On the Use of Long-Term Average Spectrum in Automatic Speaker Recognition

Tomi Kinnunen[1], Ville Hautamäki[2], and Pasi Fränti[2]

[1] Speech and Dialogue Processing Lab
Institute for Infocomm Research (I^2R)
21 Heng Mui Keng Terrace, Singapore 119613
ktomi@i2r.a-star.edu.sg
[2] Speech and Image Processing Unit
Department of Computer Science, University of Joensuu
P.O. Box 111, FIN-80101 Joensuu, Finland
{villeh, franti}@cs.joensuu.fi

Abstract. State-of-the-art automatic speaker recognition systems use mel-frequency cepstral coefficients (MFCC) features to describe the spectral properties of speakers. In forensic phonetics, the long-term average spectrum (LTAS) has been used for the same purpose. LTAS provides an intuitive graphical representation which can be used to visualize and quantify speaker differences. However, few studies have reported the use of LTAS in automatic speaker recognition. Thus, the purpose of this paper is to systematically study how to use the LTAS in automatic speaker recognition. We will also find out whether it provides additional discriminative information in respect to the MFCC-based system.

1 Introduction

Differences in our voices arise from both *physical* factors (anatomy), and *behavioral* factors (the way of speaking). Both of these factors give rise to several measurable quantities that can be used as features in speaker recognition. In state-of-the-art automatic speaker recognition systems, multiple features are used in parallel to complement each other. In this study, we focus on spectral features because they give the best accuracy among several high- and low-level features [1].

In automatic speaker recognition, spectral features are computed from short frames (20-40 milliseconds) with the rate of 50-100 frames per second. The most commonly employed features are *mel-frequency cepstral coefficients* (MFCC) [2], appended with their first and second order delta coefficients. The short-term feature computation is followed by statistical modeling of the distribution of the vectors; each speaker produces a characteristic "cloud" in the feature space. The state-of-the-art model is the *Gaussian mixture model* (GMM) [3]. In GMM, the "feature cloud" is modeled by fitting a finite set (256-2048) of Gaussian distributions to the training data so that they characterize the data as well as possible.

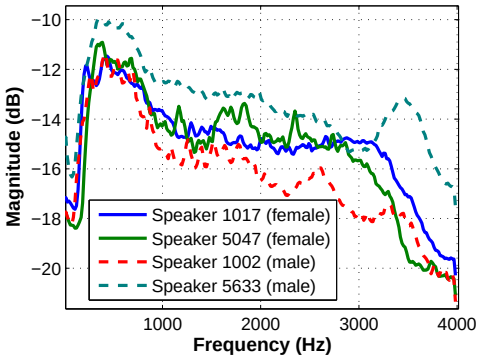

Fig. 1. Examples of LTAS computed from NIST-2001 corpus (window length = 50 ms, frequency spacing = 16 Hz).

There might be a simpler and computationally more efficient way than MFCC + GMM to describe the spectral characteristics of a speaker. In forensic phonetics [4], one approach to describe the resonance characteristics of a speaker is *long-term average spectrum* (LTAS). It is computed by time-averaging the short-term Fourier magnitude spectra, resulting in one feature vector for the whole speech sample (see Fig. 1).

The advantage of LTAS from a *forensic* perspective is that it has more or less direct physical interpretation, relating to the location of the vocal tract resonances. This makes LTAS more justified as an evidence than MFCC coefficients, which do not have direct phonetic interpretation. LTAS vectors of the questioned speech sample and the suspects speech sample can be plotted on top of each other for visual verification of the degree of similarity [5]. LTAS and other features can be complemented by auditory analysis and (semi-)automatic methods.

The advantages of LTAS from *automatic* speaker recognition perspective would be simple implementation and computational efficiency compared with the GMM. In particular, there is no separate training phase included; the extracted LTAS vector will be used as the speaker model directly and matched with the test utterance LTAS using a distance measure.

This study has two main objectives. First, although LTAS is used in forensic casework, we are not aware of systematic studies reporting the effect of the control parameters. LTAS is affected by changes in channel conditions, and robust matching and score normalization are important when LTAS is considered for telephony speaker recognition. Thus, the first goal of this paper is to study the parameters of LTAS extraction, matching, and score normalization.

The second objective of the study is to find out the usefulness of LTAS in automatic recognition. In particular, we want to answer the following questions:

- How does recognition accuracy of LTAS compare with MFCC+GMM?
- How does computational cost of LTAS compare with MFCC+GMM?

– Can LTAS and MFCC+GMM be fused for improved accuracy?
– Is there any reason to use LTAS in automatic recognition?

The rest of the paper is organized as follows. In Section 2, we describe the computation and matching of LTAS. Section 3 gives the details of the datasets used, and the experimental results are given in Section 4. Finally, conclusions are drawn in Section 5.

2 Computation and Matching of LTAS

From a signal processing point of view, LTAS computation belongs to the class of *power spectral density* (PSD) [6] estimation methods. We consider two alternative methods for computing LTAS. The first one is based on a single transformation followed by spectrum size reduction, and the second one is based on time-averaging of the short-term spectra.

In the *single-transformation LTAS*, we first compute a single fast Fourier transform (FFT) over the whole signal. This is preceded by Hamming windowing and zero padding to the length of the next power of two. Note that the length of the spectrum varies depending on the input. To get a fixed-length LTAS vector, the number of power spectrum points is reduced by averaging neighboring frequency bins, which corresponds applying a uniformly spaced filterbank. The single-transformation method is used, for instance, in the open-source *Praat*[3] speech analysis program, and it will be used here as a reference method.

Another method to compute LTAS is to divide the signal into overlapping frames, compute the power spectrum of each frame, and to average the spectra. As in the single-transformation LTAS, we apply Hamming windowing, and set the FFT size to the next power of two of the frame length. The short-term averaging method is also known as *Welch's method* [7], and it is better suited for practical applications.

The LTAS vector elements are nonlinearly compressed by applying logarithm. Log-compression balances the spectrum, and, according to our experiments, it systematically outperforms the LTAS represented in linear amplitude scale.

Finally, we need a distance measure between two LTAS vectors. We consider four simple distance measures: Euclidean distance, correlation coefficient, cosine measure and *Kullback-Leibler* (KL) divergence [8]. For the KL divergence, the LTAS vector is considered as a probability mass function. In order to meet the probabilistic constraints, the vector elements are translated to positive values, followed by normalization so that the elements sum to 1.

To increase robustness against acoustic mismatch, the raw match score is normalized by other speakers' scores. For this, we apply the *test normalization* ("T-norm") method [9]. The unknown LTAS vector is matched against a set of pseudoimpostor models (*T-norm cohort*), and the mean and standard deviation are obtained. The normalized match score is obtained from the raw score by subtracting the mean and by dividing by the standard deviation.

[3] `http://www.praat.org`

3 Experimental Setup

We use the NIST-1999 and NIST-2001 speaker recognition benchmarking corpora for our experiments [10]. Both corpora consists of conversational telephony data, NIST-1999 being recorded over the landline network and NIST-2001 over the cellular network. The NIST-1999 corpus is used for studying the effect of the feature extraction parameters, and comparing the distance measures. The NIST-2001 corpus is used for validating the results, studying score normalization, and comparing the accuracy and time consumption with the standard MFCC+GMM recognizer [3].

We use the training files of the male speakers of the NIST-1999 corpus for parameter tuning. This subset consists of 230 speakers, each represented by two audio files labeled as "a" and "b". Both samples have a duration of one minute. We consider the "a" samples as the reference samples, and the "b" samples as the unknown ones. We report both the identification and verification accuracies. We use closed-set identification error rate (IER) to measure the identification accuracy, and equal error rate (EER) to measure the verification accuracy. Equal error rate corresponds to the verification threshold for which the false acceptance and the false rejection rates are equal.

For the NIST-2001 corpus, we report verification accuracy on the 1-speaker detection task. The detection list provided by NIST consists of 9350 male trials (850 genuine + 8500 impostor) and 13068 female trials (1188 genuine + 11880 impostor). There are two minutes of training data per speaker, and the length of the test segments varies from a few seconds up to one minute. The NIST-2001 development set consisting of 60 speakers is used as the T-norm cohort set.

For the MFCC+GMM recognizer, we use the MFCC coefficients 1-12, appended with the delta and double-delta coefficients. Utterance-level mean subtraction and variance normalization are applied to the features. The universal background model of 512 components is trained from the NIST-2001 development set. The target speaker models are trained by adjusting the background model mean vectors towards the speaker's training data, see details in [3]. The fast C-top scoring algorithm described in [3] is used for matching.

4 Results

4.1 Summary of the Tuning Results

For the single-transformation LTAS, the number of FFT bins was varied in powers of two between 32-2048. For the short-term averaged LTAS, the frame length was varied between 30-320 milliseconds with a step of 10 milliseconds, and the window overlap was fixed to 50%. Table 1 summarizes the best, the worst and the average (mean $\pm$ standard deviation) accuracies. For completeness, verification accuracies of the single and short-term methods are compared in the detection error tradeoff (DET) curve of Fig. 2.

We observe that the single-transform and short-term variants are equally good. For instance, Fig. 2 shows that the short-term variant outperforms the

Table 1. Results for the tuning set.

	Eucl.	Corr.	Cos.	KL dist.
Best				
EER (single) (%)	30.0 (64 bins)	30.9 (64 bins)	18.3 (128 bins)	**18.2** (128 bins)
EER (short-term) (%)	20.4 (120 ms)	20.4 (400 ms)	19.6 (170 ms)	**18.2** (190 ms)
IER (single) (%)	76.1 (512 bins)	54.8 (512 bins)	**48.7** (128 bins)	**48.7** (128 bins)
IER (short-term) (%)	52.6 (40 ms)	**45.2** (50 ms)	47.8 (50 ms)	47.0 (4000 ms)
Average				
EER (single) (%)	31.8±1.3	22.2±1.0	**18.7**±0.5	**18.7**±0.5
EER (short-term) (%)	21.3±0.5	21.2±0.3	20.3±0.4	**19.2**±0.5
IER (single) (%)	77.8±1.9	57.1±3.3	**51.4**±3.1	**51.4**±3.1
IER (short-term) (%)	58.4±1.8	**47.8**±1.4	49.8±1.1	50.2±1.7
Worst				
EER (single) (%)	32.8 (256 bins)	23.5 (32 bins)	**19.6** (2048 bins)	**19.6** (2048 bins)
EER (short-term) (%)	22.2 (320 ms)	21.4 (110 ms)	21.2 (50 ms)	**20.0** (80 ms)
IER (single) (%)	80.9 (32 bins)	63.9 (32 bins)	**58.3** (32 bins)	**58.3** (32 bins)
IER (short-term) (%)	60.9 (200 ms)	**47.8** (250 ms)	51.0 (280 ms)	53.0 (30 ms)

single-transformation variant for low false acceptance rate (secure end) of the DET curve but the situation is reversed for low false rejection rate (user-convenience end). The equal error rates are close to each other.

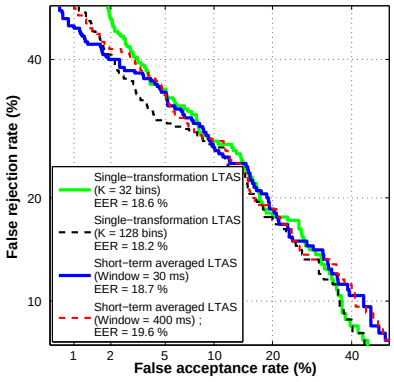

Fig. 2. Comparison of single-transformation and short-term variants for LTAS computation (KL divergence).

4.2 T-norm and Comparison with MFCC+GMM

Next, we validate our results using the NIST-2001 evaluation set. We use the short-term averaging variant with 200 millisecond window to compute LTAS. The verification results with and without score normalization are given in Table

2. The error rates are higher than for NIST-1999, for which the likely reason is that test segments are in general much shorter and trials include channel missmatch for the NIST-2001. Score normalization improves accuracy in all cases as expected. However, the KL measure does not give the best result as opposed to the NIST-1999 results. The reason for this is unknown.

Table 2. Equal error rates (%) for the NIST-2001 corpus.

Normalization	Eucl.	Corr.	Cos.	Kullb.-Leib.
None	31.7	28.0	27.2	31.7
T-norm	30.4	24.2	24.9	29.0

Next, we study the effect of channel mismatch to LTAS and give the MFCC + GMM baseline as a reference. To study the effect of channel mismatch, NIST-2001 detection list was divided into "matched" and "mismatched" trials. The matched and mismatched trials refer to the trials with same and different phone model with the training and test files, respectively.

The left and right panels of Fig. 3 show the DET curves for the matched and mismatched channel cases, respectively. Not surprisingly, channel mismatch degrades the accuracy of both the LTAS and the MFCC+GMM recognizers. The MFCC+GMM recognizer outperforms LTAS without a doubt, as expected. Note that, because of computational reasons, we did not apply T-norm for the MFCC+GMM system. The MFCC+GMM error rate would be expected to decrease further by including T-norm.

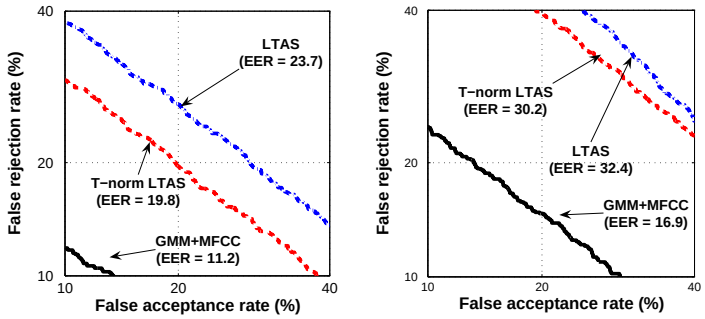

Fig. 3. Verification results for the NIST-2001 corpus, matched channel type (left), mismatched channel type (right).

4.3 Time Consumption

Next, we study the computation times of LTAS and MFCC+GMM. All the experiments are carried out in 3GHz Intel Pentium 4 with 1024 MB of memory. All algorithms were implemented and run in Matlab 7. Tests were performed by first enrolling all speakers into a database and then perfoming the NIST-2001 evaluation protocol on the enrolled speakers. Running times are reported in seconds averaged over all test cases.

The speaker enrollment times are summarized in the Table 3. The running times of the single-transformation and short-term variants are practically the same, and LTAS is about 13 times faster than the MFCC+GMM recognizer.

Verification times are summarized in Table 4. Overall matching time of LTAS without score normalization is about 10 times faster than that of the MFCC + GMM. Adding score normalization increases the processing time of LTAS, and the baseline MFCC+GMM matching is faster than LTAS + T-norm. However, even with score normalization, overall processing time of LTAS is smaller, which is due to much faster feature extraction.

Table 3. Comparison of CPU times for enrollment (seconds).

	Feature extraction	Modeling	Total
single-transf. LTAS	1.0±0.0	-	1.0
short-term avg. LTAS	0.9±0.1	-	0.9
MFCC+GMM	9.2±1.1	4.4±0.1	13.6

For identification performance, the matching times should be multiplied by the number of speakers enrolled in the database. For example, identification with the short LTAS would take on average 0.2 + 0.1 = 0.3 seconds and with the MFCC+GMM system 2.6 + 104.4 = 107.0 seconds. Thus, there is a remarkable difference in the processing time required.

Table 4. Comparison of CPU times for verification (seconds).

	Feature extraction	Matching	Total
single-transf. LTAS	0.3±0.1	< 0.01	0.3
single-transf. LTAS+T-norm	0.3±0.1	1.8±0.2	2.1
short-term avg. LTAS	0.2±0.1	< 0.01	0.2
short-term avg. LTAS+T-norm	0.2±0.1	1.8±0.2	2.0
MFCC+GMM	2.6±1.1	0.6±0.9	3.2

4.4 Combining LTAS and MFCC+GMM

Finally, we want to find out whether it is advantageous to combine LTAS and MFCC+GMM recognizers. We use weighted sum to combine the classifier output

scores so that $s_{\text{fused}} = w \cdot s_{\text{MFCC+GMM}} + (1 - w) \cdot s_{\text{LTAS}}$, where $0 \leq w \leq 1$. Here, $s_{\text{MFCC+GMM}}$ is the average log likelihood ratio from the MFCC+GMM recognizer, and s_{LTAS} is the T-normalized correlation coefficient from the LTAS recognizer. The EER as a function of w and the DET curve for $w = 0.96$ are shown in Fig. 4.

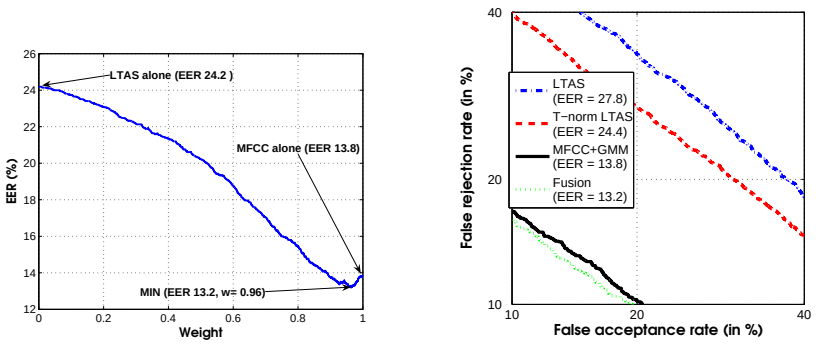

Fig. 4. ERR as a function of fusion weight (left) and Fusion results (right).

Combining LTAS and MFCC+GMM gives a slight improvement to the MFCC + GMM baseline over all detection thresholds. However, according to Fig. 4, the weight selection is critical; for this corpus, the best result is obtained in the range $[0.94 - 0.97]$, and this is likely to be different for other corpus. Moreover, as the relative gain of combining LTAS with MFCC+GMM is only marginal, we conclude that it is not worth combining these two recognizers.

5 Conclusions

In this paper, we have studied the use of long-term average spectrum feature for automatic speaker recognition. We compared two different methods for computing LTAS, a single-transformation variant and a short-term averaging variant. We also compared the LTAS performance with the baseline MFCC+GMM system, and attempted to combine the two recognizers.

Our experiments indicate that the accuracy and computational load of the single-transformation and the short-term averaging variants are practically the same. However, from the memory and real-time processing consideration viewpoints, the short-term averaging variant would be the recommended method.

The current study suggest that LTAS does not bring improvement to the standard MFCC+GMM configuration. However, the method is trivial to implement and it is computationally efficient. One possible application in automatic recognition could be speeding up speaker identification from a large database [11]. For instance, LTAS could be used to prune out speakers who have a very

large distance from the unknown sample. After this, the remaining candidate speakers could be scored more accurately by the MFCC+GMM recognizer.

To sum up, we conclude that LTAS has little use in automatic speaker recognition if the recognition accuracy is the only motivation.

References

1. Reynolds, D., Andrews, W., Campbell, J., Navratil, J., Peskin, B., Adami, A., Jin, Q., Klusacek, D., Abramson, J., Mihaescu, R., Godfrey, J., Jones, D., Xiang, B.: The SuperSID project: exploiting high-level information for high-accuracy speaker recognition. In: Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2003), Hong Kong (2003) 784–787
2. Huang, X., Acero, A., Hon, H.W.: Spoken Language Processing: a Guide to Theory, Algorithm, and System Development. Prentice-Hall, New Jersey (2001)
3. Reynolds, D., Quatieri, T., Dunn, R.: Speaker verification using adapted gaussian mixture models. Digital Signal Processing **10**(1) (2000) 19–41
4. Rose, P.: Forensic Speaker Identification. Taylor & Francis, London (2002)
5. Lindh, J.: Visual acoustic vs. aural perceptual speaker identification in a closed set of disguised voices. In: Proc. The 18th Swedish Phonetics Conference (FONETIK 2005), Göteborg, Sweden (2005) 17–20
6. Gray, R., Davisson, L.: An Introduction to Statistical Signal Processing. Cambridge University Press, Cambridge, United Kingdom (2003)
7. Welch, P.D.: The use of fast fourier transforms for the estimation of power spectra: A method based on time averaging over short modified periodograms. IEEE Transactions on Audio and Electroacoustics **15** (1967) 70–73
8. Cover, T., Thomas, J.: Elements of Information Theory. Wiley Interscience (1991)
9. Auckenthaler, R., Carey, M., Lloyd-Thomas, H.: Score normalization for text-independent speaker verification systems. Digital Signal Processing **10** (2000) 42–54
10. NIST: NIST speaker recognition evaluation. www page (2006) `http://www.nist.gov/speech/tests/spk/`.
11. Kinnunen, T., Karpov, E., Fränti, P.: Real-time speaker identification and verification. IEEE Trans. Audio, Speech, and Language Processing **14**(1) (2006) 277–288