行政院國家科學委員會專題研究計畫　期中進度報告

無線網路環境下語音處理技術之整合型研究(2/3)

計畫類別：個別型計畫
計畫編號：NSC93-2213-E-002-060-
執行期間：93 年 08 月 01 日至 94 年 07 月 31 日
執行單位：國立臺灣大學資訊工程學系暨研究所

計畫主持人：李琳山

報告類型：精簡報告

處理方式：本計畫可公開查詢

中　華　民　國 94 年 5 月 30 日

行政院國家科學委員會專題研究計畫期中報告

無線網路環境下語音處理技術之整合型研究（ 2/3 ）

計畫編號：NSC 93-2213-E-002-060-
執行期限：93 年 8 月 1 日至 94 年 7 月 31 日

主持人：李琳山　國立台灣大學資訊工程學系暨研究所
E-mail: lslee@gate.sinica.edu.tw

ABSTRACT

Cepstral mean subtraction (CMS) and cepstral normalization (CN) have been popularly used to normalize the first and the second moments of cepstral coefficients, and proved to be very helpful for robust speech recognition [1, 2]. In this paper, a unified formulation for Higher Order Cepstral Moment Normalization (HOCMN) is developed by extending the concept of CMS and CN to orders much higher than three. A whole family of normalization techniques for different orders is thus proposed. Preliminary experimental results based on Aurora 2.0 showed that the recognition accuracy can be significantly improved with this approach under all noisy conditions. For example, *HOCMN*$_{[1,5,100]}$ (normalization of the first, fifth and 100-th order cepstral moments) is shown to offer an error rate reduction of 32.83% as compared to the conventional CN with a full-utterance processing interval, or an error rate reduction of 20.78% as compared to CN with a segmental processing interval.

中文摘要

倒頻譜消去法和倒頻譜正規化法已廣為使用來正規化倒頻譜係數的一階和二階動差，而且對強健性語音辨識上有很大的幫助。在這篇論文裡，我們延伸倒頻譜消去法和倒頻譜正規化法的概念到遠高於三階，發展出一套高階倒頻譜動差正規化法的公式。我們提出了整個正規化的技術並包含了各種不同的階數。在 Aurora 2.0 基礎實驗的結果中，這個方法在各種雜訊條件下的辨識率都有顯著的進步。舉例來說，以 *HOCMN*$_{[1,5,100]}$ (正規化一階、五階和一百階倒頻譜動差)和傳統用整句話處理的倒頻譜正規化法相比可降低 32.83%的錯誤率，或和分段式處理的倒頻譜正規化法相比可降低 20.78%的錯誤率。

1. INTRODUCTION

In real world speech recognition applications, robust features are highly desired in order to offer acceptable recognition performance under various noisy conditions. Mel-frequency cepstral coefficients (MFCCs) have been well accepted as a good choice for speech features with reasonable robustness, and many advanced techniques have been developed based on them. Cepstral mean subtraction (CMS) and cepstral normalization (CN) have been two commonly used methods. A possible reason for this is that CMS effectively removes the DC component in cepstral domain, which includes the channel distortion, and avoids the low frequency noise to be further amplified. In addition, the variance normalization of CN may reduce the difference in probability density function (pdf) between the clean and noisy speech signals. It was also proposed that the normalization of the third-order cepstral moment may achieve better performance than CMS and CN [3], and with such normalization the pdf of noisy speech signals actually becomes even closer to that of the clean ones.

The pdf of cepstral coefficients of speech signals is usually regarded as a quasi-Gaussian distribution. Under this assumption, the odd order moments should be zero and the even order moments should be some specific constants. CMS is to normalize the first moment, and CN is to normalize the second moment in addition. The Higher Order Cepstral Moment Normalization (HOCMN) approaches proposed in this paper are therefore developed along this line, i.e., the order of the moments to be normalized can be more than three. Such moment normalization approaches may make the pdf of the noisy speech even closer to that of the clean one. Significant improvements in recognition accuracy under all noisy conditions were obtained with AURORA 2.0. When a higher order moment is normalized, the residual mismatch after CN can be further reduced and the feature coefficients may become more robust. One possible reason is that the higher order moments are more dominated by those samples with larger values, which are also a major source of mismatch, and normalization may suppress the values of those samples effectively. In this way, not only the signal distortion may be reduced, but also the signals can be shaped and less mismatched for both small and large value samples.

2. PROPOSED APPROACH

Here we describe the HOCMN approach in a unified formulation. The *N-th* order moment of a cepstral coefficient sequence $X(n)$ is the expectation value of $X^N(n)$, usually approximated by the time average over some interval,

$$E[X^N(n)] \triangleq \frac{1}{T}\sum_{k=0}^{T-1} X^N(k) \quad (1)$$

The purpose of moment normalization of order N is then to have

$$E[X^N_{[N]}(n)] = 0 \quad (2)$$

if N is an odd integer, where $X_{[N]}(n)$ is a transformed sequence of $X(n)$ whose *N-th* order moment has been normalized, and

$$E[X^N_{[N]}(n)] = M_N \quad (3)$$

if N is an even integer, where M_N is the *N-th* moment of a normalized Gaussian distribution, $N(0,1)$, which can be obtained by the moment generating function. With the above notation, the well-known CMS processing is

$$X_{CMS}(n) \triangleq X_{[1]}(n) = X(n) - E[X(n)]. \quad (4)$$

where $E[X(n)]$ can be approximated by equation (1) with $N=1$, and the well-known CN processing is

$$X_{CN}(n) \triangleq X_{[1,2]}(n) = X_{[1]}(n) \Big/ \sqrt{E[X_{[1]}^2(n)]}, \quad (5)$$

where $X_{[L,N]}(n)$ is a transformed sequence whose *L-th* and *N-th* moments have both been normalized as in equations (2) and (3), and so on. With the above, we now extend the moment normalization to higher orders as follows.

2.1. HOCMN for an even integer N

When performing HOCMN with an even integer N, we simply scale the first-order moment normalized coefficients $X_{[1]}(n)$ by a constant, such that the *N-th* order moment of the coefficients can also be normalized as in equation (3). Such normalization therefore usually co-exists with the first-order normalization or CMS, and can be expressed as

$$X_{[1,N]}(n) \triangleq bX_{[1]}(n) = bX_{CN}(n), \quad (6)$$

where b is the scaling factor, and the CN transformation in equation (3) is a special case of equation (6) with $N=2$. Solving equation (6) with equation (3), we have

$E[X_{[1,N]}^N(n)] = b^N E[X_{[1]}^N(n)] = M_N$, and

$$b = \left[\frac{M_N}{E[X_{[1]}^N(n)]}\right]^{\frac{1}{N}} \quad (7)$$

The value of b in equation (7) can be obtained using the approximation in equation (1). If N becomes relatively large, the value of b will be dominated by $Max_{n=0}^{(T-1)}[X_{[1]}(n)]$. In this case we can approximate the value of b by

$$b = \frac{1}{Max_{n=0}^{(T-1)}[X_{[1]}(n)]}. \quad (8)$$

It should be noted that from equation (6) it is clear that the *N-th* order moment of a coefficient sequence $X(n)$ can be normalized for only a single even number N. In other words, different N gives different values of b as in equation (7), and the scaling transformation of equation (6) can be performed only with a single value of b. For example, if $X_{[N]}(n)$ is normalized for $N=4$, it is not normalized for $N=2$ in any case.

2.2. HOCMN for an odd N

The HOCMN for an odd order can be extended from the third-order cepstral moment normalization approach previously proposed [3]. It usually also co-exists with the first-order normalization or CMS, and can be expressed as a nonlinear transformation of the coefficients normalized with the first as well as the $(N-1)-th$ order moments, $X_{[1,N-1]}(n)$,

$$X_{[1,N]}(n) \triangleq aX_{[1,N-1]}^{N-1}(n) + X_{[1,N-1]}(n) + c. \quad (9)$$

We then have $c = -aE[X_{[1,N-1]}^{N-1}(n)] \triangleq -aM_{N-1}$ because $E[X_{[1,N]}(n)] = E[X_{[1,N-1]}(n)] = 0$ from equation (2), and M_{N-1} is defined by equation (3) for $(N-1)$ being an even integer. Equation (9) can then be re-written as

$$X_{[1,N]}(n) \triangleq a(X_{[1,N-1]}^{N-1}(n) - M_{N-1}) + X_{[1,N-1]}(n) \quad (10)$$

Equation (2) for $X_{[1,N]}(n)$ is then

$$\begin{aligned} E[X_{[1,N]}^N(n)] &= E\{[a(X_{[1,N-1]}^{N-1}(n) - M_{N-1}) + X_{[1,N-1]}(n)]^N\} \\ &= a^N \cdot E\{[X_{[1,N-1]}^{N-1}(n) - M_{N-1}]^N\} + N \cdot a^{N-1} \cdot E\{[X_{[1,N-1]}^{N-1}(n) - M_{N-1}]^{N-1} X_{[1,N-1]}(n)\} \\ &\quad + \cdots + N \cdot a \cdot E\{[X_{[1,N-1]}^{N-1}(n) - M_{N-1}] X_{[1,N-1]}^{N-1}(n)\} + E[X_{[1,N-1]}^N(n)] = 0 \end{aligned} \quad (11)$$

When a is small, we can delete the higher order terms in equation (11) and keep only the last two terms. In this way the value of a can be approximated by

$$a \approx \frac{-E[X_{[1,N-1]}^N(n)]}{N \cdot E[X_{[1,N-1]}^{2(N-1)}(n) - M_{N-1} X_{[1,N-1]}^{N-1}(n)]} \quad (12)$$

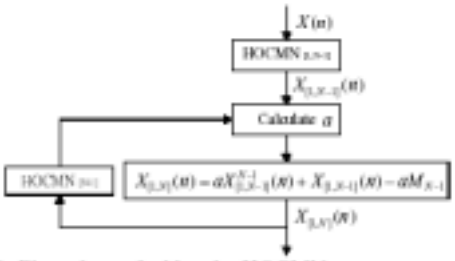

Figure 1. Flow chart of odd-order HOCMN

Figure 2. An odd order HOCMN followed by an even order HOCMN

Because equation (12) is only an approximation, a recursive procedure as shown in Figure 1 can be used to find a good solution. In Figure 1, $HOCMN_{[N]}$ or $HOCMN_{[L,N]}$ represents the processes of higher order cepstral moment normalization with order N or orders L and N both, etc.. Note that although the input to the loop in Figure 1 is normalized to both order 1 and order $N-1$, after the transformation of equation (9) the $(N-1)-th$ moment may not be normalized any longer. Therefore another normalization process of order $N-1$ is needed for the next loop. Practical experiments indicate that two iterations are usually enough to obtain a well converged solution.

2.3. HOCMN for both an even order N and an odd order L

A simple cascade method as shown in Figure 2 can be used to integrate the two techniques presented above. We can first normalize the feature coefficients by HOCMN with an odd order L and then by another HOCMN with an even order N. After such a procedure, a specific odd-order (*L-th*) moment and a specific even-order (*N-th*) moment of the feature coefficients can both be normalized. As in Figure 2, $X_{[1,L,N]}(n)$ is the feature sequence normalized to 1*st*, *L-th* and *N-th* order, and $HOCMN_{[1,N,L]}$ is the procedure to obtain such a sequence.

3. EXPERIMENTAL SETUP

The above approaches were evaluated by the AURORA 2.0 database, which is an English connected-digit string corpus. Two training conditions (clean condition/multi-condition) and three testing sets (sets A/B/C) were defined by AURORA 2.0 [4]. The clean-condition training has acoustic models trained by clean speech only, while the multi-condition training has models trained by a corpus with both clean and noisy speech. The testing set A includes four different types of noise which were used in the multi-condition training, while the testing set B includes another four different types of noise not used in the multi-condition training. The testing set C then includes noise types from both sets A and B, plus additional convolutional noise.

Whole-word HMM models were used as specified by AURORA 2.0. Each word had 16 states and 3 Gaussian mixtures per state. The speech features were extracted by the AURORA WI007 Front-end, which converted each signal frame into 13 cepstral coefficients (MFCCs, C1~C12 plus the log-energy). The first and second derivatives were then computed. In another version, the log-energy was replaced by C0.

The HOCMN approaches proposed here were implemented in two different ways, by full-utterances and by segments. In the former, the summation in (1) included the whole utterance when evaluating the moments. In the latter, or by segments with length I, the summation in equation (1) was performed by a segment of frames including the preceding $I/2$ frames and following $I/2$ frames.

4. EXPERIMENTAL RESULTS

4.1. Baseline results

The baseline experiments were performed with the clean-condition training for all the three testing sets A, B and C, and the results are listed in Table 1. In this table the word accuracy was averaged over all different noise types within each testing set and averaged over all different SNRs from 0dB to 20dB. The results in row (1) of Table 1 were obtained with the standard AURORA 2.0 front-end, MFCC features C1~C12 plus the log-energy without any extra processing. The next two rows (2) and (3) are for CN with a full-utterance processing interval, using log-energy and C0 as the energy terms respectively. Because the result in row (3) using C0 is significantly better, C0 will be used in all the following experiments. The row (4) is for CN with a segmental processing interval as mentioned previously with the segment length I being 86 frames, which gives better results. The next row (5) is for the third-order moment normalization (TMN) previously proposed [3] also with a full-utterance processing interval, in which the first, second, and the third order moments are all normalized. This approach is apparently even better. The last row (6) is exactly the same as row (5) except the TMN was performed with segments with length I being 86 frames, which offers better performance than row (5).

4.2. Initial results for HOCMN

We first performed the proposed normalization approach $HOCMN_{[I,N]}$ for N being an even integer, increasing from 2 up to 100. When full-utterance was used as the processing interval, the results are the lowest curve (a) in Figure 3. So the first point for the curve for $N=2$ is exactly the averaged result for CN, or row (3) in Table 1, 75.08%. It can be found that the recognition accuracy increases monotonically with the order N, and becomes more or less saturated when N goes up to 50 or 100. The results in curve (a) can be further improved by replacing the processing interval with segments with length $I=86$ frames. The results are the middle curve (b) in Figure 3. The first point of this curve for $N=2$ is exactly the average accuracy for CN ($I=86$) in the row (4) of Table 1. It is clear that the improvements observed for $HOCMN_{[I,N]}$ as compared to CN remain true when the full-utterance processing interval is replaced by segments. When we further cascaded this approach with a third order moment normalization, also with a segment length $I=86$ frames to perform $HOCMN_{[I,3,N]}$ as discussed in section 2.3, the results are the highest curve (c) in Figure 3. The first point of this highest curve (c) for $N=2$ is the case of third-order moment normalization, TMN or $HOCMN_{[I,2,2]}$, in the row (6) of Table 1. Apparently the higher order moment normalization does offer improvements as compared to TMN, and the cascade with a third order moment normalization does help for all even order moment normalization.

Baseline Experiments	Clean Condition			
	Set A	Set B	Set C	Ave
(1) MFCC (logE)	61.34	55.75	66.14	60.06
(2) MFCC (logE), CN (full-utterance)	70.18	70.78	66.37	69.66
(3) MFCC (C0), CN (full-utterance)	74.32	75.48	75.79	75.08
(4) MFCC (C0), CN (I=86)	78.03	79.77	78.75	78.87
(5) MFCC (C0), TMN (full-utterance)	79.62	81.47	80.10	80.46
(6) MFCC (C0), TMN (I=86)	80.23	82.70	81.48	81.47

Table 1. Recognition accuracy for the baseline experiments with clean-condition training

Best Results in Figure 3: HOCMN, N=100	Clean Condition Training				Baseline in Table 1	Relative Error Rate Reduction
	Set A	Set B	Set C	Ave		
(a) C0, $HOCMN_{[I,100]}$ (full-utterance)	80.58	82.11	80.42	81.16	(3): 75.08	24.40%
(b) C0, $HOCMN_{[I,100]}$ (I=86)	81.19	82.93	81.69	81.99	(4): 78.87	14.75%
(c) C0, $HOCMN_{[I,3,100]}$ (I_3=I_{100}=86)	81.24	83.95	82.46	82.57	(6): 81.47	5.94%

Table 2. Complete data for the best results for the curves (a)(b)(c) in Figure 3 (N=100) and relative error rate reduction as compared to the baseline (N=2) for the same curve listed in rows (3)(4)(6) of Table 1.

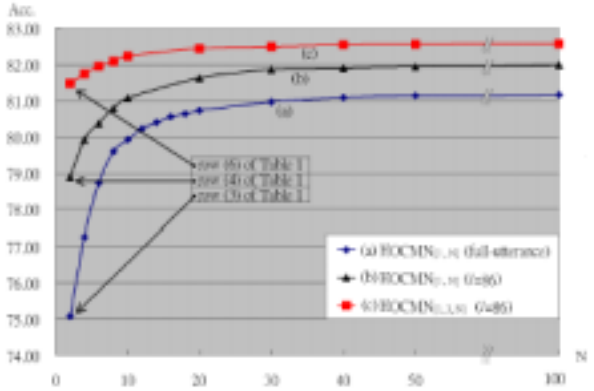

Figure 3. Recognition accuracy for HOCMN with different even orders N (a) $HOCMN_{[I,N]}$ (full-utterance) (b) $HOCMN_{[I,N]}$ (I=86) (c) $HOCMN_{[I,3,N]}$ (I_3=I_N=86)

Table 2 lists the improvements obtained by the higher order moment normalization for the three curves in Figure 3, i.e., the last point for N=100 as compared to the baseline of the first point for N=2 for each curve. For example, for the lowest curve (a) for $HOCMN_{[I,N]}$, full-utterance, N=100 produces an average accuracy of 81.16 while N=2 (CN), or row (3) of Table 1, gives only 75.08, and the error rate reduction is 24.40%, etc.. It can be found that in all the 3 cases the improvements are significant, although getting relatively less when the baseline gets better. The best result obtain here, the last point (N=100) for the highest curve (c), 82.57, represents 5.94% error rate reduction as compared to TMN as shown in the last row of Table 1. Also listed in Table 2 are the detailed data for testing sets A, B and C for the three cases of N=100.

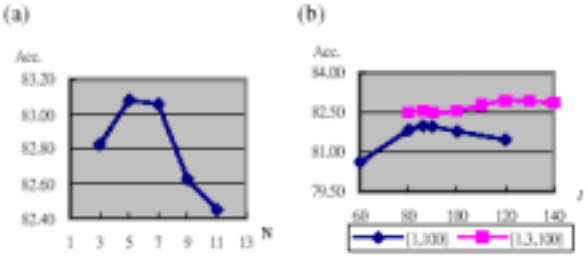

Figure 4. (a) Recognition accuracy for $HOCMN_{[I,N,100]}$ for odd order N=3,5,7,9,11, I_N=130, I_{100}=86 , (b) Recognition accuracy for $HOCMN_{[I,100]}$ with different I_{100} and for $HOCMN_{[I,3,100]}$ with different I_3 and I_{100}=86

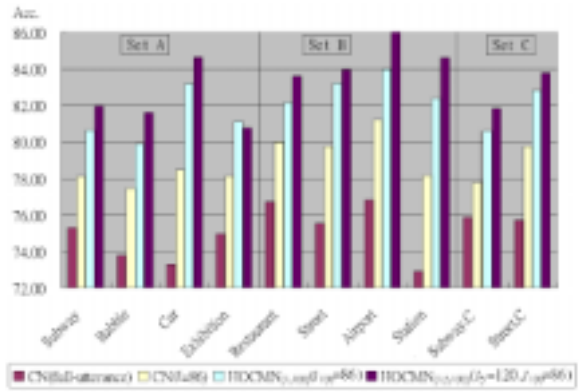

Figure 5. Comparison of several representative cases tested here for different types of noise in different testing sets, averaged over all SNR values

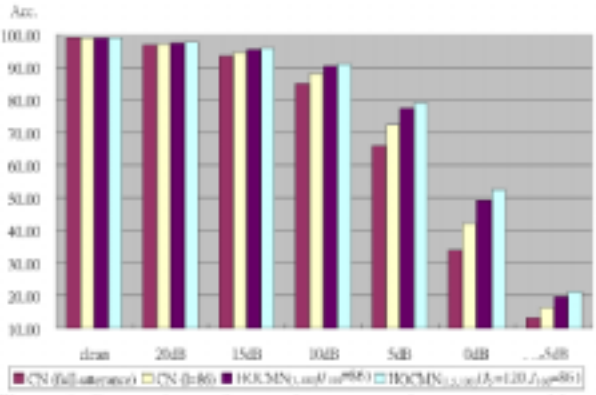

Figure 6. Comparison of several representative cases tested here for different SNR values, averaged over all different types of noise in different testing sets

4.3. Further improvements with HOCMN

The best result obtained above, the last point for N=100 for curve (c) in Figure 3, is with $HOCMN_{[I,3,N]}$, I_3=I_{100}=*86*, where I_N is the length I for the processing interval for moment normalization of order N. But from section 2.2, moment normalization can also be performed with other odd numbers. We thus further extended this best result obtained above to $HOCMN_{[1,N,100]}$, I_N=*130*, I_{100}=*86*, where N=*3, 5, 7, 9, 11*, and the results are shown in Figure 4 (a). It can be found that different N gives slightly different performance, and N=*5* is the best. On the other hand, because the segmental processing intervals does give better results than full-utterance intervals, and the length I=*86* or *130* frames used previously was obtained empirically with some initial experiments, further experiments were therefore performed. The results for $HOCMN_{[I,100]}$, with different length I_{100} was therefore obtained and plotted as the lower curve in Figure 4 (b), and those for $HOCMN_{[I,3,100]}$, I_{100}=*86* and different I_3 as the upper curve in the figure. It is clear from the two curves in the figure that the performance really depends on the segment length I, and I=*86* frames may be a good choice for N=*100* (in fact this was used for all even N as shown in Figure 3), and I=*120* may be a good number for N=*3* or other odd integers. The best result obtained here is therefore an accuracy of 83.26 with $HOCMN_{[I,3,100]}$ with I_3=*120*, I_{100}=*86*, which represents an relative error rate reduction of 32.83% as compared to the conventional CN with a full-utterance processing interval (row (3) of Table 1), or an error rate reduction of 20.78% as compared to CN with a segmental processing interval (row (4) of Table 1).

Figure 5 and 6 then compare the several representative cases tested here in this paper: CN (full-utterance), CN (I=*86*), $HOCMN_{[I,100]}$ (I_{100}=*86*), and $HOCMN_{[I,3,100]}$ (I_3=*120*, I_{100}=*86*), either averaged for all SNR values but separated for all noise types and test sets, or averaged for all noise types and test sets but separated for all SNR values. The improvements obtained with the approaches proposed here in this paper are quite obvious in all cases. In addition, from Figure 6 it is found that reasonable performance is achievable when SNR is 10dB or higher. For low SNR cases (e.g. 5dB or lower), the performance is degraded inevitably. But some noise reduction techniques can be used in the front-end and some addition robustness approaches such as temporal filtering can be used after HOCMN to provide better performance.

5. CONCLUSION

In this paper, we proposed a unified framework for higher order cepstral moment normalization. Experimental results verified that improved robustness is achievable especially under highly mismatched conditions. It is also easy to integrate this approach with other temporal filtering approaches or noise reduction front-end techniques.

6. REFERENCES

[1] S. Furui, "Cepstral Analysis Technique for Automatic Speaker Verification", IEEE Trans. Acoust. Speech Signal Process. 1981

[2] O. Viikki, K. Laurila, "Cepstral Domain Segmental Feature Vector Normalization for Noise Robust Speech Recognition", Speech Communication, Vol. 25, pp. 133-147, August 1998.

[3] Y. H. Suk, S. H. Choi, H. S. Lee, "Cepstrum Third-order Normalization Method for Noisy Speech Recognition", Electronics Letters, Vol. 35, no. 7, pp. 527-528, April 1999.

[4] H. G. Hirsch, D. Pearce, "The AURORA Experimental Framework for the Performance Evaluations of Speech Recognition Systems under Noisy Conditions", ISCA ITRW ASR2000, Paris, September 2000.