Continuous Speaker Verification in Realtime

Max Kunz[1], Klaus Kasper[2], Herbert Reininger[1], Manuel Möbius[1], Jonathan Ohms[1]

[1]atip GmbH, Daimlerstraße 32, D-60314 Frankfurt am Main
[2]Hochschule Darmstadt, Fachbereich Informatik

{max.kunz, herbert.reininger, manuel.moebius, jonathan.ohms}@atip.de
klaus.kasper@h-da.de

Abstract: Biometric speaker verification deals with the recognition of voice and speech features to reliably identify a user and to offer him a comfortable alternative to knowledge-based authentication methods like passwords. As more and more personal data is saved on smartphones and other mobile devices, their security is in the focus of recent applications. Continuous Speaker Verification during smartphone phone calls offers a convenient way to improve the protection of these sensitive data.

This paper describes an approach to realize a system for continuous speaker verification during an ongoing phone call. The aim of this research was to investigate the feasibility of such a system by creating a prototype. This prototype shows how it is possible to use existing technologies for speaker verification and speech recognition to compute segments of a continuous audio signal in real-time. In line with experiments, a simulation study was made in which 14 subjects first trained the system with a freely spoken text and then verified themselves afterwards. Additional intruder tests against all other profiles where also simulated.

1. Introduction

The protection of confidential data and the authentication of users to access these data are recently becoming more and more important. Especially with the growing amount of offers for different web portals and telephone-based services and the growing number of mobile devices, personal data tends to be saved in a distributed manner. Mostly, identification numbers or passwords are used as authentication method. With growing computing power and elaborate software to spy on passwords, they have to be longer and more complex to keep the data safe.

For this reason biometric authentication processes are very promising. They use biometric features like fingerprints, the iris of the eye, the face, the voice or other biometric features or patterns of behavior for identification. It is presupposed that these features are unique for a person, even if they are not equally distinctive for everyone.

The biometric speaker verification deals with the recognition of voice features to authenticate the user. In common practice the verification is performed with a dedicated voice application, which explicitly asks the user for certain utterances that were optimally chosen for speech verification.

An important application for such biometric verification systems is seen in mobile phones. Usually these devices are only protected by an identification number entered only at startup. Additional protection against intruders can be provided if voice verification is performed concurrent to phone calls. If a speaker cannot be matched to the speech profile of the authorized user, the device could lock and the user would be asked to enter his personal identification number once more.

In the investigation described in this paper the application scenario of a continuous real-time verification is evaluated. The aim is to verify the user during any voice input, for example during a phone call. During the conversation, a continuous rating of the identity of the speaker is computed. The text independent verification of freely spoken language proved to be a particular challenge. In opposite to explicit verification dialogues the continuous verification system does not know the text of the spoken utterance in advance.

2. System Configuration

The initial setup for the implementation of the prototype was based on the speech verification software VoxGuard created by atip GmbH. The prototype uses VoxGuard for verification of single audio segments. Therefore the signal of a specific length with a sample rate of 8 kHz was sampled and quantized with a 16-bit resolution. The features[1] that are important for speaker verification were extracted from the audio by means of spectral analysis. To identify the features, continuous HMMs (Hidden Markov Models) [1] were used. Every HMM represents a phoneme (the smallest distinctive phone) [2]. Altogether 41 different phoneme models were used. This allowed verification not just of known passphrases, but also of any possible utterance. To calculate Scores of the features with the help of HMMs, the Viterbi algorithm was used and a likelihood that a series of features build a certain HMM was determined.

Before every verification the single phonemes were individually trained for every subject. Here, several samples that contain an utterance with the needed phoneme were recorded. The extracted features were projected onto an initial model. During this, the free parameters were calculated: the emission probability (as Gaussian density), the transition probability and the start probability of the HMM. This procedure was repeated with every training data set, so that the model adopted more and more trained data.

[1] As feature for the verification, Mel- Cepstrum coefficients were used.

While it is still possible to give the speaker a certain text during training (and to transcribe this text in advance into phonemes), it is much more difficult to do so during the verification of a free text, as the system does not know the utterance before it is spoken. This problem was solved by using a phoneme recognizer before the verification system. This recognizer uses speaker independent models to recognize any phoneme well. In order to recognize an arbitrary sequence of phonemes, a loop grammar was defined. Inside this grammar, at every time, the occurrence of every phoneme of the inventory is allowed. In figure 1, the principle of this loop grammar is illustrated.

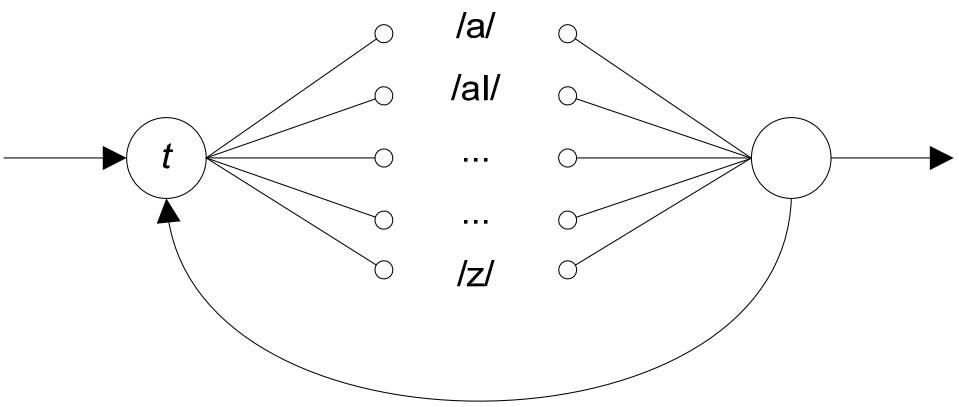

Figure 1: Loop-Grammar for recognition of phoneme sequences.

At every point in time t the scores for every phoneme model in the inventory is computed. With that, it is possible to build a two-dimensional Viterbi pattern which has exactly the width of the combined inventory of the phonemes and the length T. T is defined as the amount of blocks that were created during block-wise computing and the features that were extracted of them [3]. Then, the most probable path through this pattern is computed with the backtracking algorithm [4]. This path goes backwards along the states with the best score at any point in time.

With this loop grammar and the speech recognizer it is possible to retrieve the most probable series of phonemes for an utterance. The recognition rate for single phonemes is quite low though. It usually reaches only 30-50%, so in most cases the transcription will not be correct for the spoken utterance. Only an approximation of the voices in the audio signal to the phonemes is possible.

3. Implementation

The prototype of the continuous verification system consists of three modules. In these steps the aforementioned methods for the speaker recognition and verification are used. The following illustration shows the processing chain which comprises the speaker recognition and speaker verification. The basis for continuous verification is the audio segmentation, which cuts the running signal into smaller audio segments for the subsequent steps.

When the user makes a phone call, the audio signal is permanently recorded by the system. For further processing it is split into segments.

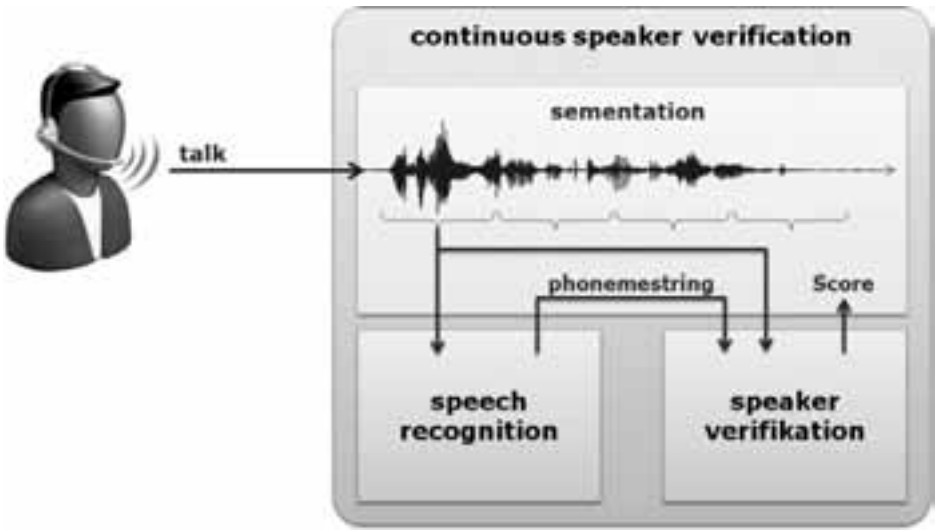

Figure 2: Modular setup of the continuous speaker verification

The phonemes contained in the audio segments are determined with the speech recognizer.

In a second step, this sequence of phonemes and the original audio segment is passed to the speaker verification system VoxGuard. In this module the verification is performed and a score is computed.

For every segment of the whole audio signal a score is computed. The continuous sequence of scores is analyzed and used to make a decision about acceptance or rejection.

4. Evaluation

During the evaluation, parameters setting are varied in order to investigate the influence. The aim is to measure the quality of the verification and the influence of different parameter settings to the quality of the verification. A closer look at the verification algorithms or the recognition of the phonemes is not part of this evaluation and will be done in further investigations.

To estimate the performance of the verification system, a simulation study with 14 subjects was made. In a controlled environment, speech samples of the subjects were recorded. Three different, freely spoken texts were recorded resulting in 3.5 minutes of audio material per subject.

Using the recordings, the 14 speech profiles of the different subjects were trained. After that every speaker attempted verification with a text of approximately 1.5 minutes length. Additionally, each of these 1.5 minute samples was used to simulate an intruder attack against all 13 profiles of the other users. In this way, 196 verifications could be simulated and the achieved scores could be logged. From these 196 trials, 14 were from authorized users and 182 were the intruder attacks, which should lead to a rejection by the system.

Figure 3 gives an example of the scores of a user whose audio is verified against his own profile. The other graph shows the score distribution of an intruder who tries to verify against the same profile. There is an obvious tendency, that the authorized user usually achieves better values. In the figure can be seen that the scores of the two speakers clearly overlap partially.

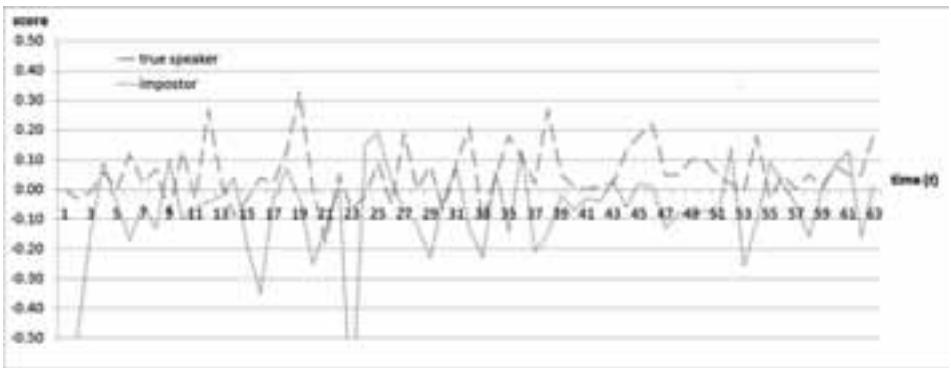

Figure 3: Score distribution of two speakers with an audio segment length of two seconds

The evaluation showed that the length of the audio segment has a significant influence on the variance of the scores. Although short segments allow a much higher frequency of retrieving verification scores for a user, fewer phonemes are included in the audio segments, which lead to a higher sensibility to variations in the pronunciation or background noise.

A further possibility to increase the robustness of the system is to use several scores and not just a single one to make a decision. As the acceptance/rejection decision does not have to be based on just a single score at any point of time, a time window that includes recent scores can be considered for the decision. As can be seen in figure 4, the variation of the score distribution of the aforementioned two speakers can be drastically reduced if an average over the last 6 scores is taken.

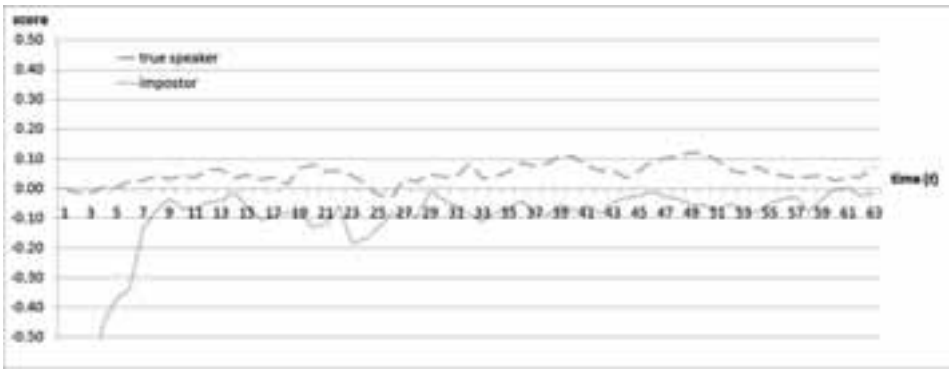

Figure 4: Average score distribution in a time window of 6 scores

The evaluation aims not only to measure the efficiency of the verification and its dependence on the segment length, but also to compare these results with the efficiency by using different window sizes.

5. Results

To rate the performance of the verification system, the Detection Error Tradeoff method (DET) was used. The DET graph shows the comparison between the „False Acceptance Rate“ (FAR) and the „False Rejection Rate“ (FRR). In this graph you can see the possible error rates independent from any decision threshold.

The following graph shows the comparison between the efficiency of the verification system and different audio segment lengths. Audio segments with lengths from 1 to 4 seconds were investigated.

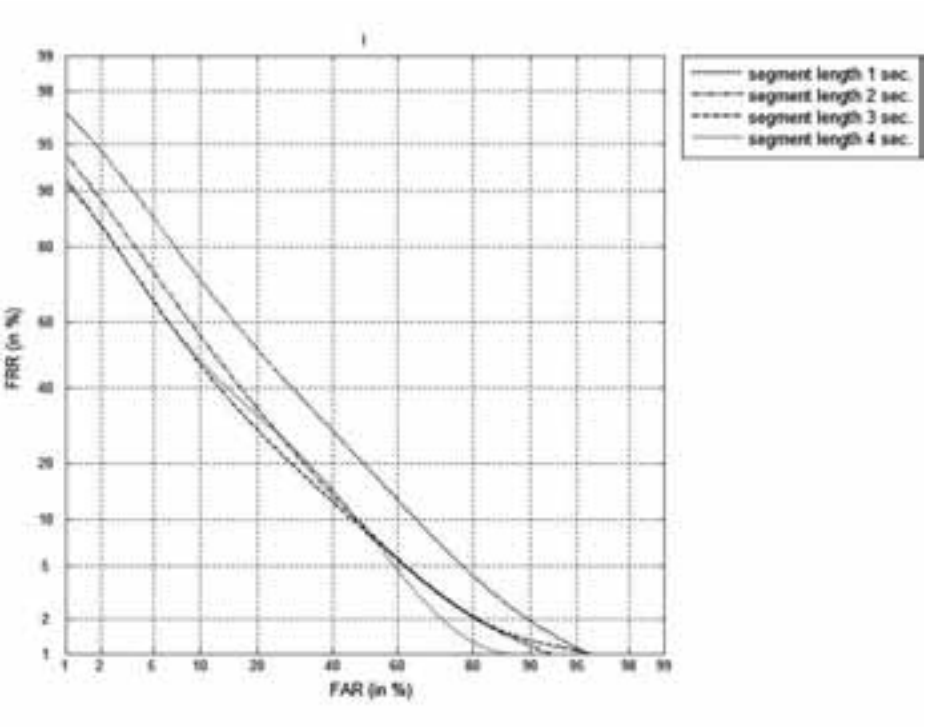

Figure 5: Comparison of rates achieved with different segment lengths

The DET graphs demonstrate that with a larger segment length a higher verification performance is achieved. Additionally, one can remark, that with a length of 2 seconds no significant improvements can be seen. Larger audio segment lengths achieve only a very little increase of the verification performance.

The last figure shows that the performance of the acceptance/rejection decision of the system is dependent on the length of the audio segments. To allow objective evaluation of system efficiency/performance, a decision strategy based on the average scores is used by averaging scores within a time window of 12 sec. For example if the segment length is 4 seconds, three scores are needed for one average value, if the segment length is two seconds, 6 values are needed to calculate an average score of 12 seconds of speech. The achieved verification performance can be seen in figure 6.

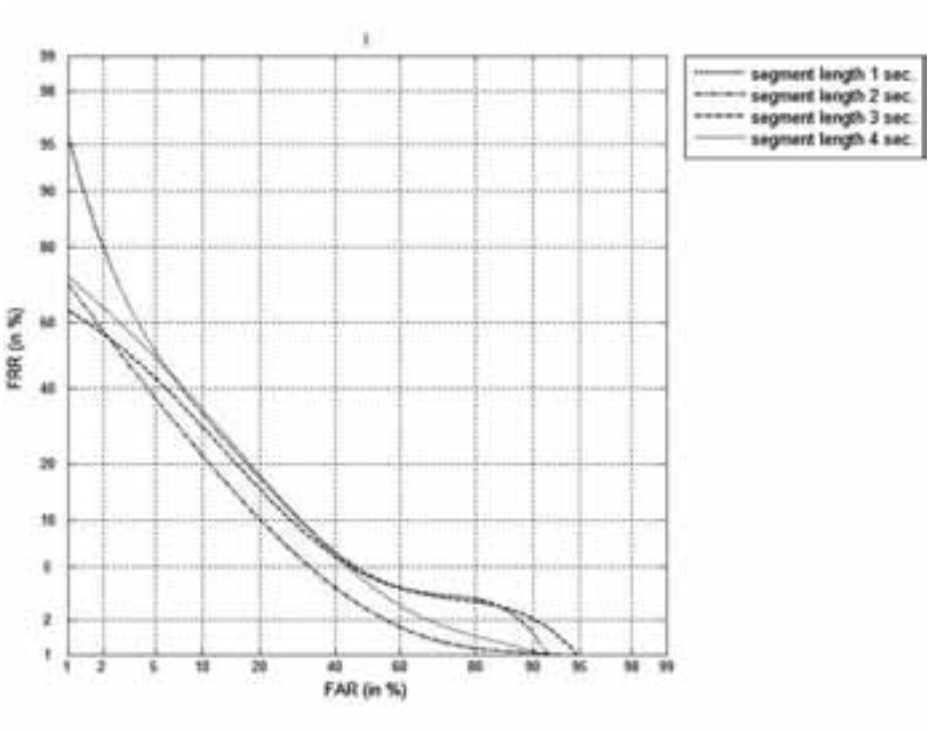

Figure 6: Comparison between the audio segment lengths for an average value of 12 seconds

It becomes obvious from this figure that it is not always an advantage to use a larger segment length to calculate the average scores. It turned out that with short segment lengths a similiar verification performance can be achieved as with larger ones.

The last evaluation shows that especially with a segment length of 2 seconds and an average value over 6 scores extraordinarily good performance can be achieved. Every two seconds the system calculates a new score and determines the distribution of the score in the 12-second time windows. In this way, an Equal Error Rate (EER) of 15% was achieved.

6. Summary and Perspectives

To evaluate on the feasibility of a continuous speaker verification system a prototype based on existing speech recognition and speaker verification technologies was implemented. This prototype allows to continuously verify a speaker in real-time. The focus of this research was the evaluation of the system's performance and its connection to varying parameter settings.

With the prototype and its evaluation it could be shown that the continuous verification of utterances with arbitrary content is feasible with existing speech recognizers and verification algorithms. Further, two parameters, segment length and frame size, were investigated. During this, some advantages of a continuous verification system became obvious: The system does not have to determine a decision about acceptance or rejection based on a single score but is able to use a score distribution over a certain time window. This way, it achieves additional robustness against intermittent bad scores as they happen in periods of loud background noises or differences in the speakers pronunciation.

In a further investigation the phoneme inventory will be optimized for verification. In opposite to a verification of a single passphrase, free spoken speech contains additional variations like e.g. the intake of breath before new sentences or longer breaks between sentences and single words. These variations can be recognized and eliminated for the following computation with help of the preceding speech recognizer. Additionally the phoneme inventory can be investigated as to whether certain phonemes are particularly well-suited for certain speakers so that the model inventory can be adapted to the speaker.

Bibliography

[1] **B.H. Juang, L.R. Rabiner.** Hidden Markov Models for Speech Recognition. *Technometrics.* 1991, Aug.

[2] **Lyon, John.** *Einführung in die moderne Linguistik.* s.l.: C.H. Beck. Bd. 8.

[3] **Beat Pfister, Tobias Kaufmann.** *Sprachverarbeitung - Grundlagen und Methoden der Sprachsynthese und Spracherkennung.* s.l.: Springer, 2008.

[4] **John R. Deller, John G. Proakis, John H. L. Hansen.** *Discrete-Time Processing of Speech Signals.* s.l.: Prentice Hall, 1993.

[5] **Fink, Gernot A.** *Mustererkennung mit Markov-Modellen - Theorie, Praxis, Anwendungsgebiete.* s.l.: Vieweg+Teubner, 2003. Bd. 1.

[6] **Wendemuth, Andreas.** *Grundlagen der stochastischen Sprachverarbeitung.* s.l.: Oldenbourg, 2004.

[7] **Manfred R. Schroeder, H. W. Strube, H. Quast.** *Computer Speech - Recognition, Compression, Synthesis.* s.l.: Springer, 2004. Bd. 2.