PAPER

# Normalization of Time-derivative Parameters for Robust Speech Recognition in Small Devices

Yasunari OBUCHI[†*], Nobuo HATAOKA[††], *Members, and* Richard M. STERN[†], *Nonmember*

**SUMMARY** In this paper we describe a new framework of feature compensation for robust speech recognition, that is suitable especially for small devices. We introduce Delta-cepstrum Normalization (DCN) that normalizes not only cepstral coefficients, but also their time-derivatives. Cepstral Mean Normalization (CMN) and Mean and Variance Normalization (MVN) are fast and efficient algorithms of environmental adaptation, and have been used widely. In those algorithms, normalization was applied to cepstral coefficients to reduce the irrelevant information from them, but such a normalization was not applied to time-derivative parameters because the reduction of the irrelevant information was not enough. However, Histogram Equalization provides better compensation and can be applied even to the delta and delta-delta cepstra. We investigate various implementation of DCN, and show that we can achieve the best performance when the normalization of the cepstra and the delta cepstra can be mutually interdependent. We evaluate the performance of DCN using speech data recorded by a PDA. DCN provides significant improvements compared to HEQ. We also examine the possibility of combining Vector Taylor Series (VTS) and DCN. Even though some combinations do not improve the performance of VTS, it is shown that the best combination gives the better performance than VTS alone. Finaly, the advantage of DCN in terms of the computation speed is also discussed.

***key words:*** *robust speech recognition, PDA, time-derivative, histogram quealization*

## 1. Introduction

Speech recognition exhibits its full value when it is used in small devices, such as PDAs, cellular phones and in-vehicle systems. Since there are no keyboards, speech is the only modality that enables easy input of long sentences. In most cases, those tools are used not in the laboratory, but in various scenes of real life. The acoustical condition is far from ideal in such scenes, and the need for robust speech recognition algorithms increases. It is well known that the performance of the standard speech recognition system degrades with additive noises, channel distortion, room reverberation, etc., and the research of robust speech recognition has a long history from 1980's.

In the HMM based speech recognition system, that is known as the most successful system so far, robustness can be acheived by two different ways. Since the system basically compares the input feature vector with the acoustic model, we can compensate either the feature vector or the acousitc model [1]. Feature vector compensation has smaller number of free parameters, therefore it tends to require less computation. Model compensation can be more precise, but generally needs more computation. In this paper, we pursue feature vector compensation because the computational resource is limited in small devices. In feature vector compensation, we start with the corrupted input signal, that is the mixture of relevant and irrelevant information, and try to remove the irrelevant information using any assumption or prior knowledge about the speech and noise model.

Spectral Subtraction (SS) [2] is an algorithm to reduce the effect of additive noises, where an assumption is made for the noise model. It is assumed that the power spectrum of noise is invariant throughout the utterance and can be estimated from the nonspeech segment. This assumption brings the conclusion that the effect of additive noises is reduced by subtracting the estimated power spectrum from the input signal. Recently SS was extended to deal with the spectral magnitude and phase [3], but the assumption is still for the noise model only.

Cepstral Mean Normalization (CMN) [4] is another successful algorithm, that compensates convolutional noises. In CMN, an assumption for the speech model is also needed, because we cannot estimate the convolutional noise using the nonspeech segment. Therefore, it is assumed that every clean utterance has the same cepstral mean, and the variation of cepstral means represents the variation of environments. Since the convolutional noise is expressed as the additive distortion in the cepstral domain, one can remove it simply by subtracting the cepstral mean. There are also some works trying to extend SS and CMN. In CDCN [5] and VTS [6], the clean speech is modeled by the Gaussian mixture, and the environment is modeled by the combination of additive and convolutional noises.

A natural extension of CMN is Mean and Variance Normalization (MVN) [7], [8], where the assumption is still stronger. More attention is paid to the clean speech model than to the environment, and it is assumed that

Manuscript received April 1, 2003.
Manuscript revised April 1, 2003.
Final manuscript received April 1, 2003.

[†]Department of Electrical and Computer Engineering and School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA
[††]Central Research Laboratory, Hitachi Ltd., Kokubunji, Tokyo, Japan
[*]Presently at Advanced Research Laboratory, Hitachi Ltd., Kokubunji, Tokyo, Japan

not only the mean but also the variance of the cepstral coefficients should be invariant for various utterances. Even though there is no simple analytical expression that implies that any distortion appears in a form of cepstral multiplication, numerical simulations support the argument, and hence the environmental noise can be removed by deviding cepstral coefficients by their variance. After all, Histogram Equalization (HEQ) [9], [10] uses the stronger assumption that the shape of the entire distribution of cepstral coefficients is invariant. In HEQ, any detail of the cepstral distribution is regarded irrelevant and to be removed.

From this perspective, we can say that any normalization can be applied to any parameter if we have an reliable assumption about the invariance related to the parameter. That is the motivation of our work, in which we try to apply normalization techniques not only to cepstral parameters, but also to their time-derivatives. Although it is true that the cepstral mean can be interpreted as the estimated convolutional noise, we do not pay much attention to the origin of the irrelevant information that is erased by normalization. Instead, we focus only on finding transformations that preserve the relevant information in speech. This paper compares and discusses such simple models and transformations, and shows that speech recognition performance can be improved by their use. More improtantly, these transformations are extended to the delta cepstra.

The remainder of this paper is organized as follows. In the next section, we describe the concept of HEQ and our implementation of it. In section 3, various versions of Delta-Cepstrum Normalization (DCN) are introduced. Parametric optimization of DCN is also discussed in this section. Section 4 presents experimental results for the speech database which we created using a PDA, and the last section gives the conclusions and future works.

## 2. Histogram Equalization

Histogram Equalization is a procedure that is commonly used in image processing[11]. Balchandran and Mammone [12] first applied it to the amplitudes of speech signals, and Dharanipragada and Padmanabhan [13] applied it to cepstral features as an adaptation method. Some more recent papers [9], [10] applied feature normalization methods for robust speech recognition.

The basic idea of HEQ is that the distribution of cepstral coefficients in the test data should be identical to that of the training data. This idea introduces the necessity of the nonlinear transformation that generates the required distribution of the output cepstral coefficients, while minimizing the total distortion between cepstral coefficients before and after the transformation. In the case where we can treat each dimension of the cepstral vector as independent, finding the transformation is easy by using the cumulative density function (CDF), the integral of the probability density function (PDF). Since the CDF is a monotonic increasing function between 0 and 1, the inverse function can be defined. Thus, the transformation of HEQ is defined as follows:

$$x_i = HEQ(y_i) = C_X^{-1}\left(C_Y\left(y_i\right)\right) \qquad (1)$$

where $C_X$ is the CDF estimated from training data, $C_Y$ is the CDF of the test data, $y_i$ is a cepstral coefficient of the $i^{th}$ frame, and $x_i$ is the corresponding transformed cepstral coefficient. Since HEQ is applied to each cepstral dimension independently, we omit the other subscript for the cepstral dimension in this paper.

Usuanlly there is a huge number of samples in the training data, and we can get an almost continuous curve of the CDF from the precise histogram. The number of samples in a test utterance is small, but we can define the CDF at sample points simply by sorting the cepstral parameters and obtaining their relative ranks, because the CDF is a function of the number of frames that have smaller values than the current point. After sorting, we calculate $C_X^{-1}(t/N)$ for $t = 0, 1, 2, \ldots, N$ (where $N$ is the number of frames) by interpolation using the pre-stored numeric table of $C_X^{-1}$. The way of calculating CDF values can be interpreted as an extension of the quantile based HEQ[14], where the number of quantiles is the same as the number of frames.

There are some issues in the implementation of HEQ. In [9], the CDF obtained from the Gaussian PDF was used as the reference. Even though the distribution of cepstral coefficients tends to be Gaussian in some cases, we made the reference CDF according to (1) to make it more precise. Another issue is whether MVN should be applied to the training data before obtaining the the reference CDF. We thought that HEQ should be a natural extension of MVN, so we applied MVN to the training data before developing the CDF. There is also a concern about the domain of HEQ. In [10], it is said that applying HEQ in the Mel-filterbank domain is better than applying it in the cepstral domain. However, our preliminary experiments showed the opposite results, so we decided to apply it in the cepstral domain.

## 3. Delta-Cepstral Normalization

### 3.1 Normalization of time-derivative parameters

It is well known that the use of time-derivative parameters such as delta and delta-delta cepstra improves speech recognition accuracy. However, there have been few previous studies that attempt to normalize these features. The RASTA method [15] and other filtering approaches make use of inter-frame information, but they do not use the entire distribution of delta parameters. Deng et al. [16] proposed an algorithm that incor-

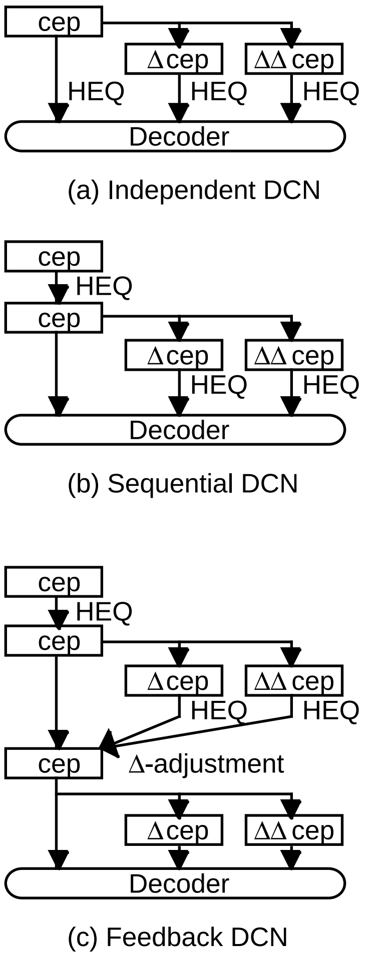

**Fig. 1** Schematic diagram of DCN. (a) Independent DCN. (b) Sequential DCN. (c) Feedback DCN.

porates the compensation of delta cepstrum. However, the compensation depends only on the prior Gaussian mixture models, and the distribution of delta cepstra among frames is not taken into account.

Mean subtraction of delta parameters does not help because the mean of delta parameters is always zero by definition. The variance of delta parameters can be non-zero, but it was reported in [8] that MVN does not need to applied to the delta and delta-delta cepstra. It is possible that the improvement obtained using MVN is smaller than the loss of relevant information. However, if compensation using HEQ, that is expected to be more efficient in noisy conditions, provides more gain than loss, we could have different results.

In the framework of normalizing delta and delta-delta cepstra, it should be noted that those parameters are not independent from the original cepstrum. Hence, there are several ways with which these parameters could be compensated. Figure 1 provides the schematic diagram of three types of Delta-Cepstrum Normalization (DCN). The simplest option is called Independent DCN, where the delta and delta-delta cepstra are calculated from the original capstrum, and then HEQ is applied to the cepstrum, the delta-cepstrum, and the delta-delta cepstrum independently. The second option is called Sequential DCN, where HEQ is applied to the original cepstrum, then time-derivative operation is carried out using the normalized cepstrum, and finally HEQ is applied to the delta and delta-delta cepstra. In this method, the delta and delta-delta cepstrum part can take advantage of the normalization of the cepstrum. The third option is called Feedback DCN, where the output of Sequential DCN is fed back to the cepstrum part, and "Δ-adjustment" is executed. Δ-adjustment is a procedure described in more detail below that reduces the mismatch between the normalized cepstrum and the normalized delta and delta-delta cepstra. By introducing Δ-adjustment, the cepstral normalization can take advantage of the normalization of the delta and delta-delta cepstra. However, even though both the delta and delta-delta cepstra are expected to be helpful, we perform Δ-adjustment using the delta cepstrum only, because it is difficult to define an appropriate Δ-adjustment procedure that makes use of both delta and delta-delta. A more detailed description of Feedback DCN including Δ-adjustment follows.

In Feedback DCN we describe the observed cepstral coefficients by $y_i$. After applying HEQ, we obtain normalized coefficients $z_i$.

$$z_i = HEQ(y_i). \tag{2}$$

Delta-cepstral coefficients are defined as follows.

$$\Delta z_i = \frac{1}{2}(z_{i+1} - z_{i-1}). \tag{3}$$

The error function is then defined to be the difference between the original delta cepstrum and the normalized delta cepstrum.

$$e_i = HEQ(\Delta z_i) - \Delta z_i. \tag{4}$$

Finally, the cepstrum is modified so that the error function decreases.

$$x_i = z_i - \alpha(e_{i+1} - e_{i-1}) \tag{5}$$

$\alpha$ is a weight parameter. Using these values of $x_i$, the delta and delta-delta cepstra are recalculated, and the resulting parameters are fed into the decoder.

### 3.2 Optimization of weight parameter

In the Δ-adjustment procedure, the weight parameter $\alpha$ plays an important role. The optimal value of $\alpha$ can be obtained by minimizing the adjusted total error function. First, the delta cepstrum after DCN is obtained as

$$\Delta x_i = \frac{1}{2}(x_{i+1} - x_{i-1})$$

$$= \Delta z_i + \alpha e_i - \frac{\alpha}{2}(e_{i+2} - e_{i-2}). \tag{6}$$

Using this, the adjusted total error function $E$ is defined as

$$E = \sum_{i=1}^{N} (HEQ\,(\Delta z_i) - \Delta x_i)^2 = \sum_{i=1}^{N} \Big( (\alpha - 1)\, e_i - \frac{\alpha}{2}\,(e_{i+2} + e_{i-2}) \Big)^2 \tag{7}$$

where we assumed the cyclic boundary condition

$$y_{i+N} = y_i \tag{8}$$

that leads to similar cyclic boundary conitions of other parameters. Then we get

$$E = (\frac{3}{2}K_0 - 2K_2 + \frac{1}{2}K_4)\alpha^2 - 2(K_0 - K_2)\alpha + K_0 \tag{9}$$

where $K_0$, $K_2$, and $K_4$ are the quadratic terms in regard to $e_i$.

$$\sum_{i=1}^{N} e_i^2 = \sum_{i=1}^{N} e_{i+2}^2 = \sum_{i=1}^{N} e_{i-2}^2 = K_0 \tag{10}$$

$$\sum_{i=1}^{N} e_i e_{i+2} = \sum_{i=1}^{N} e_i e_{i-2} = K_2 \tag{11}$$

$$\sum_{i=1}^{N} e_{i-2} e_{i+2} = K_4 \tag{12}$$

Finally, we get the value of $\alpha$ that minimizes the adjusted total error function $E$.

$$\alpha = \frac{2(K_0 - K_2)}{3K_0 - 4K_2 + K_4}. \tag{13}$$

We can also say that the cross correlation among $e_i$s is small, so $K_0$ is usually much larger than $K_2$ and $K_4$. Therefore $\alpha = 2/3$ is a good approximation of the optimal value.

### 3.3 MAP estimation

When we applied HEQ to the feature vector (either cepstrum, delta cepstrum, or delta-delta cepstrum), it was assumed that the shape of the distribution of the feature vector is completely the same as the distribution of the reference. However, the shape of the distribution can change according to the uttered sentence, although it is probably similar to the reference. Therefore, there is a risk of overfitting if one has every confidence in the normlized feature vector. The simplest way to avoid it is to introduce the idea of the MAP estimation [17] with the assumption that the prior probaility distribution of the feature vector is a Gaussian, whose mean is the output of the HEQ estimator.

$$p(x_i) = N(x_i : z_i, \Sigma_x^2) \tag{14}$$

where $z_i$ is the output of any HEQ-based estimator, and $\Sigma_x$ is the unknown variance. The variance represents the unreliability of the HEQ estimation, so it becomes small if the utterance is long or made of well-balanced phones. The observation probability distribution can also be approximated by a Gaussian,

$$p(y_i|x_i) = N(y_i : x_i, \Sigma_y^2) \tag{15}$$

where $\Sigma_y$ is also an unkown variance, that represents the distortion made by the environment. Using the Baysian rule

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} \tag{16}$$

and assuming that there is no prior knowledge about $p(y_i)$, the posterior probability distribution of $x_i$ is simply the product of two Gaussians, given by

$$p(x_i|y_i) = N(x_i : \frac{\Sigma_x^2 y_i + \Sigma_y^2 z_i}{\Sigma_x^2 + \Sigma_y^2}, \frac{\Sigma_x^2 \Sigma_y^2}{\Sigma_x^2 + \Sigma_y^2}) \tag{17}$$

that gives the MAP estimation of $x_i$ as a weighted sum of $y_i$ and $z_i$. Since we do not have any knowledge about $\Sigma_x$ and $\Sigma_y$, the optimal weight would be determined experimentally.

## 4. Experiments

### 4.1 Experimental setup

The proposed algorithms were evaluated in a series of recognition experiments. Triphone HMMs with 2000 tied states (8 Gaussians / state) were trained using the 5000-word LDC Wall Street Journal database (WSJ0). The Sphinx-III decoder developed by CMU was used for decoding, with a trigram language model. Speech input was sampled by 11.025kHz, and 13 MFCCs were computed every 10ms.

We recorded 330 utterances from eight speakers simultaneously using two microphones: the built-in microphone of the PDA (Compaq iPAQ PocketPC Model 3630) and a close-talk microphone (Optimus Nova 80). Each speaker uttered 40 to 43 sentences chosen from the WSJ0 database. The perplexity of the test set is 64.35. Recording was done in an office room with no window, where some computers were making fan noises. Using these recordings, we prepared two test sets. The first set was the read data recorded by the PDA microphone. The SNR of the first set was estimated as 18dB using NIST's stnr tool. The data are corrupted by both additive noise from computer fans in the room and the spectral tilt the PDA microphone. A second set of artificial data were obtained by digitally adding the relatively clean speech data recorded using the close-talk microphone to noise recorded by the PDA microphone

**Table 1** Recognition results for real data

| | WER (%) |
|---|---|
| Baseline (CMN) | 41.5 |
| MVN | 33.5 |
| HEQ | 30.2 |
| Independent DCN | 27.5 |
| Sequential DCN | 27.0 |
| Feedback DCN | 25.6 |
| Close-talk | 16.4 |

with varying SNR from 0dB to 25dB. The spectral tilt of the close-talk microphone is small, and the additive noise is the same as the first set except that the amplitude is adjusted to each SNR value.

### 4.2 Experiments using real data

Table 1 shows the word error rates (WERs) obtained by various methods using the real data set. Since the number of words in the hypothesis is not always the same as the number of words in the transcript, dynamic programming is used to align the hypothesis and the transcript, and the WER is defined as

$$WER = \frac{S + I + D}{N} \tag{18}$$

where $S, I, D$ are the number of substitution, insertion, and deletion errors respectively, and $N$ is the number of words in the transcript. We regard CMN as the baseline because it is a widely used algorithm, and compare it with other algorithms. It is shown that MVN and HEQ improve the recognition accuracy as expected. Independent DCN gives certain improvement from HEQ, that is 9% relative WER reduction. Sequential DCN is slightly better than Independent DCN, 11% relative WER reduction. Finally, Feedback DCN results in the best performance, that is 15% relative WER reduction. It should be noted that the value of the weight parameter $\alpha$ is set to 1 in this experiment. As the reference, the WER obtained by the close-talk microphone with no additional noise is 16.4%, that is regarded as the lower limit of any compensation method.

### 4.3 Optimizing weight and MAP parameters

To evaluate the dependency of WERs on the weight parameter $\alpha$, we made recognition experiments with various values of $\alpha$. The results are shown in Fig. 2. The point $\alpha = 0$ corresponds to HEQ. (see eq. (5)) From this point, there is a rapid WER decrease to $\alpha = 0.4$, but then the curve becomes rather flat. The difference is small from $\alpha = 0.4$ to $\alpha = 1.0$ including the case where the $\alpha$ is optimized for each utterance usgin eq.(13), so we will use $\alpha = 1.0$ in the rest of this paper for simplicity.

We also made experiments with various MAP parameters. In Figs. 3 and 4, the parameter $\beta$ is defined as $\Sigma_y^2/(\Sigma_x^2 + \Sigma_y^2)$. Figure 3 shows the results with

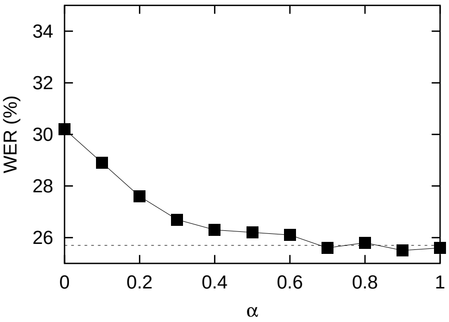

**Fig. 2** Recognition results of DCN with various alpha value. Dotted line represents the case in which $\alpha$ is optimized for each utterance using eq.(13).

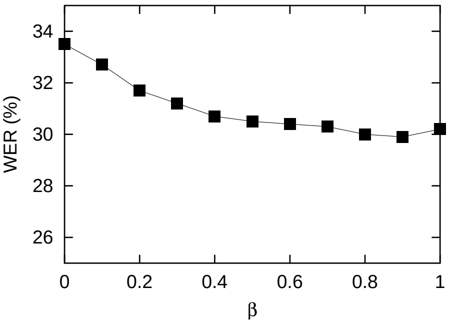

**Fig. 3** Recognition results of HEQ with various beta value

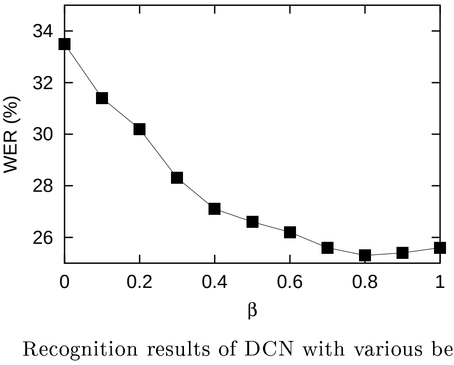

**Fig. 4** Recognition results of DCN with various beta value

**Table 2** Recognition results for real data using combination with VTS

| | WER (%) |
|---|---|
| VTS (CMN) | 23.3 |
| VTS + MVN | 25.4 |
| VTS + HEQ | 27.3 |
| VTS + Independent DCN | 23.4 |
| VTS + Sequential DCN | 23.6 |
| VTS + Feedback DCN | 22.7 |

HEQ (no Delta-Cepstrum Normalization), and Figure 4 shows the results with DCN. In both cases, WERs decrease from $\beta = 0$ to $\beta = 0.4$, but for larger $\beta$, the difference is small even though the best results were obtained with $\beta$ being slightly smaller than 1.0. The value of $\beta$ will also be set to 1.0 in the rest of the paper.

### 4.4 Combination with VTS

VTS (Vector Taylor Series) [6] is known as one of the most powerful compensation algorithms developed for quasi-stationary additive noise and linear filtering. In [18], it is reported that HEQ reduces the residual noise of VTS, so one can achieve better results by applying HEQ after VTS. To verify this result and check the extensibility to DCN, we performed some additional experiments using VTS.

Table 2 shows the word error rates obtained using VTS as well as various combination of VTS and other methods. The WER obtained by VTS alone (with CMN) is 23.3%, that is better than even the best case of DCN, but the WER becomes higher when we apply MVN after VTS. Applying HEQ after VTS makes the WER still worse, that is opposite to the result described in [18]. Independent DCN and Sequential DCN are better than MVN and HEQ, but their WERs are slightly greater than VTS only. However, if we apply Feedback DCN after VTS, we obtain a relative improvement in WER of about 3% compared to VTS alone.

### 4.5 Experiments using artificial data

To investigate the SNR-dependency of DCN, we carried out experiments using the artificial data set that was made by various SNR. The results are shown in Fig. 5. Since Feedback DCN was the best among three types of DCN in the previous experiments, we used Feedback DCN only. VTS in combination with DCN was also tested.

Obviously DCN outperformed HEQ over almost all range of SNR. The performance of DCN is noticeable especially in the lower SNR range, and it is even better by itself than VTS at 0dB. The use of DCN after VTS improves recognition accuracy over the result obtained by VTS only for SNRs below about 15dB, but it is not helpful in the higer SNR range.

Since we have estimated the SNR of the real data as 18dB, we can compare the results from the real data and the artificial data at 18dB using interpolation. Table 3 shows the comparison. The numbers of the artificial data were interpolated using 15dB and 20dB points. As seen in the table, they have similar tendencies but VTS is more effective for the artificial data, and DCN is not helpful after VTS for the artificial data at this point. That would be because the way that we made the artificail data is matched to the one assumed in the theory of VTS, while the read data include nonlinear distortion.

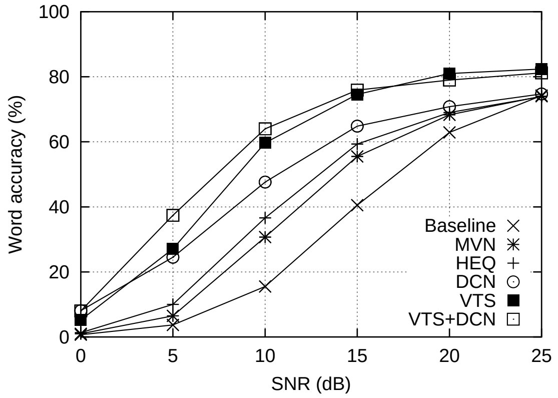

**Fig. 5** Recognition results for artificial data

**Table 3** Comparison of WERs from real and artificial data. (a) WER(%) obtained from real data (b) WER(%) obtained by interpolating results from artificial data

| | (a) | (b) |
|---|---|---|
| Baseline (CMN) | 41.5 | 46.1 |
| MVN | 33.5 | 36.8 |
| HEQ | 30.2 | 34.9 |
| Feedback DCN | 25.6 | 31.6 |
| VTS | 27.0 | 21.6 |
| VTS + Feedback DCN | 25.6 | 22.2 |

### 4.6 Computational complexity

One of the advantages of HEQ is fast execution owing the possibility of being implemented via table lookup. This makes HEQ very attractive for the use in small devices. On the other hand, EM-based algorithms such as VTS are usually very slow. To confirm the same advantage of DCN, we measured the time consumed by the CPU to compensate 330 utterances of the real data set, and calculated the average time to compensate one second of speech. The experiment was carried out with an Intel Celeron 2.0GHz processor and 256MB memory running on the Linux operating system. Execution times for the various algorithms are shown in Table 3.

Compensation by HEQ includes sorting of the cepstral coefficients and interpolation of the CDF using the pre-stored numeric table. Those procedures are simple enough and can be done in very short time. Thus, HEQ requires only 1.2ms to compensate one second speech. In Independent and Sequential DCN, there are three

**Table 4** Execution time for 1 second speech

| | WER (%) |
|---|---|
| MVN | 0.0001 |
| HEQ | 0.0012 |
| Independent DCN | 0.0033 |
| Sequential DCN | 0.0033 |
| Feedback DCN | 0.0019 |
| VTS | 2.8395 |

equalization operations for the cepstrum, the delta cepstrum, and the delta-delta cepstrum. That is why it takes approximately three times as much time as HEQ. In Feedback DCN, we did not apply HEQ to the delta-delta cepstrum, so the execution time is about twice that of HEQ. Apart from those small differences, all of three DCN algorithms ran in less than 1% of real time. In constrast, VTS requires much more than real time due to its time-consuming EM iterations.

## 5. Conclusions

In this paper, we have introduced a new feature normalization algorithm that is based on the normalization of time-derivative parameters. This procedure, referred to as Delta-Cepstrum Normalization (DCN), is quite simple to implement and provides greater recognition accuracy than either Cepstrum Mean Normalization (CMN) or Histogram Equalization (HEQ). The performance of DCN approached that of Vector Taylor Series (VTS) and with only of a small fraction of the computational cost of VTS. We investigated three implementations of DCN, Independent, Sequential, and Feedback DCN. The best implementation, Feedback DCN, provedes a relative improvement of 15% compared to standard HEQ using real data recorded by the built-in microphone of an iPAQ. We also showed that Feedback DCN can reduce recognition error rate when it is applied after VTS.

Implementation of DCN may include parametric expressions such as the weight factor of $\Delta$-adjustment, $\alpha$, and the MAP estimation parameter, $\beta$. However, experimental results showed that simple setting that $\alpha = \beta = 1$ gives the near optimal performance, even though one can acheive 1% or less relative WER reduction by using optimal value of $\alpha$ and $\beta$.

The results using artificial data showed that DCN is helpful especially in the lower SNR range. However, in the higher SNR range, VTS is effective enough because the noise model we used to make the artificial data is matched to the one assumed in the theory of VTS. In such a situation, DCN is less effective either by itself or after VTS.

Fast run times for the HEQ and DCN algorithms are observed when the algorithms are implemented using table lookup. As a result, these algorithms are attractive for small devices used in noisy conditions, such as PDAs, cellular phones, and in-vehicle systems.

So far we have examined the effect of DCN on the data with stationary noises, but our future work would include the research on the non-stationary noise environment, which small devices often face to. It is also important to combine these methods with multiple microphone techniques. After those efforts, we expect that the world with ubiquitous voice-activated small devices will apear.

### References

[1] M. J. F. Gales and S. J. Young, "Robust speech recognition in additive and convolutional noise using parallel model combination," Computer Speech and Language, vol.9, pp.289-307, 1995

[2] S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," IEEE Trans. Acoustics, Speech and Signal Processing, vol.27, pp.113-120, 1979

[3] J. Droppo, A. Acero, and L. Deng, "A nonlinear observation model for removing noise from corrupted speech log mel-spectral energies," Proc. of International Conference of Spoken Language Processing, Denver, USA, 2002

[4] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," Journal of the Acoustical Society of America, vol.55, pp.1304-1312, 1974

[5] A. Acero and R. M. Stern, "Environmental robustness in automatic speech recognition," Proc. of International Conference of Acoustics, Speech and Signal Processing, Albuquerque, USA, 1990

[6] P. J. Moreno, B. Raj, and R. M. Stern, "A vector Taylor series approach for environment independent speech recognition," Proc. of International Conference of Acoustics, Speech and Signal Processing, Atlanta, USA, 1996

[7] J. P. Openshaw and J. S. Mason, "On the limitations of cepstral features in noise," Proc. of International Conference of Acoustics, Speech and Signal Processing, Adelaide, Australia, 1994

[8] P. Jain and H. Hermansky, "Improved mean and variance normalization for robust speech recognition," Proc. of International Conference of Acoustics, Speech and Signal Processing, Salt Lake City, USA, 2001

[9] A. de la Torre, J. C. Segura, C. Benitez, A. M. Peinado, and A. J. Rubio, "Non-linear transformation of the feature space for robust speech recognition," Proc. of International Conference of Acoustics, Speech and Signal Processing, Orlando, USA, 2002

[10] S. Molau, M. Pitz, and H. Ney, "Histogram based normalization in the acoustic feature space," Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding, Trento, Italy, 2001

[11] J. C. Russ, The image processing handbook, CRC Press, 1995

[12] R. Balchandran and R. J. Mammone, "Non-parametric estimation and correction of non-linear distortion in speech systems," Proc. of International Conference of Acoustics, Speech and Signal Processing, Seattle, USA, 1998

[13] S. Dharanipragada and M. Padmanabhan, "A nonlinear unsupervised adaptation technique for speech recognition," Proc. of International Conference of Spoken Language Processing, Beijing, China, 2000

[14] F. Hilger and H. Ney, "Quantile based histogram equalization for noise robust speech recognition," Proc. of EUROSPEECH, Aalborg, Denmark, 2001

[15] H. Hermansky, N. Morgan, A. Bayya, and P. Kohn,

"RASTA-PLP speech analysis," ICSI Technical Report TR-91-069, UC Berkeley, 1991

[16] L. Deng, J. Droppo, and A. Acero, "A Bayesian approach to speech feature enhancement using the dynamic cepstral prior," Proc. of International Conference of Acoustics, Speech and Signal Processing, Orlando, USA, 2002

[17] C.-H. Lee, C.-H. Lin, and B.-H. Juang, "A study on speaker adaptation of the parameters of continuous density hidden Markov models," IEEE Trans. Signal Processing, vol.39, pp.806-814, 1991

[18] J. C. Segura, M. C. Benitez, A. de la Torre, S. Duponi, and A. J. Rubio, "VTS residual noise compensation," Proc. of International Conference of Acoustics, Speech and Signal Processing, Orlando, USA, 2002

**Yasunari OBUCHI** recieved B. S. and M. S. in physics from University of Tokyo in 1988 and 1990 respectively. Since 1992, he had been working in Hitachi Central Research Laboratory. He worked in Carnegie Mellon University as a visiting researcher from 2002 to 2003. Currently he is a senior researcher of Hitachi Advanced Research Laboratory. His research interest includes robust speech recognition, spoken dialog systems, speech recognition in small devices, and speech-to-speech translation. He is a member of ASJ.

**Nobuo Hataoka** Hitachi CRL

**Richard M. Stern** Professor of CMU