}, s_x^{ph})$, can be defined to estimate $p_0(s_x^{ph}|s_y^{e,ph})$

$$pl_{KL}(s_y^{e,ph}, s_x^{ph}) = \frac{2}{d_{KL}(s_y^{e,ph}, s_x^{ph}) + d_{KL}(s_x^{ph}, s_y^{e,ph})}, \quad (16)$$

$$p_0(s_x^{ph}|s_y^{e,ph}) = \frac{pl_{KL}(s_y^{e,ph}, s_x^{ph})}{\sum_{s_x^{ph}} pl_{KL}(s_y^{e,ph}, s_x^{ph})}. \quad (17)$$

In order to obtain $r_{0,s_x^{ph}, s_y^{e,ph}}$, the following expression is used

$$r_{0,s_x^{ph}, s_y^{e,ph}} = \frac{\sum_{t_{e,ph}} p(s_y^{e,ph}|y_{t_{e,ph}}^{e,ph}, e, ph) (y_{t_{e,ph}}^{e,ph} - \mu_{s_x^{ph}})}{\sum_{t_{e,ph}} p(s_y^{e,ph}|y_{t_{e,ph}}^{e,ph}, e, ph)}, \quad (18)$$

Note that $r_{0,s_x^{ph}, s_y^{e,ph}}$ is the correspondent PD-MEMLIN expression with stereo data, assuming that the clean feature vectors associated to a clean Gaussian, s_x^{ph}, are the mean vector, $\mu_{s_x^{ph}}$. To estimate the final initialization terms, $p_{ini}(s_x^{ph}|s_y^{e,ph})$, and $r_{ini,s_x^{ph}, s_y^{e,ph}}$, two techniques need to be applied: Forced alignment normalization of noisy training feature vectors, which obtains $p_{ini}(s_x^{ph}|s_y^{e,ph})$, and Expectation Maximization, EM, algorithm, which estimates $r_{ini,s_x^{ph}, s_y^{e,ph}}$.

Forced alignment normalization of noisy training feature vector technique consists on three steps:

• Estimate the phoneme associated to each noisy training feature vector by a forced Viterbi alignment of the training utterance.

• Normalize noisy training feature vectors only with the transformations of the associated phonemes; in this case $p_0(s_x^{ph}|s_y^{e,ph})$, and $r_{0,s_x^{ph}, s_y^{e,ph}}$.

• Estimate the new transformations with PD-MEMLIN stereo data training expressions (11), (13), where clean

$$d_{KL}(s_y^{e,ph}, s_x^{ph}) = \frac{p(s_y^{e,ph})}{2}\sum_i log(\frac{\Sigma_{s_x^{ph}}(i,i)}{\Sigma_{s_y^{e,ph}}(i,i)} + \frac{\Sigma_{s_y^{e,ph}}(i,i)}{\Sigma_{s_x^{ph}}(i,i)} - 1) + p(s_y^{e,ph})log(\frac{p(s_y^{e,ph})}{p(s_x^{ph})}). \tag{15}$$

$$r_{ini,s_x^{ph},s_y^{e,ph}} = \frac{\sum_{t_{e,ph}} p(s_y^{e,ph}|y_{t_{e,ph}}^{e,ph}, e, ph)p(s_x^{ph}|y_{t_{e,ph}}^{e,ph}, s_y^{e,ph}, \phi)(y_{t_{e,ph}}^{e,ph} - \mu_{s_x^{ph}})}{\sum_{t_{e,ph}} p(s_y^{e,ph}|y_{t_{e,ph}}^{e,ph}, e, ph)p(s_x^{ph}|y_{t_{e,ph}}^{e,ph}, s_y^{e,ph}, \phi)}, \tag{19}$$

training data will be the normalized noisy training feature vectors. In this case, only (11) will be applied in order to obtain $p_{ini}(s_x^{ph}|s_y^{e,ph})$.

In order to estimate $r_{ini,s_x^{ph},s_y^{e,ph}}$, EM algorithm is applied. The correspondent expression can be seen in (19), where $p(s_x^{ph}|y_{t_{e,ph}}^{e,ph}, s_y^{e,ph}, \phi)$, means that the probability is estimated with the correspondent noisy feature vector, $y_{t_{e,ph}}^{e,ph}$, and the Gaussian is composed by the weights and covariance matrix of $s_y^{e,ph}$, and the mean vector is $\mu_{s_x^{ph}} + r_{0,s_x^{ph},s_y^{e,ph}}$.

3.2. Adjust iterative process

Once the initialization expressions for $r_{ini,s_x^{ph},s_y^{e,ph}}$ and $p_{ini}(s_x^{ph}|s_y^{e,ph})$ have been obtained, the new iteration values for the transformation terms are estimated by forced alignment normalization of noisy training feature vector technique. Given n the correspondent iteration, with $n \in [1, N]$, where N the final number of iterations, $r_{n,s_x^{ph},s_y^{e,ph}}$ and $p_n(s_x^{ph}|s_y^{e,ph})$, will be estimated with the transformations of iteration $n-1$ (or $r_{ini,s_x^{ph},s_y^{e,ph}}$ and $p_{ini}(s_x^{ph}|s_y^{e,ph})$ if $n = 1$), following the three steps treated in 3.1:

- Estimate the phoneme associated to each noisy training feature vector by a forced Viterbi alignment of the training utterance.
- Normalize noisy training feature vectors only with the transformations of the associated phonemes: $p_{n-1}(s_x^{ph}|s_y^{e,ph})$, and $r_{n-1,s_x^{ph},s_y^{e,ph}}$.
- Estimate the new transformations with PD-MEMLIN stereo data training expressions (11), (13), where clean training data will be the normalized noisy training feature vectors.

Several experiments were carried out in order to study the behavior of forced alignment normalization of noisy training feature vector technique, and EM algorithm. The forced segmentation based algorithm converges very quickly, but the local maximum provided and the recognition results are not as good as in the EM training process. Since the EM convergence is found to be slower, a joint solution is the alternative here proposed. In the joint approach, the final transformations can be trained in a small number of iterations, obtaining good enough results as we will discuss in section 5. The joint solution enjoys the initial values of the EM algorithm, thus the parameters after the EM iteration are near a better maximum than initializing with forced alignment. Then the fast reestimation of the parameters is done thanks to the forced segmentation training recursion, leading the transformation parameters to the local maximum.

4. COEFFICIENT PROGRESSIVE PD-MEMLIN

Since the first coefficients of MFCC feature vectors are more important than the last ones in speech recognition, and in order to reduce the computational cost of the algorithm, the Coefficient Progressive PD-MEMLIN algorithm proposes normalizing only the first coefficients of the feature vectors, while the other ones are not compensated.

Note that this algorithm represents an important saving of additions and data storage. If c coefficients are not compensated, the Data Storage Saving, DSS, will be

$$DSS = N_{ph}N_{s_x^{ph}}N_{s_y^{ph}}N_e \cdot c, \tag{20}$$

where N_{ph} is the number of phonemes, $N_{s_x^{ph}}$ is the number of clean Gaussians for each phoneme, $N_{s_y^{ph}}$ is the number of noisy Gaussians for each phoneme and environment, and N_e is the number of environments. On the other hand, the Addition Saving, AS, will be $AS = DSS \cdot N_f$, where N_f is the number of frames which needs to be normalized.

5. RESULTS

A set of experiments have been carried out using the Spanish SpeechDat Car database [9]. Seven environments are defined: car stopped, motor running (E1), town traffic, windows close and climatizer off (silent conditions) (E2), town traffic and noisy conditions: windows open and/or climatizer on (E3), low speed, rough road, and silent conditions (E4), low speed, rough road, and noisy conditions (E5), high speed, good road, and silent conditions (E6), and high speed, good road, and noisy conditions (E7).

The task used is isolated and continuous digits. All the utterances are 16 KHz sampled. The clean signals (Ch0) are recorded with a close talk microphone (Shune SM-10A), and the noisy signals (Ch2) are recorded by a microphone placed on the car ceiling in front of the driver (Peiker ME15/V520-1). The SNR range for the clean signals goes from 20 to 30 dB, and for the noisy signals goes from 5 to 20 dB. 12 MFCC and energy are computed each 10 ms using a 25 ms Hamming window.

Train	Test	E1	E2	E3	E4	E5	E6	E7	MWER (%)
Ch0	Ch0	1.90	2.64	1.81	1.75	1.62	0.64	0.35	1.75
Ch0	Ch2	5.91	14.49	14.55	20.17	21.07	16.19	35.71	16.21
Ch2	Ch2	6.67	14.24	12.73	12.91	14.97	9.68	8.50	11.81

Table 1. WER baseline results, in % for different conditions of training and testing.

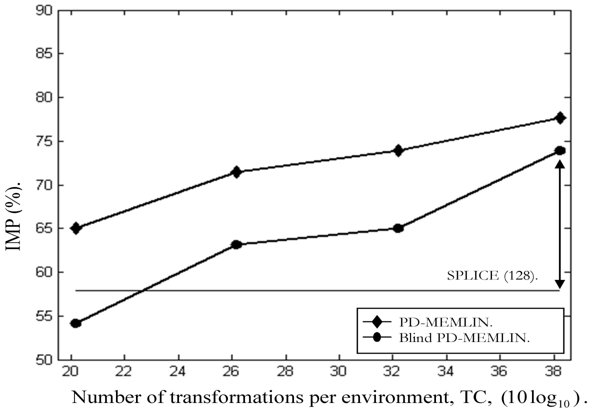

Fig. 2. Average improvement, in %, for seven environments for PD-MEMLIN, and blind PD-MEMLIN in function of the number of transformations per environment (SPLICE algorithm with 128 Gaussians for noisy space model is included to compare).

	MWER (%)	IMP (%)
PD-MEMLIN	5.30	77.67
Blind PD-MEMLIN	5.74	73.96
CPPD-MEMLIN	5.15	78.17

Table 2. Best mean WER and improvement for PD-MEMLIN, blind PD-MEMLIN, and CPPD-MEMLIN, in %.

The feature normalization techniques are applied over the 12 MFCC and delta energy, and the different used models have 26 Spanish phonemes with 2, 4, 8, or 16 gaussians for each one.

For recognition, the feature vector is composed of the 12 normalized MFCC with cepstral mean substraction, the first and second derivative and the normalized delta energy, given a feature vector of 37 coefficients. The phonetic acoustic models are composed of 25 three state continuous density HMM with 16 Gaussians per state to model Spanish phonemes and 2 silence models for long and interword silences.

The Word Error Rate, WER, baseline results for each environment are presented in Table 1. MWER represents the Mean WER, computed proportionality to the number of utterances of each environment

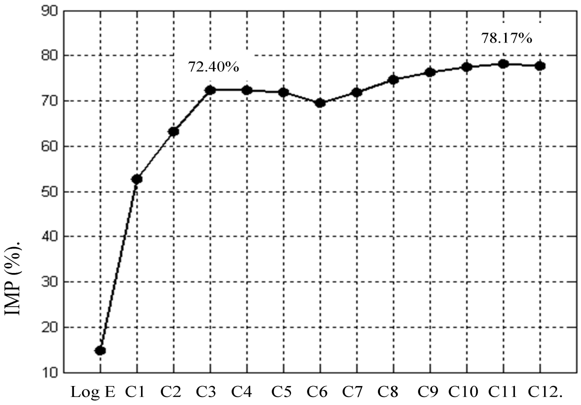

Fig. 3. Average improvement, in %, for the seven environments for CPPD-MEMLIN in function of the last normalized coefficient of the feature vectors.

$$MWER = \frac{\sum_e WER_e N_u^e}{\sum_e N_u^e}, \qquad (21)$$

where WER_e is the WER obtained for e environment and N_u^e is the number of utterances of e environment.

In order to compare PD-MEMLIN and blind PD-MEMLIN, the Transformation Cost per environment, TC, is defined as: $TC = 10log_{10}(N_{ph}N_{s_x^{ph}}N_{s_y^{ph}})$.

The comparative results between PD-MEMLIN and blind PD-MEMLIN are shown in Fig. 2, where it is presented the average improvement, IMP, which has been calculated with the improvement of each environment and proportionality to the number of utterances of each environment in a similar way as MWER. The best IMP and MWER are included in Table 2. In order to compare, the values for SPLICE [6] with 128 Gaussians for noisy model and stereo data are included, too. It can be observed that the results with blind PD-MEMLIN are close to PD-MEMLIN, especially when the number of transformations per environment is high.

The results obtained with CPPD-MEMLIN are shown in Fig. 3, where it is presented the average improvement, IMP, with 16 Gaussians per phoneme for noisy and clean models. The best average IMP and MWER are included in Table 2. It can be observed that only with 4 normalized coefficients, an improvement of 72.40% is obtained, not very far from the result with 13 normalized coefficients and stereo data training. In this case, the number addition saving is 69.23%, concerning classic PD-MEMLIN normalizing the 13 coefficients.

6. CONCLUSIONS

In this paper we have presented two approaches of PD-MEMLIN: blind PD-MEMLIN, which does not need stereo data to estimate the transformations, and Coefficient Progressive PD-MEMLIN, CPPD-MEMLIN, which only normalizes a subset of coefficients of the feature vectors. The advantages of them concerning classic PD-MEMLIN (not to use stereo data and a less number of operations), do not produce an important degradation in WER. So, classic PD-MEMLIN obtains an average improvement of 77.67%, whereas blind PD-MEMLIN reaches an improvement of 73.96%, and CPPD-MEMLIN increases it until 78.17% if 12 MFCC coefficients are normalized, but with only 4 normalized coefficients, it already obtains important results (72.40%).

7. REFERENCES

[1] A. Acero and X. Huang, "Augmented cepstral normalization for robust speech recognition," *Proc. of IEEE Automatic Speech Recognition Workshop*, pp. 146–147, Dec. 1995.

[2] A. Sankar and C. Lee, "A maximum-likelihood approach to stochastic matching for robust speech recognition," *IEEE Trans. on Speech and Audio Processing*, pp. 190–202, May 1996.

[3] R. M. Stern, B. Raj, and P. J. Moreno, "Compensation for environmental degradation in automatic speech recognition," *in Proc. ESCA Tutorial and Research Workshop on Robust Speech Recognition for Unknown Communication Channels, Apr, 1997, pp. 33-42*.

[4] M. J. F. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," *Computer Speech and Language, Vol 12*, 1998.

[5] A. Acero, "Acoustical and Environmental Robustness in Automatic Speech Recognition," *Ph.D. Thesis CMU.*, Sep. 1990.

[6] J. Droppo, L. Deng, and A. Acero, "Evaluation of the splice algorithm on the aurora2 database," *in Proc. Eurospeech*, vol. 1, Sep. 2001.

[7] L. Buera, E. Lleida, A. Miguel, and A. Ortega, "Multi-environment models based linear normalization for speech recognition in car conditions," *in Proc. ICASSP*, May. 2004.

[8] L. Buera, E. Lleida, A. Miguel, and A. Ortega, "Robust speech recognition in cars using phoneme dependent multi-environment linear normalization," *in Proc. INTERSPEECH*, Sep. 2005.

[9] A. Moreno, A. Noguiera, and A. Sesma, "Speechdat-car: Spanish," *Technical Report SpeechDat*.