TEXT-INDEPENDENT COMPRESSED DOMAIN SPEAKER VERIFICATION FOR DIGITAL COMMUNICATION NETWORKS CALL MONITORING

M. Petracca, A. Servetti, J.C. De Martin

Dipartimento di Automatica e Informatica
Politecnico di Torino
Corso Duca degli Abruzzi, 24 — I-10129 Torino, Italy
E-mail: [matteo.petracca|servetti|demartin]@polito.it

ABSTRACT

In this paper we present a text-independent automatic speaker verification system that works in the compressed domain using GSM AMR coded speech. While traditional approaches process LPC-based cepstral coefficients extracted from LPC related bitstream coefficients, our objective is to study the feasibility of a system that directly processes the raw bitstream transmitted over a digital communication network. In a text-independent closet-set task, with a database of 100 speakers, the proposed system achieves an EER equal to 5.93%, 4.41% and 3.80% for 10, 20 and 30 second long test speech segments respectively.

1. INTRODUCTION

With the large use of digital speech codecs in professional communication networks, growing interest is arising in speaker recognition technology for call monitoring in such context. In [1] Automatic Speaker Verification (ASV) is proposed for on-line vocal-based identity monitoring to assess a nominal use of the mobile terminals in a digital network. On-line monitoring of speech communications needs a specific implementation of ASV systems when compared to off-line traditional solutions that work with uncompressed and complete speech utterances. As analyzed in the next section, this kind of application should consider that i) speech must be processed on a frame-by-frame basis that depends on the codec speech segmentation policy, ii) speech transits over the network after lossy coding where low-bitrate algorithms (e.g. GSM AMR 12.2 kb/s) are used for reducing bandwith consumption at the expense of audio quality, iii) call monitoring of high traffic networks requires reduced complexity algorithms and low memory usage for fast and scalable processing of speech bitstreams.

In this work we introduce an alternative approach for ASV with respect to state of the art systems. Instead of extracting speech features through the conversion of internal speech coder parameters on the mobile terminal (as defined in the ETSI Aurora standard) or on a remote platform (as in [1]), we apply statistical analysis directly on the compressed bitstream values, i.e., without parameters decoding and cepstral information computation.

This low-complexity approach, coupled with low memory pattern matching algorithms, may enable scalable bitstream processing of hundreds of calls and it does not require any additional complexity or software changes on the user's devices. Although the performance of this system cannot outperform the results achieved with state of the art approaches, in practical scenarios it may be combined with a second form of authentication (e.g., monitor of user calling behaviour) to enhance the system strength as in [2].

The rest of this paper is organized as follows. An overview of the ASV architecture for call monitoring in a digital speech communication scenario is presented in Section 2. In Section 3 the Compressed-Domain Automatic Verification system (CD-ASV) is analyzed. Performance results for the GSM Adaptive Multi-Rate (AMR) speech coder are illustrated in Section 4. Conclusions follow in Section 5.

2. THE NETWORK CALL MONITORING SYSTEM

Speaker verification with coded speech is here proposed for call monitoring in a professional telecommunication network. The architecture of this system is illustrated in Fig. 1. Feature extraction and speaker verification can be applied at four different locations in the network. Stream processing can take place (1) on the terminal with uncoded speech, at the receiver either with (2) parameters extracted from the re-synthesized speech or (3) from the prediction coefficients of the coded speech. Additionally we propose a fourth approach, where stream processing is performed by the appliances of the network provider without the extraction of speech related features, but with a statistical analysis of the compressed bitstream (that has been shown to still reveal speaker discriminant information [3]).

The *uncoded speech* case (1) is clearly the most traditional in ASV. The input sequence is a digitalized PCM representation of the voice waveform that is usually transformed into the frequency domain for processing. In this domain, speaker

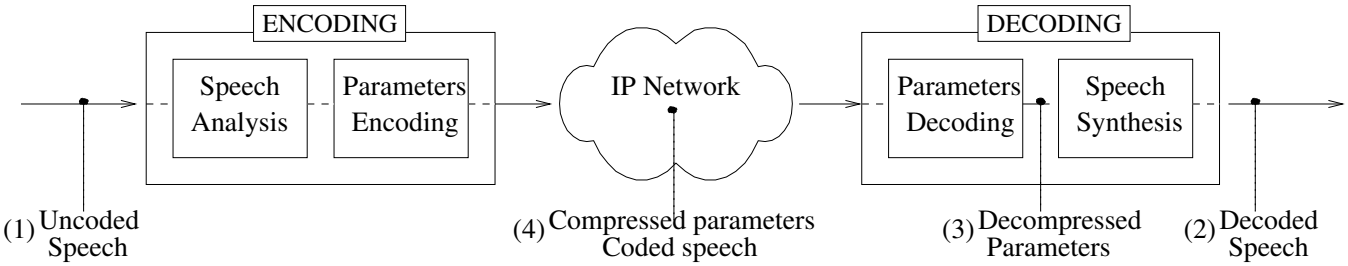

Fig. 1. Classification of ASV approaches in a digital speech communication scenario.

dependent features such as cepstral coefficients and their time variation are extracted from short-time speech frames, then they are processed using a classification algorithm such as the Gaussian Mixture Models [4].

More recently the effect of speech coding/decoding on speaker and language recognition tasks has been analyzed applying traditional techniques to the *decoded speech* (2). In [5] several codecs and a wide range of bit rates have been considered. These studies showed that straightforward application of traditional ASV on the re-synthesized speech generally degrades with the codec bit rate, with respect to the uncoded baseline.

Moreover, with the focus on reducing the computational load introduced at the receiver by the speech synthesis process, a parametric approach has been investigated in [6]. In the parametric approach, the goal is to perform ASV using a feature vector consisting of *decompressed parameters* (3). In particular, speech spectrum coefficients are obtained decompressing the bitstream values (i.e., *parameters decoding* in Fig. 1) , and then converting extracted parameters into much useful features. Despite a gain in complexity reduction with respect to the previous approach, recognition performance noticeably decreases.

CD-ASV, instead, works directly in the compressed domain with *coded speech and compressed parameters* (4), no decoding is applied to the speech bitstream, thus lowering the computational requirements with respect to previous mentioned approaches. In this fourth approach traditional pattern matching techniques have been rejected in favour of lightweight clustering algorithms [7] or medium-term statistical analysis [3], more suitable for an implementation on devices with reduced computational capabilities as network devices or user's equipments.

3. SPEAKER VERIFICATION SYSTEM

The basic structure of a speaker verification system is depicted in Fig. 2. A verification system essentially receives as input a test speech segment and a claimed speaker identity, giving as output an answer at the question "Is he the test speaker who he claims to be?" choosing between two possible hypotheses: the test speech segment has been produced by the claimed speaker or by an impostor. In the front-end processing, speaker-dependent features are extracted from test speech segments, then they are processed and compared with claimed speaker and impostor models, previously created by means of an enrollment procedure, in order to obtain similarity measures. A final decision about the initial hypothesis is taken according to the evaluated similarity measures and an imposed security threshold.

3.1. Speaker-Dependent Features in the Compressed Bitstream

In the literature there are several studies on the choice of acoustic features in speaker verification tasks. Spectral-related parameters such as cepstral coefficients, have been proved to be useful discriminating feature, as well average fundamental frequency [8] and gain measurements [9], although this last two set of parameters are not extensively adopted.

In the approach under investigation, speaker-dependent features are derived from bitstream values of compressed speech. In this work we consider the compressed speech parameters of the bitstream generated by the widely used GSM AMR speech coder, the default speech coder for GSM 2+ and WCDMA third generation wireless systems. GSM AMR compressed parameters have been previously studied, for the particular 12.2 kb/s coding rate, in [3], where their discriminant power has been studied and the parameters able to discriminate among speakers were selected. These parameters contains: speech spectrum (i.e., line spectral frequencies), excitation (i.e., adaptive codebook index) and gain related features (i.e., adaptive and fixed codebook gains). Among the GSM AMR 12.2 kb/s parameters, listed in Table 1, the fixed codebook index has not been taken into account for speaker discrimination since it has shown to be almost uniformly distributed and statistically identical among different speakers.

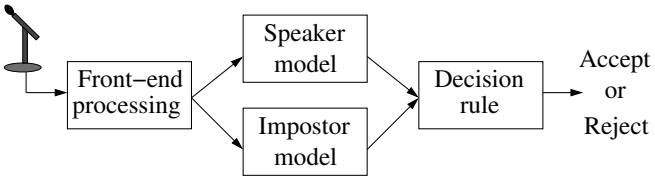

Fig. 2. Basic components of speaker verification systems.

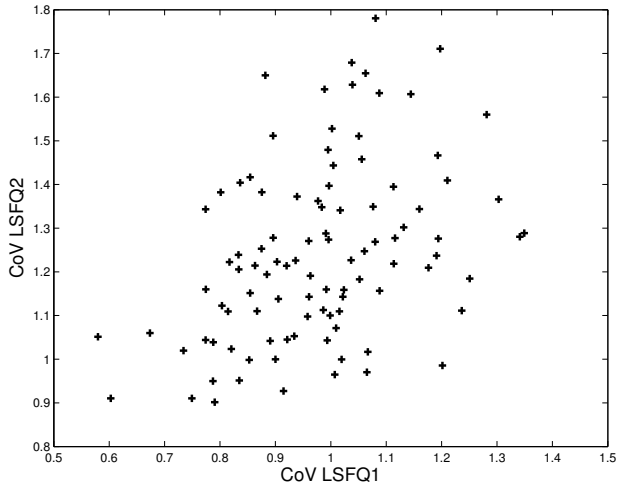

Fig. 3. Two-dimensional representation of speaker models in the LSFQ1 and LSFQ2 coefficient of variation space using training speech segments of 90 seconds.

3.2. Speaker and Impostor models

As previously introduced, the speaker and impostor models are created by means of an enrollment procedure. The choice of a particular model depends on the particular application in which the verification system has to be used. In the selected scenario, digital communications networks, the system should have the possibility to be implemented on devices using a limited amount of resources, thus an economical design in both memory size and computation requirements is needed.

According to the results of our previous work on low-complexity compressed-domain speaker recognition [3], the speaker model is created using medium-term statistics of the compressed speech parameters. In particular the Coefficient of Variation (CoV[1]) and Skewness (Skew[2]) for the nine selected parameters of the GSM AMR at 12.2 kb/s are evaluated using 90 second long training speech segments of active speech, thus each speaker model is composed by a template of 18 float values. Figure 3 shows a representation of the speaker models in a two-dimensional reduced feature space obtained from the coefficient of variation values of the LSFQ1 and LSFQ2 parameters in Table 1.

Parameter	**Subframe**			
	1	**2**	**3**	**4**
Line Spectral Frequencies (LSFQ 1-5)	7 + 8 + 9 + 8 + 6			
Adaptive Codebook index	9	6	9	6
Adaptive Codebook gain	4	4	4	4
Fixed Codebook index	35	35	35	35
Fixed Codebook gain	5	5	5	5

Table 1. Bit allocation of the GSM AMR 12.2 kb/s speech coding standard.

[1] $CoV(x) = E[x]/\sqrt{E[(x - E[x])^2]}$
[2] $Skew(x) = E[(x - E[x])^3]/(E[(x - E[x])^2])^{3/2}$

The impostor model is then dynamically generated for each test as a function of the declared speaker identity, i.e., its value is computed using all speaker template models stored in the database except the one that belongs to the declared speaker. This procedure ensures no correlation between the impostor and claimed speaker models and it can be implemented with negligible complexity as follows. A global impostor model is stored in the system database as the average of all known speaker models, then the specific impostor model for each test is obtained subtracting the normalized contribution of the declared speaker to the overall average.

3.3. Decision rule

Speaker verification system decision to accept or reject the declared identity is then based on testing which one of the declared speaker or impostor model is a closer match to the model derived from the test speech segment. Denoting by X the reference models (either of the declared speaker X_{dec}, or of the impostor X_{imp}) and by Y the measured model from the test segment, the similarity measure between the templates is expressed as:

$$D(Y, X) = d(\delta_Y, \delta_X) + d(\xi_Y, \xi_X), \qquad (1)$$

where $d(a, b)$ is the squared Euclidean distance between feature vectors a and b, δ and ξ are respectively the coefficient of variation and skewness calculated over the nine selected compressed parameter values of the speech frames.

Finally, a decision is made that the declared identity is true as a function of a security threshold, θ, that enables a trade-off between two types of error: (1) that the test speaker is rejected incorrectly (false rejection) and (2) that the test speaker is erroneously accepted as the declared one (false acceptance). The decision rule is as follows:

$$\begin{cases} D_{dec} < D_{imp} & \text{AND} & D_{dec} < \theta & \text{Accept} \\ D_{dec} \geq D_{imp} & \text{OR} & D_{dec} \geq \theta & \text{Reject} \end{cases}. \qquad (2)$$

The declared identity is accepted as true if i) the distance between the test and the declared speaker (D_{dec}) is lower than the distance between the test and the impostor model (D_{imp}) and ii) the test model distance from the declared model (D_{dec}) is lower than the threshold (θ). In any other case, the declared identity is considered an impostor and the original hypothesis is rejected. The imposed threshold value reflect the system security, low values guarantee low probabilities of false acceptance, which is a strong requirement for a security system, at the cost of higher probabilities of false rejection.

4. PERFORMANCE ANALYSIS

In this section we evaluate the potential use of compressed parameter values for speaker verification tasks on GSM AMR 12.2 kb/s coded speech material. Experiments consist of text-independent closed-set tasks over a speech corpora of 100

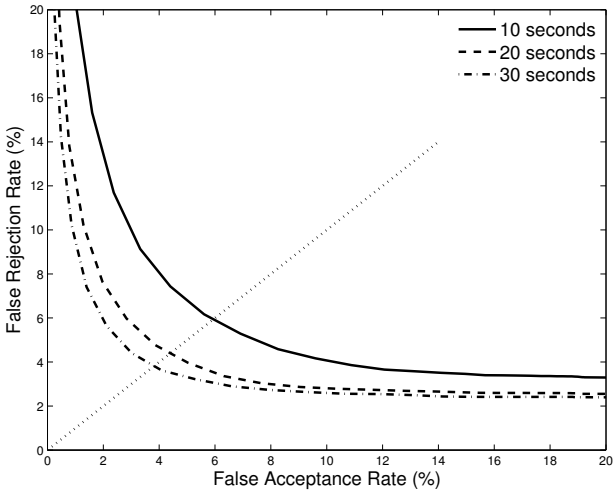

Fig. 4. Detection error trade-off curves for test speech segments of 10, 20 and 30 seconds.

speakers recorded with different microphones in normal room noise conditions. The GSM AMR voice activity detection algorithm, that works in real-time, is used to detect and select only active speech frames both on the training and test material. In the enrollment phase speaker models are extracted from 90 second long training traces (as in [3]) and inserted in a model database, while in the test phase we considered shorter speech segments of 10, 20 and 30 seconds from a separate set. For each test segment all the possible speaker identities belonging to the model database have been used as claimed speaker identity, moreover test segments have been partially overlapped in order to further increase the total number of tests.

Performance results are presented in Fig. 4 in terms of False Acceptance Rate (FAR) and False Rejection Rate (FRR) by means of a Detection Error Trade-Off (DET) graph, the various FAR and FRR values have been evaluated varying the θ threshold.

Clearly increased length of the test segments shows a better verification performance, but with progressively reduced gain. In Fig. 4 the axis bisector represents Equal Error Rate (EER) values, i.e., equal values of false acceptance and false rejection rates, commonly used as performance index for speaker verification systems. The proposed tentative CD-ASV system achieves an EER of 5.93%, 4.41% and 3.80% for 10, 20 and 30 second long test segments respectively.

The results obtained in this work appear to be consistent with similar experiments on coded speech reported in [10], where traditional LPC-cepstral coefficients plus additional residual information are employed for ASV from decoded G.729 parameters (approach number 3 as identified in Fig. 1). In this work, verification performance reduction due to the processing in the compressed domain can be compared to the performance reduction reported in [10] because of additive noise imposed on the test speech material.

5. CONCLUSIONS

In this paper we presented a low-complexity speaker verification system working in the compressed speech domain suitable for real implementation in digital communication networks appliances for call monitoring. Speaker-dependent information is derived directly from compressed bitstreams without decoding or re-synthesis of the speech signal, thus reducing computational requirements and memory usage. For the selected GSM AMR speech coder, in a text-independent closed-set task with a database of 100 speakers, the proposed system reaches an equal error rate of 5.93% for 10-second long test speech sequences. For ASV this approach has shown better performance and scalability with respect to speaker identification tasks presented in previous works.

6. REFERENCES

[1] A. Preti, B. Ravera, F. Capman, and J.-F. Bonastre, "An application constrained front end for speaker verification," in *Proc. of 16th European Signal Processing Conference*, Lousanne, Switzerland, August 2008.

[2] M.J. Carey and R. Auckenthaler, "User validation for mobile telephones," in *Proc. IEEE Intl. Conf. on Signal Acoustics, Speech, and Signal Processing*, Instambul, Turkey, 2000, vol. 2, pp. 1093–1096.

[3] M. Petracca, A. Servetti, and J.C. De Martin, "Low-complexity automatic speaker recognition in the compressed GSM-AMR domain," in *Proceedings of IEEE ICME*, Amsterdam, The Netherlands, July 2005, pp. 662–665.

[4] D.A. Reynolds and R.C. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, January 1995.

[5] R.B. Dunn, T.F. Quatieri, D.A. Reynolds, and J.P. Campbell, "Speaker recognition from coded speech in matched and mismatched conditions," in *A Speaker Odyssey. The Speaker Recognition Workshop*, Crete, Greece, June 2001, pp. 72–83.

[6] E.W.M. Yu, M. Mak, C. Sit, and S. Kung, "Speaker verification based on G.729 and G.723.1 coder parameters and handset mismatch compensation," in *Proceedings of EUROSPEECH*, Geneva, Switzerland, September 2003, pp. 1681–1684.

[7] C. Aggarwal, D. Olshefski, D. Saha, Z.-Y. Shae, and P. Yu, "CSR: Speaker recognition from compressed VoIP packet stream," in *Proceedings of IEEE ICME*, Amsterdam, The Netherlands, July 2005, pp. 970–973.

[8] B.S. Atal, "Automatic speaker recognition based on pitch contours," *The Journal of the Acoustical Society of America*, vol. 52, no. 6B, pp. 1687–1697, December 1972.

[9] R.C. Lummis, "Speaker verification by computer using speech intensity for temporal registration," *IEEE Transactions on Audio and Electroacoustics*, vol. 21, no. 2, pp. 80–89, April 1973.

[10] A. Moreno-Daniel, B.-H. Juang, and J.A. Nolazco-Flores, "Speaker verification using coded speech," *Progress in Pattern Recognition, Image Analysis and Applications*, vol. LNCS 3287, pp. 366–373, November 2004.