Published in Proceedings of the International COST 254 Workshop on
Friendly Exchanging through the Net, 117-222, 2000
which should be used for any reference to this work

SPEAKER RECOGNITION ON COMPRESSED SPEECH

S. Grassi[1], *A. Dufaux*[1], *L. Besacier*[2], *M. Ansorge*[1], *F. Pellandini*[1]

(1) Institute of Microtechnology, University of Neuchâtel,
Rue A.-L. Breguet 2, 2000 Neuchâtel, Switzerland
Phone: +41 32 7183432; Fax: +41 32 7183402; e_mail: Sara.Grassi@imt.unine.ch

(2) CLIPS Lab., GEOD team, University Joseph Fourier, BP 53, 38041 Grenoble, France

ABSTRACT

We have investigated the influence of GSM speech coding in the performance of a text-independent speaker recognition system based on Gaussian Mixture Models (GMM) classifiers.

The performance degradation due to the utilization of the three GSM speech coders was first assessed, using three transcoded databases, obtained by passing the TIMIT through each GSM coder / decoder. The coded databases were used for training and testing the speaker identification system. The speaker recognition performance was also assessed using the original TIMIT and its 8 kHz downsampled version.

Then, different experiments aimed to explore feature calculation directly from the encoded parameters, and to measure the degradation introduced by different aspects of the coders were carried out.

Keywords: *Speaker recognition, Speech coding, GSM speech coding.*

1. INTRODUCTION

Automatic speaker recognition (ASR) is the use of a machine to recognize a person from a spoken phrase [1]. It includes verification and identification. Automatic speaker verification (ASV) is the use of a machine to verify a person's claimed identity from his voice. In automatic speaker identification (ASI), there is no "a priori" identity claim, and the system decides who the person is, or what group the person is a member of, or (in an open set case) that the person is unknown. Speaker recognition has applications such as banking over telephone network, telephone shopping, database access services and security control for confidential information. Due to the increasing demand for mobile communications, it is expected that in a near future many of these transactions will take place through the mobile cellular network. Therefore, the motivation for this work is to study the effect of speech coding on recognition performance in the GSM cellular network, but it could also apply in the context of speech transmission over packet-based multimedia communications systems (H.323 terminals) where speech is compressed before its transmission.

Three speech coders[1] are standardized for use in the GSM wireless communication network. They are referred to as the full rate (FR), half rate (HR) and enhanced full rate (EFR) coder (see Section 2).

Preliminary work we did using only a speaker *identification* system is reported in [2]. The TIMIT database [3] was passed through each GSM coder / decoder obtaining three transcoded databases, which were used for training and testing the speaker identification system. The performance was also assessed using the original TIMIT and its 8 kHz downsampled version. Results showed significant performance degradation when using the GSM transcoded databases. Similar investigations reported in literature [4], [5], using a speaker *verification* system, suggest that GSM coding does not introduce major degradations. This motivated us to repeat the experiments using both *verification and identification* systems [6], to have a means of comparison.

Two different experiments are presented in [6]. In the first experiment the recognition performance degradation due to the utilization of the three GSM speech coders was assessed (see Section 5). In the second experiment, the features for the speaker recognition system were calculated directly from the information available in the GSM FR encoded bit stream (see Section 6). This allowed a measurement of the degradation introduced by the different aspects of the coder, and gave some guidelines for a better use of the information available in the bit stream, for speaker recognition purposes. It was found that the low (8-th) order LPC of the FR coder is responsible for most performance degradations. Thus, better results are expected in experiences using the EFR, which has a 10-th order LPC.

In this paper we present latest experiments, carried out using the EFR coder (see Section 7). Additionally we explore usage of Line Spectrum Pairs (LSP) instead of cepstral coefficients, and the calculation of higher order LPC information that has "leaked" in other encoded parameters (LTP lags and gains, and stochastic pulses and gain) from the decoded speech.

This paper is organized as follows. The three GSM speech coders are briefly explained in Section 2, and the

[1] Recently, another speech coder, named the Adaptive Multirate (AMR) coder, was standardized by ETSI

construction of the GSM transcoded databases is described in Section 3. The speaker recognition system used in all the experiments is presented in Section 4. Speaker recognition experiments conducted on original and GSM transcoded speech are given in Section 5.

Experiments on using features extracted directly from the GSM FR encoded parameters are described in Section 6, whereas similar experiences carried out with the EFR coder are given in Section 7. Finally, conclusions and future work are drawn in Section 8.

2. GSM SPEECH CODERS

There exist three different GSM speech coders, referred to as the full rate (FR), half rate (HR) and enhanced full rate (EFR) GSM coders. These coders work on a 13 bit uniform PCM speech input signal, sampled at 8 kHz, which is processed on a frame-by-frame basis, with a frame size of 20 ms (160 samples). This frame is divided into four subframes of 5 ms each.

2.1 Full Rate (FR) Speech Coder

The GSM FR coder was standardized in 1987 and belongs to the class of Regular Pulse Excitation-Long Term Prediction (RPE-LTP) linear predictive coders. A frame of 160 speech samples is encoded as a block of 260 bits, for a bit-rate of 13 kbps. The GSM full-rate channel supports 22.8 kbps. Thus, the remaining 9.8 kbps are used for error protection. The FR coder is described in GSM 06.10 [7] down to the bit level, enabling its verification by means of a set of digital test sequences, also given in GSM 06.10. A public domain bit exact C-code implementation of this coder is available [8]. Spectral analysis is performed once per frame, as explained next.

2.1.1 Spectral Analysis in the FR Coder

The input speech signal is first pre-processed for offset removal and pre-emphasis. Then it is segmented into 20 ms non-overlapping frames. Linear Predictive analysis (LPC) is done for each frame with 8^{th}-order autocorrelation and Schur recursion. The obtained reflection coefficients, k_1-k_8, are converted to log-area-ratio (LAR) and quantized using 36-bit independent nonuniform scalar quantization. The quantized LAR are linearly interpolated and converted back to LPC, to be used in the calculation of the other encoded parameters (LTP lags and gain, and RPE pulses and gain).

2.2 Half Rate (HR) Speech Coder

The HR coder standard was established to cope with the increasing number of subscribers. This coder is a 5.6 kbps VSELP (Vector Sum Excited Linear Prediction) coder from Motorola. The half rate channel supports 11.4 kbps. Therefore, 5.8 kbps are used for error protection. The measured output speech quality for the HR coder is comparable to the quality of the FR coder in all tested conditions, except for tandem and background noise conditions. The normative GSM 06.06 [7] gives the bit-exact ANSI-C code for this algorithm, while GSM 06.07 gives a set of digital test sequences for compliance verification.

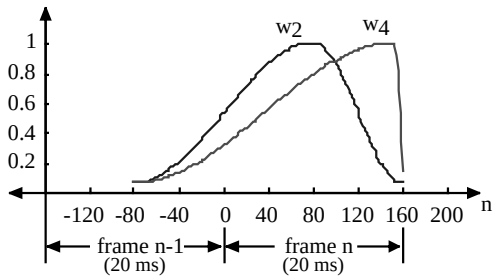

Figure 1: LPC analysis windows in the EFR coder.

2.3 Enhanced Full Rate (EFR) Speech Coder

The EFR coder was the latest to be standardized. It is intended for utilization in the full rate channel, and it provides a substantial improvement in quality compared to the FR coder. This coder is based on Algebraic Code Excited Linear Prediction (ACELP) and uses 12.2 kbps for speech coding and 10.6 kbps for error protection. The bit exact ANSI-C code for the EFR coder is given in GSM 06.53 [7] and the verification test sequences are given in GSM 06.54.

2.3.1 Spectral Analysis in the GSM EFR Coder

The input speech signal is filtered using a 2^{nd} order high-pass filter with 80 Hz cutoff frequency. LPC analysis is performed twice per speech frame using auto-correlation and Levinson-Durbin recursion, with the two different 30 ms asymmetric windows, w_2 and w_4, shown in Figure 1. The first window, w_2, has its weight concentrated at the second subframe, whereas the second window, w_4, has its weight concentrated at the fourth subframe. Both LPC analyses are performed on the same set of speech samples. The windows are applied to 80 samples from past speech frame in addition to the 160 samples of the present speech frame. The auto-correlations are lag-windowed with a 60 Hz bandwidth expansion factor and the energy is multiplied by a white noise correction factor of 1.0001.

The 2 sets of LPC coefficients are converted to 2 sets of Line Spectrum Pairs (LSP) for quantization and interpolation. The two calculated LSP sets correspond to the second and fourth subframes whereas LSPs for the first and third subframes are interpolated from the LSPs in the adjacent subframes. The interpolated LSP vectors are converted to LPC, obtaining a different LPC filter for each subframe, which is used for calculation of other encoded parameters, such as adaptive and stochastic codevectors and gains.

3. GSM TRANSCODED DATABASES

The whole TIMIT database [3] was downsampled from 16 kHz to 8 kHz, using a 158th-order linear-phase FIR half-band filter, with a very steep transition band (150 Hz of transition band), a very flat passband (passband ripple < 0.1 dB), and more than 97 dB of attenuation in the stop band. Thus, the downsampled speech files

contain basically all the frequencies of the original TIMIT in the 0-4 kHz range. Hereafter, the downsampled database will be referred to as TIMIT8k, while the original will be referred to as TIMIT16k. TIMIT8k was transcoded using the three GSM speech coders. The public domain C-code implementation of the FR coder was used (see Section 2.1), as well as the ANSI-C code for the HR and the EFR provided by ETSI (see Section 2.2 and 2.3). These C-code implementations were compiled, and verified using the test vectors provided by ETSI [7] before their utilization.

4. SPEAKER RECOGNITION SYSTEM

All the experiments were performed using a speaker recognition system based on Gaussian Mixture Models (GMM) classifiers [9]. A GMM classifier of N=16 mixtures was used, as a good compromise between complexity and performance. Diagonal covariance matrices were used for gaussian densities, since the correlation between coefficients is no strong when using cepstral or LSP parameters. The speaker recognition system was programmed in Matlab, using h2m [10], a set of Matlab functions designed by O. Cappe.

4.1 Speaker identification and verification

Given a sequence $(x_t)_{1\le t\le T}$ of feature vectors from a speaker signal, maximum likelihood estimates of the model parameters are obtained using the Expectation-Maximization (EM) algorithm. Given an unknown sequence of signal $(y_t)_{1\le t\le T'}$, the recognized speaker $\hat{s}$ is then obtained with the maximum likelihood (ML) decision rule:

$$\hat{s} = \arg\max_{s} \frac{1}{T'} \sum_{t=1}^{T'} \log p(y_t \mid \lambda_s) \qquad (1)$$

where λ_s is the gaussian mixture speaker model. For speaker verification, a world model is constructed, in order to normalize the scores, which are then compared to a threshold in order to accept or reject the speaker.

4.2 Protocol

The protocol used is the "long training / short test" protocol [11] for speaker identification and verification on TIMIT. The features corresponding to the 5 SX sentences are concatenated for training each speaker model. The average total duration is 14.4 seconds. During the testing of the speaker identification system, the two SA and the three SI sentences of every speaker are tested separately. 430 speakers of the database (147 women and 283 men) are used. Thus, the whole test set consists of 430x5=2150 test patterns of 3.2 seconds each, in average. Even though the SA sentences are the same for each speaker, these sentences are used in the test set. Therefore, the experiments can be considered as totally text independent.

The remaining 200 speakers of the database are used to train the world model needed for the speaker verification experiments. 2150 client accesses and 2150 impostor accesses are made (for each client access, an impostor speaker is randomly chosen among the 429 remaining speakers).

Original		*GSM Transcoded*		
TIMIT16k	*TIMIT8k*	*FR*	*HR*	*EFR*
2.2%	13.1%	31.5%	38.5%	28.2%

Table 1: Speaker identification results (% errors).

Original		*GSM Transcoded*		
TIMIT16k	*TIMIT8k*	*FR*	*HR*	*EFR*
1.1%	5.1%	7.3%	7.8%	6.6%

Table 2: Speaker verification results (% EER).

4.3 Feature Extraction

Feature extraction varies for the different experiments, thus it will be explained as part of the experimental setup (see Section 5, 6 and 7). In all the experiments, the same type of feature extraction and same databases are used for training and testing (matching condition).

5. EXPERIMENTS USING THE TIMIT AND TRANSCODED DATABASES

In this experiment the speech analysis module extracts 16 cepstral coefficients (c_0-c_{15}) from the speech signal, using real cepstrum calculation based on DFT [12]. The frame length is 30 ms and the frame rate is 10 ms.

Table 1 and 2 show the identification and verification errors respectively obtained with the speaker recognition system on TIMIT16k, TIMIT8k, and the GSM transcoded TIMIT (FR, HR and EFR).

The results show significant performance degradation when using GSM transcoded databases, compared to the normal and downsampled versions of TIMIT even if training and testing were both performed with transcoded speech (matched conditions). The results obtained are in correspondence with the perceptual speech quality of each coder. That is, the higher the speech quality is the higher the measured recognition performance.

We see that the degradation of the performance is less important for speaker verification than for speaker identification, but is still significant. These results are equivalent to those obtained in [13], whereas [4] and [5] suggest that the GSM coding does not introduce major degradations. From our point of view, the performance achieved using GSM transcoded speech is not sufficient in a practical context. Thus, in the following sections we investigate the source of the degradation for the FR and the EFR coders, as well as the possibility of performing recognition using directly coder parameters rather than parameters extracted from resynthesized speech.

6. EXPERIMENTS USING THE GSM FR ENCODED PARAMETERS

The goal of these experiments is to explore the possibility of using features extracted directly from the

	Coefficients	*id. error*	*EER*
(1)	**Baseline: resynthesized speech FR**	**31.5%**	**7.3%**
(2)	LPC8 → c0-c15	31.8%	7.0%
(3)	LPC8 → c1-c15	38.0%	7.8%
(4)	LPC12 → c0-c15	24.0%	5.5%
(5)	FR (no q) → c1-c15	43.7%	7.5%
(6)	FR (no q) → c1-c16	43.6%	7.5%
(7)	FR (with q) → c1-c15	40.8%	8.4%
(8)	**Codec param. FR (with q) → ĉ0-c15**	**35.7%**	**7.0%**

Table 3: Speaker verification and identification results for the experiments using the GSM FR encoded parameters (id. = identification).

FR encoded parameters, without resynthesize the decoded speech. Results are given in Table 3. Line (1) lists the values reported from the TIMIT FR experiment in Table 1 and 2 (results obtained by extracting the features from decoded speech). All the experiences (lines (2) to (8)) were carried out using TIMIT8k, but the feature extraction was made compatible with the FR coder characteristics (see Section 2.1.1): 20 ms segmentation, calculation of 8-th order LPC (LPC8), calculation of cepstral coefficients c1-c15 from the LPC using the well known recursion for minimum phase signals [12], and calculation of c0 using log (E), where E is the energy of the LPC residual. The results obtained with this feature extraction are given in line (2) of Table 3. For lines (2) to (4) the feature extraction is done with a C-program, using double-precision floating-point arithmetic:

(3) Uses only cepstral coefficients c1-c15 (no energy term c0).
(4) Uses an LPC model order of 12 instead of 8.

Feature extraction for lines (5) to (8), is done from the FR C-program, which uses a simulated 16-bit fixed-point arithmetic:

(5) Uses c1-c15, from unquantized LPC (before LPC coding/decoding).
(6) Uses c1-c16, from unquantized LPC.
(7) Uses c1-c15, from quantized LPC.
(8) Uses c1-c15, from quantized LPC, and ĉ0, which is calculated using log (Ê), where Ê is the energy of the reconstructed LPC residual.

6.1 Comments on Pair-wise Comparison on Table 3

(1)-(2): The use of the new feature extraction (more compatible with the FR characteristics) does not introduce significant distortion.

(2)-(3): The use of c0 (more laborious to calculate from the bit-stream) is crucial for good performance.

(2)-(4): The low (8-th) LPC order in GSM FR coding is responsible for most performance degradations.

(5)-(6): No performance improvement is expected by retaining cepstral coefficients beyond c15 without increasing the LPC order.

	Coefficients	*id. error*	*EER*
(1)	**Baseline: resynthesized speech EFR**	**28.2 %**	**6.6 %**
(2a)	LPC10 (w2) → c0-c15	25.5 %	6.3 %
(2b)	LPC10 (w4) → c0-c15	25.1 %	6.7 %
(2c)	LPC10 (w2-4) → c0-c15	24.2 %	6.1 %
(3a)	LPC10 (w2) → c1-c15	31.4 %	6.7 %
(3b)	LPC10 (w4) → c1-c15	32.4 %	7.4 %
(3c)	LPC10 (w2-4) → c1-c15	30.0 %	6.8 %
(4a)	LPC12 (w2) → c0-c15	23.4 %	6.1 %
(4b)	LPC12 (w4) → c0-c15	22.8 %	6.1 %
(4c)	LPC12 (w2-4) → c0-c15	22.2 %	5.9 %
(5a)	LPC10 (w2) → ω1-ω10	32.7 %	7.1 %
(5b)	LPC10 (w4) → ω1-ω10	32.0 %	7.0 %
(5c)	LPC10 (w2-4) → ω1-ω10	29.7 %	6.3 %
(6a)	LPC10 (w2) → c1-c10	34.1 %	7.4 %
(6b)	LPC10 (w4) → c1-c10	34.7 %	7.0 %
(6c)	LPC10 (w2-4) → c1-c10	31.4 %	7.0 %
(7a)	EFR (no q) (w2) → c1-c15	33.3 %	7.1 %
(7b)	EFR (no q) (w4) → c1-c15	34.3 %	7.1 %
(7c)	EFR (no q) (w2-4) → c1-c15	32.5 %	6.9 %
(8a)	EFR (no q) (w2) → c1-c16	33.2 %	7.3 %
(8b)	EFR (no q) (w4) → c1-c16	33.1 %	6.9 %
(8c)	EFR (no q) (w2-4) → c1-c16	31.1 %	6.8 %
(9a)	EFR (no q) (w2) → c1-c20	35.6 %	7.3 %
(9b)	EFR (no q) (w4) → c1-c20	34.4 %	8.0 %
(9c)	EFR (no q) (w2-4) → c1-c20	32.7 %	7.2 %
(10a)	EFR (with q) (w2) → c1-c15	35.9 %	7.1 %
(10b)	EFR (with q) (w4) → c1-c15	34.7 %	7.3 %
(10c)	EFR (with q) (w2-4) → c1-c15	33.3 %	7.1 %
(11a)	EFR (with q) (w2) → c1-c15 + ĉ0	31.5 %	7.2 %
(11b)	EFR (with q) (w4) → c1-c15 + ĉ0	30.3 %	6.7 %
(11c)	EFR (with q) (w2-4) → c1-c15 + ĉ0	34.0 %	7.4 %
(12a)	EFR (with q) (w2) → ω1-ω10	34.5 %	7.0 %
(12b)	EFR (with q) (w4) → ω1-ω10	34.7 %	7.1 %
(12c)	EFR (with q) (w2-4) → ω1-ω10	33.5 %	6.2 %
(13a)	**EFR (with q) (w2) → ω1-ω10 + ĉ0**	**29.3 %**	**6.7 %**
(13b)	EFR (with q) (w4) → ω1-ω10 + ĉ0	29.5 %	6.7 %
(13c)	EFR (with q) (w2-4) → ω1-ω10 + ĉ0	32.4 %	7.1 %

Table 4: Speaker verification and identification results for the experiments using the GSM EFR encoded parameters.

(5)-(7): LPC quantization in the FR coder decreases the performance in the verification and improves in the identification. Not conclusive.

(7)-(8): The ĉ0 calculated from the reconstructed residual improves the performance.

(1)-(8): By extracting the features directly from the information in the encoded bit-stream, we have managed to obtain a speaker verification system that is slightly better than the baseline.

7. EXPERIMENT USING THE GSM EFR ENCODED PARAMETERS

Results of these experiments are given in Table 4 and 5. Line (1) lists the values reported from the TIMIT EFR

experiment in Table 1 and 2. As two sets of LPC are calculated every 20-ms frame in the EFR coder (see Section 2.3.1), three possibilities are considered for each type of feature extraction:

(a) Features calculated using window w_2.
(b) Features calculated using window w_4.
(c) Two sets of features per 20 ms frame, calculated using window w_2 and w_4.

All the experiences (line (2a) to (13c)) were carried out using TIMIT8k, but the feature extraction was made compatible with the EFR coder spectral analysis (see Section 2.3.1). For all the experiences, LPC coefficients are converted to cepstral coefficients c1-cn using the recursion for minimum phase signals [12]. The cepstral coefficient c0 (energy term) is calculated using log (E), where E is the energy of the LPC residual. When E is not available (features calculated from the coder parameters) the energy term is calculated using $\hat{c}0 = \log(\hat{E})$, where $\hat{E}$ is the energy of the reconstructed LPC residual. Conversion from LPC to LSP is done using the Matlab function poly2lsf. For lines (2) to (6) the feature extraction is done with a C-program, using double-precision floating-point arithmetic:

(2) Uses c1-c15, from 10-th order LPC, and the energy term c0.
(3) Uses c1-c15 from 10-th order LPC.
(4) Uses c1-c15 from 12-th order LPC, and c0.
(5) Uses LSPs, ω1-ω10, from 10-th order LPC.
(6) Uses only 10 cepstral coefficients, c1-c10, from 10-th order LPC.

Feature extraction for lines (7) to (13), is done from the EFR C-program, which uses a simulated 16-bit fixed-point arithmetic:

(7) Uses c1-c15, from unquantized LPC.
(8) Uses c1-c16, from unquantized LPC.
(9) Uses c1-c20, from unquantized LPC.
(10) Uses c1-c15, from quantized LPC.
(11) Uses c1-c15, from quantized LPC, and $\hat{c}0$, from the energy of the reconstructed LPC residual.
(12) Uses ω1-ω10, from quantized LPC.
(13) Uses ω1-ω10, from quantized LPC, and $\hat{c}0$.

7.1 Comments on Comparisons on Table 4

(1)-(2): The use of the feature extraction compatible with the EFR coder improves the performance with respect to the baseline. Besides, cases (a) and (b) use a feature vector every 20 ms, which is half the amount used in (1). The use of twice as many feature vectors in (c) than in (a),(b) may not be justified by the small performance improvement.

(2)-(3): The use of c0 is crucial for good performance.

(2)-(4): Increasing LPC order from 10 to 12 improves the performance by a modest amount (~2% on identification) compared with the improvement obtained passing from LPC8 to LPC12 in the FR coder (~8 %).

(3)-(5): The use of LSPs ω1-ω10 instead of c1-c15 slightly degrades the performance for (a) and improves it for (b),(c) but the dimension of the vectors is decreased from 15 to 10.

(5)-(6): Use of LSP gives better results than cepstral coefficients, for the same dimension, when using unquantized LPC. This suggests that LSPs are a more compact representation of spectral features.

(10)-(12): Equivalent or better performance is achieved using LSP ω1-ω10 compared with c1-c15, when using quantized LPC, in spite of the dimension reduction from 15 to 10. This positive result may be due to the fact that the EFR coder does LPC quantization in the LSP domain.

(7)-(8)-(9): Increasing the number of cepstral coefficients beyond c15 does not significantly improve, and may actually degrade, the performance.

(3)-(6): Reducing the number of cepstral coefficients from 15 to 10 decreases the performance in most cases.

(3)-(7): Calculations using 16-bit fixed-point arithmetic in the EFR coder decrease the performance in most cases.

(7)-(10): LPC quantization slightly decreases the performance.

(10)-(11) & (12)-(13): $\hat{c}0$ calculated from reconstructed residual improves the performance in cases (a) and (b). Performance degradation in (c) may be due to the way $\hat{c}0$ is calculated: subframes 1-2 (80 samples) are used for calculating $\hat{c}0$ in w_2 while subframes 3-4 are used for $\hat{c}0$ in w_4. We should find a way to calculate $\hat{c}0$ in case (c) that is more consistent with the actual LPC windows.

Coefficients	*id. error*	*EER*
(1) k1-k10 from encoded parameters, k11-12 from transcoded speech → ω1-ω12 + $\hat{c}0$	29.2 %	6.1 %
(2) k1-12 from transcoded speech → ω1-ω12 + $\hat{c}0$	29.8 %	6.9 %
(3) k1-12 from transcoded speech → ω1-ω12 + c0	32.0 %	6.6 %
(4) k1-k12 from original speech → ω1-ω12 + c0	24.7 %	5.9 %

Table 5: Speaker verification and identification results for the experiment on the use of higher order LPC.

7.2 Use of Higher order LPC

In Table 4 it is observed that an increase in LPC order improves the performance, but only 10-th order LPC is available in the EFR encoded parameters. From (1)-(3) we hypothesize that higher order LPC information "leaks" in other encoded parameters (LTP lags and gains, and stochastic pulses and gain) and is thus available in the resynthesized speech, improving recognition. We investigated the use of this higher order LPC information for case (a). The goal is to improve upon (13a), the best result obtained using encoded parameters. Results are given in Table 5. The features used are explained as follows.

(1) LPC from encoded parameters is converted to reflection coefficients k1-k10 and concatenated with

reflection coefficients k11-k12 calculated from resynthesized speech. These concatenated k1-k12 are converted to LSPs ω1-ω12, and used as features, together with ĉ0 calculated from encoded parameters.

(2) Uses ω1-ω12 calculated from resynthesized speech, and ĉ0 from encoded parameters.

(3) Uses ω1-ω12, calculated from resynthesized speech, and c0 from resynthesized speech.

(4) For comparison purposes: Uses ω1 - ω12 from original (TIMIT8k) speech, and c0 from original speech.

In Table 5 it is observed that best results are obtained by using information extracted from the encoded parameters, rather than from resynthesized speech. Nevertheless the performance is still better when extracting features from the original speech.

We have improved upon (13a) in Table 4, got close to the baseline for speaker identification and improved upon the baseline for verification.

8. CONCLUSIONS AND FUTURE WORK

In this paper we have investigated the influence of GSM speech coding on a text-independent speaker recognition system based on GMM classifiers. The recognition performance when extracting features from GSM transcoded speech was measured, and it was found that the achieved performance is not acceptable for practical applications.

Different experiments were carried out, using the FR and EFR coders. It was found that the performance can be improved by using feature extraction directly from encoded parameters rather than from resynthesized speech. The degradation in performance introduced by different aspects of the coders was also measured.

The results obtained also showed that using LSP coefficients improves recognition performance while decreasing complexity (by reducing the dimension of the feature vectors).

From experiments conducted using original and GSM transcoded speech it is observed that major sources of performance degradation are the down-sampling from 16 kHz to 8 kHz and the use of transcoded speech. By extracting features from encoded parameters we have managed to get close to the baseline (features extracted from transcoded speech), but not to improve upon it. Thus, future work should include finding ways of improving the baseline, varying either the speaker recognition system, or the feature extraction. For the latter, we would like to explore the use of mel-cepstral coefficients and of LSP weighting functions to emphasize formant structure and attenuate broad-bandwidth components that introduce undesired variability due to environmental factors.

When extracting features from the encoded parameters, it was found that the performance can be enhanced by the contribution of the residual (reconstructed from encoded parameters other than LPC). In our experiences this contribution was taken into account by using the energy of the reconstructed residual (ĉ0), and higher order LPC information from resynthesized speech. Possible direction of future work is to find effective means to parameterize encoded parameters other than LPC in order to improve recognition performance.

9. ACKNOWLEDGEMENTS

This work was partially supported by the Swiss National Science Foundation under Grant FN 20-53'843, and by the Swiss Federal Office for Education and Science under Grant OFES C97.0050 (COST 254 project).

10. REFERENCES

[1] J.P., Jr. Campbell, *Speaker Recognition: a Tutorial*, Proc. of the IEEE, Vol. 85, no. 9, pp. 1437 –1462, Sep. 1997.

[2] S. Grassi, L. Besacier, A. Dufaux, M. Ansorge, F. Pellandini, *Influence of GSM Speech Coding Algorithms on the Performance of Text-Independent Speaker Identification*, Proc. of Int'l. COST 254 Workshop on Intelligent Communication Technologies and Applications, with Emphasis on Mobile Communications, Neuchâtel, Switzerland, (in print), May 5-7, 1999.

[3] W. Fisher, V. Zue, J. Bernstein, D. Pallet, *An acoustic-phonetic database*, JASA, suppl. A, Vol. 81(S92), 1986.

[4] M. Kuitert and L. Boves, *Speaker verification with GSM coded telephone speech*, Proc. Eurospeech'97, Vol.2, pp. 975-978, 1997.

[5] M. El-Maliki, P. Renevey and A. Drygajlo, *Speaker verification for noisy GSM quality speech*, Proc. of Int'l. COST 254 Workshop on Intelligent Communication Technologies and Applications, with Emphasis on Mobile Communications, Neuchâtel, Switzerland, (in print), May 5-7, 1999.

[6] L. Besacier, S. Grassi, A. Dufaux, M. Ansorge, F. Pellandini, *GSM Speech Coding and Speaker Recognition*, Accepted for publication at ICASSP'00, Istanbul, Turkey, June 5-9, 2000.

[7] http://www.etsi.org

[8] http://kbs.cs.tu-berlin.de/~jutta/toast.html

[9] D. Reynolds, *Speaker Identification and Verification using Gaussian Mixture Speaker Models*, Workshop on Automatic Speaker Recognition, Identification and Verification, Martigny, Switzerland, pp. 27-30, April 5-7, 1994.

[10] O. Cappe, *h2m : A set of MATLAB functions for the EM estimation of hidden Markov models with Gaussian state-conditional distributions*, ENST/Paris.

http://sig.enst.fr/~cappe/h2m/html/.

[11] F. Bimbot, I. Magrin-Chagnolleau, L. Mathan, *Second-order Statistical Methods for Text-Independent Speaker Identification*, Speech Communication, n°.17 (1-2), pp. 177-192, Aug. 1995.

[12] J. R. Deller, J. G. Proakis and J. H. L. Hansen, *Discrete-Time Processing of Speech Signals*, Prentice Hall, New Jersey, USA, 1993.

[13] T.F. Quatieri, E. Singer, R.B. Dunn, D.A. Reynolds, J.P. Campbell, *Speaker and Language Recognition Using Speech Codec Parameters*, Proc. Eurospeech'99, Vol. 2, pp. 787-790, 1999.