EFFICIENT JOINT COMPENSATION OF SPEECH FOR THE EFFECTS OF ADDITIVE NOISE AND LINEAR FILTERING

Fu-Hua Liu[1], Alejandro Acero[2], and Richard M. Stern[1]

[1]Department of Electrical and Computer Engineering
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

[2]Apple Computer, Inc.
MS 76-7F
20525 Mariani Avenue
Cupertino, CA 95014

ABSTRACT

As automatic speech recognition systems are finding their way into practical applications it is becoming increasingly clear that they must be able to accommodate a variety of acoustical environments. This paper describes two algorithms that provide robustness for automatic speech recognition systems in a fashion that is suitable for real-time environmental normalization for workstations of moderate size. The first algorithm is a modification of the previously-described SDCN and FCDCN algorithms, except that unlike these algorithms it provides computationally-efficient environmental normalization *without* prior knowledge of the acoustical characteristics of the environment in which the system will be operated. The second algorithm is a modification of the more complex CDCN algorithm that enables it to perform environmental compensation in better than real time. We compare the recognition accuracy, computational complexity, and amount of training data needed to adapt to new acoustical environments using these algorithms with several different types of headset-mounted and desktop microphones.

1. INTRODUCTION

Results of several studies have demonstrated that even automatic speech recognition systems that are designed to be speaker independent can perform very poorly when they are tested using a different type of microphone or acoustical environment from the one with which they were trained (*e.g.* [1, 2, 3]). For example, the recognition accuracy of the SPHINX speech recognition system on a speaker-independent alphanumeric task dropped from 85% correct to less than 20% correct when the close-talking Sennheiser HMD-414 microphone (CLSTLK) used in training was replaced by the omnidirectional Crown PZM6FS desktop microphone (PZM6FS) [1].

We have found that two major factors that degrade the performance of speech recognition systems using desktop microphones in normal office environments are additive noise and unknown linear filtering. We showed in [1] that simultaneous *joint* compensation for the effects of additive noise and linear filtering is needed to achieve maximal robustness with respect to acoustical differences between the training and testing environments of a speech recognition system. We described in [1] two algorithms that can perform such joint compensation, based on additive corrections to the cepstral coefficients of the speech waveform.

The first compensation algorithm, *SNR-Dependent Cepstral Normalization* (SDCN), applies an additive correction in the cepstral domain that depends exclusively on the instantaneous SNR of the signal. This correction vector equals the average difference in cepstra between simultaneous "stereo" recordings of speech samples from both the training and testing environments at each SNR of speech in the testing environment. At high SNRs, this correction vector primarily compensates for differences in spectral tilt between the training and testing environments (in a manner similar to the blind deconvolution procedure first proposed by Stockham *et al.* [4]), while at low SNRs the vector provides a form of noise subtraction (in a manner similar to the spectral subtraction algorithm first proposed by Boll [5]). The SDCN algorithm is simple and effective, but for every new acoustical environment encountered it must be calibrated with a new stereo database that contains samples of speech simultaneously recorded in the training and testing environments. In many situations such a database is impractical or unobtainable, and SDCN is clearly not able to model a non-stationary environment since only long-term averages are used.

The second compensation algorithm, *Codeword-Dependent Cepstral Normalization* (CDCN), uses EM techniques to compute ML estimates of the parameters characterizing the contributions of additive noise and linear filtering that when applied in inverse fashion to the cepstra of an incoming utterance produce an ensemble of cepstral coefficients that best match (in the ML sense) the cepstral coefficients of the incoming speech in the testing environment to the locations of VQ codewords in the training environment. Use of the CDCN algorithm improved the recognition accuracy obtained when training on the CLSTLK microphone and testing with the PZM6FS to the level observed when the system is both trained and tested on the PZM6FS. The CDCN algorithm has the advantage that it does not require *a priori* knowledge of the testing environment (in the form of stereo training data in the training and testing environments), but it is much more computationally demanding than the SDCN algorithm. Compared to the SDCN algorithm, the CDCN algorithm uses a greater amount of structural knowledge about the nature of the degradations to the speech signal in order to achieve good recognition accuracy. The SDCN algorithm, on the other hand, derives its compensation vectors entirely from empirical observations of differences between data obtained from the training and testing environments.

More recently we presented, along with several other algorithms, the *fixed CDCN* (FCDCN) algorithm [6]. FCDCN combines some of the more attractive features of the CDCN and SDCN algorithms: like SDCN, the correction factor equals the difference in cepstra between the training and testing environments, but like CDCN, the correction factor is different for different VQ codewords as well. This algorithm is also simple and efficient, and it can achieve a level of recognition accuracy comparable to that of CDCN. Unfortunately, FCDCN (like SDCN) also requires the use of a training database of simultaneously-recorded speech samples in the training and testing environments. Hence, the FCDCN algorithm also cannot adapt to unknown environments.

Table 1 compares the environmental specificity, computational complexity, and recognition accuracy of these algorithms when evaluated on the alphanumeric database described in [1]. Recognition accuracy is somewhat greater than the figures reported in [1] and [6] because the current version of SPHINX incorporates a fourth codebook which describes the second-order difference cepstrum for each speech frame. In addition, the current version of SPHINX includes between-word triphones in the phonetic models [7], while previous evaluations used a recognition system that included only within-word models.

ALGORITHM	ENVIRN. SPECIFIC?	COM-PLEXITY	ACCU-RACY
NONE	NO	NONE	31.4%
SDCN	YES	MINIMAL	72.4%
CDCN	NO	MAJOR	75.7%
FCDCN	YES	MINIMAL	78.6%

Table 1: Comparison of recognition accuracy of SPHINX with no processing and the CDCN, SDCN, and FCDCN algorithms. In each case the system was trained using the CLSTLK microphone and tested using the PZM6FS microphone. Training and testing on the CLSTLK produces a recognition accuracy of 86.9%, while training and testing on the PZM6FS produces 76.2%

The ultimate goal of a robust speech recognition system is to be able to adapt to new environments with high recognition accuracy, with low computational complexity, and without environment-specific training. The CDCN, SDCN, and FCDCN algorithms all fall short in at least one of these attributes, as the SDCN and FCDCN algorithms require environment-specific training and the CDCN algorithm is more computationally complex. In this paper we describe a new algorithm, the *blind SDCN algorithm* (BSDCN), which performs a cepstral normalization very similar to that of the SDCN algorithm, except *without* the need for specfic *a priori* training to each new microphone or acoustical environment. We then describe a new implementation of the CDCN algorithm that permits the environmental compensation to take place in real time, while the environmental parameters used to perform the compensation are computed in the background during time intervals between utterances. We compare the performance of the BSDCN algorithm and that of the "real-time" implementation of the CDCN algorithm in terms of recognition accuracy and the amount of environment-specific testing data needed to perform the compensation effectively.

2. THE BSDCN ALGORITHM

As in our previous work on environmental compensation [1, 6], we assume that the speech signal $x[m]$ is passed through an unknown linear filter $h[m]$ whose output is then corrupted by uncorrelated additive noise $n[m]$. We characterize the power spectral density (PSD) of the processes involved as

$$P_y(\omega) = P_x(\omega)\,|H(\omega)|^2 + P_n(\omega) \tag{1}$$

If we let the cepstral vectors $\mathbf{x}$, $\mathbf{n}$, $\mathbf{y}$ and $\mathbf{q}$ represent the Fourier series expansion of $\ln P_x(\omega)$, $\ln P_n(\omega)$, $\ln P_y(\omega)$ and $\ln|H(\omega)|^2$ respectively, Eq. (1) can be rewritten as

$$\mathbf{y} = \mathbf{x} + \mathbf{q} + \mathbf{r}(\mathbf{x}, \mathbf{n}, \mathbf{q}) \tag{2}$$

where the correction vector $\mathbf{r}(\mathbf{x}, \mathbf{n}, \mathbf{q})$ is given by

$$\mathbf{r}(\mathbf{x}, \mathbf{n}, \mathbf{q}) = IDFT\ \{\ln\ (1 + e^{DFT\ [\mathbf{n} - \mathbf{q} - \mathbf{x}]})\} \tag{3}$$

We can obtain an estimate $\hat{P}_y(\omega)$ of the PSD $P_y(\omega)$ from a sample function of the process $y[m]$ (*i.e.* a frame of degraded speech that is assumed to be locally stationary). If $\mathbf{z}$ represents the Fourier expansion of $\ln \hat{P}_y(f)$, our goal is to estimate the uncorrupted vectors $\mathbf{X} = \mathbf{x}_0, \ldots \mathbf{x}_{N-1}$ of an utterance given the observations $\mathbf{Z} = \mathbf{z}_0, \ldots \mathbf{z}_{N-1}$.

In the original SDCN algorithm, it was assumed that the correction vector depends only on $\mathbf{z}[0] - \mathbf{n}[0]$ (*i.e.* that we can apply an average correction to all spectral shapes with the same SNR), and an estimate for $\hat{\mathbf{x}}$ was obtained by the expression

$$\hat{\mathbf{x}} = \mathbf{z} - \mathbf{w}(SNR) \tag{4}$$

This procedure subtracts from the observed vector $\mathbf{z}$ a correction $\mathbf{w}$ that depends only on the instantaneous SNR of the observed signal, $\mathbf{z}[0] - \mathbf{n}[0]$. In the original SDCN algorithm these compensation vectors $\mathbf{w}(SNR)$ were estimated by computing the average difference between cepstral vectors from the training and testing environments, and they must be "calibrated" by collecting long-term statistics from a database containing these simultaneously-recorded speech samples.

2.1. The Blind SDCN Algorithm

In the BSDCN algorithm the need for stereophonic data is circumvented by lumping all data at each SNR together. A correspondence is established between SNRs in the training and testing environments by use of traditional nonlinear warping techniques [8] on histograms of SNRs from each of the two environments. The histograms of SNR values are first normalized for equal area, to avoid having the mapping be dominated by the environment from which more data had been collected. The minimum and maximum slopes of the warping path are limited to 0.2 dB/dB and 5 dB/dB, respectively, and the warping procedure seeks to minimize the Euclidean distance between the two histograms.

The SNR-warping procedure is illustrated in schematic form in Fig. 1. The left and lower panels of Fig. 1 show typical histograms of SNRs of speech collected using the PZM6FS and CLSTLK microphones, respectively. The central panel of Fig. 1 shows the warping path used to match SNRs from the two microphones. As can be seen in Fig. 1, the mode in the SNR histogram for the CLSTLK microphone at 26 dB is approximately matched to the mode in the SNR histogram for the PZM6FS microphone which actually occurs at 9 dB.

Since the alignment obtained by dynamically warping the histograms of the training and the testing data is not perfect, we have found that it is beneficial to smooth the correction vectors using the simple function

$$\text{Smoothed } \mathbf{v}(SNR) = .40\,\mathbf{v}(SNR) + .24\,\mathbf{v}(SNR{+}1) + {} \\ + .24\,\mathbf{v}(SNR{-}1) + .06\,\mathbf{v}(SNR{+}2) + .06\,\mathbf{v}(SNR{-}2)$$

where $\mathbf{v}$ refers to an arbitrary cepstral vector from either environment, and SNR is in dB. Applying the above smoothing function

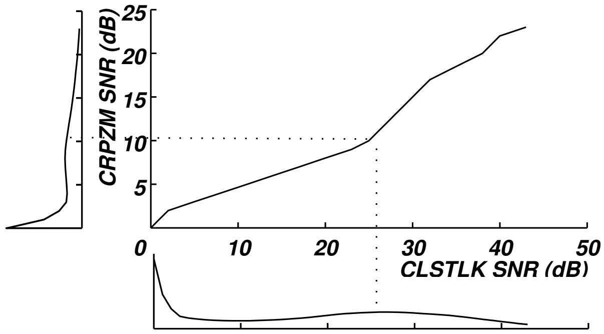

Figure 1: Nonlinear mapping of SNRs for the CLSTLK and PZM6FS microphones based on histograms of SNR values. The unlabeled graphs along the horizontal and vertical axes indicate the relative likelihood of observing various SNRs for the two microphones. The central panel indicates the warping path that best matches the two functions.

to cepstra from both the training and testing data, we have reduced recognition error rate by an average of about 10 percent.

Once a correspondence is established between the SNRs in the training and testing environments, correction vectors are computed as the difference between average cepstra for every SNR in the testing environment and its corresponding SNR in the training environment.

2.2. Experimental Results

Table 2 compares the recognition accuracy obtained when the BSDCN algorithm is evaluated using the alphanumeric census database described in [1]. We note that the environment-independent BSDCN algorithm achieves a level of recognition accuracy when trained on the CLSTLK microphone and tested on the PZM6FS microphone that is approximately equal to the recognition accuracy achieved by the environment-dependent SDCN algorithm on the same task.

TEST	CLSTLK	PZM6FS
BASE	86.9%	31.4%
BSDCN	86.4	70.0%
SDCN	N/A	72.4%
CDCN	85.7%	75.7%
FCDCN	N/A	78.6%

Table 2: Performance of the BSDCN algorithm compared with the baseline, SDCN, and CDCN algorithms, using testing data from two microphones. The system was trained using speech from the CLSTLK microphone.

Figure 2 compares the recognition accuracy obtained using the BSDCN, SDCN, and FCDCN algorithms for four microphones: the omnidirectional desktop PZM6FS, the Crown PCC160 cardioid desktop microphone (PCC160), the Sennheiser ME80 supercardioid electret microphone (ME80), and the Sennheiser 518 handheld dynamic cardioid microphone (SE518). (Different speech samples were used from those used to compile Tables 1 and 2.) The system was trained with speech from the CLSTLK microphone in all cases. We found again that accuracy obtained using the environment-independent BSDCN algorithm was comparable to that of the environment-dependent SDCN algorithm. The environment-dependent FCDCN algorithm produces greater recognition accuracy, especially for microphones such as the PZM6FS, which provides a lower intrinsic SNR. This is to be expected, since the value of the optimal cepstral correction vector varies much more from one VQ codeword to the next when the SNR is low. We are currently working to develop a *Blind FCDCN* algorithm (BFCDCN) that is similar in philosophy to the BSDCN algorithm, but that can also exploit the additional information that is made available by allowing for the compensation vectors to vary for different VQ codewords at each SNR, as in FCDCN.

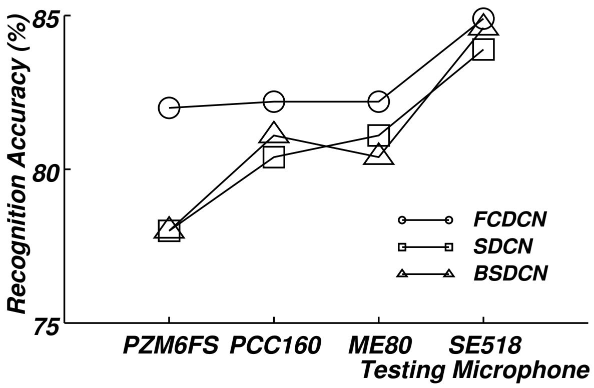

Figure 2: Comparison of recognition accuracy for SPHINX on the alphanumeric task using four microphones and the BSDCN and FCDCN algorithms. The system was trained using speech from the CLSTLK microphone.

3. REAL-TIME IMPLEMENTATION OF THE CDCN ALGORITHM

We have also produced a real-time implementation of the original CDCN algorithm. As described in [1], the CDCN algorithm compensates for unknown additive noise and linear filtering by use of a parametric model of environmental distortion, rather than by direct estimation of cepstral vectors, as is done with the SDCN, FCDCN, and similar algorithms. Although the CDCN algorithm is intrinsically more computationally costly than either the SDCN or FCDCN algorithms, we integrated a version of this algorithm into a real-time spoken language system [9] without any apparent additional processing time to the user. This was accomplished in two ways. First, the compensation and normalization parameters **n** and **q** are computed in the background during the silent intervals between the speaker's utterances. (This computation presently takes approximately 15 seconds on a 15-MIPS NeXT workstation.) Second, compensation of the incoming speech is expedited by normalizing only the first several cepstral coefficients rather than the entire vector, and by computing cepstral distances only for those codewords that are most similar to the incoming speech vector. The actual cepstral compensation is presently accomplished in better than real time using the Motorola 56001 DSP chip on the NeXT workstation.

Figure 3 shows how the recognition accuracy of the BSDCN algorithm and the real-time implementation of the CDCN algorithm depend on the amount of environment-specific speech data available for adaptation. The recognition accuracy of the real-time CDCN algorithm converges with only about 2 seconds of adapting speech, while the BSDCN algorithm requires at least 60 seconds of adapting speech to reach asymptotic levels of recognition accuracy. This is consistent with intuition, as the CDCN algorithm imposes more structure on the compensation process (from knowledge of how speech is likely to be degraded), while the BSDCN algorithm is entirely data driven.

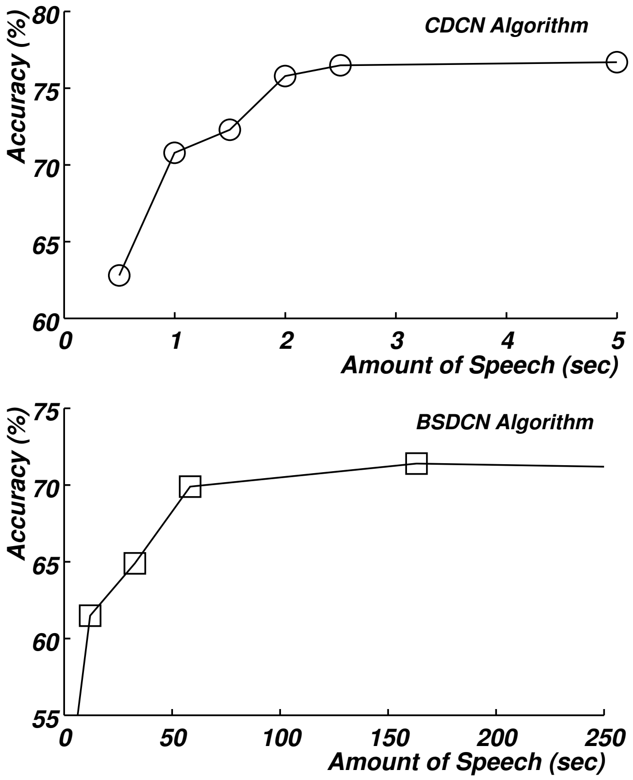

Figure 3: Dependence of recognition accuracy of the CDCN and BSDCN algorithms on the amount of speech in the testing environment available for adaptation. The system was trained using the CLSTLK microphone and tested using the PZM6FS.

4. SUMMARY

We described two algorithms for robust speech recognition that compensate incoming speech for the effects of additive noise and linear filtering. The first algorithm, *Blind SNR-dependent cepstral normalization (BSDCN*), differs from previous algorithms we have discussed in that it provides good recognition accuracy using an extremely simple compensation algorithm, and without the need for simultaneously-recorded training data in which speech is matched between the training and testing environments on a frame-by-frame basis. The second algorithm discussed was an implementation of the more complex CDCN algorithm, which estimates compensation parameters in the background on an ongoing basis, and then applies the compensation vectors in better than real time. The BSDCN algorithm is simpler and provides good speech recognition accuracy, even when the acoustical characteristics of the training and testing environments are quite different. The "real-time" CDCN algorithm is more computationally complex, but it is able to exploit *a priori* structural knowledge about the nature of the acoustical degradation to estimate compensation parameters on the basis of far less speech from the unknown testing environment.

ACKNOWEDGEMENTS

This research was sponsored by the Defense Advanced Research Projects Agency (DOD), ARPA Order No. 5167, under contract number N00039-85-C-0163. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US Government. We thank Hsiao-Wuen Hon, Xuedong Huang, Kai-Fu Lee, Raj Reddy, Eric Thayer, Bob Weide, and the rest of the speech group for their contributions to this work.

1. Acero, A. and Stern, R. M., ‘‘Environmental Robustness in Automatic Speech Recognition’’, *ICASSP-90*, April 1990, pp. 849-852.

2. Erell, A. and Weintraub, M., ‘‘Estimation Using Log-Spectral-Distance Criterion for Noise-Robust Speech Recognition’’, *ICASSP-90*, April 1990, pp. 853-856.

3. Juang, B. H., ‘‘Speech Recognition in Adverse Envoronments’’, *Comp. Speech and Lang.*, Vol. 5, 1991, pp. 275-294.

4. T. G. Stockham, T. M. Cannon and R. B. Ingebretsen, ‘‘Blind Deconvolution Through Digital Signal Processing’’, *Proc. IEEE*, Vol. 63, 1975, pp. 678-692.

5. Boll, S. F. , ‘‘Suppression of Acoustic Noise in Speech Using Spectral Subtraction’’, *ASSP*, Vol. 27, 1979, pp. 113-120.

6. Acero, A. and Stern, R. M., ‘‘Robust Speech Recognition by Normalization of the Acoustic Space’’, *ICASSP-91*, May 1991, pp. 893-896.

7. Hwang, M.Y., Hon, H.W., Lee, K.F., ‘‘Between-Word Coarticulation Modeling for Continuous Speech Recognition’’, Technical Report, Carnegie Mellon University, April 1989.

8. Sakoe, H., Chiba, S., ‘‘Dynamic Programming Algorithm Optimization for Spoken Word Recognition’’, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 26, 1978, pp. 43-49.

9. Ward, W., ‘‘Understanding Spontaneous Speech: The Phoenix System’’, *ICASSP-91*, April 1991.