Developing Speaker Recognition System: from Prototype to Practical Application

P. Fränti, J. Saastamoinen, I. Kärkkäinen*, T. Kinnunen, V. Hautamäki, I. Sidoroff

Speech & Image Processing Unit,
Dept. of Computer Science and Statistics,
University of Joensuu, FINLAND

*Institute for Infocomm Research (I^{2}R),
Agency for Science, Technology and
Research (A*STAR), SINGAPORE

Abstract. In this paper, we summarize the main achievements made in the 4-year PUMS project during 2003-2007. The emphasis is on the practical implementations, how we have moved from Matlab and Praat scripting to C/C++ implemented applications in Windows, UNIX, Linux and Symbian environments, with the motivation to enhance technology transfer. We summarize how the baseline methods have been implemented in practice, how the results are utilized in forensic applications, and compare recognition results to the state-of-art and existing commercial products such as ASIS, FreeSpeech and VoiceNet.

1 Introduction

Voice-based person identification can be a useful tool in *forensic research* where any additional piece of information can guide the inspections to the correct track. Even if 100% matching cannot be reached by the current technology, it may be enough to get the correct speaker ranked high enough among the tested ones.

A state-of-art *speaker recognition system* consists of components shown in Fig. 1. The methods are based on short-term features such as *mel-frequency cepstral coefficients* (MFCCs), but two longer term features are considered here as well: *long-term average spectrum* (LTAS) and *long-term distribution of the fundamental frequency* (F0). After feature extraction, the similarity of a given test sample is measured to previously trained models stored in a speaker database. In person authentication applications, the similarity is measured relative to a known or estimated *universal background model* (UBM) which represents speech in general, and draw conclusion whether the sample should be accepted or rejected. Sometimes a match confidence measure is also desired. In forensics, it may be enough to find a small set (say 3-5) of the best matching speakers for further investigations by a specialist phonetician.

In this paper, we overview the results of *speaker recognition* (SRE) research done within the Finnish nationwide *PUMS*[1] project funded by TEKES[2]. The focus has been to transfer research results into practical applications. We studied the existing SRE methodology and proposed several new solutions with practical usability and real-

[1] Puheteknologian uudet menetelmät ja sovellukset – New methods and applications of speech technology (http://pums.fi)

[2] National Technology Agency of Finland (http://www.tekes.fi)

time processing as our main motivations. As results of the project, we developed two pieces of software: *WinSProfiler* and *EpocSProfiler*. The first one is used by forensic researchers in the National Bureau of Investigations (NBI) in Finland, and the second is tailored to work in mobile environment.

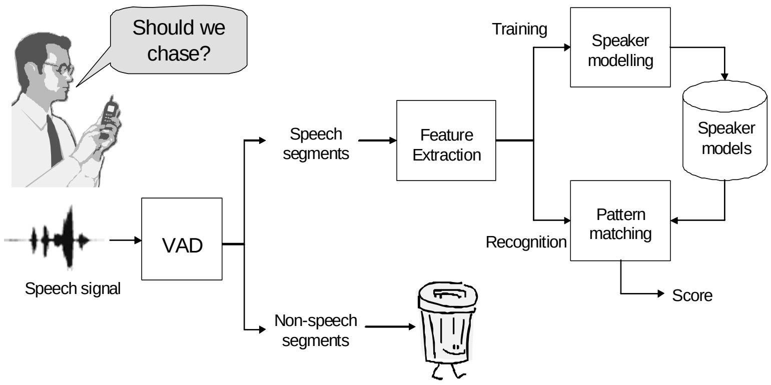


Fig. 1. Overall system diagram for speaker recognition.

The rest of the paper is organized as follows. In Section 2, we review the feature extraction and speaker modeling components used in this study, and study the effect of the voice activity detection by experimenting with several existing techniques and new ones developed during the project. Implementation aspects are covered in Section 3, and results of the implemented software are given in Section 4. The implemented methods are compared against two prototype systems developed for the *NIST*[3] speaker recognition evaluation (SRE) competition[4] in 2006. Conclusions are drawn in Section 5.

2 Speaker recognition

2.1 Short-term spectral features

Our *baseline method* is based on the *mel-frequency cepstral coefficients* (MFCCs), which is a representation of an approximation of the short-term spectrum (Fig. 2). The audio signal is first divided into 30 ms long frames with 10 ms overlap. Each segment is then converted into spectral domain by the *fast Fourier transform* (FFT), filtered according to a psycho-acoustically motivated *mel-scale* frequency warping, where lower frequency components are emphasized more than the higher ones. The feature vector consists of 12 DCT magnitudes of filter output logarithms. The corresponding

[3] National institute of standards and technology
[4] http://www.nist.gov/speech/tests/spk/2006

1st and 2nd temporal differences are also included to model the rate and acceleration of changes in the spectrum. The lowest MFCC coefficient (referred to as C0) represents the log-energy of the frame, and is removed as a form of energy normalization. Mean subtraction and variance normalization is then performed for each coefficient to have zero mean and unit variance over the utterance.

The main benefit of using MFCC is that it is also used in speech recognition, and the same signal processing components can therefore be used for both. This is also its main drawback: the MFCC feature tends to capture information related to the speech content better than the personal speaker characteristics. If the MFCC features are applied as such, there is a danger that the recognition is mostly based on the content instead of the speaker identity. Another similar feature, *linear prediction cepstral coefficients* (LPCC), was also implemented and tested but the MFCC remained our choice of practice.

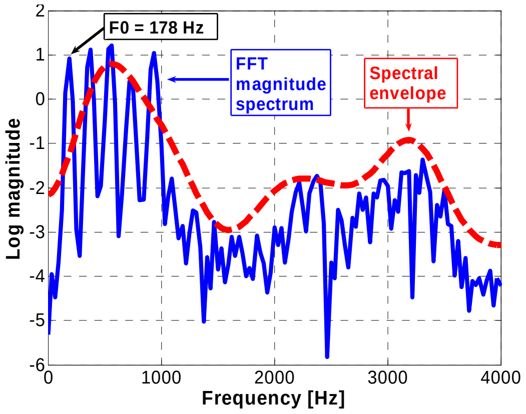


Fig. 2. Illustration of a sample spectrum and its approximation by cepstral coefficients.

2.2 Long term features

Besides the short-term features, two longer-term features were studied: *Long-term average spectrum* (LTAS) and *long-term distribution of the fundamental frequency* (F0). The first one is motivated by the facts that it includes more spectral detail than MFCC and as a long time average it should be more robust on changing conditions. On the other hand, it is also criticized by the same reasons: it represents only averaged information over time and all information about variance of the utterance is evidently lost.

Results in [14] showed that LTAS provides only marginal additional improvement when fused with the stronger MFCC features, but at the cost of making the overall system more complex in terms of implementation and parameter tuning, see Fig. 3. Even though LTAS is used in forensic research for visual examination, its use in automatic analysis has no proven motives.

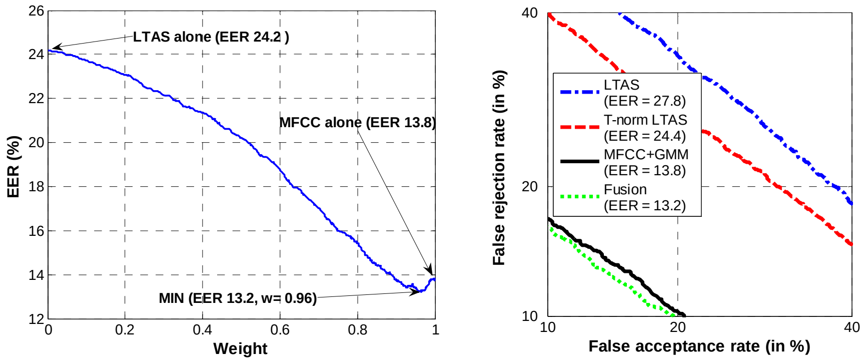


Fig. 3. An attempt to improve the baseline by adding LTAS via classifier fusion. The difficulty of tuning the fusion weights is shown on left. The corresponding results of the best combination are shown on right for NIST 2001 corpus.

Fundamental frequency, on the other hand, does contain speaker-specific information, which is expected to be independent of the speech content. Since this information is not captured by MFCCs, it can potentially improve recognition accuracy of the baseline system. However, it is not trivial to extract the F0 feature and use it in the matching process. These issues were extensively studied using combination of F0, its derivative (delta), and the *log-energy* of the frame. This combination is referred to as *prosody vector*, and it was implemented in WinSProfiler 2.0.

The results support the claim that the recognition accuracy of F0 is consistent under changing conditions. In clean conditions, no improvement was obtained in comparison to the MFCC baseline. In noisy conditions (additive factory noise with 10 dB SNR), the inclusion of F0 improved the results according to our tests [12]. It is open whether this translates to real-life applications. With the NIST corpora (see Section 4) the effect of F0 is mostly insignificant, or even harmful, probably because the SNR of the NIST files is better than the 10 dB noise level of our simulations.

2.3 Speaker modeling and matching

After feature extraction, the similarity or dissimilarity of a given test sample to the trained models in a speaker database must be measured. We implemented the traditional *Gaussian mixture model* (GMM), where the speaker model is represented as a set of cluster means, covariance matrixes, and mixture weights, and a simpler solution based on *vector quantization* (VQ): estimated cluster centroids represent the speaker model. In [7], we found out that the simpler VQ model provides similar results with significantly less complex implementation than GMM. Nevertheless, both methods have been used and implemented in WinSProfiler 2.0. In the mobile implementation, only the VQ model was implemented at first. Later a new compact *feature histogram* model has been implemented as well.

The background normalization (UBM) is crucial for successful verification. Existing solution known as *maximum a posteriori* (MAP) adaptation was originally formulated for the GMM [21]. The essential difference to clustering-based methods is that the model is not constructed from scratch to approximate the distribution of feature vectors. Instead it is an iteration which starts from the background model. Similar solution for the VQ model was then formulated during the project [7].

In addition to modeling a single feature set, a solution is needed to combine the results of independent classifiers. A *linear weighting* scheme optimized using *Fisher's criterion* and *majority voting* have been implemented. On the other hand, fusion is not necessarily wanted in practical solutions because the additional parameter tuning is non-trivial. In this sense, the performance of the method in WinSProfiler 2.0 could be improved but it is uncertain if it is worth it, or whether it would work in practical application at all. The use of data fusion is more or less experimental and is not considered as a part of the baseline.

2.4 Voice activity detection

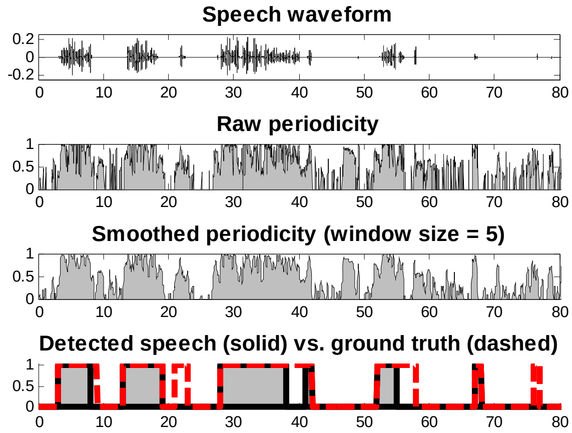
The goal of *voice activity detection* (VAD) is to divide a given input signal into parts that contain speech and the parts that contain background. In speaker recognition, we want to model the speaker only from the parts of a recording that contain speech.

We carried out extensive study of several existing solutions, and developed a few new ones during the course of the project. Real-time operation is necessary in VAD applications such as speaker recognition where latency is an important issue in practice. The methods can also be classified according to whether separate training material is needed (trained) or not (adaptive). Methods that operate without any training are typically based on short-term signal statistics. We consider the following non-trained methods: *Energy*, LTSD, *Periodicity* and the current telecommunication standards: G729B, AMR1 and AMR2, see Table 1.

Trained VAD methods construct separate speech and non-speech models based on annotated training data. The methods differ in both the type of used feature and model. We consider two methods based on MFCC features (SVM, GMM), and one based on *short-term time series* (STS). All of these methods were developed during the PUMS project. We also modified the LTSD method to adapt the noise model from a separate training material instead of using the beginning of the sound signal.

Figure 4 shows an example of the process, where the speech waveform is transformed frame by frame to the speech/non-speech decisions using the Periodicity-based method [8]. First, features of the signal are calculated, and smoothed by taking into account the neighboring frames (five frames in our tests). The final decisions (speech or non-speech) are made according to a user select threshold. In real applications, the problem of selecting the threshold should also be issued.

The classification accuracy of the tested VAD methods is summarized in Table 1 for the four datasets as documented in [25]. For G729B, AMR, and STS, we set the threshold when combining individual frame-wise decisions to one second resolution decisions, by counting the speech and non-speech frame proportions in each segment.


Speech waveform

Raw periodicity

Smoothed periodicity (window size = 5)

Detected speech (solid) vs. ground truth (dashed)

Fig. 4. Demonstration of voice activity detection from frame-wise scores to longer segments using Periodicity method [8].

Table 1. Speech detection rate (%) comparison of the VAD methods with the four data sets.

	VAD method	NIST 2005	Bus stop	Lab	NBI
Adaptive	Energy [24]	1.5	14.6	16.8	30.0
	LTSD [20]	40.0	19.2	14.4	31.8
	Periodicity [8]	3.2	21.9	9.9	21.4
	G729B [9]	8.9	6.5	7.9	**13.3**
	AMR1 [5]	5.5	5.7	7.2	21.8
	AMR2 [5]	8.4	7.4	**5.1**	16.1
Trained	SVM [14]	11.6	5.2	19.5	---
	GMM [10]	8.8	7.5	9.7	---
	LTSD [20]	**1.3**	6.2	14.9	---
	STS (unpublished)	7.1	**3.9**	8.6	---

For the NIST 2005 data, the simple energy-based and the trained LTSD provide the best results. This is not surprising since the parameters of the method have been optimized for earlier NIST corpuses through extensive testing, and because the energy of the speech and non-speech segments is clearly different in most samples. Moreover, the trained LTSD clearly outperforms its adaptive variant because the noise model initialization failed on some of the NIST files, and caused high error values.

The NBI data is the most challenging, and all adaptive methods have values higher than 10%. The best method is G729B with the error rate of 13%. It is an open question how much better results could be reached if the trained VAD could be used for these data. However, in this case the training protocol and the amount of trained material needed should be studied more closely.

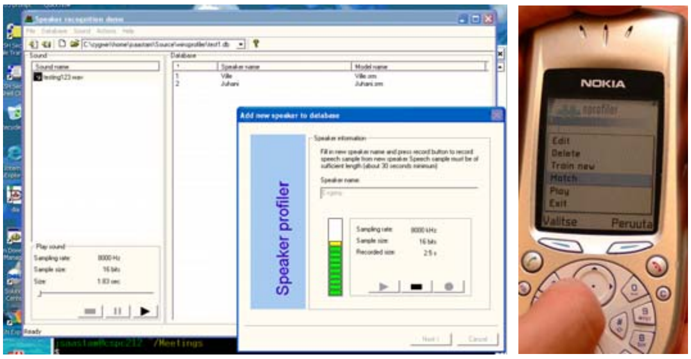
For WinSProfiler 2.13, we have implemented the three VAD methods that performed best in NIST data: *LTSD*, *Energy* and *Periodicity*. Their effect on speaker verification accuracy is reported in Table 2. The advantage of using VAD in this application with NIST 2006 corpus is obvious, but the choice between *Energy* and *Periodicity* is unclear.

Table 2. Effect of VAD in speaker verification performance (error rate %).

	NIST 2001		NIST 2006
	Model size 512	Model size 64	Model size 512
No VAD	13.6	16.0	44.4
LTSD	12.4	13.7	35.8
Energy	9.3	10.4	**16.6**
Periodicity	**8.5**	**9.6**	16.8

3 Methods implemented and tested

Experimentation using Praat and Matlab is rather easy and convenient for quick testing of new ideas, but that is not true for technology transfer or larger scale development. Our aim was to have the baseline methods implemented in C/C++ language for software integration with real products, and also for performing large scale tests. Applications were therefore built for three platforms: UNIX/Linux (*SProfiler*), Windows (*WinSProfiler*) and Symbian (*EpocSProfiler*), see Fig. 5.



#	Speaker name	Model name
1	Ville	Ville.sm
2	Juhani	Juhani.sm

Fill in new speaker name and press record button to record speech sample from new speaker. Speech sample must be of sufficient length (about 30 seconds minimum)

Sampling rate: 8000 Hz
Sample size: 16 bits
Size: 1.83 sec

Sampling rate: 8000 kHz
Sample size: 16 bits
Recorded size: 2.5 s

Fig. 5. Constructed applications where the developed SRE system was implemented during the project: *SProfiler* (not shown), *WinSProfiler* (left), and *EpocSProfiler* (right).

3.1 Windows application: WinSProfiler

First applications (WinSProfiler 1.0 and EpocSProfiler 1.0) were developed based on speaker recognition library called *Srlib2*, which had clear specifications of the functionalities of the training and matching operations. However, the functionality was too much tied with the user interface making porting to other platforms complicated.

In order to avoid multiple updates for all software, the library was then reconstructed step-by-step, ending up to a significant upgrade in 2006 and 2007, which was renamed to PSPS2 (*portable speech processing system 2*). Main motivation of this large but invisible work was that the software should be maintainable, modular, and portable. The following life cycle of the recognition library appeared during the project: Srlib1 (2003) → Srlib2 (2004) → Srlib3 (2005-2006) → PSPS2 (2006-2007).

As a consequence, all the functionality in WinSProfiler was re-written to support the new architecture of the PSPS2 library so that all unnecessary dependencies between the user interface and the library functionality were finally cleared, and above all that the software would be flexible and configurable for testing new experimental methods. This happened as a background project during the last project year (2006-07). Eventually a new version (WinSProfiler 2.0) was released in Spring 2007, and a series of upgrades were released since then: 2.1 (June-07) → 2.11 (July-07) → 2.12 (Aug-07) → 2.13 (Oct-07) → 2.14 (June-08).

The current version (WinSProfiler 2.14) is written completely using C++ language, consisting of the following components:

- Database library to handle storage of the speaker profiles.
- Audio processing library to handle feature extraction and speaker modelling.
- Recognition library to handle matching feature streams against speaker models.
- Configurable audio processing and recognition components.
- Graphical user interface.

The GUI part is based on 3rd party C++ development library *wxWidgets*. Similarly, 3rd party libraries *libsndfile* and *portaudio* were used for the audio processing, and *SQLite3* was used for the database. The rest of the system is implemented by us: signal processing, speaker modeling, matching and graphical user interface. The new version was extensively tested, and the functioning of the recognition components was verified step-by-step with the old version (WinSProfiler 1.0). The new library architecture is show in Fig. 6.

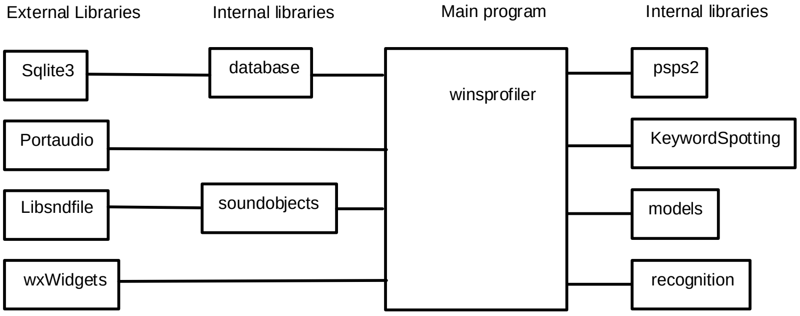


Fig. 6. Technical organization of the WinSProfiler 2.0 software.

3.2 Symbian implementation: EpocSProfiler

During the first project year, the development of a *Symbian* implementation was also started with the motivation to implement a demo application for Nokia *Series 60* phones. Research was carried on for faster matching techniques by speaker pruning, quantization and faster search structures [13]. The existing baseline (Srlib 2) was converted to Symbian environment (Srlib 3) in order to have real-time MFCC signal processing, as well as instant on-device training, identification, and text-independent verification from spoken voice samples.

The development of the *EpocSProfiler* software was made co-operatively with Nokia Research Center during the first project year, and the first version (EpocSProfiler 1.0, based on Srlib 2) was published in April 2004. The Symbian development was then separated from PUMS and further versions of the software (EpocSProfiler 2.0) were developed separately, although within the same research group, using the same core library code, and mostly by the same people.

The main challenge was that the CPU was limited to fixed-point arithmetic. Conversion of floating point algorithms to fixed-point itself was rather straightforward but the accuracy of the fixed-point MFCC was insufficient. Improved version was developed [22] by fine-tuned intermediate signal scaling, and more accurate 22/10 bit allocation scheme of the FFT.

Two voice model types were implemented: centroid model with MSE-based matching as the baseline and a new faster experimental *feature histogram modelling* with entropy-based matching was developed for EpocSProfiler 2.0. In identification, training and recognition response of the new histogram models on a Nokia 6630 device is about 1 second for a database of 45 speakers, whereas the training and identification using the centroid model are both more than 100 times slower.

3.3 Prototype solutions for NIST competition

In addition to the developed software, two prototype systems were also considered based on the NIST 2006 evaluation. NIST organizes annually or bi-annually a speaker recognition evaluation (NIST SRE) competition. The organizers have collected speech material and then release part of it for benchmarking. Each sample has an identity label, gender, and other information like, for example, the spoken language. At the time of evaluation, NIST then sends to the participants a set of verification trials (about 50.000 in the main category alone) with claimed identities of listed sound files. The participants must send their recognition results (accept or reject claim, and likelihood score) within 2-3 weeks. The results are released in a workshop and are available for all participants.

For this purpose, we developed a prototype method for the NIST 2006 competition in collaboration with Institute for Infocomm Research (IIR) at Singapore[5]. This method is referred here as *IIRJ*. The main idea was to include three independent classifiers, and calculate overall result by classifier fusion. A variant of the baseline (SVM-LPCC) [4] with T-norm [1] was one component, F0 another one, and GMM tokenization [18] the third one (Fig. 9). In this way, different levels of speaker cues are extracted: spectral (SVM-LPCC), prosodic (F0), and high-level (GMM tokenization). The LPCC feature showed slightly better results at IIR and replaced MFCC.

As a state-of-art, we consider the method reported in [2]. It provided the best recognition performance in the main category (1conv-1conv) and is used here as a benchmark. This system was constructed by a combination of several MFCC-based

[5] Institute for Infocomm Research (I^{2}R)

subsystems similar to ours, combined by SVM-based data fusion [2]. Based on analytical comparison with our MFCC baseline, the main components missing from our software are *heteroscedastic linear discriminant analysis* (HLDA) [17], [3] and *eigenchannel normalization* [11].

The authors at the Brno University of Technology (BUT) later reported simplified variant of the method [3], showing that similar result can be achieved based on the carefully tuned baseline method without fusion and using multiple sub-systems. The authors of the method in [11] have also expressed the same motivation, i.e. to keep the method simple and avoid the use data fusion. The problem of data fusion in practical applications is that the additional parameter tuning is non-trivial, and its role is more or less for demonstrating theoretical limits that given system can reach. The fusion implemented in *WinSProfiler* is therefore mainly for experimental purposes and not considered here as a part of the baseline.

4 Summary of the main results

Even though usability and compatibility are important issues for a practical application, an important question is the identification accuracy the system can provide. We have therefore collected here the main recognition results of the methods developed during the project, and made an attempt to compare them with the state-of-the-art (according to NIST evaluation), and provide indicative results from comparisons with existing commercial programs. The corpora used are summarized in Table 3.

Table 3. Databases that have been used in the evaluation.

Corpus	Trials	Speakers	Length of training data	Length of test data
NIST 2001 (core test)	22,418	174	2 min	2-60 s
NIST 2006 (core test)	53,966	731	5 min	5 min
Sepemco	494	45	12-60 s	9-60 s
TIMIT	184,900	430	15-35 s	5-15 s
NBI data	62	62	42-150 s	10-93 s

4.1 Recognition results

The following methods have been included in the tests reported here:

- WinSProfiler 1.0: An early demo version from 2005 using only the raw MFCC coefficients without deltas, normalization, and VAD. VQ model of size 64 is used.
- WinSProfiler 2.0: A new version released in May 2007 based on the PSPS2 recognition library developed already in late 2006. Main differences were use of GMM-UBM, deltas, and normalization. The first version did use neither VAD nor gender information (specific for NIST corpus).
- WinSProfiler 2.11: Version released in June 2007, now included gender information (optional) and several VADs, of which the periodicity-based method [8] has been used for testing.

- EpocSProfiler 2.1: Symbian version from October 2006. Corresponds to WinSProfiler 1.0 except that the histogram models are used instead of VQ.
- NIST-IIRJ: Our joint submission with IIR to NIST competition based on the LPCC-SVM, GMM tokenization and F0 features, and fusion by NN and SVM, using Energy-based VAD. This system does not exist as a program, but the results have been constructed manually using scripting.
- NIST state-of-the-art: The results released by the authors providing the winning method in NIST 2006 competition as a reference.

The main results (verification accuracy) are summarized in Table 4 as far as available. The challenging NIST 2001 corpus has been used as the main benchmark since summer 2006. Most remarkable lesson is that, even though the results were reasonable for the easier datasets (TIMIT), they are devastating for the *WinSProfiler* 1.0 when NIST 2006 was used. The most remarkable improvements have been achieved in the latter stage of the project since the release of the PSPS2 library used in *WinSProfiler* 2.11.

Another observation is that the role of VAD was shown to be critical for NIST 2006 evaluation (45% vs. 17%), but this did not generalize to *Sepemco* data (7% vs. 13%). This arises the questions whether the database could be too specific, and how much the length of training material would change the design choices and parameters used (model sizes, use of VAD). Although NIST 2006 has a large number of speakers and huge amount of test samples, the length of the samples is typically long (5 minutes). Moreover, the speech samples are usually easy to differentiate from background by a simple energy-based VAD. The background noise level is also rather low.

Table 4. Summary of verification (equal error rate) results (0 % is best) using the NIST 2001, NIST 2006 and the *Sepemco* database.

Method and version	Sepemco	TIMIT	NIST 2001	NIST 2006
EpocSProfiler 2.1 (2006)	12 %	8 %	---	46 %
WinSProfiler 1.0 (2005)	24 %	---	33 %	48 %
WinSProfiler 2.0 (no-vad)	7 %	3 %	16 %	45 %
WinSProfiler 2.11 (2007)	13 %	9 %	11 %	17 %
NIST submission (IIRJ)	---	---	---	7 %
State-of-art [2]	---	---	---	4 %

4.2 Comparisons with commercial products

Speaker identification comparisons with three selected commercial software (*ASIS*, *FreeSpeech*, *VoiceNet*) are summarized in Table 5 using NBI material obtained by phone tapping (with permission). Earlier results with WinSProfiler 1.0 for different dataset have been reported in [19]. The current data (TAP) included two samples from 62 male speakers: the longer sample was used for model training and the shorter one for testing. The following software has been tested:

- WinSProfiler, Univ. of Joensuu, Finland, www.cs.joensuu.fi/sipu/
- ASIS, Agnitio, Spain, http://www.agnitio.es
- FreeSpeech, PerSay, Israel, http://www.persay.com
- VoiceNet, Speech Technology Center, Russia, http://www.speechpro.com
- Batvox, Agnitio, Spain, http://www.agnitio.es

The results have been provided by Tuija Niemi-Laitinen at the Crime laboratory in National Bureau of Investigation, Finland. The results are summarized as how many times the correct speaker is found as the first match, and how many times among the top-5 in the ranking. WinSProfiler 2.11 performed well in the comparison, which indicates that it is at par with the commercial software (Table 5).

Besides the recognition accuracy, *WinSProfiler* was highlighted as having good usability in the NBI tests, especially due to its ease of use, fast processing, and the capability to add multiple speakers into the database in one run. Improvements could be made for more user-friendly processing and analysis of the output score list though.

Overall, the results indicated that there is large gap between the recognition accuracy obtained by the latest methods in research, and the accuracy obtained by available software (commercially or via the project). In NIST 2006 benchmarking, accuracy of about 4 to 7% could be reached by the state-or-the-art methods such as in [2], and by our own submission (IIRJ).

Direct comparisons to our software WinSProfiler 2.11, and indirect comparisons to the commercial software gave us indications of how much is the difference between “*what is*” (commercial software, our prototype) and “*what could be*”. It demonstrates the fast development of the research in this area, but also shows the problem that tuning towards one data can set lead undesired results for another data set.

Table 5. Recognition accuracies (100% is best) of WinSProfiler 2.11 and the commercial software for NBI data (TAP).

Software	Used samples	Failed samples	Top-1	Top-5
ASIS	51	11	67 %	92 %
WinSProfiler 2.11 (*)	51	11	53 %	100 %
WinSProfiler 2.11	62	0	53 %	98 %
FreeSpeech	61	1	74 %	98 %
VoiceNet	38	24	29 %	52 %

(*) Selected sub-test with those 51 samples accepted by ASIS.

5 Conclusions

Voice-based recognition is technically not mature, and the influence of background noise and changes in recording conditions affects too much the recognition accuracy to be used for access control as such. The technology, however, can already be used in forensic research where any additional piece of information can guide the inspections to the correct track. Even if 100% matching cannot currently be reached, it can be enough to detect the correct suspect high in ranking.

In this paper, we have summarized our work that resulted in software called *WinSProfiler* that serves as a practical tool supporting the following features:

- Speaker recognition and audio processing.
- Speaker profiles in database.
- Several models per speaker.
- Digital filtering of audio files.
- MFCC, F0 + energy and LTAS features.
- GMM and VQ models (with and w/o UBM).
- Voice activity detection by energy, LTSD and periodicity-based methods.
- Keyword search (support for Finnish and English languages).
- Fully portable (Windows, Linux and potentially Mac OS X).

Extended version of this report appears in [6].

Acknowledgements

The work has been supported by the *National Technology Agency of Finland* (TEKES) as the four year project *New Methods and Applications of Speech Technology* (PUMS) under the contracts 40437/03, 49398/04 40356/05, 40195/06.

References

1. R. Auckenthaler, M. Carey and H. Lloyd-Thomas, "Score normalization for text-independent speaker verification systems", *Digital Signal Processing,* 10(1-3), pp. 42--54, January 2000.
2. N. Brummer, L. Burget, J. Cernocky, O. Glembek, F. Grezl, M. Karafiat, D.A. van Leeuwen, P. Matejka, P. Schwarz, A. Strasheim, "Fusion of Heterogeneous Speaker Recognition Systems in the STBU Submission for the NIST Speaker Recognition Evaluation 2006", *IEEE Trans. Audio, Speech and Language Processing*, 15(7), 2072--2084, 2007.
3. L. Burget, P. Matejka, P. Schwarz, O. Glembek, J.H. Cernocky, "Analysis of Feature Extraction and Channel Compensation in a GMM Speaker Recognition System", *IEEE Trans. Audio, Speech and Language Processing*, 15(7), 1979--1986, Sept. 2007.
4. W.M. Campbell, J.P. Campbell, D.A. Reynolds, E. Singer and P.A. Torres-Carrasquillo, "Support vector machines for speaker and language recognition", *Computer Speech and Language,* 20(2-3), pp. 210--229, April 2006.
5. ETSI, “Voice Activity Detector (VAD) for Adaptive Multi-Rate (AMR) Speech Traffic Channels”, *ETSI EN 301 708 Recommendation*, 1999.
6. P. Fränti, J. Saastamoinen, I. Kärkkäinen, T. Kinnunen, V. Hautamäki, I. Sidoroff, "Implementing speaker recognition system: from Matlab to practice", Research Report A-2007-4, Dept. of Comp. Science, Univ. of Joensuu, Finland, November 2007. (http://cs.joensuu.fi/sipu/pub.htm)
7. V. Hautamäki, T. Kinnunen, I. Kärkkäinen, J. Saastamoinen, M. Tuononen and P. Fränti, "Maximum a posteriori adaptation of the centroid model for speaker verification", *IEEE Signal Processing Letters*, 15, 162--165, 2008.

8. V. Hautamäki, M. Tuononen, T. Niemi-Laitinen and P. Fränti, "Improving speaker verification by periodicity based voice activity detection", *Int. Conf. on Speech and Computer (SPECOM'07),* Moscow, Russia, vol. 2, 645--650, October 2007.
9. ITU, "A Silence Compression Scheme for G.729 Optimized for Terminals Conforming to Recommendation V.70", *ITU-T Recommendation G.729-Annex B*, 1996.
10. S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume 2: Detection Theory,* Prentice Hall. 1998.
11. P. Kenny, P. Ouellet, N. Dehak, V. Gupta and P. Dumouchel, "A study of inter-speaker variability in speaker verification", *IEEE Transactions on Audio, Speech and Language Processing*, 16 (5), 980--988, July 2008.
12. T. Kinnunen, R. Gonzalez-Hautamäki, "Long-Term F0 Modeling for Text-Independent Speaker Recognition", *Int. Conf. on Speech and Computer (SPECOM'2005),* Patras, Greece, 567--570, October 2005.
13. T. Kinnunen, E. Karpov and P. Fränti, "Real-time speaker identification and verification", *IEEE Trans. on Audio, Speech and Language Processing*, 14 (1), 277--288, January 2006.
14. T. Kinnunen, V. Hautamäki and P. Fränti, "On the use of long-term average spectrum in automatic speaker recognition", *Int. Symposium on Chinese Spoken Language Processing (ISCSLP'06)*, Singapore, 559--567, December 2006.
15. T. Kinnunen, E. Chernenko, M. Tuononen, P. Fränti and H. Li, "Voice activity detection using MFCC features and support vector machine", *Int. Conf. on Speech and Computer (SPECOM'07)*, Moscow, Russia, vol. 2, 556--561, October 2007.
16. T. Kinnunen, J. Saastamoinen, V. Hautamäki, M. Vinni and P. Fränti, "Comparative evaluation of maximum a posteriori vector quantization and Gaussian mixture models in speaker verification", *Pattern Recognition Letters,* (accepted)
17. N. Kumar, A. G., Andreou, **"**Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition", *Speech Communication,* 26(4), pp. 283--297, December 1998.
18. B. Ma, D. Zhu, R. Tong, H. Li, "Speaker Cluster based GMM tokenization for speaker recognition", *Proc. Interspeech* 2006, pp. 505--508, Pittsburg, USA, September 2006.
19. T. Niemi-Laitinen, J. Saastamoinen, T. Kinnunen, and P. Fränti, "Applying MFCC-based automatic speaker recognition to GSM and forensic data", *2^{nd} Baltic Conf. on Human Language Technologies* (*HLT'05*), 317--322, Tallinn, Estonia, April 2005.
20. J. Ramirez, J.C. Segura, C. Benitez, A. de la Torre and A. Rubio, "Efficient voice activity detection algorithms using long-term speech information", *Speech Communications*, 42(3-4), pp. 271--287, 2004.
21. D.A. Reynolds and T.F. Quatieri and R.B. Dunn, "Speaker Verification Using Adapted Gaussian Mixture Models", *Digital Signal Processing,* 10(1), pp. 19--41, January 2000.
22. J. Saastamoinen, E. Karpov, V. Hautamäki and P. Fränti, "Accuracy of MFCC based speaker recognition in series 60 device", *Journal of Applied Signal Processing*, (17), 2816--2827, September 2005.
23. J. Saastamoinen, Z. Fiedler, T. Kinnunen and P. Fränti, "On factors affecting MFCC-based speaker recognition accuracy", *Int. Conf. on Speech and Computer (SPECOM'05),* Patras, Greece, 503--506, October 2005.
24. R. Tong, B. Ma, K. A. Lee, C. H. You, D. L. Zhou, T. Kinnunen, H. W. Sun, M. H. Dong, E. S. Ching, and H. Z. Li, "Fusion of acoustic and tokenization features for speaker recognition," *5th In. Symp. on Chinese Spoken Language Proc.,* pp. 566--577, Singapore, 2006.
25. M. Tuononen, R. González Hautamäki and P. Fränti, "Automatic voice activity detection in different speech applications", *Int. Conf. on Forensic Applications and Techniques in Telecommunications, Information and Multimedia (e-Forensics'08),* Adelaide, Australia, Article No.12, January 2008.