TABLE IV
RECOGNITION RATES FOR THE OSALPC-II TECHNIQUE FOR SEVERAL PREDICTION ORDER VALUES AND CEPSTRAL LIFTERS

ORDER	LIFTERING / SNR(dB)	CLEAN	20	10	0
8	BANDPASS	97.3	95.5	82.6	44.2
	ISD	97.0	96.4	86.4	52.5
	SLOPE	97.6	97.0	92.5	76.0
12	BANDPASS	98.8	97.2	94.1	71.1
	ISD	98.8	98.3	93.3	68.4
	SLOPE	99.4	98.4	94.7	72.2
16	BANDPASS	99.3	98.7	94.4	76.8
	ISD	99.1	98.1	92.4	72.7
	SLOPE	99.1	98.1	90.7	68.3

a block diagram for the calculation of the LPC, SMC, OSALPC-I, and OSALPC-II cepstra that permits comparison of their respective algorithms.

The OSALPC and SMC representations clearly outdo the conventional LPC technique in severe noisy conditions: OSALPC-I and OSALPC-II rates are better than LPC ones at 10 and 0 dB, and SMC outperforms LPC at 0 dB. Moreover, OSALPC-I and OSALPC-II representations outperform the SMC technique in all noisy conditions. For the OSALPC representation, the use of the conventional biased autocorrelation estimator for computing the OSA sequence (version OSALPC-I) is convenient in severe noisy conditions, i.e., for an SNR of 10 or 0 dB.

However, in noise-free conditions, there is a loss of recognition performance in the OSALPC and SMC approaches with respect to the conventional LPC technique due to the imperfect deconvolution of the speech signal performed by those techniques. This effect seems to be minimized by using the coherence estimator to compute the OSA sequence, as in the case of OSALPC-II and SMC.

Finally, Table IV shows the recognition rates corresponding to OSALPC-II for the same model orders and cepstral lifters as in Table I. It can be noticed that the new technique is less sensitive to changes in both the model order and the type of cepstral lifter than the conventional LPC approach, provided that the model order is not too low.

IV. CONCLUSIONS

In this correspondence, several LPC-based techniques that work in the autocorrelation domain are presented and compared in noisy speech recognition. The OSALPC technique, which is based on the application of the (windowed) autocorrelation method of linear prediction to the one-sided autocorrelation sequence, yields the best results among all the compared LPC-based techniques in severe noisy conditions.

REFERENCES

[1] F. Itakura, "Minimum prediction residual principle applied to speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-23, pp. 67–72, 1975.

[2] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-28, pp. 357–366, 1980.

[3] D. Mansour and B. H. Juang, "The short-time modified coherence representation and its application for noisy speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 37, pp. 795–804, 1989.

[4] J. A. Cadzow, "Spectral estimation: An overdetermined rational model equation approach," *Proc. IEEE,* vol. 70, pp. 907–939, 1982.

[5] J. Hernando and C. Nadeu, "Speech recognition in noisy car environment based on OSALPC representation and robust similarity measuring techniques," in *Proc. ICASSP'94,* Adelaide, Apr. 1994, pp. 69–72.

[6] M. A. Lagunas and M. Amengual, "Non-linear spectral estimation," in *Proc. ICASSP'87,* Dallas, Apr. 1987, pp. 2035–2038.

[7] D. P. McGinn and D. H. Johnson, "Reduction of all-pole parameter estimation bias by successive autocorrelation," in *Proc. ICASSP'83,* Boston, Apr. 1983, pp. 1088–1091.

[8] S. L. Marple, Jr., Ed., *Digital Spectral Analysis with Applications.* Englewood Cliffs, NJ: Prentice-Hall, 1987.

[9] C. Nadeu, J. Pascual, and J. Hernando, "Pitch determination using the cepstrum of the one-sided autocorrelation sequence," in *Proc. ICASSP'91,* Toronto, Canada, May 1991, pp. 3677–3680.

[10] B. H. Juang, L. R. Rabiner, and J. G. Wilpon, "On the use of band-pass liftering in speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-35, pp. 947–954, 1987.

[11] Y. Tohkura, "A weighted cepstral distance measure for speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-35, pp. 1414–1422, 1987.

[12] B. A. Hanson and H. Wakita, "Spectral slope distance measures with linear prediction analysis for word recognition in noise," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-35, pp. 968–973, 1987.

A Fast Algorithm for Finding the Adaptive Component Weighted Cepstrum for Speaker Recognition

Mihailo S. Zilovic, Ravi P. Ramachandran, and Richard J. Mammone

***Abstract*— In speaker recognition systems, the adaptive component weighted (ACW) cepstrum has been shown to be more robust than the conventional linear predictive (LP) cepstrum. The ACW cepstrum is derived from a pole-zero transfer function whose denominator is the pth-order LP polynomial $A(z)$. The numerator is a $(p-1)$th-order polynomial that is up to now found as follows. The roots of $A(z)$ are computed, and the corresponding residues obtained by a partial fraction expansion of $1/A(z)$ are set to unity. Therefore, the numerator is the sum of all the $(p-1)$th-order cofactors of $A(z)$. In this correspondence, we show that the numerator polynomial is merely the derivative of the denominator polynomial $A(z)$. This greatly speeds up the computation of the numerator polynomial coefficients since it involves a simple scaling of the denominator polynomial coefficients. Root finding is completely eliminated. Since the denominator is guaranteed to be minimum phase and the numerator can be proven to be minimum phase, two separate recursions involving the polynomial coefficients establishes the ACW cepstrum. This new method, which avoids root finding, reduces the computer time significantly and imposes negligible overhead when compared with the approach of finding the LP cepstrum.**

I. INTRODUCTION

Speaker recognition is the task of identifying a speaker by his or her voice [1]. A common problem in realizing robust speaker recognition systems is that a mismatch in training and testing conditions seriously degrades the performance [2]. One of the pursued approaches to

Manuscript received February 3, 1995; revised July 13, 1996. The associate editor coordinating the review of this paper and approving it for publication was Dr. Joseph Campbell.

M. S. Zilovic is with Bell Communications Research, Red Bank, NJ USA.

R. P. Ramachandran and R. J. Mammone are with the CAIP Center, Department of Electrical Engineering, Rutgers University, Piscataway, NJ 08855 USA.

Publisher Item Identifier S 1063-6676(97)00762-1.

1063–6676/97$10.00 © 1997 IEEE

keep the performance high is to make the features used for speaker recognition more robust in that these features would show relatively less variation for different conditions. Recently, the use of an adaptive component weighted (ACW) cepstrum has been shown to be less susceptible to channel effects than the conventionally used linear predictive (LP) cepstrum [3]. Unlike the LP cepstrum, the ACW cepstrum involves polynomial root finding, which demands a much greater computational burden and can be numerically difficult if two or more roots are close together. In this correspondence, we show that the ACW cepstrum can be efficiently computed without any root finding. This much faster algorithm is based on an important mathematical result concerning the derivative of the LP polynomial.

II. Concept of ACW

A pth-order LP analysis of speech leads to an LP polynomial $A(z)$ and an all-pole model $H(z) = 1/A(z)$ of the speech spectrum. The polynomial $A(z)$ is expressed as

$$A(z) = 1 - \sum_{k=1}^{p} a_k z^{-k} = \prod_{k=1}^{p} (1 - f_k z^{-1}) \tag{1}$$

which in turn can be guaranteed to be minimum phase by the autocorrelation method of LP analysis. The conventional LP cepstrum $c_{lp}(n)$ is defined for $n > 0$ and can be found by a recursion involving the coefficients a_k as given by [4]

$$c_{lp}(n) = a_n + \sum_{k=1}^{n-1} \left(\frac{k}{n}\right) c_{lp}(k) a_{n-k}. \tag{2}$$

The approach in [3] is to first perform a partial fraction expansion of $H(z)$ to get

$$\begin{aligned} H(z) &= \sum_{k=1}^{p} \frac{\lim_{z \to f_k} [(1 - f_k z^{-1})/A(z)]}{1 - f_k z^{-1}} \\ &= \sum_{k=1}^{p} \frac{r_k}{1 - f_k z^{-1}}. \end{aligned} \tag{3}$$

The experiments in [3] reveal that the residues r_k show considerable variations, especially for nonformant poles when the speech is degraded by a channel. Therefore, the variations in r_k were removed by forcing r_k to be equal to 1 for every k. Hence, we get a pole-zero system function of the form

$$H_{\text{acw}}(z) = \frac{N(z)}{A(z)} = \sum_{k=1}^{p} \frac{1}{1 - f_k z^{-1}} \tag{4}$$

where

$$N(z) = \sum_{k=1}^{p} \prod_{i=1, i \neq k}^{p} (1 - f_i z^{-1}) \tag{5}$$

which can be further written as

$$N(z) = p\left(1 - \sum_{k=1}^{p-1} b_k z^{-k}\right). \tag{6}$$

Note that $N(z)$ is the sum of all the $(p-1)$th-order cofactors of $A(z)$. Let $c_{\text{acw}}(n)$ be the ACW cepstrum (corresponding to $H_{\text{acw}}(z)$) and $C_{\text{acw}}(z)$ be the z-transform of $c_{\text{acw}}(n)$. From (4), we get

$$C_{\text{acw}}(z) = \log H_{\text{acw}}(z) = \log \frac{N(z)}{A(z)} = \log \frac{1}{A(z)} - \log \frac{1}{N(z)} \tag{7}$$

where log refers to the natural logarithm throughout this correspondence. It will be justified later in this correspondence that $N(z)$ is minimum phase. Therefore, $H_{\text{acw}}(z)$ has all its poles and zeros inside the unit circle. This implies that the ACW cepstrum is causal [4] and given by $c_{\text{acw}}(0) = \log\ p$, and

$$c_{\text{acw}}(n) = c_{lp}(n) - c_{nn}(n) \tag{8}$$

for $n > 0$, where $c_{nn}(n)$ is the cepstrum corresponding to $1/N(z)$. Note that $c_{nn}(n)$ can be found by a recursion [4] involving the coefficients b_k, just as in (2).

It must be noted that the present method of finding $c_{\text{acw}}(n)$ from $A(z)$ involves the following steps [3]:

1) Find $c_{lp}(n)$ from a_k.
2) Determine the roots of $A(z)$.
3) Find all the cofactors of $A(z)$ of order $p-1$, and add them up to get $N(z)$.
4) Find $c_{nn}(n)$ from b_k.
5) Find $c_{\text{acw}}(n) = c_{lp}(n) - c_{nn}(n)$.

Steps 2 and 3 are mainly responsible for the increase in computational burden over merely finding $c_{lp}(n)$. As we shall see later, this increase is by a factor of 1.4. With the fast algorithm, we propose, the increase in computation is a very small factor of 1.02.

III. Mathematical Definition of Numerator Polynomial

Theorem: Every single coefficient b_k of $N(z)$ in (6) is of the form

$$b_k = \frac{p-k}{p} a_k \tag{9}$$

$\forall k, 1 \leq k \leq p-1$, where a_k is the kth coefficient of the LP polynomial $A(z)$ (see (1)).

Proof: Let us rewrite $A(z)$ in terms of positive powers of z as

$$A(z) = z^p - \sum_{k=1}^{p} a_k z^{p-k} = \prod_{k=1}^{p} (z - f_k). \tag{10}$$

Then, $\log A(z)$ can be expressed as

$$\log A(z) = \sum_{k=1}^{p} \log(z - f_k). \tag{11}$$

Differentiating both sides of the equation gives

$$\frac{A'(z)}{A(z)} = \sum_{k=1}^{p} \frac{1}{(z - f_k)} \tag{12}$$

where $A'(z)$ is the derivative of $A(z)$ with respect to z. By comparing (4) and (12), we see that $N(z)$, when written in terms of positive powers of z, is equal to $A'(z)$, which can be written as

$$\begin{aligned} N(z) &= pz^{p-1} - \sum_{k=1}^{p-1} (p-k) a_k z^{p-k-1} \\ &= p\left[z^{p-1} - \sum_{k=1}^{p-1} \frac{(p-k)}{p} a_k z^{p-1-k}\right]. \end{aligned} \tag{13}$$

Therefore, the coefficients b_k in (6) are given as in the theorem.

IV. Minimum-Phase Property of Numerator Polynomial

In order to define a causal ACW cepstrum as in (8), it is necessary and sufficient that $N(z)$ be minimum phase [4]. The minimum phase property of $N(z)$ is clearly established by quoting the theorem below.

Theorem [5]: Any circle that encloses all the zeros of a polynomial also encloses all the zeros of its derivative.

From the theorem, the minimum-phase property of $A(z)$ ensures the minimum-phase nature of $N(z)$.

V. Improved Algorithm and Computer Time

For the improved fast method of finding $c_{\rm acw}(n)$, the following steps are involved:

1) Find $c_{lp}(n)$ from a_k.
2) Find b_k from a_k. This gives $N(z)$.
3) Find $c_{nn}(n)$ from b_k.
4) Find $c_{\rm acw}(n) = c_{lp}(n) - c_{nn}(n)$.

Speech sampled at 8 kHz served as input to a system that does LP analysis and converts the LP coefficients to either the conventional LP cepstrum or the ACW cepstrum. An optimized software code that implements the above system was run on a SPARC10. Three different scenarios were compared in terms of CPU time. In scenario 1, the LP coefficients were transformed into $c_{lp}(n)$ via the well-known recursion. In scenario 2, the LP coefficients were transformed into $c_{\rm acw}(n)$ by the method offered in this correspondence for finding $N(z)$ and employing two separate recursions on $N(z)$ and $A(z)$ to get the respective cepstra. In scenario 3, the LP coefficients were again transformed into $c_{\rm acw}(n)$, but unlike scenario 2, $N(z)$ was found (as suggested in [3]) by a standard polynomial root-finding program [6]. The ratio of the required computer time for going from speech to cepstral features through scenarios 1, 2, and 3 is 1:1.02:1.40. This shows that our proposed method is much faster than doing polynomial root finding. In addition, the more robust ACW cepstrum can be obtained by a negligible overhead, as compared with the conventional LP cepstrum.

VI. Summary and Conclusions

The contribution of this correspondence is in achieving a fast procedure to calculate the ACW cepstrum from the LP coefficients. The ACW cepstrum is found from a pole-zero transfer function in which the denominator is the LP polynomial, and the numerator is what we show to be the derivative of the LP polynomial. The numerator polynomial is found by a simple scaling of the denominator polynomial coefficients and does not necessitate any root-finding method. In addition, the numerator polynomial is guaranteed to be minimum phase. The ACW cepstrum is computed by two separate recursions based on the polynomial coefficients. In fact, these recursions are independent and can be implemented in parallel. Simulations show that the additional computer time needed for finding the ACW cepstrum as compared with the LP cepstrum is negligible. A more robust feature is obtained with very little extra computation.

References

[1] G. R. Doddington, "Speaker recognition—Identifying people by their voices," *Proc. IEEE*, vol. 73, pp. 1651–1664, Nov. 1985.

[2] A. E. Rosenberg and F. K. Soong, "Recent research in automatic speaker recognition," in *Advances in Speech Signal Processing*, S. Furui and M. M. Sondhi, Eds. New York: Marcel Dekker, 1991, pp. 701–738.

[3] K. T. Assaleh and R. J. Mammone, "New LP-derived features for speaker identification," *IEEE Trans. Speech Audio Processing*, vol. 2, pp. 630–638, Oct. 1994.

[4] L. R. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[5] M. Marden, *Geometry of Polynomials*. New York: Amer. Math. Soc., 1966.

[6] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*. Cambridge, UK: Cambridge Univ. Press, 1992.