ENVIRONMENTAL ADAPTATION BASED ON FIRST ORDER APPROXIMATION[++]

C. Cerisara[+], *L. Rigazio*[*], *R. Boman*[*] *and J.-C. Junqua*[*]
[+] LORIA UMR 7503
Campus Scientifique BP 239 - F54506 Vandœuvre-lès-Nancy, France
[*] Panasonic Speech Technology Laboratory
3888 State St., Suite 202, Santa Barbara, CA, 93105, USA

ABSTRACT

In this paper, we propose an algorithm that compensates for both additive and convolutional noise. The goal of this method is to achieve an efficient environmental adaptation to realistic environments both in terms of computation time and memory. The algorithm described in this paper is an extension of an additive noise adaptation algorithm presented in [1]. Experimental results are given on a realistic database recorded in a car. This database is further filtered by a low pass filter to combine additive and channel noise. The proposed adaptation algorithm reduces the error rate by 75 % on this database, when compared to our baseline system without environmental adaptation.

1. INTRODUCTION

Automatic speech recognition systems must face the problem of unknown testing environments. Three approaches might be used:

1. Training speech models on a training corpus that is representative of most of the possible testing environments;
2. Training speech models on “clean” conditions, and adapting these models to any new detected environment.
3. Enhancing the speech signal by removing the noise.

The first solution is very difficult to achieve, as it requires a very large training corpus. Such a corpus would be difficult to collect. Furthermore, the resulting models would be very large to encode such an important quantity of information and it would be computationally expensive to use them. The second solution builds lighter models, but rely heavily on the adaptation stage. The adaptation method must first correctly estimate the testing environment and then accurately adapt the models to it. The method described in this paper belongs to this second set of solutions.

Like most of the related works, we assume the following simplified model of the environment:

$$Z = H \cdot S + N \qquad \text{(Eq 1)}$$

Equation 1 describes the corruption of the original clean speech signal S by a convolutional (or channel) noise vector H and an additive noise vector N in the spectral domain. Let N_{tar} and N_{ref} be the target (or testing) and reference (or training) additive noises, and H_{tar} and H_{ref} the target and reference channel noises.

Based on an estimation of the additive and channel bias $N_{tar} - N_{ref}$ and $H_{tar} - H_{ref}$, the method proposed in this paper adapts the models to the target environment using a first order approximation of equation 1 in the cepstral domain. In a previous paper [1], we described a low-cost adaptation method for additive noise only. In this work, we extend this method to joint additive and channel noise adaptation.

We briefly review the additive noise adaptation method in section 2 and then extend its principle to channel noise in section 3. Section 4 presents some experimental results and section 5 concludes the paper.

2. ADDITIVE NOISE ADAPTATION

Let $C(S)$ be the function that transforms the spectral vector S into the cepstral domain, and $f(C(S))$ the adaptation function that transforms the signal (or model) from the reference to the target environment.

When only additive noise is considered, f is equal to:

$$f(C(S+N_{ref})) = C\left(C^{-1}\left(C(S+N_{ref})\right) + N_{tar} - N_{ref}\right)$$

2.1. Jacobian adaptation

Jacobian adaptation [4] computes the Jacobian approximation of $f(C(S+N_{ref})) = C(S+N_{tar})$:

$$C(S+N_{tar}) = C(S+N_{ref}) + \frac{\partial C(S+N_{ref})}{\partial C(N_{ref})}\left(C(N_{tar}) - C(N_{ref})\right)$$

[++] This work was done while the first author was at Panasonic Speech Technology Laboratory.

The main goal of Jacobian adaptation is thus to replace the non-linear function f by a computationally less expensive linear adaptation function. Indeed, speech models are often composed of more than 20000 Gaussian densities. After each new estimate of the target environment, all these densities have to be adapted. The resulting cost might thus considerably be reduced when using a linear function instead of f.

Experimental results reported in [4] as well as in [1] show that Jacobian adaptation can provide good results for realistic testing environments.

2.2. Proposed algorithm

In [1] we propose to use another linear approximation of f than the Jacobian one. The basic principle of our method is to parameterize the set of possible linear adaptations and to select the best one, by training the chosen parameter on a development environment which is as close as possible to the real testing environment[1].

The chosen set of parameterized linear adaptations is defined by the following adaptation equations:

$$C(S+N_{tar}) = C(S+N_{ref}) + F \cdot \frac{\alpha N_{ref}}{S+\alpha N_{ref}} \cdot F^{-1}\left(C(N_{tar}) - C(N_{ref})\right)$$

where the parameter is α and F is the Discrete Cosine Transform (DCT) matrix.

Such an algorithm is a generalization of the Jacobian adaptation, because the exact Jacobian adaptation is obtained when the development environment is close to the training environment. In such a case, $\alpha = 1$ and the linear adaptation matrix is the Jacobian matrix:

$$J_S = F \cdot \frac{N_{ref}}{S+N_{ref}} \cdot F^{-1}$$

The only problem of the method may come from the mismatch between the development and the real testing environments. However, this mismatch is especially important when the development environment is “clean”, i.e. close to the training conditions. Even in such a case, experiments reported in [4] suggest that adaptation greatly improves the recognition accuracy. Our idea was thus to reduce this mismatch, by using a more realistic development environment. Consequently, our goal is that the accuracy provided by our adaptation scheme should be greater than the Jacobian one, at no extra cost.

Experimental results reported in [1] confirm this hypothesis, and further suggest that the dependency between the resulting adaptation and the development environment is very weak. It is then possible to use the same linear adaptation for every possible testing environment – even a clean one. We have also proposed a method in [1] to further reduce the computational and memory costs of the adaptation.

3. EXTENSION TO CHANNEL ADAPTATION

3.1. Environmental adaptation equation

Let us now consider that the clean speech signal is corrupted by both additive and convolutional noise. The first order approximation, when applied to the corrupted speech signal, gives:

$$\begin{aligned} C(H_{tar}S+N_{tar}) = {} & C\left(H_{ref}S+N_{ref}\right) + \\ & \frac{\partial C\left(H_{ref}S+N_{ref}\right)}{\partial C(N_{ref})}\left(C\left(N_{tar}\right) - C\left(N_{ref}\right)\right) + \\ & \frac{\partial C\left(H_{ref}S+N_{ref}\right)}{\partial C(H_{ref})}\left(C\left(H_{tar}\right) - C\left(H_{ref}\right)\right) \end{aligned}$$

The term $\frac{\partial C(H_{ref}S+N_{ref})}{\partial C(N_{ref})}$ is the Jacobian matrix J_S. Similarly, we can compute:

$$\begin{aligned} \frac{\partial C(H_{ref}S+N_{ref})}{\partial C(H_{ref})} = {} & \frac{\partial C(H_{ref}S+N_{ref})}{\partial \log(H_{ref}S+N_{ref})} \cdot \\ & \frac{\partial \log(H_{ref}S+N_{ref})}{\partial (H_{ref}S+N_{ref})} \cdot \frac{\partial (H_{ref}S+N_{ref})}{\partial (H_{ref})} \cdot \\ & \frac{\partial (H_{ref})}{\partial \log(H_{ref})} \cdot \frac{\partial \log(H_{ref})}{\partial C(H_{ref})} \end{aligned}$$

which gives:

$$\frac{\partial C(H_{ref}S+N_{ref})}{\partial C(H_{ref})} = F \cdot \frac{1}{H_{ref}S+N_{ref}} \cdot S \cdot H_{ref} \cdot F^{-1}$$

As the vector multiplication $S \cdot H_{ref}$ is commutative, we get:

$$\frac{\partial C(H_{ref}S+N_{ref})}{\partial C(H_{ref})} = I - J_S$$

where I is the identity matrix. Thus,

[1] Three corpora are used in our method: the training corpus on which the HMMs are trained, the development corpus on which α is trained, and the testing corpus.

$$C(H_{tar}S+N_{tar})=C(H_{ref}S+N_{ref})+ \\ J_S\left(C(N_{tar})-C(N_{ref})\right)+ \\ (I-J_S)\left(C(H_{tar})-C(H_{ref})\right) \qquad \text{(Eq 2)}$$

Note that we have described the adaptation equation with the Jacobian approximation, but it is of course possible to use the linear adaptation defined in section 2 as well.

3.2. Estimation of the environmental bias

Equation 2 uses two additive and convolutional bias, respectively $C(N_{tar})-C(N_{ref})$ and $C(H_{tar})-C(H_{ref})$.

The additive bias is estimated during the background segments of the speech signal. For the following experiments, we are using the first 150 ms of each sentence, as these segments do not contain any speech signal.

We propose two solutions to estimate the channel bias. Both of them assume that the alignment between the frames of the signal and the models is known. In practice, we have computed this alignment on the previous sentence, using the previously adapted models. This method implies that the models used to recognize the current sentence are adapted to the channel based on the previous sentence. We have then to assume that the channel noise does not vary very much from one sentence to another. This is a very strong hypothesis, but it allows us to adapt the models without any additional pass on the signal. Thus, real-time implementation of the adaptation algorithm can still be realized. Let us now describe the two proposed solutions:

1. ***First solution***

If we assume that $H_{tar}S >> N_{tar}$, then averaging the speech frames of one sentence of the test corpus gives:

$$\hat{C}(S_{tar})=\frac{1}{T}\sum_{t=1}^{T}C(H_{tar}\cdot S_t)=C(H_{tar})+\frac{1}{T}\sum_{t=1}^{T}C(S_t)$$

where T is the number of speech frames of the sentence. Similarly, we can average the Gaussian means of the models aligned with the same speech frames:

$$\hat{C}(S_{ref})=\frac{1}{T}\sum_{t=1}^{T}C(H_{ref}\cdot S_t)=C(H_{ref})+\frac{1}{T}\sum_{t=1}^{T}C(S_t)$$

By subtracting the two previous equations, we get:

$$\hat{C}(S_{tar})-\hat{C}(S_{ref})=C(H_{tar})-C(H_{ref})$$

2. ***Second solution***

If we do not want to assume that $H_{tar}S >> N_{tar}$, then:

$$\hat{C}(S_{tar})=\frac{1}{T}\sum_{t=1}^{T}C(H_{tar}\cdot S_t+N_{tar}) \\ =\frac{1}{T}\sum_{t=1}^{T}C\left(H_{tar}\cdot\left(S_t+\frac{N_{tar}}{H_{tar}}\right)\right)$$

which simplifies into:

$$\hat{C}(S_{tar})=C(H_{tar})+\frac{1}{T}\sum_{t=1}^{T}C\left(S_t+\frac{N_{tar}}{H_{tar}}\right)$$

The adapted models can be decomposed as follows:

$$C(H_{tar}\cdot S+N_{tar})=C\left(\frac{H_{tar}}{H_{ref}}\right)+C\left(H_{ref}\cdot S+\frac{H_{ref}}{H_{tar}}\cdot N_{tar}\right)$$

where $C\left(H_{ref}\cdot S+\frac{H_{ref}}{H_{tar}}\cdot N_{tar}\right)$ are the "partially adapted" models:

$$C\left(H_{ref}\cdot S+\frac{H_{ref}}{H_{tar}}\cdot N_{tar}\right)=C(H_{ref}\cdot S+N_{ref})+ \\ J_S\left(C(N_{tar})-C(N_{ref})\right)-J_S\left(C(H_{tar})-C(H_{ref})\right)$$

These "partially adapted" models can thus be computed as intermediate models during the adaptation process. Instead of averaging the adapted models, we can average these intermediate models on the speech frames of the adaptation sentence:

$$\hat{C}(S_{ref})=\frac{1}{T}\sum_{t=1}^{T}C\left(H_{ref}\cdot S_t+\frac{H_{ref}}{H_{tar}}N_{tar}\right) \\ =C(H_{ref})+\frac{1}{T}\sum_{t=1}^{T}C\left(S_t+\frac{N_{tar}}{H_{tar}}\right)$$

As previously, we can then compute:

$$\hat{C}(S_{tar})-\hat{C}(S_{ref})=C(H_{tar})-C(H_{ref})$$

Other methods to estimate the channel bias can be found in [2].

4. EXPERIMENTS

4.1. Experimental setup

The task is the recognition of digit strings. The database used has been recorded at Panasonic Speech Technology Laboratory (PSTL). The training corpus is composed of 3803 sentences recorded by 80 speakers. Three testing corpora, composed of approximately 500 digit sequences each, have been recorded by 20 speakers: A clean corpus, a noisy one recorded in a car at 30 mph, and another one recorded in a car at 60 mph. These three corpora are then filtered by a smooth low-pass filter to simulate the effects of a convolutional noise.

Each 20 ms window of the speech signal is coded into 13 MFCC coefficients, plus 13 delta coefficients. The window shift is 10 ms. 12 context-independent digit word models are built: the numbers from one to nine, plus the models "o" and "zero", and the silence. The silence is modeled by an HMM with three emitting states, whereas all the other models use thirteen emitting states. Each state of all the HMMs uses four Gaussian densities. Recognition is performed using a simple loop grammar, with equal transition probabilities. Accuracy is computed only on the ten digits, without taking into account the silences.

Adaptation is performed only on the first 13 static mean coefficients. N_{ref} is computed using the Gaussian density of the middle state of the silence model with the highest weight.

4.2. Experimental results

In practice, the parameter α has not been trained on a development corpus, but has rather been manually chosen equal to 10. As explained in [1], this is possible because of the stability of the chosen parameterization for any value of α between 5 and 15 on any tested environment. Table 1 presents the results on the non-filtered corpora and Table 2 presents the same experiments on the three filtered corpora. The following systems are tested:

- *None* refers to the reference recognition system, without any noise adaptation;
- α*-JAC* refers to the system when only additive noise adaptation is used;
- *CMA* (Cepstral Mean Adaptation) refers to the system when only convolutional noise adaptation is used;
- α*-JAC + CMA* refers to the whole system which compensates for both additive and channel noise.

System	**Clean**	**30 mph**	**60 mph**
None	99.2%	63.2%	44.1%
α*-JAC*	99.1%	97.7%	94.5%
CMA	99.1%	71.2%	58.8%
α*-JAC+CMA*	99.1%	98.4%	95.4%

Table 1: Experimental results without channel noise

System	**Clean Filtered**	**30 mph filtered**	**60 mph filtered**
None	91.7%	39.0%	23.2%
α*-JAC*	98.5%	82.0%	70.8%
CMA	98.7%	74.2%	64.8%
α*-JAC+CMA*	98.8%	86.1%	80.6%

Table 2: Experimental results on joint additive and channel noise adaptation

5. CONCLUSION

We have proposed in this paper an extension of our additive noise adaptation method presented in [1] that takes into account both additive and channel noise. Experimental results on a database recorded at PSTL show that the resulting adaptation method is very efficient and reduces the error rate by 75%, when compared to the reference system without any adaptation algorithm and when both types of noise are present. Some of these experiments have been reproduced on the TIDIGITS corpus, and confirm the good results presented in this paper. However, experiments are still needed to compare our algorithm with other joint additive and channel noise adaptation methods, like the Vector Taylor Series approach presented in [3].

6. REFERENCES

[1] Cerisara, C., Rigazio, L., Boman, R., and Junqua, J.-C. *Transformation of Jacobian matrices for noisy speech recognition*, ICSLP'2000, Beijing, China, Vol. I, pp. 369-372, October 2000.

[2] Gales, M. *Predictive model-based compensation schemes for robust speech recognition*. Speech Communication, Vol. 25, pp. 49-74, 1998.

[3] Moreno, P. J., Raj B., and Stern, R. M. *A Vector Taylor Series Approach for Environment-Independent Speech Recognition*. ICASSP'96, pp. 733-736, 1996.

[4] Sagayama, S., Yamaguchi, Y., Takahashi, S., and Takahashi, J. *Jacobian approach to fast acoustic model adaptation*, ICASSP'97, Munich, Germany, pp. 835-838, 1997.