MULTI-ENVIRONMENT MODELS BASED LINEAR NORMALIZATION FOR SPEECH RECOGNITION IN CAR CONDITIONS

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

University of Zaragoza, Spain
{lbuera,lleida,amiguel,ortega}@unizar.es

ABSTRACT

In this paper a multi-environment adaptation technique based on minimum mean squared error estimation is proposed. MEMLIN, Multi-Environment Models based LInear Normalization, consists on a feature adaptation using stereo data and several basic defined environments. The target of this algorithm is to learn the difference between clean and noisy feature vectors associated to a pair of gaussians (one for a clean model, and the other one for a noisy model), for each basic environment. This knowledge, the gaussians associated, the conditional probability between clean and noisy gaussians, and the environments are the data used to compensate the mismatch between clean and noisy vectors. This algorithm obtains important improvements regarding other techniques that look for similar targets. The experimental results with the SpeechDat Car database shows an average improvement of more than 68 %, concerning the baseline, over 7 different defined environments.

1. INTRODUCTION

It is well known that changes between the testing and training environments deteriorate the performance of the speech recognition systems. Many algorithms have been developed to compensate the environment mismatch, but all of them can be grouped into two rough categories [1]: feature compensation or normalization, that modifies the feature vectors, and model adaptation, in which the acoustic model parameters are adjusted. Hybrid techniques exist [2], and they have proved to be effective. The use of one or other sort of algorithms depends on the circumstances: normalization needs less data and time to compensate than model adaptation, whereas model adaptation can be more specific [3].

Feature compensation algorithms based on Minimum Mean Squared Error, MMSE, estimation constitute a very important research line. Techniques like multivariate gaussian based cepstral normalization algorithm, RATZ [4] and Stereo based Piecewise LInear Compensation for Environments, SPLICE [5] are a good example of MMSE based feature compensation. In this paper a multi-environment models based linear normalization, MEMLIN, is proposed and compared against a SPLICE version for multi-enviroments, named here SPLIC-ME. MEMLIN introduces a correction factor which dependents on a clean and noisy gaussian models and the conditional probability of the clean model given the noisy model and the noisy vector. MEMLIN learns one transformation vector for each pair of clean and noisy gaussians, however SPLICE defines only a transformation vector for each noisy gaussian.

This paper is organized as follows: in section 2, the MMSE estimator is presented, and the expression for SPLICE and MEMLIN are obtained pointing out the main differences between both techniques. The calculation of the different parameters needed in the estimator is studied in section 3 for the different algorithms. The results with SpeechDat Car database [6] are presented and discussed in the section 4. Finally, the conclusions are included in section 5.

2. MMSE ESTIMATOR

Given the clean feature vector x, and the noisy one, y, the clean estimation vector, $\hat{x}$, can be calculated by MMSE estimation:

$$\hat{x} = E[x|y] = \int_x xp(x|y)dx \tag{1}$$

The problem is how the probability density function (PDF) of x given y, $p(x|y)$, can be obtained. In order to calculate it, some approximations can be applied. The kind of algorithm and the final results depend on these assumptions.

MEMLIN and SPLIC-ME suppose that noisy feature vector follows the distribution of mixture of gaussians for each basic environment:

$$p_e(y) = \sum_{s_y^e} p(y|s_y^e)p(s_y^e) \tag{2}$$

This work has been supported by MCyT under contract TIC2002-04103-C03

$$p(y|s_y^e) = N(y; \mu_{s_y^e}, \Sigma_{s_y^e}) \quad (3)$$

Where e represents the environment index, s_y^e denotes the correspondent gaussian of the noisy model for the e environment, $\mu_{s_y^e}$, $\Sigma_{s_y^e}$, and $p(s_y^e)$ are the mean vector, the diagonal covariance matrix, and the weight associated to s_y^e.

MEMLIN assumes that clean feature vector model follows the distribution of mixture gaussians:

$$p(x) = \sum_{s_x} p(x|s_x)p(s_x) \quad (4)$$

$$p(x|s_x) = N(x; \mu_{s_x}, \Sigma_{s_x}) \quad (5)$$

Where s_x denotes the correspondent gaussian of the clean model, μ_{s_x}, Σ_{s_x}, and $p(y|s_x)$ are the mean, diagonal covariance matrix, and the weight associated to s_x.

On the one hand, MEMLIN approximates the PDF of x given y, s_y^e, and s_x, as gaussian whose covariance matrix, Σ_{s_x,s_y^e}, depends on s_x, and s_y^e, and the mean vector is a linear transformation of the noisy vector that depends on s_y^e, s_x, and α_e, which is the weight associated to each environment. r_{s_x,s_y^e} is called the transformation vector, and represents the difference between clean and noisy data given a clean model gaussian, and a noisy model one of an environment:

$$p(x|y, s_y^e, s_x) = N(x; y - \sum_e \alpha_e r_{s_x,s_y^e}, \Sigma_{s_x,s_y^e}) \quad (6)$$

On the other hand, SPLIC-ME assumes that the PDF of x given y, and s_y^e, is gaussian whose covariance matrix, $\Sigma_{s_y^e}$, depends on s_y^e, and the mean vector, is a linear transformation of the noisy vector that depends on s_y^e, and α_e. $r_{s_y^e}$ is, in this case, the transformation vector:

$$p(x|y, s_y^e) = N(x; y - \sum_e \alpha_e r_{s_y^e}, \Sigma_{s_y^e}) \quad (7)$$

Approximating x for the mean of (6) or (7), according to the algorithm, (1) can take the following forms for MEMLIN (8), and SPLIC-ME (9):

$$\hat{x}_t \simeq y_t - \sum_{s_x} \sum_e \sum_{s_y^e} \alpha_{e,t} r_{s_x,s_y^e} p(s_y^e|y_t) p(s_x|s_y^e, y_t) \quad (8)$$

$$\hat{x}_t \simeq y_t - \sum_e \sum_{s_y^e} \alpha_{e,t} r_{s_y^e} p(s_y^e|y_t) \quad (9)$$

Where t is a temporal index, $p(s_y^e|y_t)$ is the probability of s_y^e given y_t, and $p(s_x|s_y^e, y_t)$ is the probability of the clean model gaussian given the noisy one, and y_t.

If the environment is known, it can be considered that there is only one environment and the index e can be avoided in the expressions before. This modification in MEMLIN equations is what we have defined as Multivariate Model based Cepstral Normalization, MMCN, and the transformation of SPLIC-ME is SPLICE [5].

The use of several environments in MEMLIN and SPLIC-ME is a great advantage from the single environment techniques (MMCN and SPLICE). If the environments are well defined and cover the main part of the feature space, it is very difficult to find noisy phrases that only belong to one environment, and a linear combination of environments is a better representation of the phrase. In this sense, to consider the clean vector estimation as a multi-environment linear combination brings better results.

On the other hand, the use of clean and noisy models in the MEMLIN and MMCN adaptation is another advantage concerning to SPLIC-ME and SPLICE. Real contamination produces a nonlinear shift [4] over the clean feature vectors. So, clean vectors associated to a certain gaussian of the clean model do not always have the same associated gaussian in noisy speech model when they are contaminated. MEMLIN and MMCM model this effect by using a conditional probability model between noisy and clean gaussians, $p(s_x|s_y^e, y_t)$. So, defining a transformation vector for each noisy gaussian (this is what SPLICE and SPLIC-ME do), is not the best solution. However, MEMLIN and MMCN learn one transformation vector for each pair of gaussians (one for the clean model, and the other one for the noisy model). It is reasonable to think that MEMLIN, which benefits from the use of several environments and two models, will obtain better recognition results than MMCN, SPLIC-ME, or SPLICE.

3. MMSE PARAMETERS ESTIMATION

In order to calculate $\hat{x}_t$, it is necessary to estimate: $\alpha_{e,t}$, and $p(s_y^e|y_t)$ for MEMLIN and SPLIC-ME, which have to be calculated each time instant with each noisy feature vector we want to normalize, and r_{s_x,s_y^e}, $p(s_x|s_t^e, y_t)$, for MEMLIN, and $r_{s_y^e}$, for SPLIC-ME, which need a training process with stereo data for each environment.

In order to calculate $\alpha_{e,t}$, an iterative solution is considered. Each moment, t, a noisy feature vector is available, y_t. The calculation of the environment weight in this moment will be:

$$\alpha_{e,t} = \beta \cdot \alpha_{e,t-1} + (1 - \beta) \frac{p_e(y_t)}{\sum_e p_e(y_t)} \quad (10)$$

Where β is the memory constant. $\alpha_{e,0}$ for all environments are considered uniform. Also, $p(s_y^e|y_t)$ can be calculated using (3) and Bayes:

$$p(s_y^e|y_t) = \frac{p(y_t|s_y^e)p(s_y^e)}{\sum_{s_y^e} p(y_t|s_y^e)p(s_y^e)} \quad (11)$$

The available stereo data for each environment for the training process are: $X_e = \{x_1^e, ..., x_{T_e}^e\}$, for clean feature

	E1	E2	E3	E4	E5	E6	E7	MWER
C0-C0	1.90	2.64	1.81	1.75	1.62	0.64	0.35	1.75
C0-C2	5.91	14.49	14.55	20.17	21.07	16.19	35.71	16.21
C2-C2	10.39	19.38	16.78	16.41	17.73	13.65	9.86	15.56

Table 1. WER baseline results

vectors and $Y_e = \{y_1^e, ..., y_{T_e}^e\}$ for noisy ones. With these data, r_{s_x,s_y^e}, and $r_{s_y^e}$ can be obtained with the Maximum Likelihood algorithm, ML. The maximization function is (12) for MEMLIN and (13) for SPLIC-ME, and the optimal solutions using the Expectation Maximization algorithm EM [7] are (14), and (15).

$$L(Y_e) = \sum_{t_e} \log(\sum_{s_y^e} p(s_y^e) N(y; \mu_{s_y^e} + r_{s_x,s_y^e}, \Sigma_{s_x,s_y^e})) \tag{12}$$

$$L(Y_e) = \sum_{t_e} \log(\sum_{s_y^e} p(s_y^e) N(y; \mu_{s_y^e} + r_{s_y^e}, \Sigma_{s_y^e})) \tag{13}$$

$$r_{s_x,s_y^e} = \frac{\sum_{t_e} p(s_x|x_{t_e}^e) p(s_y^e|y_{t_e}^e)(y_{t_e}^e - x_{t_e}^e)}{\sum_{t_e} p(s_x|x_{t_e}^e) p(s_y^e|y_{t_e}^e)} \tag{14}$$

$$r_{s_y^e} = \frac{\sum_{t_e} p(s_y^e|y_{t_e}^e)(y_{t_e}^e - x_{t_e}^e)}{\sum_{t_e} p(s_y^e|y_{t_e}^e)} \tag{15}$$

Where $t_e = (1, ..., T_e)$. $p(s_x|x_{t_e}^e)$ is the probability of s_x given the clean feature vector. It can be calculated in a similar way of (11).

The conditional probability, $p(s_x|s_y^e, y_t)$, can be estimed with the training phrases set by relative frecuency. For each stereo pair of vectors, the most probable pair of gaussians is obtained. After that, the conditional probability model between gaussians can be obtained:

$$p(s_x|s_y^e, y_t) = \frac{C_N(s_x|s_y^e)}{N} \tag{16}$$

where $C_N(s_x|s_y^e)$ is the number of times that the most probable pair of gaussians is s_x, and s_y^e. N is the number of times that the most probable gaussian for noisy vector is s_y^e.

The expressions for MMCN and SPLICE can be obtained from (11), (14), and (15), avoiding the e index.

4. RESULTS

A set of experiments have been carried out using the Spanish SpeechDat Car database [6]. 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 phrases are 16 KHz sampled. The clean signals are recorded with a close talk microphone (Shune SM-10A), which we will call C0, and the noisy signals are recorded by a microphone placed on the car ceiling in front of the driver (Peiker ME15/V520-1): it is called C2. 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.

The feature normalization techniques are applied over the 12 MFCC and delta energy. The clean and noisy models are built for these feature vectors with 8, 16, or 32 gaussians.

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 silent models for long and interword silents.

The baseline results for each environment are presented in table 1. C0-C0 represents training and testing with clean phrases, C0-C2 represents training with clean phrases and testing with noisy ones. C2-C2 represents the results with noisy signals and models, when they have been trained with all the environments. It can be seen that C2-C2 obtains better results than C0-C2 in most noisy environments, and worse when the environment is not as noisy. This is the compromise of the all environment noisy models.

The comparative results between the different techniques can be seen in table 2. Next to the technique, appears the number of model gaussians, 8, 16 or 32 (in MMCN and MEMLIN the first number represents the number of clean model gaussians, and the second is for the noisy one). The improvement is calculated between C0-C0 and C0-C2, and the mean of the improvement (MIMP) and the mean error rate (MWER) are presented in table 2 . MEMLIN and SPLIC-ME use all environments to normalize (E1,...,E7).

SPLICE obtains an improvement around 50% with 32 gaussians. Otherwise, MMCN obtains better results than SPLICE for each environment, except E6 and E3. The improvement, in mean, is always bigger for the same number of gaussians than SPLICE, obtaining almost

	E1	E2	E3	E4	E5	E6	E7	MWER	MIMP
SPLICE 8	4.79	10.10	7.83	11.15	15.82	11.75	15.64	10.51	37.24
SPLICE 16	4.75	9.86	7.83	8.90	14.11	8.25	13.50	9.33	43.96
SPLICE 32	4.70	9.50	6.29	8.77	11.44	7.46	12.92	8.42	49.60
MMCN 8-8	4.79	8.06	8.39	11.53	14.11	11.11	15.31	9.81	42.06
MMCN 16-16	3.64	8.49	7.55	8.27	11.15	8.57	13.50	8.21	54.74
MMCN 32-32	3.35	8.74	6.57	7.64	9.89	7.94	12.58	7.65	58.85
SPLIC-ME 8	3.16	8.74	6.15	9.27	12.58	9.21	15.65	8.58	54.99
SPLIC-ME 16	3.45	8.23	5.87	7.77	10.77	7.78	13.95	7.71	58.94
SPLIC-ME 32	2.59	7.98	6.15	7.52	9.34	6.67	12.59	7.04	65.56
MEMLIN 8-8	3.16	8.49	6.43	9.27	11.91	9.05	14.97	8.39	56.00
MEMLIN 16-16	3.26	8.06	5.45	7.64	10.01	7.78	12.92	7.37	61.49
MEMLIN 32-32	2.49	7.80	5.03	6.64	9.25	6.51	11.22	6.62	68.50

Table 2. WER results with SPLICE, MMCN, SPLIC-ME, and MEMLIN techniques

59% with 32 gaussians. This improvement is produced by the use of the conditional probability model between noisy and clean gaussians in MMCN. The results of MEMLIN and SPLIC-ME are better than MMCN or SPLICE; this is because sometimes, test feature vectors from an environment can be better represented by a linear combination of several environments. The mean WER with MEMLIN is always better than SPLIC-ME for the same number of gaussians. These results show the improvement that can be obtained by using multi-environment models and modeling the conditional probability between noisy and clean gaussians.

5. CONCLUSIONS

In this paper we have presented a two multi-environment normalization technique using the MMSE estimator, MEMLIN, and SPLIC-ME, which is a derived version of SPLICE for multi-environments. MMCN, a simplification for controlled environments of MEMLIN has been also presented. Experiments have been carried out with the Spanish SpeechDat Car database. Seven different environments have been defined according to the acoustic conditions. Important error rate reductions have been obtained with all the normalization techniques used. Multi-environment techniques, MEMLIN and SPLIC-ME, reduce the error rate in more than 10% regarding the controlled environment techniques, MMCN and SPLICE. Also, modeling the conditional probability between noisy and clean gaussians reduce the error rate in more than 9% for controlled environments and a 3% in multi-environments. A global improvement of 68.5% in the error rate is finally obtained with MEMLIN using 32 gaussians to model noisy and clean speech.

6. REFERENCES

[1] S. Sagayama, K. Shimoda, M. Nakai, and H. Shimodaira, "Analytic methods for acoustic model adaptation: a review," *in Proc. Isca ITR-Workshop2001*, pp. 67–76, Aug 2001.

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

[4] P. Moreno, "Speech recognition in noisy environments," *Ph. D. Thesis*, ECE Department, Carnegie-Mellon University. Apr. 1996.

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

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

[7] J. Bilmes, "A gentle tutorial on the EM algorithm and its application to parameter estimation for gaussian mixture and hidden markov models," University of Berkeley, ICSI-TR-97-021, 1997.