ISCAS 2000 - IEEE International Symposium on Circuits and Systems, May 28-31, 2000, Geneva, Switzerland

FAST POLE FILTERING FOR SPEAKER RECOGNITION

Ravi P. Ramachandran

Electrical Engineering, Rowan University
201 Mullica Hill Road
Glassboro, New Jersey 08028
ravi@rowan.edu

Kevin R. Farrell

T-NETIX Inc.
67 Inverness Drive East
Englewood, Colorado 80112
kevin.farrell@t-netix.com

ABSTRACT

Mismatched training and testing conditions for speaker recognition exist when speech is subjected to a different channel for both cases. This results in diminished speaker recognition performance. Estimating and removing the channel filtering effect will make speaker recognition systems more robust. It has been shown that a reliable estimate is obtained by taking the mean of the pole filtered linear predictive (LP) cepstrum. Finding the pole filtered mean requires factorization of the LP polynomial which is computationally intensive especially for real time applications. In this paper, we examine a fast method of doing pole filtering that avoids polynomial factorization. This method is much more computationally efficient and gives equal or better performance than the conventional way of doing pole filtering. Experimental results are given for four databases having a variety of mismatched conditions.

1. INTRODUCTION

Speaker recognition refers to the concept of recognizing a speaker by his/her voice or speech samples [1][2]. Some of the important applications of speaker recognition include customer verification for bank transactions, access to bank accounts through telephones, control on the use of credit cards, and for security purposes in the army, navy and airforce. The two main tasks within speaker recognition are *speaker identification* and *speaker verification.* Speaker identification (ID) deals with a situation where the person has to be identified as being one among a set of persons by using his/her voice samples. The objective of speaker verification is to verify the claimed identity of that speaker based on the voice samples of that speaker alone. A claimant speaker is either accepted or rejected by the system.

The speaker ID problem may further be subdivided into *closed set* and *open set*. The closed set speaker ID problem refers to a case where the speaker is known *a priori* to belong to a set of M speakers. In the open set case, the speaker may be out of the set and hence, a "none of the above" category is necessary. Another distinguishing aspect of speaker recognition systems is that they can either be text-independent or text-dependent depending on the application. In the text-independent case, there is no restriction on the sentence or phrase to be spoken, whereas in the text-dependent case, the input sentence or phrase is fixed for each speaker. A text-dependent scenario is commonly encountered in speaker verification systems in which a person's password must be the same for enrollment and verification and is critical for verifying his/her identity.

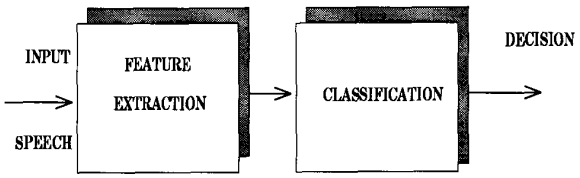


Figure 1: A general diagram of a recognition system

Speaker recognition consists of two stages, namely, *Feature extraction* and *Classification* as shown in Fig. 1. Feature extraction is associated with obtaining the characteristic patterns of the signal that are representative of the speaker in question. The parameters or features used in speaker recognition are a transformation of the speech signal into a compact acoustic representation that contains information useful for the identification of the speaker. This is often done using short-time linear predictive (LP) [3] analysis which leads to an all-pole LP vocal tract model. The LP coefficients are converted to the LP cepstrum [3] which in turn, is the feature vector. The classifier uses the features to render a decision as to the speaker identity or verifies the claimed identity of the speaker.

The recognition task is highly successful if the environmental conditions for training and testing are the same (known as matched conditions). Studies have shown that recognition performance degrades when the training and testing conditions are not the same (known as mismatched conditions) [4]. This occurs if the speaker is trained on one type of telephone (handset, cordless or speakerphone) and during the testing phase, a different type of telephone is used. In this particular case, channel mismatch is encountered and this contributes to the degradation in the performance. Channels have a filtering effect on the speech and alter the overall spectral envelope of the speech signal. Assuming that the speech and channel spectra are well approximated by the all-pole LP model, it is observed that a channel influence on the speech leads to an additive component on the LP cepstrum. Estimating and removing this additive channel component will mitigate the channel effect and make speaker recognition systems more robust. One method of estimating the additive channel component is to take the mean of the LP cepstrum vectors over an utterance [5]. It has been shown that a better estimate is obtained by taking the mean of the pole filtered LP cepstrum (described in detail later) [6]. Finding the pole filtered mean requires factorization of the LP polynomial which is computationally intensive especially for real time applications. In this pa-

0-7803-5482-6/99/$10.00 ©2000 IEEE

per, we examine a fast method of doing pole filtering that avoids polynomial factorization [7]. This method is much more computationally efficient and gives equal or better performance than the conventional way of doing pole filtering.

2. FEATURE EXTRACTION

The autoregressive LP model for speech is given by the difference equation [3]

$$s(n) = \sum_{i=1}^{p} a_i s(n-i) + e(n) \tag{1}$$

where $s(n)$ is the speech signal, $e(n)$ is the prediction error and a_i are the predictor coefficients. It can be noted that $s(n)$ is predicted as a linear combination of the previous p samples. The all-pole LP transfer function is given by

$$H(z) = \frac{S(z)}{E(z)} = \frac{1}{A(z)} = \frac{1}{1 - \sum_{i=1}^{p} a_i z^{-i}}. \tag{2}$$

where $S(z)$ and $E(z)$ are the z-transforms of $s(n)$ and $e(n)$ respectively. In practice, the predictor coefficients a_i are computed over short intervals (typically 10 ms to 30 ms) called frames during which the vocal tract configuration is assumed to be stationary. This is done using the autocorrelation method [3][8] which guarantees that $H(z)$ is a stable function.

The predictor coefficients a_i are converted to the LP cepstrum $clp(n)$ $(n \geq 1)$ by [8]

$$clp(n) = \frac{1}{n} \sum_{i=1}^{p} p_i^n \tag{3}$$

where p_i are the poles of $H(z)$ $(|p_i| < 1)$. A more efficient recursive relation between the LP cepstrum and the predictor coefficients is given as [3][8]

$$clp(n) = a_n + \sum_{i=1}^{n-1} (\frac{i}{n}) clp(i) a_{n-i} \tag{4}$$

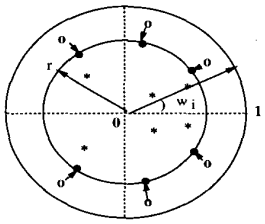
Since $clp(n)$ is of infinite duration, the feature vector of dimension p consists of the components $clp(1)$ to $clp(p)$ which are the most significant due to the decay of the sequence with increasing n.

2.1. Cepstral Mean

As mentioned earlier, when speech is subjected to channel interference, an additive component due to the channel manifests itself on the LP cepstrum. To compensate for the channel effect, this component is estimated as the mean of the LP cepstrum and removed by subtraction (known as cepstral mean subtraction (CMS)). The new feature vector is

$$ccms(n) = clp(n) - E[clp(n)] \tag{5}$$

where the expectation is taken over an utterance consisting of a number of frames.



r : threshold radius
o : old poles
● : filtered poles
* : poles within the threshold radius

Figure 2: Concept of pole filtering.

2.2. Conventional Pole Filtering

The LP poles p_i with narrow bandwidths that lie close to the unit circle usually represent the formants and are less sensitive to channel and noise effects. Hence, these poles do not contribute to the channel estimate as they contain much speech information. In contrast, the broad bandwidth poles model the spectral tilt, sub-glottal variation and the channel effects. These poles offer a better estimate of the channel. Pole filtering, modifies the LP poles so as to broaden the bandwidth of the formant poles [6]. Bandwidth broadening is accomplished by moving the formant poles radially away from the unit circle towards the origin. The pole frequency is left intact. Figure 2 illustrates the concept of pole filtering. The cepstrum (denoted as $cmlp(n)$) formed from these filtered or modified poles has less speech information and more channel information. Hence, a much better channel estimate in the form of $E[cmlp(n)]$ is found due to the deemphasis of the formant poles.

The technique of forming the feature vector is known as pole filtered cepstral mean subtraction (PFCMS) and the details are given below.

- Select a threshold radius α.
- For each frame of speech:
  - Calculate LP poles p_i for $i = 1$ to p.
  - For each pole p_i:
    - If $|p_i| > \alpha$, modify p_i such that its magnitude is α and its angle is unaltered.
  - Calculate $cmlp(n)$ based on the modified or filtered poles using Eq. (3).
- Find the channel estimate $E[cmlp(n)]$ over all speech frames.
- Find the feature vector $cpfcms(n) = clp(n) - E[cmlp(n)]$.

2.3. Fast Pole Filtering

Broadening the bandwidth of the formant poles can also be performed by transforming the LP polynomial so as to weight the predictor coefficients as given by

$$H(z/\gamma) = \frac{1}{A(z/\gamma)} = \frac{1}{1 - \sum_{i=1}^{p} a_i \gamma^i z^{-i}}. \tag{6}$$

where $0 < \gamma \leq 1$ [7]. Given the original LP poles p_i, the new set of poles are γp_i. In contrast to conventional pole filtering, all the poles move radially inward by a factor γ. The cepstrum formed from these modified poles (denoted as $cflp(n)$) is merely related to the LP cepstrum as

$$cflp(n) = \frac{1}{n}\sum_{i=1}^{p}(\gamma p_i)^n = \frac{\gamma^n}{n}\sum_{i=1}^{p}p_i^n = \gamma^n clp(n) \quad (7)$$

Results (next section) show that even though the poles that model the channel estimate are perturbed, the channel estimate is not affected. Moreover, the performance improves synergistically with lower computational burden especially since no polynomial factorization is required.

The much simpler technique of forming the feature vector is known as fast pole filtered cepstral mean subtraction (FPFCMS) and the details are given below.

- Select the parameter γ.
- For each frame of speech, calculate $clp(n)$ and $cflp(n)$ using Eq. (7).
- Find the channel estimate $E[cflp(n)]$ over all speech frames.
- Find the feature vector $cfpfcms(n) = clp(n) - E[cflp(n)]$

3. EXPERIMENTAL RESULTS

The first experiment is on comparing the computational complexity of the conventional and fast pole filtering approaches. All computations were carried out using MATLAB which gives us the number of floating point operations (flops). A 12th order LP analysis was done on 1 second of 8 kHz sampled speech partitioned into 30 ms frames having an overlap of 20 ms. Both fast and conventional pole filtering was done using the predictor coefficients. The ratio of the number of flops for doing LP analysis and finding the feature vectors using the conventional and fast methods for the entire speech signal is about 3.1:1. The ratio of the number of flops for finding the feature vectors from the predictor coefficients using the conventional and fast methods for the entire speech signal is about 1600:1. The fast method is clearly more efficient especially since no polynomial factorization is involved.

Closed set, text-independent speaker identification experiments are carried out using the TIMIT database. Thirty eight speakers from the New England dialect are considered. The speech is downsampled from 16 kHz to 8 kHz. For each speaker, there are 10 sentences. The first five are used for training a vector quantizer (VQ) classifier using the Linde-Buzo-Gray (LBG) method [9] and the squared Euclidean distance as the distortion measure. A VQ codebook is designed for each of the 38 speakers. The training conditions include clean speech and speech subjected to representative bandpass telephone channels [10]: (1) the Continental Mid Voice (CMV) channel, (2) the Continental Poor Voice (CPV) channel, (3) the European Mid Voice (EMV) channel and (4) the European Poor Voice (EPV) channel. The remaining five sentences are individually used for testing thereby giving 190 test utterances.

The testing conditions correspond to channel corrupted speech. Consider a particular test feature vector. This is quantized by each of the 38 codebooks. The quantized vector is that which is closest (according to the squared Euclidean distance) to the test feature vector. Hence, 38 different distances are recorded, one for each codebook. This process is repeated for every test feature vector.

Training Condition	Testing Condition	CMS ISR	PFCMS Conventional α, ISR	FPFCMS Fast γ, ISR
Clean	CMV	53.7	0.95, 56.8	0.90, 58.4
Clean	CPV	53.2	0.85, 58.9	0.90, 59.5
CMV	CPV	61.6	0.85, 70.0	0.85, 66.3
CPV	CMV	58.4	0.80, 64.2	0.80, 66.8
Clean	EMV	57.9	0.80, 65.8	0.80, 65.8
Clean	EPV	57.9	0.85, 68.9	0.85, 68.9
EMV	EPV	61.1	0.80, 69.5	0.80, 74.2
EPV	EMV	56.8	0.80, 71.1	0.80, 71.1

Table 1: Speaker identification success rate (ISR) as a percent. The best values of α and γ are shown. The acronym CMS is for cepstral mean subtraction. The acronym PFCMS is for pole filtered cepstral mean subtraction (conventional approach). The acronym FPFCMS is for fast pole filtered cepstral mean subtraction (the fast approach).

The distances are accumulated over the entire set of feature vectors. The codebook which renders the smallest accumulated distance identifies the speaker. The identification success rate (ISR) is the number of utterances for which the speaker is identified correctly divided by the total number of utterances tested. The VQ codebook size is 64. For both training and testing, silent or low-energy frames are discarded by energy thresholding. Also, a 12th order LP analysis is used with 30 ms frames having an overlap of 20 ms.

Table 1 shows the results for speaker identification. Different values of α (conventional approach) and γ (fast approach) were tried. The best values of α and γ are from 0.80 to 0.95. In Table 1, the result for the best values of α and γ are given for each training/testing combination. The performance of the fast method is equal to or better than the conventional approach (except for the CMV/CPV combination) and simultaneously offers great computational savings.

The T-NETIX Voiceprint text-dependent speaker verification system based on a user supplied password [11] is used to further test our fast pole filtering approach. Both the neural tree network (NTN) [12] and Gaussian mixture model (GMM) [13] classifiers are used. During training, the password is segmented into subwords using the blind segmentation algorithm in [14]. An NTN and GMM model is trained for each subword. A leave-one-out strategy [15] is deployed in that a password is repeated N times to train N subword models for both classifiers. Each subword model is trained with $N-1$ repetitions with a different repetition "left-out" for each model. In our experiments, $N = 3$. The left-out repetition is applied as a test utterance to get an unbiased score that is used to set the threshold for accepting or rejecting a claimant speaker. The mean and diagonal covariance matrix of the feature are used to obtain the GMM parameters. Hence, the GMM is trained only using the speech of the speaker being trained. For the NTN, both speaker and anti-speaker speech data are used to obtain the hyperplanes that partition the feature space into feature and anti-speaker feature vectors. The anti-speaker data corresponds to the subwords of other speakers enrolled in the database from which the extracted feature vectors are close to the feature vectors of the subword of the speaker being trained.

During testing, a test utterance and a claimed speaker identity

Database	# Enrolled Speakers	# Development Speakers	# True Trials	# Impostor Trials
Landline	56	47	195	11,129
Multimedia	50	30	90	1,638
Wireless	26	15	273	6,825

Table 2: Database specifications.

Database	Mean Removal EER	Conventional Pole Filtering α, EER	Fast Pole Filtering γ, EER
Landline	1.33	0.85, 1.09	0.95, 1.13
Multimedia	3.08	0.85, 1.75	0.90, 1.44
Wireless	7.44	0.95, 6.36	0.80, 6.07

Table 3: The EER as a percent for the three databases.

are the inputs. The utterance is segmented [14] into the same number of subwords as for the claimed speaker that was done during training. Each subword is scored by (1) the corresponding N subword NTN models to obtain an overall average NTN score and by (2) the corresponding N subword GMM models to obtain an overall average GMM score. Both the scores are in the range 0 to 1 (like a probability). The average of the overall NTN and GMM scores is the final score which is compared against a threshold to decide upon acceptance or rejection. For both training and testing, the feature vectors are computed for each subword using a 12th order LP analysis with 30 ms frames having a 20 ms overlap. Energy thresholding eliminates the silent frames.

Three databases are used to obtain the results. The landline database is configured by collecting speech data over a standard telephone having an electret microphone. Each speaker has the same password "open sesame". The multimedia database is configured by collecting speech data over a Lucent noise canceling microphone connected to a personal computer. Each speaker has the same password "open the door". The wireless database is configured by collecting speech data over a cellular telephone and hence, has the most severe channel effect. Each speaker has the same password "Al Capone". Table 2 gives, for each database, the number of enrolled speakers, the number of development speakers (used as anti-speaker data in training the NTN), the number of true trials (test speaker matches the claimed identity) and the number of impostor trials (test speaker does not match the claimed identity). Note that the development speakers are not used as impostor trials.

Two types of errors occur in speaker verification. The first is known as a *false accept* (FA) and occurs when the user is accepted as the claimed speaker but in fact, is not the claimed speaker. This is when an imposter breaks in to the system. The second is known as a *false reject* (FR) and occurs when the user is rejected as the claimed speaker but is in fact the claimed speaker. In the experiments, the threshold for acceptance/rejection is varied to get different FA and FR rates. Then, a receiver operating curve (ROC) curve is obtained as a plot of the FA rate versus FR rate for various thresholds. The point on the ROC curve when the FA rate equals the FR rate is known as the equal error rate (EER) and is the performance measure used. Table 3 gives the EER results for the best values of α and γ which again occur between 0.80 and 0.95. The fast pole filtering approach is generally better especially for a severe channel corruption manifested in the wireless database. Our extensive experimental results show that the fast method of pole filtering gives equal or better performance than the conventional method.

4. REFERENCES

1. G. R. Doddington, "Speaker recognition - identifying people by their voices" *Proc. IEEE*, vol. 73, pp. 1651–1664, November 1985.
2. A. E. Rosenberg, "Automatic speaker verification: A review", *Proc. IEEE*, vol. 64, pp. 475–487, April 1976.
3. L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Prentice-Hall, 1978.
4. A. E. Rosenberg and F. K. Soong, "Recent research in automatic speaker recognition", in *Advances in Speech Signal Processing*, edited by S. Furui and M. M. Sondhi, Marcel Dekker, pp. 701–738, 1991.
5. B. S. Atal, ""Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification", *Jour. of the Acoust. Soc. of Amer.*, vol. 55, pp. 1304–1312, June 1974.
6. D. Naik and R. J. Mammone, "Pole filtered cepstral mean subtraction", *IEEE Int. Conf. on Acoust., Speech and Sig. Proc.*, Detroit, Michigan, pp. 157–160, May 1995.
7. D. Naik and R. J. Mammone, "Channel normalization using pole filtered cepstral mean subtraction", *SPIE Int. Symp. on Optics, Imaging and Instrumentation*, vol. 2277, San Diego, California, pp. 99–110, July 1994.
8. L. R. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*, Prentice-Hall, 1993.
9. Y. Linde, A. Buzo and R. M. Gray, "An algorithm for vector quantizer design", *IEEE Trans. on Comm.*, vol. COM-28, pp. 84–95, Jan. 1980.
10. J. Kupin, "A wireless simulator (software)," CCR-P, April 1993.
11. K. R. Farrell, R. P. Ramachandran and R. J. Mammone, "An analysis of data fusion methods for speaker verification", *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Seattle, Washington, pp. II-1129–II-1132, May 12–15, 1998.
12. A. Sankar and R. J. Mammone, "Growing and pruning neural tree networks", *IEEE Transactions on Computers*, vol. C-42, pp. 221–229, March 1993.
13. D. A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models", *Speech Communication*, vol. 17, pp. 91–108, March 1995.
14. M. Sharma and R. J. Mammone, "Blind speech segmentation: Automatic segmentation of speech without linguistic knowledge", *Int. Conf. on Spoken Language Proc.*, Philadelphia, Pennsylvania, October 1996.
15. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis* John Wiley and Sons, 1973.