A NEW SPEAKER IDENTIFICATION ALGORITHM FOR GAMING SCENARIOS

Hoang Do[1]*, Ivan Tashev*[2]*, and Alex Acero*[2]

[1]LEMS, School of Engineering, Brown University, USA
[2]Microsoft Research, Redmond, WA, USA

hdo@lems.brown.edu, {ivantash,alexac}@microsoft.com

ABSTRACT

Speaker identification is a well-established research problem but has not been a major application used in gaming scenarios. In this paper, we propose a new algorithm for the open-set, text-independent, speaker ID problem, applied as an important component (among other cues) of a game player identification system. This scenario poses new challenges: far-field, limited training and very short test data, and almost real-time processing. To tackle this, we introduce new and more informative feature sets. The scores given by these feature sets are then combined in an optimal way to construct the final score. Experimental results on the gaming device's processed reverberated-speech show the effectiveness of the new features, and that reliable decisions can be made after very short (2 - 5 second) test utterances required by the gaming scheme.

Index Terms: acoustic arrays, games, speaker recognition

1. INTRODUCTION

Recently developed gaming devices for controller-free gaming and hands-free sound-capturing has made voice-identification of the game players an important component in building games for a natural and interactive environment. Speaker identification (speaker ID) is a well-studied research problem. The main focus of this paper is to introduce a new application of speaker ID as one of the cues for game player identification systems, apart from the applications found in conventional scenarios using recorded telephone speech. In our gaming scenario, each player registers with the gaming system by speaking some sentences (training data). In the game stage, a player speaks briefly for a few seconds (test data). Using the test data, a score against each model built from the training data is produced. Based on the scores, the tested player is identified as one of the registered players or as an impostor. This identification process can be used in many cases. For example, to prevent unauthorized players to log in or to participate in the game. In another example, when the players speak simultaneously in a trivia game, we would like the gaming system to be able to pick the player whose answer is correct. In such case, the system separates the players' voices, runs through a speech recognizer, and finally applies the speaker ID algorithm. All the speech data is captured by a 4-element microphone array of the gaming device.

In the conventional scenarios, the speaker population is often quite large (500+). However, the amount of training data and test data are abundant (can be a few minutes). Off-line processing and latency can be acceptable. On the other hand, speaker ID for gaming scenarios operates on a relatively small speaker population (typically fewer than 20), has limited training data ($\leq$ 10 seconds), very short test data (2-5 second test utterances), and demands almost real-time processing. In addition, the computational resources of the gaming system are shared by many tasks, such as: sound source localization, beamforming, noise reduction, speech recognition, and video processing, thus, a computationally simple yet effective speaker ID algorithm is desired. The speech data remotely captured by the microphone array also raises the problem of reverberation and background noise from the gaming environment. These are the challenges that are not found in the conventional scenarios using telephone speech. A general speaker ID system often uses Gaussian Mixture Models (GMM) with the Universal Background Model (UBM) and a scoring normalization technique, such as the adaptive T-normalization (aT-norm) [1]. The feature set used in speaker ID typically is the Mel-frequency cepstral coefficients (MFCC) [2]. In this paper, we investigate features that are more informative than the traditional MFCC, and combine their scores in an optimal way that minimizes the identification error rate. The focus of the proposed method is on the feature sets, hence, we fix the speaker ID system in both the baseline and our proposed method to GMM-UBM with aT-norm. Although GMM-UBM with aT-norm using MFCC as the feature set is not the top performing system today, it is a relatively simple, yet commonly used baseline [3].

For experiments we used the TIMIT database and simulated the effects of reverberation and background noise by convolving with the carefully measured room's impulse responses and adding real background noise. The contaminated TIMIT was then enhanced by the audio processing tools utilized by the gaming system, and the output was used as the data for both algorithms (baseline and proposed method). TIMIT was chosen because of its manageable size and appropriateness to the gaming scenarios:

- Speaker population is small
- Data is sampled at 16 KHz, the sampling rate used by our gaming device

Experimental results show that our algorithm, using new features, achieves 5.5% equal error rate (EER) compared to 10.5% of the baseline GMM-UBM with aT-norm using the traditional MFCC for 2-second test utterances.

2. REVERSED MEL-FREQUENCY CEPSTRAL COEFFICIENTS (RMFCC)

The traditional MFCC is a representation defined as the real cepstrum of a windowed short-time signal derived from the FFT of that signal [2, 4]. This real cepstrum utilizes a nonlinear frequency scale, Mel-scale, which approximates the behavior of the human auditory system [2]. The Mel-scale warping is done by using a Mel-filterbank, where the filters space linearly at low frequencies ($f \leq f_{cutoff} = 700$ Hz) and logarithmically at high frequencies

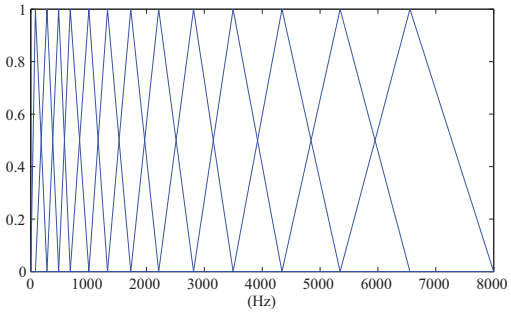

Fig. 1. *The traditional Mel-filterbank.*

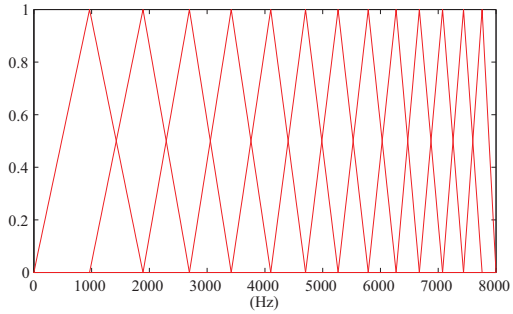

Fig. 2. *The reversed Mel-filterbank.*

(700 Hz to the Nyquist frequency), see Eq. 1 and Fig. 1.

$$f_{Mel} = 2595\log_{10}\left(1 + \frac{f_{linear}}{700}\right). \quad (1)$$

The design of this filterbank benefits speech recognition tasks [2], where the first and second formant frequencies (F1 & F2) carry the most important information of what is being said. It has a high resolution for lower formant frequencies, and a low resolution in the upper frequency range. The implicit effect of the Mel-filterbank is that it enhances the most useful spectral information (lower formant frequencies) that a speech recognizer needs, and smears out the differences (upper formant frequencies) among different speakers so that the same content spoken by anyone would be perceived similarly, and thus, can be recognized easier. On the other hand, a speaker identification system aims to extract as much speaker-dependent information as possible from the speech representation. It has been shown that the upper formant frequencies carry a lot of speaker-dependent characteristics, which are very beneficial to speaker identification tasks [5, 6, 7, 8]. Hence, the MFCC obtained from the traditional Mel-filterbank is not the optimal feature set for such tasks. Recently, Lu and Dang [9] proposed using a non-uniform filterbank based on the F-ratio and achieved 20.1% error rate reduction. However, the F-ratio-based training process to construct the filterbank is quite computationally expensive. In 2009, Lei and Gonzalo [10] proposed an antiMel filterbank in which the Mel-filters from 300 Hz to 3400 Hz were flipped about 1550 Hz. However, the reported performance was poorer than using the traditional MFCC on all speech. We suppose this poor performance is due to the limited bandwidth of the telephone speech, in which the upper formant frequencies are not fully present.

Taking advantage of the 16 KHz sampled speech signal given by the gaming device's microphone array, we propose a reversed Mel-filterbank similar to the idea of [10]. In this filterbank, we would like to have more resolution in the mid-to-upper high frequencies, say 5000 Hz to the Nyquist frequency, 8000 Hz. To fulfill this goal, we first design a traditional Mel-filterbank with $f_{cutoff} = 8000 - 5000 = 3000$ Hz. This filterbank will be linearly scaled from 0 to 3000 Hz (high resolution) and logarithmically scaled (low resolution) in the rest. We then flip the filterbank about its center frequency, i.e., f_{center} = 4000 Hz. The result is a reversed Mel-filterbank, which scales linearly from 5000 Hz to 8000 Hz, and logarithmically scaled in the lower frequencies, see Fig. 2. Thus, the reversed Mel-filterbank, which is very simple to design, enhances the resolution for the upper formant frequencies which are more informative to speaker recognition. The Mel-frequency cepstral coefficients obtained from this reversed Mel-filterbank are labeled as rMFCC.

3. LINEAR PREDICTIVE CODING (LPC) COEFFICIENTS AND FUNDAMENTAL FREQUENCY (F0)

A simple yet effective model of speech production is the source-filter model [11], in which the combination of the *source* (vocal folds) and the linear *filter* (vocal tract + radiation characteristics) produces the speech signal. In this section, we present the use of two features: the fundamental frequency (F0), which indicates the vibration rate of the vocal folds (*source*), and the linear predictive coding (LPC) coefficients, which models the vocal tract's transfer function (*filter*), to represent the speech information of a speaker, which are beneficial to speaker ID tasks.

3.1. Linear predictive coding (LPC)

As we discussed in Sec.2, the upper formant frequencies carry important speaker-dependent information. Roughly speaking, formant frequencies are the vocal tract's resonances. Hence, the LPC coefficients, which model the vocal tract, are good representations of the formants. Using all-pole autoregressive modeling [11], the vocal tract's transfer function in the z-domain, $H(z)$, is:

$$H(z) = \frac{X(z)}{E(z)} = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}} = \frac{1}{A(z)}, \quad (2)$$

where $X(z), E(z)$ are the z-domain representations of the output and excitation signals respectively, p is the LPC order, and $A(z)$ is the inverse filter. Taking the inverse z-transform and after some algebraic manipulation, we have:

$$e[n] = x[n] - \sum_{k=1}^{p} a_k x[n-k]. \quad (3)$$

From this, we can estimate the LPC coefficients a_k, $k = 1, ..., p$ by minimizing $e[n]$, using the autocorrelation method [4].

3.2. Fundamental frequency (F0)

Another good speaker-dependent feature is the fundamental frequency (F0). F0, at least, gives us additional, gender an age related information. The effect of F0 is even more profound in tonal languages, such as Chinese, Thai, etc. Because test utterances are required to be very short in gaming applications and fundamental frequencies can only be extracted from voiced frames of each utterance, only a single value of F0 (median value) is used as our feature. F0's of the voiced frames can be estimated using a cepstrum-based technique. In this paper, we used the one proposed in [12].

4. FEATURE SCORING

In our algorithm, we utilize three sets of features: rMFCC, LPC, and F0. The scores for the 2 features, rMFCC and LPC, are the standard log-likelihood scores computed as:

$$\Gamma(X) = \log p(X|\lambda_{target}) - \log p(X|\lambda_{UBM}), \quad (4)$$

where X is a feature vector, $p(X|\lambda)$ is the likelihood of feature vector X belonging to the GMM λ. The fundamental frequency score is defined as,

$$\begin{aligned}\Pi(\text{F0}_X) \equiv \max\,(&P_{voiced} \times |\text{F0}_{target} - \text{F0}_X|) \\ &- P_{voiced} \times |\text{F0}_{target} - \text{F0}_X|\,,\end{aligned} \quad (5)$$

where P_{voiced} is the probability of voiced frames in the test utterance:

$$P_{voiced} = \frac{\text{No. voiced frames}}{\text{Total no. frames}}. \quad (6)$$

Note that in Eq. 5, the fundamental frequency score is weighted by the probability of voiced frames because the fundamental frequencies were estimated from the voiced frames only. Also, the weighted score is subtracted from the maximum value over all genuine speakers so that the better match between F0_X and F0_{target} is, the larger the score is. In this way, the fundamental frequency score varies coherently with the log-likelihood scores of the first two features.

Next, we would like to properly assign weights to the feature scores so that the resulting final score would minimize the error rate. The final score of a given test utterance u is defined as,

$$S(u) = \hat{w}_1\Gamma(\text{rMFCC}) + \hat{w}_2\Gamma(\text{LPC}) + \hat{w}_3\Pi(\text{F0}). \quad (7)$$

The optimization criterion for the weight vector $\mathbf{W} = \{w_1, w_2, w_3\}$ is,

$$\hat{\mathbf{W}} = \underset{\mathbf{W}}{\text{argmin}} \left(\frac{\sqrt{\text{FRR}^2 + \text{FAR}^2}}{2} \right), \quad (8)$$

where FRR and FAR are the false rejection rate and false acceptance rate, respectively. The optimized $\mathbf{W}$ are obtained from the training set (created from the TRAIN directory of TIMIT, see Sec. 5.1) using Gaussian optimization method with the criterion of minimizing the equal error rate, and will be applied to final scores of the test set.

5. EXPERIMENTAL EVALUATION

5.1. Data preparation

From the TRAIN directory of TIMIT, we put aside 132 speakers (equal gender proportions) for training the UBM, 166 speakers for the adaptive T-norm models [1], and 20 speakers to create impostor messages for the adaptive T-norm's cohort selection procedure. The rest of the speakers in the TRAIN directory were used to create the training set. The training set was used to obtain the optimal weight vector $\hat{\mathbf{W}}$. The entire TEST directory of TIMIT was used to create the test set. Both training and test sets consisted of 10 data groups. In each data group, there were 8 genuine speakers (speakers who were learned by the gaming device during the training session) and 8 impostors (speakers who were not learned by the device), making up a total of 16 speakers per data group. All 10 sentences of each speaker were utilized. About 10 seconds of speech data from each of the 8 genuine speakers were used in training session, and 2-second test utterances were used by all 16 speakers. The reported results are average of the results on the test set (10 data groups, or 160 test speakers).

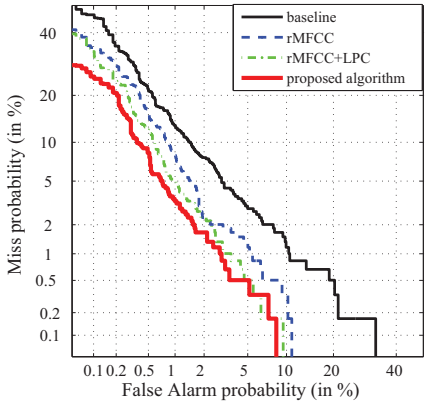

Fig. 3. *DET curves (2s test utterances): baseline vs. feature contributions of the proposed algorithm.*

5.2. Audio processing

Audio data processing included three steps:

- We converted the clean TIMIT data to 4-channel reverberated data by convolving them with the carefully measured room's impulse responses at 12 different positions in the active gaming zone in front of the device, and then added real background noise.
- The 4-channel simulated data was then processed by the audio pipeline used in the gaming device to generate a single-channel"enhanced" data. This pipeline includes an adaptive beamformer, a spatial filter, a noise suppressor [13].
- Sampling rate of 16 KHz, frame length of 25 ms, overlapped 10 ms, and Hamming window were used. We extracted 13 MFCC's and their delta's for the baseline algorithm; 13 rMFCC's, 13 LPC's, and F0 for our proposed algorithm. Both algorithms used 14-component GMM's for speaker models. In both the baseline and our algorithm, GMM's were trained using maximum likelihood (ML) criterion with EM algorithm.

Because the gaming scenario requires a low computational cost processing, a simple EM algorithm was more suitable to use than the UBM-MAP or discriminative training methods.

5.3. Experimental results

Fig. 3 shows the Detection Error Tradeoff (DET) curves of the proposed algorithm with the contribution of each new feature, and the baseline algorithm (MFCC) when using 10-second training data and 2-second test utterances. The equal error rates (EER) and minimum decision cost functions (DCF) of these algorithms are shown in Fig. 4. It can be seen that in our scenario, when test utterances are 2-second long, the proposed algorithm performs better than the baseline algorithm. Next, we tested the performance of the proposed algorithm for different test utterance lengths (1.5, 2, 3, 4, and 5 seconds). The results (EER and minimum DCF) are shown in Fig. 5. The performance of the system drops when going below 2 seconds and seems to saturate when the test utterances are longer than 4 seconds. We also tested the case when the test utterance length is fixed at 2 seconds, and we vary the length of the training data (5, 7.5, 10, 12, 15 seconds). The results are shown in Fig. 6. Clearly, increasing the amount of training data improves the performance.

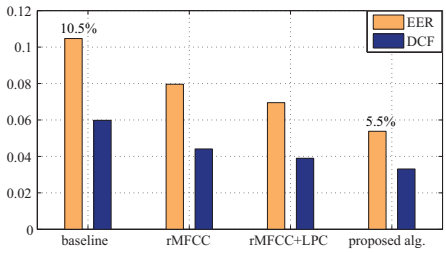

Fig. 4. *EER and DCF (2s test utterances): baseline vs. feature contributions of the proposed algorithm.*

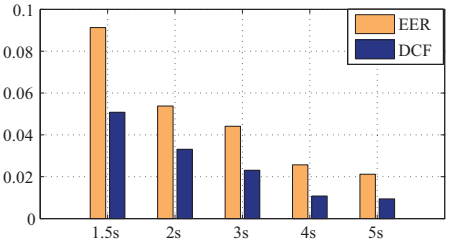

Fig. 5. *EER and DCF (10s training data): performance vs. test utterance length.*

6. CONCLUSIONS

In this paper we presented a new application of speaker identification for gaming scenarios. We proposed the use of three features, reversed MFCC, LPC, and F0 that extract more speaker-dependent information from the speech signals than the traditional MFCC. These features were then combined optimally to give the final score. Under the specific conditions of gaming applications, which are limited training data, very short test utterances and a low computational cost, our proposed algorithm achieved 5.5% EER when using 10-second training data, and 2-second test utterances, whereas the baseline algorithm (GMM-UBM with adaptive T-norm, MFCC as the feature) was at 10.5% EER, see Fig. 4. The performance at this level, when combined with other cues (biometrics, face detection, etc.), would create a complete game player identification system. Our next step in the future would be evaluating the proposed algorithm on a standard data corpus, such as the NIST SRE data.

7. REFERENCES

[1] D. E. Sturim and D. A. Reynolds, "Speaker adaptive cohort selection for tnorm in text-independent speaker verification," in *Proc. IEEE Int. Conf. Acoust. Speech, Signal Process.*, Philadelphia, PA, March 2005, vol. 1, pp. 745–748.

[2] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoust., Speech, and Signal Process.*, vol. 28, no. 4, pp. 357–366, 1980.

[3] F. Bimbot, J. F. Bonastre, C. Fredouille, G. Gravier, Magrin I. Chagnolleau, S. Meignier, T. Merlin, Ortega J. Garcia, Petrovska-Delacretaz, and D. A. Reynolds, "A tutorial on text-independent speaker verification," *EURASIP Journal on Applied Signal Processing*, vol. 4, pp. 430–451, 2004.

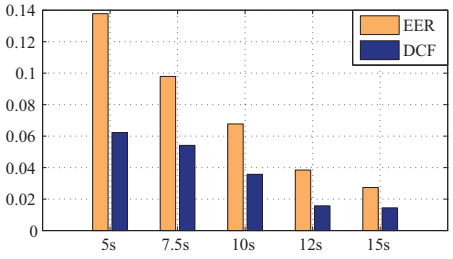

Fig. 6. *EER and DCF (2s test utterances): performance vs. training data length.*

[4] X. Huang, A. Acero, and H. Hon, *Spoken language processing: a guide to theory, algorithm, and system development*, chapter Speech signal representations, Prentice Hall, 2001.

[5] A. J. Compton, "Effects of filtering and vocal duration upon the identification of speakers, aurally," *Journal of the Acoustical Society of America*, vol. 35, pp. 1748–1752, 1963.

[6] S. Furui and M. Akagi, "Perception of voice individuality and physical correlates," *Journal of the Acoustical Society of Japan*, vol. J66-A, pp. 311–318, 1985.

[7] Y. Lavner, I. Gath, and J. Rosenhouse, "The effects of acoustic modifications on the identification of familiar voices speaking isolated vowels," *Speech Communication*, vol. 30, pp. 9–26, 2000.

[8] S. Hayakawa and F. Itakura, "Text-dependent speaker recognition using the information in the higher frequency," in *Proc. IEEE Int. Conf. Acoust. Speech, Signal Process.*, Adelaide, Australia, 1994, pp. 137–140.

[9] X. Lu and J. Dang, "Physiological feature extraction for text-independent speaker identification using non-uniform subband processing," in *Proc. IEEE Int. Conf. Acoust. Speech, Signal Process.*, Honolulu, HI, 2007, pp. 461–464.

[10] H. Lei and E. Lopez-Gonzalo, "Mel, linear, and antimel frequency cepstral coefficients in broad phonetic regions for telephone speaker recognition," in *Proc. Interspeech*, Brighton, UK, 2009, pp. 2323–2326.

[11] B. S. Atal and S. L. Hanauer, "Speech analysis and synthesis by linear prediction of speech wave," *Journal of the Acoustical Society of America*, vol. 50, pp. 637–655, 1971.

[12] S. Ahmadi and A. S. Spanias, "Cepstrum-based pitch detection using a new statistical v/uv classification algorithm," *IEEE Trans. Speech, Audio Process.*, vol. 7, no. 3, pp. 333–338, 1999.

[13] I. Tashev, *Sound Capture and Processing: Practical Approaches*, p. 388, Wiley, July 2009.