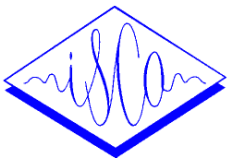

2001: A Speaker Odyssey
The Speaker Recognition Workshop
Crete, Greece
June 18–22, 2001

Speaker Recognition and the ETSI Standard Distributed Speech Recognition Front-End

Charles C. Broun[†], *William M. Campbell*[†], *David Pearce*[‡], *Holly Kelleher*[‡]

[†]Motorola Human Interface Lab, Tempe, Arizona 85284, USA
[‡]Motorola Limited, Basingstoke, UK

Abstract

With the advent of Wireless Application Protocol (WAP) and 2.5/3G communication systems, the mobile device has become a window to the Internet. A natural interface to this mobile device is through speech. To address this need, a new European Telecommunications Standards Institute (ETSI) standard front-end has evolved for Distributed Speech Recognition (DSR). The goal of the ETSI DSR front-end is to standardize client-server speech recognition applications with a common feature set and quantization method. Although originally evolved as a *speech* recognition standard, we propose it is also a method of standardizing distributed *speaker* recognition authentication. To this end, we perform experiments using the DSR parameterization for a speaker recognition application. Results indicate excellent preservation of speaker identity in the DSR standard. This testing shows that DSR brings the potential for a promising new era of portable authentication for applications in personalization and security.

1. Introduction

Biometrics [1] is a maturing field with outstanding potential in many modern authentication systems. Biometrics simplifies the interface to the human user by eliminating the need for passwords and personal identification numbers (PIN). They are cumbersome at best because of various practices currently in use. By their nature, passwords and PINs are difficult to remember, must be changed frequently, and are subject to "cracking". Biometrics solves these problems by the use of various distinguishing characteristics of individuals. Authentication methods commonly used are voice, fingerprints, hand geometry, iris structure, facial characteristics, etc. Access is controlled through a verification process that determines whether a claimant's characteristics match those of the claimed identity.

Biometrics solutions for networked environments must address three criteria: 1) the security must be as good as, or better than, the existing password system, 2) the authentication mechanism must be accepted by the end users, and 3) the biometric solution must be inexpensive to implement. The final criterion comprises several considerations. The verification transaction rate must be scalable, and not tax the existing servers. The user models must have a sufficiently small memory footprint to allow for efficient storage and transmission across the network. Lastly, the barriers to entry must be minimized. It is unreasonable to expect a customer to incur the cost of retina scanners for mobile devices.

The concept of Distributed Speech Recognition (DSR) is powerful. DSR separates the structural and computational components of recognition into two parts – front end processing on the client system and the speech recognition engine on the back-end. This separation of tasks enables a flexible architecture with great potential.

There are several advantages of the DSR structure. First, a standard front-end increases accuracy by minimizing mismatch. Currently, a variety of vocoders exist for wireless systems. The proliferation of different coding methods introduces *different* artifacts for different vocoders causing mismatch problems for speech recognizers based directly on the coded speech. Second, DSR is based on a data network. Because of the new standards in WAP, DSR fits naturally into the wireless Internet architecture. Third, DSR is attractive since it focuses on speech recognition alone. That is, standards in this area are produced to work well with modern speech recognition systems. Also, standards are implemented to minimize the impact of bit errors over standard communication channels. Finally, DSR is attractive because of the limited processing power of modern cell phones. DSR puts the main burden of computation and upgradability on the server side. This feature avoids costly and difficult attempts to squeeze speech recognition code into a small platform.

Our goal in this paper is to explore the potential of using the ETSI DSR front-end for speaker recognition. The benefits are numerous. One of the main difficulties in speaker recognition is the cross-channel problem; i.e., enrolling on speech from one channel (e.g., vocoders and communication channel) and then testing on speech from another channel with different characteristics has the potential to cause severe degradation in performance. To be acceptable as a standard for security, speaker recognition must guarantee a reasonable level of accuracy. DSR circumvents this problem by avoiding the telephone system. The speech is coded at the input source and the features are transmitted across a data channel. A second advantage of DSR for speaker recognition is that, as in the speech recognition case, it avoids the need for multiple designs and strategies for different vocoders. Also, we can reduce computational load if we need to perform *only* verification and not send voice content. A third advantage of DSR is that it is integrated in the data network. Thus, it is easy to envision integrating authentication with Internet security.

The choice of the polynomial classifier, as opposed to other classifiers such as hidden Markov models (HMM) and neural networks, is based on our earlier work [2]. We have shown the utility of this particular solution for applications that dictate low-computational complexity and minimal memory resources. The *Cipher***VOX** system, [2], is based on a client-server approach where high transaction rates are desired.

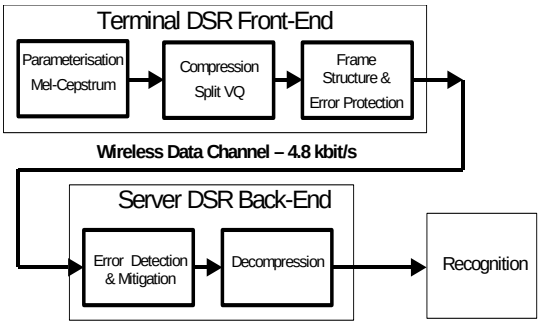

Figure 1: Block Diagram of DSR System

Since the DSR standard is based on a similar concept, we draw upon our experience in implementing this system.

The outline of this paper is as follows. In Section 2, we introduce the ETSI DSR standard front-end. We give basic design criteria and the data format. In Section 3, we show the structure of the classification system used for speaker recognition. Section 4 has experiments showing the effect of the ETSI DSR front-end on speaker recognition on the YOHO database.

2. ETSI DSR standard front-end

To enable widespread applications using DSR in the market place, a standard for the front-end is needed to ensure compatibility between the terminal and the remote recognizer. The Aurora DSR Working Group within ETSI has been actively developing this standard over the last two years. The first DSR standard was published by ETSI in February 2000. This standard and some aspects of its performance are described as follows.

The overall DSR system is illustrated in *Figure 1*. The mel-cepstrum was chosen as the feature set for the first standard because of its widespread use throughout the speech recognition industry. *Figure 2* shows a block diagram of the processing stages for the DSR front-end. At the terminal the speech signal is sampled and parameterized using a mel-cepstrum algorithm to generate 12 cepstral coefficients together with C0 and a log energy parameter. These are then compressed using a split vector quantizer to obtain a lower data rate for transmission. To be suitable for today's wireless networks a data rate of 4800 b/s was chosen as the requirement. The compressed parameters are formatted into a defined bit stream for transmission.

It is anticipated that the DSR bit stream will be used as a payload in other higher-level protocols when deployed in specific systems supporting DSR applications. Thus the standard does not cover the areas of data transmission or any higher-level application protocols that may run over them. In this respect it is similar to speech codec standards where the codec is specified separately to the systems that use it.

The defined bit stream is sent over a wireless or wire line transmission link to the remote server where parameters received with transmission errors are detected and the front-end parameters are decompressed to reconstitute the DSR mel-cepstrum features. These are passed to the recognition decoder residing on the central server. The recognizer back-end is not part of the standard.

Since the data channels used for the transport of the DSR bit stream may be subject to errors (transparent data channels), special attention has been given to make the whole system robust to the types of burst errors that occur on wireless channels. To achieve this, error detection bits are added in the terminal DSR encoder as part of the bit stream and a special error mitigation algorithm is used at the decoder.

When developing the standard the following requirements were met:

- Mel-Cepstrum feature set consisting of 12 cepstral coefficients logE and C0
- Data transmission rate of 4800 b/s

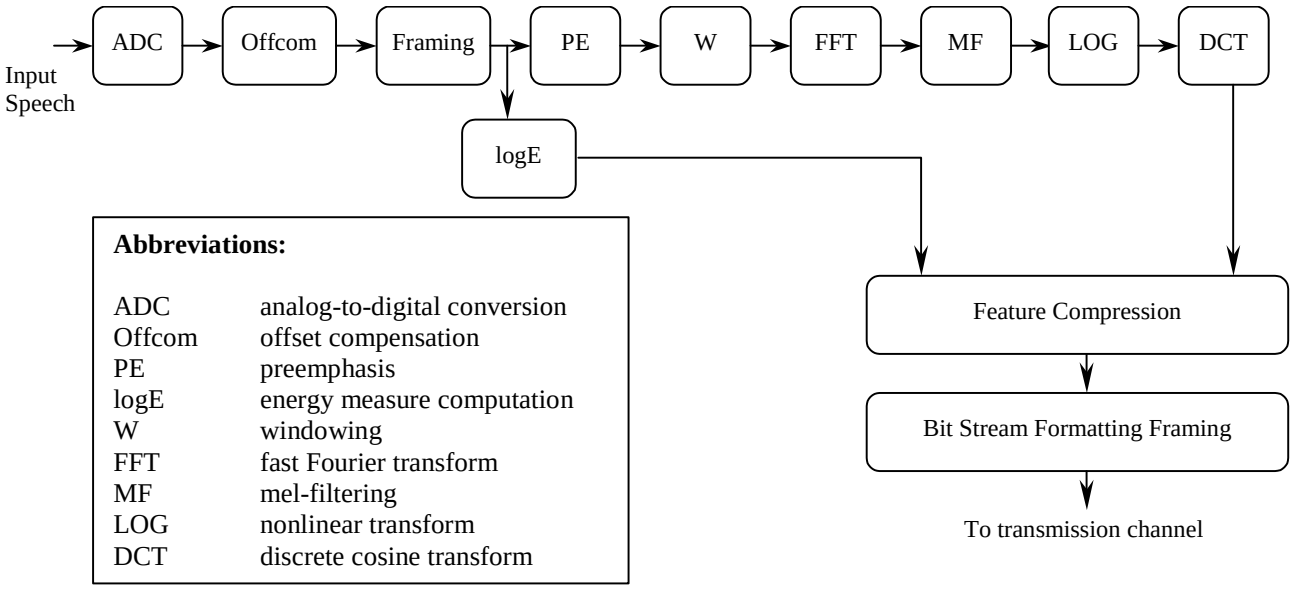

Figure 2: DSR front-end.

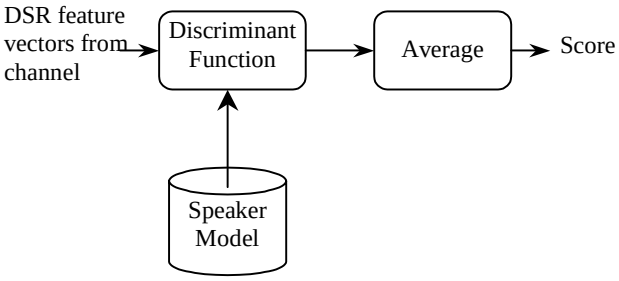

Figure 3: Classifier structure.

- Low computational and memory requirements for implementation in the mobile terminals
- Low latency
- Robustness to transmission errors

Full details of the algorithms and architecture are given in the paper [3] and the standards document [4] (accessible from the ETSI standards web site).

3. Classifier structure

The basic structure of our classifier, on the server DSR back-end, is shown in *Figure 3*. The feature vectors, $\mathbf{x}_1 \ldots \mathbf{x}_M$, produced from the DSR front-end and transmitted across the channel, are presented to the classifier. A discriminant function [5] is applied to each feature vector, $\mathbf{x}_k$, using a speaker model, $\mathbf{w}$, producing a scalar output, $f(\mathbf{x}_k,\mathbf{w})$, representing the frame score. The final score for the speaker model is then computed as the average of the individual frame scores,

$$s = \frac{1}{M}\sum_{k=1}^{M} f(\mathbf{x}_k, \mathbf{w}). \qquad (1)$$

Comparing the final score to a threshold, T, performs the accept/reject decision for the system. If $s < T$, then the claim is rejected; otherwise the claim is accepted.

Our pattern classifier uses a polynomial discriminant function [6],

$$f(\mathbf{x}, \mathbf{w}) = \mathbf{w}^t \mathbf{p}(\mathbf{x}). \qquad (2)$$

The polynomial discriminant function is composed of two parts. The first part, $\mathbf{w}$, is the speaker model. The second part, $\mathbf{p}(\mathbf{x})$, is a polynomial basis vector constructed from input feature vector $\mathbf{x}$. This basis vector is the monomial terms up to degree K of the input features. For example, for a two dimensional feature vector, $\mathbf{x}=[x_1\ x_2]^t$, and $K=2$, we have

$$\mathbf{p}(\mathbf{x}) = \begin{bmatrix} 1 & x_1 & x_2 & x_1^2 & x_1 x_2 & x_2^2 \end{bmatrix}. \qquad (3)$$

Thus, the polynomial discriminant function output is a linear combination of the polynomial basis elements.

For a server-based application, a high transaction rate is desired. For speaker recognition applications, the server load is typically determined by the complexity of the discriminant function evaluation. For the polynomial classifier, this evaluation can be simplified as follows. Since $\mathbf{w}$ does not depend on the frame index, the computational complexity is reduced as illustrated in equation (4).

$$s = \mathbf{w}^t \frac{1}{M}\sum_{k=1}^{M} \mathbf{p}(\mathbf{x}_k) = \mathbf{w}^t \overline{\mathbf{p}} \qquad (4)$$

Thus, only a single vector, $\overline{\mathbf{p}}$, representing the input speech is generated from the feature vectors transmitted across the network. A single transaction equates to computing an inner product between a speaker model and this vector. The number of floating point operations (FLOPS) is

$$2N_{\text{model}} - 1, \qquad (5)$$

where N_{model} is the length of $\mathbf{w}$. Thus for 12 features and a 3rd order (K=3) polynomial expansion, $\mathbf{w}$ is of length 455, resulting in only 909 flops per transaction, and a model size of 1820 bytes for a floating point representation.

A method of training the classifier is given in [7].

Table 1: Performance of the CipherVOX engine for YOHO and the DSR Standard. The numeric quantity for each entry is the average EER in % for a one-phrase test.

Verify / Enroll	Un-quantized	Error-Free	EP1	EP2	EP3
Unquantized	1.18	-	-	-	-
Error-Free	-	1.22	1.22	1.26	1.67
EP1	-	1.22	1.22	1.26	1.67
EP2	-	1.22	1.22	1.27	1.66
EP3	-	1.26	1.26	1.30	1.70

4. Experiments

We evaluated the performance of the ETSI DSR front-end on YOHO. The YOHO database is a publicly available speaker verification database. It is a natural choice since it is live microphone speech collected with no telephone line distortion (other than band-limiting); also, several comparisons are available in the literature [8]. The YOHO database consists of 138 speakers enrolled in 4 separate sessions. Each session has 24 enrollment phrases of the form "23-45-56" (3 doublets). For verification, there are 10 sessions consisting of 4 phrases of the same form. Additional details of the YOHO database are available in [8].

For verification, we used the classification system and testing methodology described in [7]. A polynomial of degree 3 was constructed from the 12 input mel-cepstral features from the DSR front-end; this results in a model size of 455 coefficients. Endpointing was performed on a per utterance basis using the DSR front-end's log energy parameter. All four enrollment sessions in the YOHO database were used for training the classifier. Verification was performed using one-phrase tests for a total of 40 tests per speaker. Performance is gauged in terms of the average equal error rate (EER) across all speakers.

We tested the ETSI standard in several configurations. First, as a baseline, we found the average EER for the *unquantized* parameter set; this situation would never occur in practice, but it is needed to show the degradation from quantization. Second, we tested the ETSI standard across several GSM channels: EP1, EP2, and EP3. These channels show the effects of bit errors on enrollment and verification.

Table 1 shows the results of using the ETSI DSR front-end standard with the configuration described. We note several items of interest. First, the results show that the loss of accuracy when going from the unquantized situation to the quantized error-free situation is negligible; the average EER increases only from 1.18% to 1.22%. This shows that the speaker identity is well maintained by the front-end. Second, the results show that the channel degradation has only modest effect on the EER. As in reference [3], the worst situation is seen for EP3. As mentioned in [3], EP3 is an unusual channel situation and represents an extreme. Thus, our increase from 1.22% to 1.70% average EER is quite encouraging; we would probably not notice this kind of increase in a typical application scenario.

5. Conclusions

We have tested the performance of speaker recognition using the new ETSI DSR front-end. Although originally designed for *speech* recognition applications, we have demonstrated that the standard works well for *speaker* recognition applications. This attribute opens the exciting potential for the use of the ETSI standard as a new robust method for authentication through wireless and wire line devices attached to the Internet.

6. References

[1] *Biometrics: Personal Identification in a Networked Society*, A. K. Jain, R. Bolle, and S. Pankanti eds., Kluwer Academic Publishers, 1999.

[2] Fette, B. A., Broun, C. C., Campbell, W. M., and Jaskie, C., "CipherVOX: Scalable Low-Complexity Speaker Verification," in *Proceedings of The IEEE International Conference on Acoustics, Speech and Signal Processing*, 2000.

[3] Pearce, David, "Enabling New Speech Driven Services for Mobile Devices: An Overview of the ETSI standards activities for Distributed speech Recognition Front-ends," *Proceedings AVIOS 2000*.

[4] ETSI ES 201 108 v1.1.2 Distributed Speech Recognition; Front-end Feature Extraction Algorithm; Compression Algorithm, April 2000, http://www.etsi.org.

[5] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. Academic Press, 1990.

[6] J. Schürmann, *Pattern Classification*. John Wiley and Sons, Inc., 1996.

[7] W. M. Campbell and K. T. Assaleh, "Polynomial classifier techniques for speaker verification," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 321-324, 1999.

[8] J. P. Campbell, Jr., "Testing with the YOHO CD-ROM voice verification corpus," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 341-344, 1995.