Speaker Recognition Using Auditory Features and Polynomial Classifier

Pawan K. Ajmera
Vishwakarma Institute of Technology
Pune, India

Raghunath S. Holambe
S. G. G. S. I. E & T. Vishnupuri,
Nanded, India

ABSTRACT
This paper presents a speaker recognition method which makes use of auditory features and polynomial classifier for speaker recognition. Auditory features based on an auditory periphery model extract significant speaker characteristics. Polynomial classifier has been used to accomplish speaker recognition task. Polynomial classifier has several advantages over the conventional classifiers such as computational scalability with the number of speakers, discriminative training allowing it to use out of class data and the statistical interpretation of scoring allowing it to combine with HMM and GMM. This approach achieves substantial performance improvement in a speaker identification task compared with state-of-the-art in a wide range of signal to noise conditions.

Categories and Subject Descriptors
I.2.7 [**Artifical Intelligence**]: Natural Language Processing – *Speech Recognition and synthesis*.

General Terms
Algorithms, Design, Verification.

Keywords
Speaker recognition, Auditory features, Polynomial classifier.

1. INTRODUCTION
The most widely used speaker recognition feature is Mel Frequency Cepstrum Coefficients (MFCCs). MFCCs are computed from the log energies in frequency bands distributed over a mel scale. The wide spread use of the MFCCs is due to the low complexity of the estimation algorithm and their good performance for automatic speaker recognition tasks under clean matched conditions [3]. However, MFCCs are easily affected by common frequency localized random perturbations, to which human perception is largely insensitive [11]. MFCC performance degrades rapidly in presence of noise and performance degradation is directly proportional to signal to noise ratio [16]. MFCCs lack of robustness in noisy or mismatched conditions have led many researchers to investigate robust variants of MFCCs or novel feature extraction algorithm altogether. Much of these research is motivated by models of human perception, e.g., the RASTA [8] and PLP features [7].

Many classifiers have been proposed for speaker recognition. The two most popular techniques are statistical and connectionist based methods. For the former, Gaussian mixture models (GMM) [13] and HMM systems with cohort normalization [9], [14] or background normalization [2], [15] are the techniques of choice. For connectionist systems, artificial neural networks have been commonly used; e.g., neural tree networks and multilayer perceptrons [4].

In this paper, we design a robust front-end that is motivatedfrom auditory perception and uses a dense (in frequency) bank of Gammatone filters. The filter bandwidths are proportional to the auditory Equivalent Rectangular Bandwidth (ERB) function as described in [5], [6], [10]. Specifically, a Gammatone feature (GF) is obtained from a bank of Gammatone filters, which was originally proposed to model human cochlear filtering. Then, Gammatone frequency cepstral coefficients (GFCC) are derived from GF. We find that such features achieve comparable performance under both clean and noisy conditions

In this paper, we propose text-independent speaker recognition system based upon a discriminatively trained polynomial classifier. The method is discriminative, easily handles large datasets, has low memory usage, and computationally separates the training on independent class basis. The scoring method makes the recognition problem highly computationally scalable with the number of speakers. The training framework is easily adapted to support new class addition, adaptation, and iterative methods.

2. AUDITORY FEATURES
Human auditory processing relies on a set of dense (in frequency) asymmetrical filters that estimate the activity in each frequency band. The notion of ERB can be used to quantify the bandwidth of asymmetrical filters like the auditory ones. Specifically, given the magnitude of a filters frequency response $|H\ f\ |$ and the filters maximum gain $|H\ f_{\max}\ |$ at frequency $f_{\max}$ the filters *ERB* (in *Hz*) is defined as

$$ERB = \frac{\int |H\ f\ |^2 df}{|H\ f_{\max}\ |^2} \quad (1)$$

The ERB is the equivalent bandwidth of an orthogonal filter with constant gain $|H\ f_{\max}\ |$ and energy equal to the original filters energy (the filters energy is defined as the integral of the filters frequency response squared).

Recent studies [5], [6], [10] present the ERB(*f*) function as follows

$$ERB\ \ f\ \ = 6.23\ \ f/1000\ \ ^2 + 93.39\ \ f/1000\ \ + 28.52 \quad (2)$$

where *f* is the filter center frequency in Hz. Moreover, the filter placing is equidistant in the critical (bark) frequency scale

$$bark\ \ f\ \ = \frac{26.81j}{f + 3920} - 0.53 \quad (3)$$

where $0 \le f \le F_s$ and F_s is the sampling frequency of the signal. A good approximation of the auditory filters are the asymmetrical Gammatone filters with impulse response

$$g(t) = At^{n-1} \exp\ -2\pi bERB\ \ f_c\ \ t\ \cos\ 2\pi f_c t \quad (4)$$

where *A*, *b*, *n* are the Gammatone filter design parameters and f_c is the center frequency of the filter. In [10], it is proposed that the auditory filters should have *b* = 1.019 and *n* = 4. Thus, the filter frequency response $G\ \ \varpi\ \ $ is given by

$$G\ \ \varpi\ \ = \frac{A}{2}\frac{6}{2\pi bERB\ \ f_c\ \ + j\ \ \varpi - \varpi_c\ \ ^4} + \frac{A}{2}\frac{6}{2\pi bERB\ \ f_c\ \ + j\ \ \varpi + \varpi_c\ \ ^4} \quad (5)$$

moreover, the filter gain A is set taking under consideration that $\left|H\ \ \varpi_c\ \ \right| = 1$ and is equal to

$$A = \frac{1}{\sum_{k=1}^{N} t^{n-1} \exp\ -2\pi bERB\ \ f_c\ \ t} \quad (6)$$

where *N* is the length of the discrete impulse response.

The auditory filterbank is not constant-Q and emphasizes the lower part of the frequencies where the main part of the acoustic information is located. Mel-spaced filterbanks used for MFCC feature extraction in speech recognition tasks [3] use symmetric filters and constant-Q filterbanks. The main differences between the proposed filterbank and the used for MFCC estimation are the type of filters used and their corresponding bandwidths.

The Gammatone filterbank presented above, with filters placed according to the bark scale and with bandwidths given by the *ERB(f)* is a good approximation of the human auditory system [7], [8], [5]. The human ear employs several thousand filters and the corresponding filterbank is very dense (in frequency). In this paper, we use 30 filters in filterbank (filterbank density) and 1.5 as the bandwidth multiplying factor *F*.

3. TRAINING METHOD

3.1 Gammatone Frequency Cepstral Coefficients

The Gammatone frequency cepstral coefficients (GFCC) are extracted from the speech signal according to the following steps:

1) Use the Gammatone filterbanks defined in Eqs. (4), (5) with 30 filters and the bandwidth multiplying factor *f* = 1.5 to bandpass the speech signal. The filter spacing is linear in the bark scale.

2) Estimate the logarithm of the short-time average of the energy operator for each one of the bandpass signals. The short-time averaging window duration and window shift are 20 and 10 msec respectively.

3) Estimate the cepstrum coefficients of the short-time average using the discrete cosine transform (DCT).

4) Truncate the cepstrum coefficients to keep the first 13 coefficients (including the zeroth coefficient C0).

The first two steps are the main differences between GFCC and MFCC feature extraction, namely the auditory filterbank and the short-time energy computation. The standard MFCC front-end uses filters with frequency response that is triangular in shape and constant-Q (50% filter frequency response overlap). The proposed auditory GFCCs use filters that are smoother and broader than the MFCC triangular filterbank [12] (the bandwidth of the filter is controlled by the *ERB* curve and the bandwidth multiplication factor *F*). Also, the GFCC filterbank is denser in frequency (controlled by the number of filters parameter) and spaced according to the barkscale rather than the mel-scale.

3.2 Polynomial Classifier

Training the classifier is accomplished by obtaining the optimum speaker model for each speaker. The basic method of optimizing the performance of the classifier is to use discriminative training with a mean square error criterion [1]. For the speaker's feature vectors an output of '1' is desired and for the imposter's feature vectors an output of '0' is desired. For discussion purpose lets consider a two class problem with speakers feature vectors as x1, $x_1, x_2, ..., x_{Nspk}$ and imposters features vectors as $y_1, y_2, ...y_{Nimp}$. The resulting problem then becomes,

$$W_{spk} = \arg_w \min E\ \ w^t p\ \ x\ - y\ \ \omega \quad (7)$$

where W_{spk} be the optimum speaker model, ω be the class label, and *y(ω)* be the ideal output i.e. *y(spk)* = 1 and *y(imp)* = 0. *E* denotes expectation over *x* and *w*. Using the training set this criterion can be approximated as,

$$W^* = \arg_w \min \left[\sum_{i=1}^{Nspk} \left| w^t p\ \ x_i - 1 \right|^2 + \sum_{i=1}^{Nimp} \left| w^t p\ \ y_i \right|^2 + \right] \quad (8)$$

The training method in the above form looks cumbersome and difficult to implement. Rewriting it in the matrix form makes it computationally easy to implement. An input matrix of a speaker of the following form is considered,

$$X = \begin{bmatrix} x_{1_1} & x_{2_1} & \cdots & x_{Nframe_1} \\ \vdots & \vdots & \cdots & \vdots \\ x_{1_D} & x_{2_D} & \cdots & x_{Nframe_D} \end{bmatrix} \quad (9)$$

where *D* is the number of features and *Nframes* is the number of feature vectors, equal to the number of frames we divide our sample data into. A matrix M_{spk} is the matrix whose rows are the vectors of the polynomial basis terms of the speakers feature vectors is defined as follows,

$$M_{spk} = \begin{bmatrix} p\ x_1^{\ t} \\ p\ x_2^{\ t} \\ \vdots \\ p\ x_{Nframe}^{\ t} \end{bmatrix} \quad (10)$$

A similar matrix M_{imp} is defined for the imposter feature matrix. Combining both M_{spk} and M_{imp} we define,

$$M = \begin{bmatrix} M_{spk} \\ M_{imp} \end{bmatrix} \quad (11)$$

Thus the training problem in Eq. (8) then becomes,

$$W^* = \arg_w \min \left\| Mw - O \right\|_2 \quad (12)$$

where *O* is the vector consisting of N_{spk} (*Nframes* for speaker data) ones followed by N_{imp} (*Nframes* for imposter data) zeros (i.e. the ideal output).

This problem can be solved using the method of normal equations, resulting in,

$$M^t M w = M^t o \quad (13)$$

Substituting matrix *M* we get,

$$M_{spk}^t M_{spk} + M_{imp}^t M_{imp} \quad w = M_{spk}^t 1 \quad (14)$$

where 1 is the vector of all ones. Now, $R_{spk} = M_{spk}^t M_{spk}$ is defined and R_{imp} is defined similarly, thus Eq. (14) reduces to,

$$R_{spk} + R_{imp} \quad w = M_{spk}^t 1 \quad (15)$$

$R = R_{spk} + R_{imp}$, is defined reducing the training method to

$$Rw = M_{spk}^t 1 \quad (16)$$

Thus training of the system boils down to the calculation of *R*. The terms in the right hand side of Eq. (16) can be calculated as a submatrix of R_{spk} or *R*. This is denoted as A_{spk}. Therefore the training method i.e. the determination of the optimum speaker model thus is,

$$W_{spk} = R^{-1} A_{spk} \quad (17)$$

Extending this to the multiclass problem,

$$R = \sum_{j=1}^{Nspk} R_j = \sum_{j=1}^{Nspk} M_j^t M_j \quad (18)$$

It can be noted that the problem is now separable. Thus R_j for each speaker *j* can be individually calculated. This feature makes the addition of new speakers easier and faster. By substituting $A_{spk} \quad M_{spk}^t 1$ for each speaker we can calculate W_{spk}. The major part of the training algorithm is the calculation of *R*. The matrix *R* or R_{spk} and R_{imp} consists exactly of sums of polynomial of order $\leq$2k. It has many redundant terms. Storage space and computation time can be reduced by calculating only the unique terms. These unique terms are denoted by $p_2 \quad x$.

The vector of polynomial basis terms can be computed iteratively as discussed further. Suppose one has the polynomial basis terms of order k, and wishes to calculate the terms for order k + 1. Assuming that every term is of the form,

$$x_{i1}, x_{i2}, \ldots\ldots, x_{ik} \quad (19)$$

where $i_1 \leq i_2 \leq \ldots\ldots \leq i_k$. Now if one has the k^{th} order terms of the polynomial basis terms of order *k* with end terms having $i_k = l$ as a vector u_l then the $k+1^{\ th}$ order terms ending with $i_{k+1} = l$ can be obtained as

$$\begin{bmatrix} x_l u_1 \\ x_l u_2 \\ \vdots \\ x_l u_l \end{bmatrix} \quad (20)$$

Using the above iterative process one $p \quad x$ can be constructed as follows. Starting with the basis terms of first order i.e. 1 and the terms of the feature vector, then iteratively calculating $k+1^{\ th}$ order terms from the k^{th} terms and concatenating the different order terms the desired $p \quad x$ of a given order can be computed. To obtain vector of polynomial basis terms of order 2 for the two dimensional feature vectors $x_1 \quad x_2^{\ t}$ is explained below.

Step 1: Terms of order 1: $1 \quad x_1 \quad x_2 \quad ^t$

Step 2: 1st order terms: $x_1 \quad x_2$

Step 3: $u_1 = x_1 \ , \ u_2 = x_2$

Step 4: For *l*=1 the $k+1^{th}$ terms with $i_{k+1} = l = 1,$ is given by

$$x_1 u_1 = x_1 u_1$$

For *l*=2 the $k+1^{th}$ terms with $i_{k+1} = l = 2$ is given by

$$\begin{bmatrix} x_2 u_1 \\ x_2 u_2 \end{bmatrix} = \begin{bmatrix} x_2 x_1 \\ x_2 x_2 \end{bmatrix}$$

Thus the second order terms are,

$$\begin{bmatrix} x_1^2 & x_1 x_2 & x_2^2 \end{bmatrix}$$

Step 5: Concatenating all the terms we get the polynomial basis terms of order 2 as,

$$\begin{bmatrix} 1 & x_1 & x_2 & x_1^2 & x_1 x_2 & x_2^2 \end{bmatrix}^t$$

For computational simplicity the terms for a given order can also be calculated with a nested loop structure using the semi group property of the monomials. The following steps show the extension of algorithm for multiclass problem.

1) For $i = 1 \ to \ N_{classes}$ [for a multiclass problem]

2) Let $r_i = 0$ and $A_i = 0$ [r_i corresponds to $p_2 \quad x$, A_i corresponds to M_i^t 1]

3) For $j = 1 \ to \ N_{frames}$ [corresponds to the number of feature vectors]

4) $r_i = r_i + p_2 \quad x_{ij}$ [x_{ij} refers to the feature vector]

5) $A_i = A_i + p \quad x_{ij}$

6) Next j

7) Next i

8) Compute $r = \sum_{i=1}^{Nframes} r_i$

9) Map r to R

10) For $i = 1 \ to \ N_{classes}$

11) $W_i = R^{-1} A_i$

12) Next i

The mapping in step 9 is based upon the fact that it is not necessary to compute the sum of outer products. Instead one can compute the subset of unique entries (i.e. the vector $p_2 \quad x$), and then map this result to the final matrix using the semigroup property of monomials.

The method of recognition is as follows. An unknown speaker is introduced. The model of the unknown speaker is tested with all the speaker models present in the library. The speaker model for which the unknown speaker model produces the maximum score is said to be that speaker. Feature Vectors $x_1, x_2, ..., x_N$ each D-dimensional, of the unknown speaker are introduced into the classifier. They are processed by a polynomial discriminant function. Every speaker i has a model W_i. The output of the discriminant function is averaged over time resulting in a score S_i for every W_i. The score is then given by,

$$S_i = \frac{1}{N} \sum_{j=1}^{N} W_i^t \, p \quad x_i \qquad (21)$$

Here x_i is the j^{th} input feature vector, W_i is the i^{th} speaker model and $p \quad x_j$ is the vector of polynomial basis terms of the j^{th} input feature vectors. The speaker with the maximum score is selected as the best match.

Table 1. % Recognition rate of various speaker recognition Techniques

4. EXPERIMENT AND RESULT

Technique	TIMIT Database 630 speakers	SGGS Database 43 speakers
MFCC-GMM	96.0	74.4
GFCC-GMM	95.7	74.4
MFCC-Polynomial	95.9	79.0
GFCC-Polynomial	96.2	83.7

The speaker recognition experiments presented in this paper were conducted using TIMIT and own created SGGS databases. The TIMIT database allows examination of speaker identification performance under almost ideal conditions. This corpus contains 630 speakers (438 male and 192 female) with 10 files for each speaker. The speech signal is recorded through a high quality microphone with a sampling frequency of 16 KHz in quiet environment. The own created English language SGGS database consist of 43 speakers (23 male and 20 female in the age group of 16-50 years) with 10 files each of 3 sec duration. The training set is recorded in laboratory using the software 'Sound Forge Version 5.0' at a sampling frequency of 22050 Hz while testing set was recorded in classroom with head mounted microphone at sampling frequency of 22050 Hz.

Ten different combinations with seven sentences per speaker (approximately 21 sec) for training and three sentences per speaker (approximately 9 sec) for testing were used. Training and testing data in any set is not overlapping. The recognition rates of the algorithms were computed using these combinations and the average recognition rates are presented. These sentences were selected randomly. Using the above dataset GFCC feature vectors were computed as explained in Section 3. A second order

polynomial classifier is used to recognize the speaker. The proposed technique is compared with MFCC (13 coefficients) [3] GMM (32 mixtures) [13]. Table 1 shows the percentage recognition rate of the above experiment. The noise robustness of the proposed approach was tested using TIMIT test speaker set of 168 speakers. White noise with different variance was added electronically to each test utterance to make its signal to noise ratio (SNR) between 0-30 dB (increasing 5dB every step). No noise was added during the training. The performances of different techniques on the noisy speeches are shown in Fig. 1. The results show that the proposed algorithm is robust to noise. However performance deteriorates when signal to noise ratio drops below 5dB.

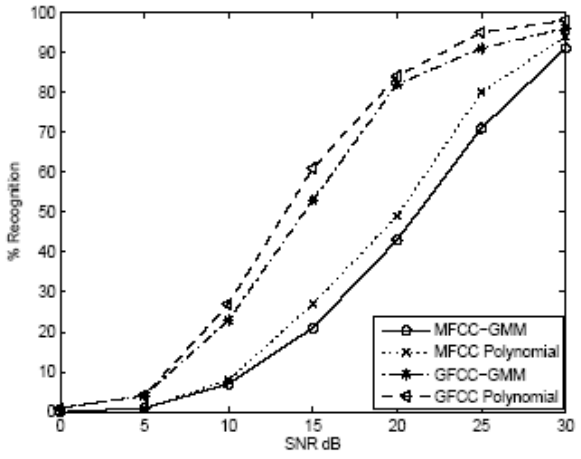

Figure 2. Recognition performance of different techniques under noisy conditions.

5. CONCLUSION

In this paper, we have proposed a general solution to robust speaker recognition under additive noise conditions. Speaker features are derived from auditory filtering and cepstral analysis. Additionally, use of polynomial classifier make the proposed system computationally efficient with low memory usage and simply multiply and add architecture. Our systematic evaluation shows that the proposed auditory features with Polynomial classifier achieve substantial performance improvement over not only typical speaker features but also in the noisy and channel variation condition.

6. REFERENCES

[1] Campbell W. M., Assaleh K. T. and Broun C. C., 2002, Speaker recognition with polynomial classifiers, IEEE Trans. on Speech and Audio Processing, 10 (4), 205-212.

[2] Carey M. J., Parris E. S, and Bridle J. S., 1991, A speaker verification system using alpha-nets, in Proc. Int. Conf. Acoustics, Speech, Signal Processing, , 397-400.

[3] Davis S. B. and Mermelstein P., 1980, Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences, IEEE Trans. ASSP, 28(4), 357-366.

[4] Farrell K. R., Mammone R. J., and Assaleh K. T., 1994, Speaker recognition using neural networks and conventional classifiers, IEEE Trans. Speech Audio Processing, 2, 194-205.

[5] Ghitza O., 1994, Auditory models and human performance in tasks related to speech coding and speech recognition, IEEE Trans. SAP, 2 (1), 115-132.

[6] Glasberg B. R. and Moore B. C. J., 1990, Derivation of auditory filter shapes from notched-noise data, Hear. Res., 47, 103-138.

[7] Hermansky H., 1990, Perceptual linear predictive (PLP) analysis of speech, J. Acoust. Soc. Am., 87(4), 1738-1752.

[8] Hermansky H. and Morgan N., 1994, RASTA processing of speech, IEEE Trans. SAP, 2 (4), 578-589.

[9] Higgins A., Bahler L., and Porter J., 1991, Speaker verification using randomized phrase prompting, Digital Signal Process., 1, 89- 106.

[10] Irino T. and Patterson R. D., 1997, A time-domain, level-dependent auditory filter: the gammachirp, J. Acoust. Soc. Am., 101, 412-419.

[11] Shao Y. and. Wang D. L, 2006, Robust speaker recognition using binary time-frequency masks, in Proc. ICASSP, 1, 645-648.

[12] Skowronski M. D. and Harris J. G., 2002, Increased MFCC filter bandwidth for noise-robust phoneme recognition, in Proc. ICASSP-02, Florida.

[13] Reynolds D. A., 1995, Automatic speaker recognition using Gaussian mixture speaker models, Lincoln Lab. J., 8(2), 173-192.

[14] Rosenberg A. E., DeLong J., Lee C.-H., Juang B.-H., and Soong F. K., 1992, the use of cohort normalized scores for speaker verification, in Proc. Int. Conf. Spoken Language Processing, 599-602.

[15] Rosenberg A. E. and Parthasarathy S., 1996, Speaker background models for connected digit password speaker verification, in Proc. Int. Conf. Acoustics, Speech, Signal Processing, 81-84.

[16] Yoma N. B. and Villar M., 2002, Speaker verification in noise using a stochastic version of the weighted Viterbi algorithm, IEEE Trans. Speech Audio Process., 10(3), 158.