Available online at www.sciencedirect.com
ScienceDirect
Speech Communication 52 (2010) 12–40
SPEECH COMMUNICATION
www.elsevier.com/locate/specom

An overview of text-independent speaker recognition: From features to supervectors

Tomi Kinnunen [a,*], Haizhou Li [b]

[a] Department of Computer Science and Statistics, Speech and Image Processing Unit, University of Joensuu, P.O. Box 111, 80101 Joensuu, Finland
[b] Department of Human Language Technology, Institute for Infocomm Research (I^2R), 1 Fusionopolis Way, #21-01 Connexis, South Tower, Singapore 138632, Singapore

Received 4 November 2008; received in revised form 1 July 2009; accepted 20 August 2009

Abstract

This paper gives an overview of automatic speaker recognition technology, with an emphasis on text-independent recognition. Speaker recognition has been studied actively for several decades. We give an overview of both the classical and the state-of-the-art methods. We start with the fundamentals of automatic speaker recognition, concerning feature extraction and speaker modeling. We elaborate advanced computational techniques to address robustness and session variability. The recent progress from vectors towards supervectors opens up a new area of exploration and represents a technology trend. We also provide an overview of this recent development and discuss the evaluation methodology of speaker recognition systems. We conclude the paper with discussion on future directions.

Keywords: Speaker recognition; Text-independence; Feature extraction; Statistical models; Discriminative models; Supervectors; Intersession variability compensation

1. Introduction

Speaker recognition refers to recognizing persons from their voice. No two individuals sound identical because their vocal tract shapes, larynx sizes, and other parts of their voice production organs are different. In addition to these *physical* differences, each speaker has his or her characteristic *manner of speaking*, including the use of a particular accent, rhythm, intonation style, pronunciation pattern, choice of vocabulary and so on. State-of-the-art speaker recognition systems use a number of these features in parallel, attempting to cover these different aspects and employing them in a complementary way to achieve more accurate recognition.

An important application of speaker recognition technology is *forensics*. Much of information is exchanged between two parties in telephone conversations, including between criminals, and in recent years there has been increasing interest to integrate automatic speaker recognition to supplement auditory and semi-automatic analysis methods (Alexander et al., 2004; Gonzalez-Rodriguez et al., 2003; Niemi-Laitinen et al., 2005; Pfister and Beutler, 2003; Thiruvaran et al., 2008b).

Not only forensic analysts but also ordinary persons will benefit from speaker recognition technology. It has been predicted that telephone-based services with integrated speech recognition, speaker recognition, and language recognition will supplement or even replace human-operated telephone services in the future. An example is automatic password reset over the telephone.[1] The advantages of such automatic services are clear – much higher capacity compared to

* Corresponding author.
E-mail addresses: tkinnu@cs.joensuu.fi (T. Kinnunen), hli@i2r.a-star.edu.sg (H. Li).
URLs: http://cs.joensuu.fi/sipu/ (T. Kinnunen), http://hlt.i2r.a-star.edu.sg/ (H. Li).

[1] See e.g. http://www.pcworld.com/article/106142/visa_gets_behind_voice_recognition.html.

0167-6393/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.specom.2009.08.009

human-operated services with hundreds or thousands of phone calls being processed simultaneously. In fact, the focus of speaker recognition research over the years has been tending towards such telephony-based applications.

In addition to telephony speech data, there is a continually increasing supply of other *spoken documents* such as TV broadcasts, teleconference meetings, and video clips from vacations. Extracting *metadata* like topic of discussion or participant names and genders from these documents would enable automated information searching and indexing. *Speaker diarization* (Tranter and Reynolds, 2006), also known as "who spoke when", attempts to extract speaking turns of the different participants from a spoken document, and is an extension of the "classical" speaker recognition techniques applied to recordings with multiple speakers.

In forensics and speaker diarization, the speakers can be considered non-cooperative as they do not specifically wish to be recognized. On the other hand, in telephone-based services and access control, the users are considered cooperative. Speaker recognition systems, on the other hand, can be divided into *text-dependent* and *text-independent* ones. In text-dependent systems (Hébert, 2008), suited for cooperative users, the recognition phrases are fixed, or known beforehand. For instance, the user can be prompted to read a randomly selected sequence of numbers as described in (Higgins et al., 1991). In text-independent systems, there are no constraints on the words which the speakers are allowed to use. Thus, the reference (what are spoken in training) and the test (what are uttered in actual use) utterances may have completely different content, and the recognition system must take this phonetic mismatch into account. Text-independent recognition is the much more challenging of the two tasks.

In general, phonetic variability represents one adverse factor to accuracy in text-independent speaker recognition. Changes in the acoustic environment and technical factors (transducer, channel), as well as "within-speaker" variation of the speaker him/herself (state of health, mood, aging) represent other undesirable factors. In general, any variation between two recordings of the same speaker is known as *session variability* (Kenny et al., 2007; Vogt and Sridharan, 2008). Session variability is often described as mismatched training and test conditions, and it remains to be the most challenging problem in speaker recognition.

This paper represents an overview of speaker recognition technologies, including a few representative techniques from 1980s until today. In addition, we give emphasis to the recent techniques that have presented a paradigm shift from the traditional vector-based speaker models to so-called supervector models. This paper serves as a quick overview of the research questions and their solutions for someone who would like to start research in speaker recognition. The paper may also be useful for speech scientists to have a glance at the current trends in the field. We assume familiarity with basics of digital signal processing and pattern recognition.

We recognize that a thorough review of the field with more than 40 years of active research is challenging. For the interested reader we therefore point to other useful surveys. Campbell's tutorial (Campbell, 1997) includes in-depth discussions of feature selection and stochastic modeling. A more recent overview, with useful discussions of normalization methods and speaker recognition applications, can be found in (Bimbot et al., 2004). Recent collection of book chapters on various aspects of speaker classification can also be found in (Müller, 2007a; Müller, 2007b). For an overview of text-*dependent* recognition, refer to (Hébert, 2008).

Section 2 provides fundamentals of speaker recognition. Sections 3 and 4 then elaborate feature extraction and speaker modeling principles. Section 5 describes robust methods to cope with real-life noisy and session mismatched conditions, with the focus on feature and score normalization. Section 6 is then devoted to the current supervector classifiers and their session compensation. In Section 7, we discuss the evaluation of speaker recognition performance and give pointers to software packages as well. Finally, possible future horizons of the field are outlined in Section 8, followed by conclusions in Section 9.

2. Fundamentals

Fig. 1 shows the components of an automatic speaker recognition system. The upper is the enrollment process, while the lower panel illustrates the recognition process. The *feature extraction* module first transforms the raw signal into feature vectors in which speaker-specific properties are emphasized and statistical redundancies suppressed. In the enrollment mode, a *speaker model* is trained using the feature vectors of the target speaker. In the recognition mode, the feature vectors extracted from the unknown person's utterance are compared against the model(s) in the system database to give a similarity score. The decision module uses this similarity score to make the final decision.

Virtually all state-of-the-art speaker recognition systems use a set of *background speakers* or *cohort speakers* in one form or another to enhance the robustness and computational efficiency of the recognizer. In the enrollment phase, background speakers are used as the negative examples in the training of a discriminative model (Campbell et al., 2006a), or in training a *universal background model* from which the target speaker models are adapted (Reynolds et al., 2000). In the recognition phase, background speakers are used in the normalization of the speaker match score (Furui, 1997; Higgins et al., 1991; Li and Porter, 1988; Reynolds, 1995; Reynolds et al., 2000; Sivakumaran et al., 2003b).

2.1. Selection of features

Speech signal includes many features of which not all are important for speaker discrimination. An ideal feature would (Rose, 2002; Wolf, 1972)

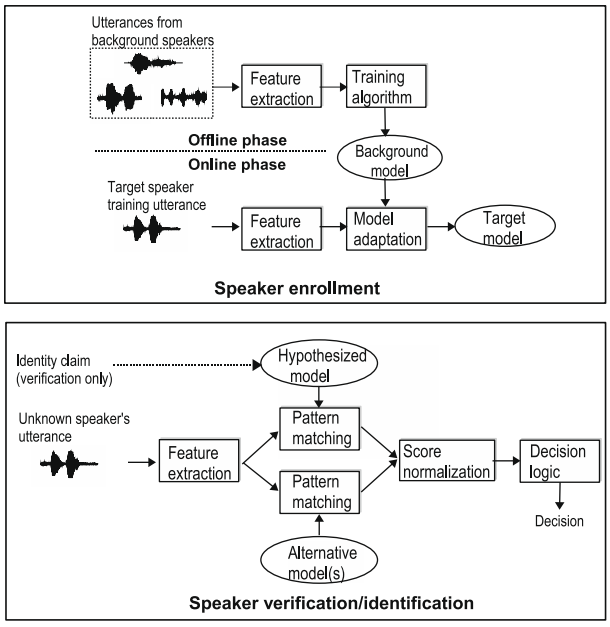

Fig. 1. Components of a typical automatic speaker recognition system. In the enrollment mode, a speaker model is created with the aid of previously created background model; in recognition mode, both the hypothesized model and the background model are matched and background score is used in normalizing the raw score.

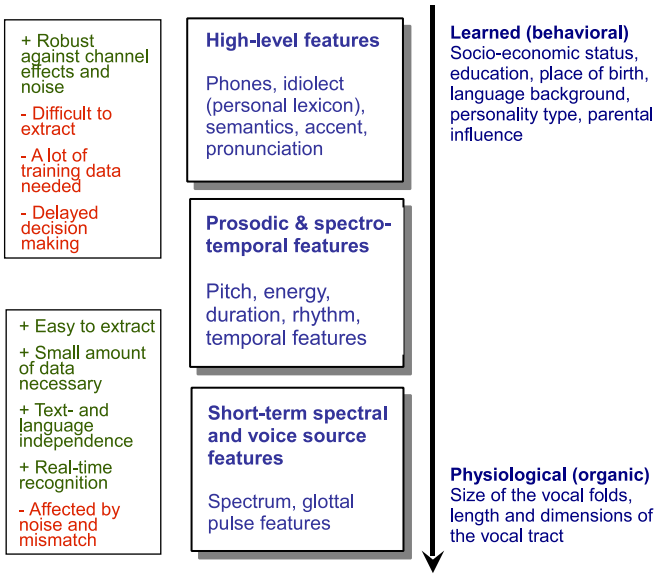

Fig. 2. A summary of features from viewpoint of their physical interpretation. The choice of features has to be based on their discrimination, robustness, and practicality. Short-term spectral features are the simplest, yet most discriminative; prosodics and high-level features have received much attention at high computational cost.

- have large between-speaker variability and small within-speaker variability
- be robust against noise and distortion
- occur frequently and naturally in speech
- be easy to measure from speech signal
- be difficult to impersonate/mimic
- not be affected by the speaker's health or long-term variations in voice.

The number of features should be also relatively low. Traditional statistical models such as the Gaussian mixture model (Reynolds et al., 2000; Reynolds and Rose, 1995) cannot handle high-dimensional data. The number of required training samples for reliable density estimation grows exponentially with the number of features. This problem is known as *the curse of dimensionality* (Jain et al., 2000). The computational savings are also obvious with low-dimensional features.

There are different ways to categorize the features (Fig. 2). From the viewpoint of their physical interpretation, we can divide them into (1) *short-term spectral features*, (2) *voice source features*, (3) *spectro-temporal features*, (4) *prosodic features* and (5) *high-level features*. Short-term spectral features, as the name suggests, are computed from short frames of about 20–30 ms in duration. They are usually descriptors of the short-term *spectral envelope* which is an acoustic correlate of *timbre*, i.e. the "color" of sound, as well as the resonance properties of the supralaryngeal vocal tract. The voice source features, in turn, characterize the voice source (glottal flow). Prosodic and spectro-temporal features span over tens or hundreds of milliseconds, including intonation and rhythm, for instance. Finally, high-level features attempt to capture conversation-level characteristics of speakers, such as characteristic use of words ("uh-huh", "you know", "oh yeah", etc.) (Doddington, 2001).

Which features one should use? It depends on the intended application, computing resources, amount of speech data available (for both development purposes and in run-time) and whether the speakers are cooperative or not. For someone who would like to start research in speaker recognition, we recommend to begin with the short-term spectral features since they are easy to compute and yield good performance (Reynolds et al., 2003). Prosodic and high-level features are believed to be more robust, but less discriminative and easier to impersonate; for instance, it is relatively well known that professional impersonators tend to modify the overall pitch contour towards the imitated speaker (Ashour and Gath, 1999; Kitamura, 2008). High-level features also require considerably more complex front-end, such as automatic speech recognizer. To conclude, there does not yet exist globally "best" feature but the choice is a trade-off between speaker discrimination, robustness, and practicality.

2.2. Speaker modeling

By using feature vectors extracted from a given speaker's training utterance(s), a speaker model is trained and stored into the system database. In text-dependent mode, the model is utterance-specific and it includes the temporal dependencies between the feature vectors. Text-dependent speaker verification and speech recognition do share

similarities in their pattern matching processes, and these can also be combined (BenZeghiba and Bourland, 2003; Heck and Genoud, 2002).

In text-independent mode we often model the feature distribution, i.e. the shape of the "feature cloud" rather than the temporal dependencies. Note that, in text-dependent recognition, we can temporally align the test and training utterances because they contain (are assumed to contain) the same phoneme sequences. However, in text-independent recognition, since there are little or absolutely no correspondence between the frames in the test and reference utterances, alignment at the frame level is not possible. Therefore, *segmentation* of the signal into phones or broad phonetic classes can be used as a pre-processing step, or alternatively, the speaker models can be structured phonetically. Such approaches have been proposed in (Faltlhauser and Ruske, 2001; Hansen et al., 2004; Gupta and Savic, 1992; Hébert and Heck, 2003; Park and Hazen, 2002; Kajarekar and Hermansky, 2001). It is also possible to use data-driven units instead of the strictly linguistic phonemes as segmentation units (Hannani et al., 2004).

Classical speaker models can be divided into *template models* and *stochastic models* (Campbell, 1997), also known as *nonparametric* and *parametric* models, respectively. In template models, training and test feature vectors are directly compared with each other with the assumption that either one is an imperfect replica of the other. The amount of distortion between them represents their degree of similarity. *Vector quantization* (VQ) (Soong et al., 1987) and *dynamic time warping* (DTW) (Furui, 1981) are representative examples of template models for text-independent and text-dependent recognition, respectively.

In stochastic models, each speaker is modeled as a probabilistic source with an unknown but fixed probability density function. The training phase is to estimate the parameters of the probability density function from a training sample. Matching is usually done by evaluating the likelihood of the test utterance with respect to the model. The *Gaussian mixture model* (GMM) (Reynolds and Rose, 1995; Reynolds et al., 2000) and the *hidden Markov model* (HMM) (BenZeghiba and Bourland, 2006; Naik et al., 1989) are the most popular models for text-independent and text-dependent recognition, respectively.

According to the training paradigm, models can also be classified into *generative* and *discriminative* models. The generative models such as GMM and VQ estimate the feature distribution *within* each speaker. The discriminative models such as *artificial neural networks* (ANNs) (Farrell et al., 1994; Heck et al., 2000; Yegnanarayana and Kishore, 2002) and *support vector machines* (SVMs) (Campbell et al., 2006a), in contrast, model the *boundary* between speakers. For more discussions, refer to (Ramachandran et al., 2002).

In summary, a speaker is characterized by a speaker model such as VQ, GMM or SVM. At run-time, a unknown voice is first represented by a collection of feature vectors or a supervector – a concatenation of multiple vectors, then evaluated against the target speaker models.

3. Feature extraction

3.1. Short-term spectral features

The speech signal continuously changes due to articulatory movements, and therefore, the signal must be broken down in short *frames* of about 20–30 ms in duration. Within this interval, the signal is assumed to remain stationary and a spectral feature vector is extracted from each frame.

Usually the frame is *pre-emphasized* and multiplied by a smooth *window function* prior to further steps. Pre-emphasis boosts the higher frequencies whose intensity would be otherwise very low due to downward sloping spectrum caused by glottal voice source (Harrington and Cassidy, 1999, p. 168). The window function (usually Hamming), on the other hand, is needed because of the finite-length effects of the discrete Fourier transform (DFT); for details, refer to (Harris, 1978; Deller et al., 2000; Oppenheim et al., 1999). in practice, choice of the window function is not critical. Although the frame length is usually fixed, *pitch-synchronous analysis* has also been studied (Nakasone et al., 2004; Zilca et al., 2006; Gong et al., 2008). The experiments in (Nakasone et al., 2004; Zilca et al., 2006) indicate that recognition accuracy reduces with this technique, whereas (Gong et al., 2008) obtained some improvement in noisy conditions. Pitch-dependent speaker models have also been studied (Arcienega et al., 2001; Ezzaidi et al., 2001).

The well-known *fast Fourier transform* (FFT), a fast implementation of DFT, decomposes a signal into its frequency components (Oppenheim et al., 1999). Alternatives to FFT-based signal decomposition such as non-harmonic bases, aperiodic functions and data-driven bases derived from independent component analysis (ICA) have been studied in literature (Gopalan et al., 1999; Imperl et al., 1997; Jang et al., 2002). The DFT, however, remains to be used in practice due to its simplicity and efficiency. Usually only the magnitude spectrum is retained, based on the belief that phase has little perceptual importance. However, Paliwal and Alsteris (2003) provides opposing evidence while (Hedge et al., 2004) described a technique which utilizes phase information.

The global shape of the DFT magnitude spectrum (Fig. 3), known as *spectral envelope*, contains information about the resonance properties of the vocal tract and has been found out to be the most informative part of the spectrum in speaker recognition. A simple model of spectral envelope uses a set of bandpass filters to do energy integration over neighboring frequency bands. Motivated by psycho-acoustic studies, the lower frequency range is usually represented with higher resolution by allocating more filters with narrow bandwidths (Harrington and Cassidy, 1999).

Although the subband energy values have been used directly as features (Besacier et al., 2000; Besacier and Bonastre, 2000; Damper and Higgins, 2003; Sivakumaran et al., 2003a), usually the dimensionality is further reduced

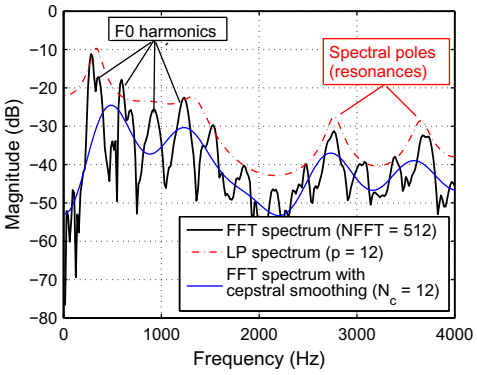

Fig. 3. Extraction of spectral envelope using cepstral analysis and linear prediction (LP). Spectrum of NFFT = 512 points can be effectively reduced to only $N_c = 12$ cepstral coefficients or $p = 12$ LP coefficients. Both the cepstral and LP features are useful and complementary to each other when used in speaker recognition.

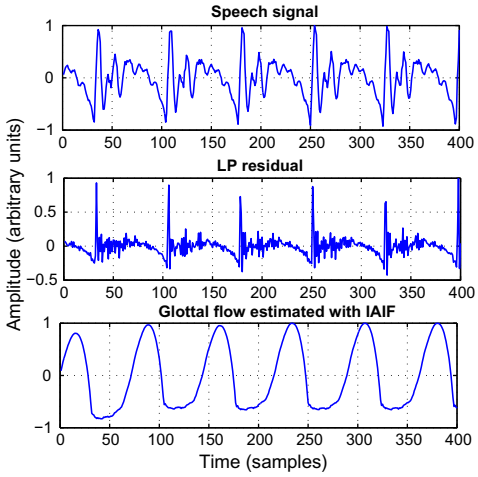

Fig. 4. Glottal feature extraction (Kinnunen and Alku, 2009). Speech frame (top), linear prediction (LP) residual (middle), and glottal flow estimated via inverse filtering (bottom). © 2009 IEEE.

using other transformations. The so-called *mel-frequency cepstral coefficients* (MFCCs) (Davis and Mermelstein, 1980) are popular features in speech and audio processing. MFCCs were introduced in early 1980s for speech recognition and then adopted in speaker recognition. Even though various alternative features, such as spectral subband centroids (SSCs) (Kinnunen et al., 2007; Thian et al., 2004) have been studied, the MFCCs seem to be difficult to beat in practice.

MFCCs are computed with the aid of a psychoacoustically motivated filterbank, followed by logarithmic compression and discrete cosine transform (DCT). Denoting the outputs of an M-channel filterbank as $Y(m), m = 1, \ldots, M$, the MFCCs are obtained as follows:

$$c_n = \sum_{m=1}^{M} [\log Y(m)] \cos\left[\frac{\pi n}{M}\left(m - \frac{1}{2}\right)\right]. \quad (1)$$

Here n is the index of the cepstral coefficient. The final MFCC vector is obtained by retaining about 12–15 lowest DCT coefficients. More details of MFCCs can be found in (Deller et al., 2000; Huang et al., 2001). Alternative features that emphasize speaker-specific information have been studied in (Charbuillet et al., 2006; Miyajima et al., 2001; Kinnunen, 2002; Orman and Arslan, 2001). For study of speaker-discriminative information in spectrum, refer to (Lu and Dang, 2007). Finally, some new trends in feature extraction can be found in (Ambikairajah, 2007).

Linear prediction (LP) (Makhoul, 1975; Mammone et al., 1996) is an alternative spectrum estimation method to DFT that has good intuitive interpretation both in time domain (adjacent samples are correlated) and frequency domain (all-pole spectrum corresponding to the resonance structure). In time domain, LP predictor equation is defined as,

$$\tilde{s}[n] = \sum_{k=1}^{p} a_k s[n-k]. \quad (2)$$

Here $s[n]$ is the observed signal, a_k are the *predictor coefficients* and $\tilde{s}[n]$ is the predicted signal. The prediction error signal, or *residual*, is defined as $e[n] = s[n] - \tilde{s}[n]$, and illustrated in the middle panel of Fig. 4. The coefficients a_k are usually determined by minimizing the residual energy using the so-called *Levinson–Durbin* algorithm (Harrington and Cassidy, 1999; Huang et al., 2001; Rabiner and Juang, 1993). The spectral model is defined as,

$$H(z) = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}}, \quad (3)$$

and it consists of spectral peaks or *poles* only (dash-dotted line in Fig. 3).

The predictor coefficients $\{a_k\}$ themselves are rarely used as features but they are transformed into robust and less correlated features such as *linear predictive cepstral coefficients* (LPCCs) (Huang et al., 2001), *line spectral frequencies* (LSFs) (Huang et al., 2001), and *perceptual linear prediction* (PLP) coefficients (Hermansky, 1990). Other, somewhat less successful features, include *partial correlation coefficients* (PARCORs), *log area ratios* (LARs) and *formant frequencies and bandwidths* (Rabiner and Juang, 1993).

Given all the alternative spectral features, which one should be used for speaker recognition and how should the parameters (e.g. the number of coefficients) be selected? Some comparisons can be found in (Atal, 1974; Kinnunen, 2004; Kinnunen et al., 2004; Reynolds and Rose, 1995), and it has been observed that in general channel compensation methods are much more important than the choice of the base feature set (Reynolds and Rose, 1995). Different spectral features, however, are complementary and can be combined to enhance accuracy (Brümmer et al., 2007; Campbell et al., 2006a; Kinnunen et al., 2004). In summary, for practical use we recommend any of the following features: MFCC, LPCC, LSF, PLP.

3.2. Voice source features

Voice source features characterize the glottal excitation signal of voiced sounds such as glottal pulse shape and fundamental frequency, and it is reasonable to assume that they carry speaker-specific information. Fundamental frequency, the rate of vocal fold vibration, is popular and will be discussed in Section 3.4. Other parameters are related to the shape of the glottal pulse, such as the degree of vocal fold opening and the duration of the closing phase. These contribute to voice quality which can be described for example, as modal, breathy, creaky or pressed (Espy-Wilson et al., 2006).

The glottal features are not directly measurable due to the vocal tract filtering effect. By assuming that the glottal source and the vocal tract are independent of each other, vocal tract parameters can be first estimated using, for instance, the linear prediction model, followed by *inverse filtering* of the original waveform to obtain an estimate of the source signal (Kinnunen and Alku, 2009; Murty and Yegnanarayana, 2006; Plumpe et al., 1999; Prasanna et al., 2006; Thévenaz and Hügli, 1995; Zheng et al., 2007). An alternative method uses *closed-phase covariance analysis* during the portions when the vocal folds are closed (Gudnason and Brookes, 2008; Plumpe et al., 1999; Slyh et al., 2004). This leads to improved estimate of the vocal tract but accurate detection of closed phase is required which is difficult in noisy conditions. As an example, Fig. 4 shows a speech signal together with its LP residual and glottal flow estimated with a simple inverse filtering method (Alku et al., 1999).

Features of the inverse filtered signal can be extracted, for instance, by using an auto-associative neural network (Prasanna et al., 2006). Other approaches have used parametric glottal flow model parameters (Plumpe et al., 1999), wavelet analysis (Zheng et al., 2007), residual phase (Murty and Yegnanarayana, 2006), cepstral coefficients (Gudnason and Brookes, 2008; Chetouani et al., 2009; Kinnunen and Alku, 2009) and higher-order statistics (Chetouani et al., 2009) to mention a few.

Based on the literature, voice source features are not as discriminative as vocal tract features but fusing these two complementary features can improve accuracy (Murty and Yegnanarayana, 2006; Zheng et al., 2007). Experiments of (Chan et al., 2007; Prasanna et al., 2006) also suggest that the amount of training and testing data for the voice source features can be significantly less compared to the amount of data needed for the vocal tract features (10 s vs 40 s in (Prasanna et al., 2006)). A possible explanation for this is that vocal tract features depend on the phonetic content and thus require sufficient phonetic coverage for both the training and test utterances. Voice source features, in turn, depend much less on phonetic factors.

3.3. Spectro-temporal features

It is reasonable to assume that the spectro-temporal signal details such as formant transitions and energy modulations contain useful speaker-specific information. A common way to incorporate some temporal information to features is through first- and second-order time derivative estimates, known as *delta* (Δ) and *double-delta* (Δ^2) coefficients, respectively (Furui, 1981; Huang et al., 2001; Soong and Rosenberg, 1988). They are computed as the time differences between the adjacent vectors feature coefficients and usually appended with the base coefficients on the frame level (e.g. 13 MFCCs with Δ and Δ^2 coefficients, implying 39 features per frame). An alternative, potentially more robust, method fits a regression line (Rabiner and Juang, 1993) or an orthogonal polynomial (Furui, 1981) to the temporal trajectories, although in practice simple differentiation seems to yield equal or better performance (Kinnunen, 2004). *Time–frequency principal components* (Magrin-Chagnolleau et al., 2002) and *data-driven temporal filters* (Malayath et al., 2000) have also been studied.

In (Kinnunen, 2006; Kinnunen et al., 2008), we proposed to use *modulation frequency* (Atlas and Shamma, 2003; Hermansky, 1998) as a feature for speaker recognition as illustrated in Fig. 5. Modulation frequency represents the frequency content of the subband amplitude envelopes and it potentially contains information about speaking rate and other stylistic attributes. Modulation frequencies relevant for speech intelligibility are approximately in the range 1–20 Hz (Atlas and Shamma, 2003; Hermansky, 1998). In (Kinnunen, 2006), the best recognition result was obtained by using a temporal window of 300 ms and by including modulation frequencies in the range 0–20 Hz. The dimensionality of the modulation frequency vector depends on the number of FFT points of the spectrogram and the number of frames spanning the FFT computation in the temporal direction. For the best parameter combination, the dimension of the feature vector was 3200 (Kinnunen, 2006).

In (Kinnunen et al., 2006c; Kinnunen et al., 2008) we studied reduced-dimensional spectro-temporal features. The *temporal discrete cosine transform* (TDCT) method, proposed in (Kinnunen et al., 2006c) and illustrated in Fig. 6, applies DCT on the temporal trajectories of the cepstral vectors rather than on the spectrogram magnitudes. Using DCT rather than DFT magnitude here has an advantage that it retains the relative phases of the feature coefficient trajectories, and hence, it can preserve both phonetic and speaker-specific information. This, however, requires more research. In (Kinnunen et al., 2008), DCT was used in a different role: reducing the dimensionality of the modulation magnitude spectra. The best results in (Kinnunen, 2006; Kinnunen et al., 2008) were obtained by using a time context of 300–330 ms, which is significantly longer compared with the typical time contexts of the delta features.

Even though we obtained some improvement over the cepstral systems by fusing the match scores of the cepstral and temporal features (Kinnunen, 2006; Kinnunen

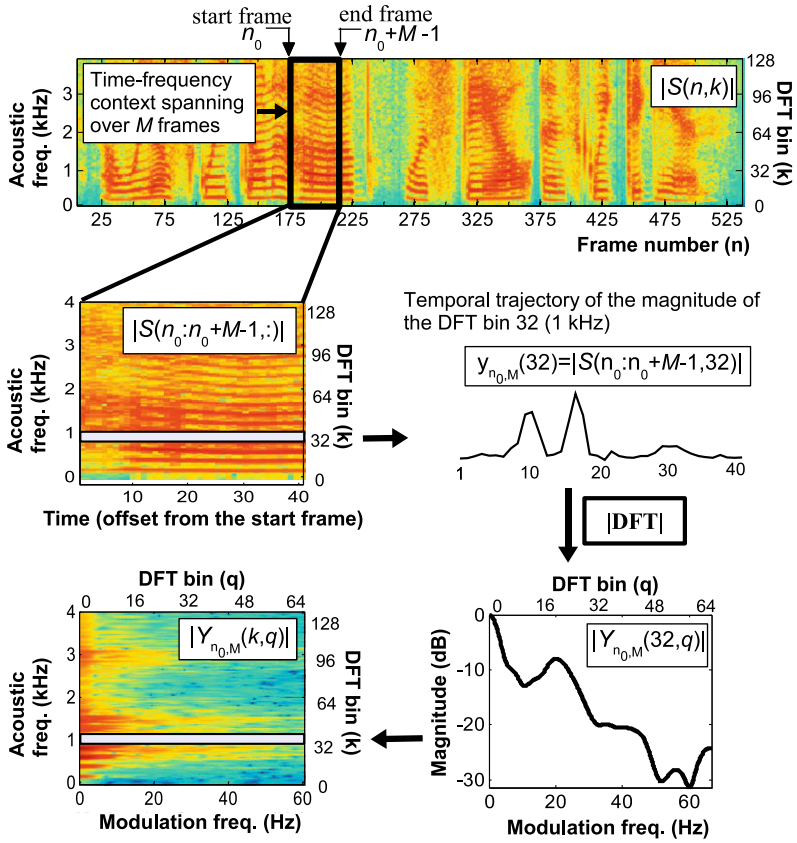

Fig. 5. Extracting modulation spectrogram features (Kinnunen et al., 2008). A time–frequency context, including M short-term spectra over the interval $[n_0 \cdots n_0 + M - 1]$, is first extracted. The DFT magnitude spectra of all feature trajectories are then computed and stacked as a feature vector with high dimensionality (here $129 \times 65 = 8385$ elements).

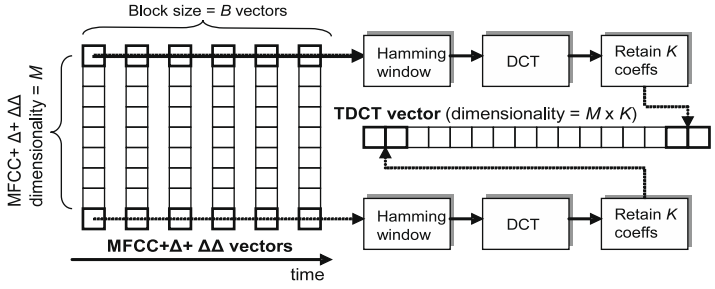

Fig. 6. Temporal discrete cosine transform (TDCT) (Kinnunen et al., 2006c). Short-term MFCC features with their delta features are taken as input; their low-frequency modulation characteristics are then represented by computing discrete cosine transform (DCT) over a context of B frames. The lowest DCT coefficients are retained as features.

et al., 2006c), the gain was rather modest and more research is required before these features can be recommended for practical applications. One problem could be that we have applied speaker modeling techniques that are designed for short-term features. Due to larger temporal context, the number of training vectors is usually less compared with short-term features. Furthermore, as the short-term and longer-term features have different frame rates, they cannot be easily combined at the frame level. Perhaps a completely different modeling and fusion technique is required for these features.

An alternative to amplitude-based methods considers *frequency modulations* (FM) instead (Thiruvaran et al., 2008a). In FM-based methods, the input signal is first divided into subband signals using a bank of bandpass filters. The dominant frequency components (such as the frequency centroids) in the subbands then capture formant-like features. As an example, the procedure described in (Thiruvaran et al., 2008a) uses second-order all-pole analysis to detect the dominant frequency. The FM features are then obtained by subtracting the center frequency of the subband from the pole frequency, yielding

a measure of deviation from the "default" frequency of the bandpass signal. This feature was applied to speaker recognition in (Thiruvaran et al., 2008b), showing promise when fused with conventional MFCCs.

3.4. Prosodic features

Prosody refers to non-segmental aspects of speech, including for instance syllable stress, intonation patterns, speaking rate and rhythm. One important aspect of prosody is that, unlike the traditional short-term spectral features, it spans over long segments like syllables, words, and utterances and reflects differences in speaking style, language background, sentence type, and emotions to mention a few. A challenge in text-independent speaker recognition is modeling the different levels of prosodic information (instantaneous, long-term) to capture speaker differences; at the same time, the features should be free of effects that the speaker can voluntarily control.

The most important prosodic parameter is the fundamental frequency (or F0). Combining F0-related features with spectral features has been shown to be effective, especially in noisy conditions. Other prosodic features for speaker recognition have included duration (e.g. pause statistics, phone duration), speaking rate, and energy distribution/modulations among others (Adami et al., 2003; Bartkova et al., 2002; Reynolds et al., 2003; Shriberg et al., 2005). Interested reader may refer to (Shriberg et al., 2005) for further details. In that study, it was found out, among a number of other observations, that F0-related features yielded the best accuracy, followed by energy and duration features in this order. Since F0 is the predominant prosodic feature, we will now discuss it in more detail.

Reliable F0 determination itself is a challenging task. For instance, in telephone quality speech, F0 is often outside of the narrowband telephone network passband (0.3–3.4 kHz) and the algorithms can only rely on the information in the upper harmonics for F0 detection. For a detailed discussion of classical F0 estimation approaches, refer to (Hess, 1983). More recent comparison of F0 trackers can be found in (Cheveigné and Kawahara, 2001). For practical use, we recommend the YIN method (DeCheveigne and Kawahara, 2002) and the auto-correlation method as implemented in Praat software (Boersma and Weenink, 2009).

For speaker recognition, F0 conveys both physiological and learned characteristics. For instance, the mean value of F0 can be considered as an acoustic correlate of the larynx size (Rose, 2002), whereas the temporal variations of pitch are related to the manner of speaking. In text-dependent recognition, temporal alignment of pitch contours have been used (Atal, 1972). In text-independent studies, long-term F0 statistics – especially the mean value – have been extensively studied (Carey et al., 1996; Kinnunen and González-Hautamäki, 2005; Markel et al., 1977; Nolan, 1983; Sönmez et al., 1998; Sönmez et al., 1997). The mean value combined with other statistics such as variance and kurtosis can be used as speaker model (Bartkova et al., 2002; Carey et al., 1996; Kinnunen and González-Hautamäki, 2005), even though histograms (Kinnunen and González-Hautamäki, 2005), latent semantic analysis (Chen et al., 2004) and support vector machines (Shriberg et al., 2005) perform better. It has also been found through a number of experiments that $\log(F0)$ is a better feature than F0 itself (Kinnunen and González-Hautamäki, 2005; Sönmez et al., 1997).

F0 is a one-dimensional feature, therefore mathematically, not expected to be very discriminative. Multidimensional pitch- and voicing-related features can be extracted from the auto-correlation function without actual F0 extraction as done in (Laskowski and Jin, 2009; Ma et al., 2006a; Wildermoth and Paliwal, 2000) for example. Another way to improve accuracy is modeling both the local and long-term temporal variations of F0.

Capturing local F0 dynamics can be achieved by appending the delta features with the instantaneous F0 value. For longer-term modeling, F0 contour can be segmented and presented by a set of parameters associated with each segment (Adami, 2007; Adami et al., 2003; Mary and Yegnanarayana, 2006; Shriberg et al., 2005; Sönmez et al., 1998). The segments may be syllables obtained using automatic speech recognition (ASR) system (Shriberg et al., 2005). An alternative, ASR-free approach, is to divide the utterance into syllable-like units using, for instance, vowel onsets (Mary and Yegnanarayana, 2008) or F0/energy inflection points (Adami, 2007; Dehak et al., 2007) as the segment boundaries.

For parameterization of the segments, prosodic feature statistics and their local temporal slopes (tilt) within each segment are often used. In (Adami et al., 2003; Sönmez et al., 1998), each voiced segment was parameterized by a piece-wise linear model whose parameters formed the features. In (Shriberg et al., 2005), the authors used N-gram counts of discretized feature values as features to an SVM classifier with promising results. In (Dehak et al., 2007), prosodic features were extracted using polynomial basis functions.

3.5. High-level features

Speakers differ not only in their voice timbre and accent/pronunciation, but also in their lexicon – the kind of words the speakers tend to use in their conversations. The work on such "high-level" conversational features was initiated in (Doddington, 2001) where a speaker's characteristic vocabulary, the so-called *idiolect*, was used to characterize speakers. The idea in "high-level" modeling is to convert each utterance into a sequence of *tokens* where the co-occurrence patterns of tokens characterize speaker differences. The information being modeled is hence in categorical (discrete) rather than in numeric (continuous) form.

The tokens considered have included words (Doddington, 2001), phones (Andrews et al., 2002; Campbell et al.,

2004), prosodic gestures (rising/falling pitch/energy) (Adami et al., 2003; Chen et al., 2004; Shriberg et al., 2005), and even articulatory tokens (manner and place of articulation) (Leung et al., 2006). The top-1 scoring Gaussian mixture component indices have also been used as tokens (Ma et al., 2006b; Torres-Carrasquillo et al., 2002; Xiang, 2003).

Sometimes several parallel tokenizers are utilized (Campbell et al., 2004; Jin et al., 2002; Ma et al., 2006b). This is partly motivated by the success of parallel phone recognizers in state-of-the-art spoken language recognition (Zissman, 1996; Ma et al., 2007). This direction is driven by the hope that different tokenizers (e.g. phone recognizers trained on different languages or with different phone models) would capture complementary aspects of the utterance. As an example, in (Ma et al., 2006b) a set of parallel GMM tokenizers (Torres-Carrasquillo et al., 2002; Xiang, 2003) were used. Each tokenizer was trained from a different group of speakers obtained by clustering.

The baseline classifier for token features is based on *N-gram* modeling. Let us denote the token sequence of the utterance by $\{\alpha_1, \alpha_2, \ldots, \alpha_T\}$, where $\alpha_t \in V$ and V is a finite vocabulary. An N-gram model is constructed by estimating the joint probability of N consecutive tokens. For instance, $N = 2$ gives the *bigram* model where the probabilities of token pairs (α_t, α_{t+1}) are estimated. A *trigram* model consists of triplets $(\alpha_t, \alpha_{t+1}, \alpha_{t+2})$, and so forth. As an example, the bigrams of the token sequence `hello_world` are (`h`,`e`), (`e`,`l`), (`l`,`l`), (`l`,`o`), (`o`,`_`), (`_`,`w`), (`w`,`o`), (`o`,`r`), (`r`,`l`) and (`l`,`d`).

The probability of each N-gram is estimated in the same way as N-gram in statistical language models in automatic speech recognition (Ney et al., 1997). It is the *maximum likelihood* (ML) or *maximum a posteriori* (MAP) estimate of the N-gram in the training corpus (Leung et al., 2006). The N-gram statistics have been used in vector space (Campbell et al., 2004; Ma et al., 2006b) and with entropy measures (Andrews et al., 2001; Leung et al., 2006) to assess similarity between speakers.

4. Speaker modeling: classical approaches

This section describes some of the popular models in text-independent speaker recognition. The models presented here have co-evolved with the short-term spectral features such as MFCCs in the literature.

4.1. Vector quantization

Vector quantization (VQ) model (Burton, 1987; Hautamäki et al., 2008b; He et al., 1999; Karpov et al., 2004; Kinnunen et al., 2006b; Soong et al., 1987; Soong and Rosenberg, 1988), also known as *centroid model*, is one of the simplest text-independent speaker models. It was introduced to speaker recognition in the 1980s (Burton, 1987; Soong et al., 1987) and its roots are originally in data compression (Gersho and Gray, 1991). Even though VQ is often used for computational speed-up techniques (Louradour and Daoudi, 2005; Kinnunen et al., 2006b; Roch, 2006) and lightweight practical implementations (Saastamoinen et al., 2005), it also provides competitive accuracy when combined with background model adaptation (Hautamäki et al., 2008b; Kinnunen et al., 2009). We will return to adaptation methods in Section 4.2.

In the following, we denote the test utterance feature vectors by $\mathscr{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T\}$ and the reference vectors by $\mathscr{R} = \{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_K\}$. The *average quantization distortion* is defined as,

$$D_Q(\mathscr{X}, \mathscr{R}) = \frac{1}{T}\sum_{t=1}^{T} \min_{1 \leqslant k \leqslant K} d(\boldsymbol{x}_t, \boldsymbol{r}_k), \tag{4}$$

where $d(\cdot,\cdot)$ is a distance measure such as the Euclidean distance $\|\boldsymbol{x}_t - \boldsymbol{r}_k\|$. A smaller value of (4) indicates higher likelihood for $\mathscr{X}$ and $\mathscr{R}$ originating from the same speaker. Note that (4) is not symmetric (Karpov et al., 2004): $D_Q(\mathscr{X}, \mathscr{R}) \neq D_Q(\mathscr{R}, \mathscr{X})$.

In theory, it is possible to use all the training vectors directly as the reference template $\mathscr{R}$. For computational reasons, however, the number of vectors is usually reduced by a clustering method such as K-means (Linde et al., 1980). This gives a reduced set of vectors known as *codebook* (Fig. 7). The choice of the clustering method is not as important as optimizing the codebook size (Kinnunen et al., 2000).

4.2. Gaussian mixture model

Gaussian mixture model (GMM) (Reynolds and Rose, 1995; Reynolds et al., 2000) is a stochastic model which has become the *de facto* reference method in speaker recognition. The GMM can be considered as an extension of the VQ model, in which the clusters are overlapping. That is, a feature vector is not assigned to the nearest cluster as in (4), but it has a nonzero probability of originating from each cluster.

A GMM is composed of a finite mixture of multivariate Gaussian components. A GMM, denoted by λ, is characterized by its probability density function:

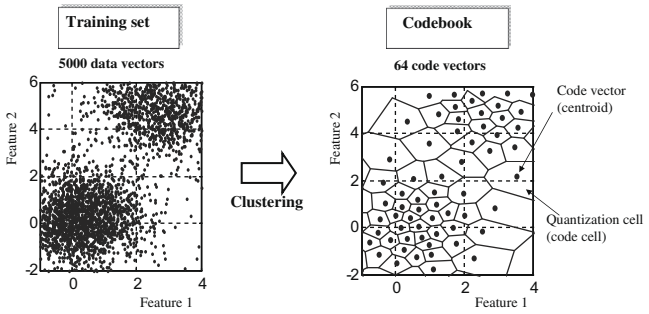

Fig. 7. Codebook construction for vector quantization using the K-means algorithm. The original training set consisting of 5000 vectors is reduced to a set of $K = 64$ code vectors (centroids).

$$p(\boldsymbol{x}|\lambda) = \sum_{k=1}^{K} P_k \, \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k). \quad (5)$$

In (5), K is the number of Gaussian components, P_k is the prior probability (mixing weight) of the kth Gaussian component, and

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = (2\pi)^{-\frac{d}{2}} |\boldsymbol{\Sigma}_k|^{-\frac{1}{2}} \exp\left\{ -\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_k) \right\} \quad (6)$$

is the d-variate Gaussian density function with mean vector $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$. The prior probabilities $P_k \geqslant 0$ are constrained as $\sum_{k=1}^{K} P_k = 1$.

For numerical and computational reasons, the covariance matrices of the GMM are usually diagonal (i.e. variance vectors), which restricts the principal axes of the Gaussian ellipses in the direction of the coordinate axes. Estimating the parameters of a full-covariance GMM requires, in general, much more training data and is computationally expensive. As an example for estimating the parameters of a full-covariance GMM, refer to (Yuo and Wang, 1999).

Monogaussian model uses a single Gaussian component with a full covariance matrix as the speaker model (Besacier and Bonastre, 2000; Besacier et al., 2000; Bimbot et al., 1995; Campbell, 1997; Zilca, 2002). Sometimes only the covariance matrix is used because the cepstral mean vector is affected by convolutive noise (e.g. due to the microphone/handset). The monogaussian and covariance-only models have a small number of parameters and are therefore computationally efficient, although their accuracy is clearly behind GMM.

Training a GMM consists of estimating the parameters $\lambda = \{P_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{K}$ from a training sample $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T\}$. The basic approach is *maximum likelihood* (ML) estimation. The average log-likelihood of $\mathcal{X}$ with respect to model λ is defined as,

$$\mathrm{LL}_{\mathrm{avg}}(\mathcal{X}, \lambda) = \frac{1}{T} \sum_{t=1}^{T} \log \sum_{k=1}^{K} P_k \mathcal{N}(\boldsymbol{x}_t|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k). \quad (7)$$

The higher the value, the higher the indication that the unknown vectors originate from the model λ. The popular *expectation–maximization* (EM) algorithm (Bishop, 2006) can be used for maximizing the likelihood with respect to a given data. Note that K-means (Linde et al., 1980) can be used as an initialization method for EM algorithm; a small number or even no EM iterations are needed according to (Kinnunen et al., 2009; Kolano and Regel-Brietzmann, 1999; Pelecanos et al., 2000). This is by no means a general rule, but the iteration count should be optimized for a given task.

In speech applications, *adaptation* of the acoustic models to new operating conditions is important because of data variability due to different speakers, environments, speaking styles and so on. In GMM-based speaker recognition, a speaker-independent *world model* or *universal background model* (UBM) is first trained with the EM algorithm from tens or hundreds of hours of speech data gathered from a large number of speakers (Reynolds et al., 2000). The background model represents speaker-independent distribution of the feature vectors. When enrolling a new speaker to the system, the parameters of the background model are adapted to the feature distribution of the new speaker. The adapted model is then used as the model of that speaker. In this way, the model parameters are not estimated from scratch, with prior knowledge ("speech data in general") being utilized instead. Practice has shown that it is advantageous to train two separate background models, one for female and the other one for male speakers. The new speaker model is then adapted from the background model of the same gender as the new speaker. Let us now look how the adaptation is carried out.

As indicated in Fig. 8, it is possible to adapt all the parameters, or only some of them from the background model. Adapting the means only has been found to work well in practice (Reynolds et al., 2000) - this also motivates for a simplified adapted VQ model (Hautamäki et al., 2008b; Kinnunen et al., 2009). Given the enrollment sample, $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T\}$, and the UBM, $\lambda_{\mathrm{UBM}} = \{P_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{K}$, the adapted mean vectors $(\boldsymbol{\mu}_k')$ in the *maximum a posteriori* (MAP) method (Reynolds et al., 2000) are obtained as weighted sums of the speaker's training data and the UBM mean:

$$\boldsymbol{\mu}_k' = \alpha_k \tilde{\boldsymbol{x}}_k + (1 - \alpha_k)\boldsymbol{\mu}_k, \quad (8)$$

where

$$\alpha_k = \frac{n_k}{n_k + r}, \quad (9)$$

$$\tilde{\boldsymbol{x}}_k = \frac{1}{n_k} \sum_{t=1}^{T} P(k|\boldsymbol{x}_t)\boldsymbol{x}_t, \quad (10)$$

$$n_k = \sum_{t=1}^{T} P(k|\boldsymbol{x}_t), \quad (11)$$

$$P(k|\boldsymbol{x}_t) = \frac{P_k \mathcal{N}(\boldsymbol{x}_t|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{m=1}^{K} P_m \mathcal{N}(\boldsymbol{x}_t|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)}. \quad (12)$$

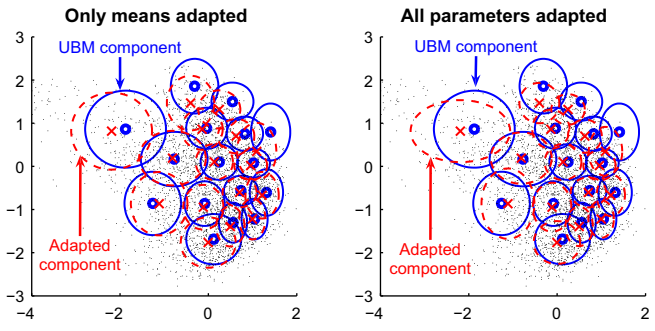

Fig. 8. Examples of GMM adaptation using *maximum a posteriori* (MAP) principle. The Gaussian components of a universal background model (solid ellipses) are adapted to the target speaker's training data (dots) to create speaker model (dashed ellipses).

The MAP adaptation is to derive a speaker-specific GMM from the UBM. The relevance parameter r, and thus α_k, controls the effect of the training samples on the resulting model with respect to the UBM.

In the recognition mode, the MAP-adapted model and the UBM are coupled, and the recognizer is commonly referred to as *Gaussian mixture model – universal background model*, or simply "GMM–UBM". The match score depends on both the target model (λ_{target}) and the background model (λ_{UBM}) via the average log likelihood ratio:

$$\text{LLR}_{\text{avg}}(\mathcal{X}, \lambda_{\text{target}}, \lambda_{\text{UBM}}) = \frac{1}{T}\sum_{t=1}^{T}\{\log p(\boldsymbol{x}_t|\lambda_{\text{target}}) - \log p(\boldsymbol{x}_t|\lambda_{\text{UBM}})\}, \quad (13)$$

which essentially measures the *difference* of the target and background models in generating the observations $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T\}$. The use of a common background model for all speakers makes the match score ranges of different speakers comparable. It is common to apply test segment dependent normalization (Auckenthaler et al., 2000) on top of UBM normalization to account for test-dependent score offsets.

There are alternative adaptation methods to MAP, and selection of the method depends on the amount of available training data (Mak et al., 2006; Mariéthoz and Bengio, 2002). For very short enrollment utterances (a few seconds), some other methods have shown to be more effective. *Maximum likelihood linear regression* (MLLR) (Leggetter and Woodland, 1995), originally developed for speech recognition, has been successfully applied to speaker recognition (Karam and Campbell, 2007; Mak et al., 2006; Mariéthoz and Bengio, 2002; Stolcke et al., 2007). Both the MAP and MLLR adaptations form a basis for the recent supervector classifiers that we will cover in Section 6.

Gaussian mixture model is computationally intensive due the frame-by-frame matching. In the GMM–UBM framework (Reynolds et al., 2000), the score (13) can be evaluated fast by finding for each test utterance vector the top-C (where usually $C \approx 5$) scoring Gaussians from the UBM (Reynolds et al., 2000; Saeidi et al., 2009; Tydlitat et al., 2007). Other speed-up techniques include reducing the numbers of vectors, Gaussian component evaluations, or speaker models (Auckenthaler and Mason, 2001; Kinnunen et al., 2006b; Louradour et al., 2005; McLaughlin et al., 1999; Pellom and Hansen, 1998; Roch, 2006; Saeidi et al., 2009; Xiang and Berger, 2003; Xiong et al., 2006).

Unlike the hidden Markov models (HMM) in speech recognition, GMM does not explicitly utilize any phonetic information – the training set for GMM simply contains all the spectral features of different phonetic classes pooled together. Because the features of the test utterance and the Gaussian components are not phonetically aligned, the match score may be biased due to different phonemes in training and test utterances.

This phonetic mismatch problem has been attacked with phonetically-motivated tree structures (Chaudhari et al., 2003; Hébert and Heck, 2003) and by using a separate GMM for each phonetic class (Castaldo et al., 2007a; Faltlhauser and Ruske, 2001; Hansen et al., 2004; Park and Hazen, 2002) or for parts of syllables (Bocklet and Shriberg, 2009). As an example, *phonetic GMM* (PGMM) described in (Castaldo et al., 2007a) used neural network classifier for 11 language independent broad phone classes. In the training phase, a separate GMM was trained for each phonetic class and in run-time the GMM corresponding to the frame label was used in scoring. Promising results were obtained when combining PGMM with feature-level intersession combination and with conventional (non-phonetic) GMM. Phonetic modeling in GMMs is clearly worth further studying.

4.3. Support vector machine

Support vector machine (SVM) is a powerful discriminative classifier that has been recently adopted in speaker recognition. It has been applied both with spectral (Campbell et al., 2006a; Campbell et al., 2006b), prosodic (Shriberg et al., 2005; Ferrer et al., 2007), and high-level features (Campbell et al., 2004). Currently SVM is one of the most robust classifiers in speaker verification, and it has also been successfully combined with GMM to increase accuracy (Campbell et al., 2006a; Campbell et al., 2006b). One reason for the popularity of SVM is its good generalization performance to classify unseen data.

The SVM, as illustrated in Fig. 9, is a *binary* classifier which models the decision boundary between two classes as a *separating hyperplane*. In speaker verification, one class consists of the target speaker training vectors (labeled as +1), and the other class consists of the training vectors from an "impostor" (background) population (labeled as −1). Using the labeled training vectors, SVM optimizer finds a separating hyperplane that maximizes the *margin* of separation between these two classes.

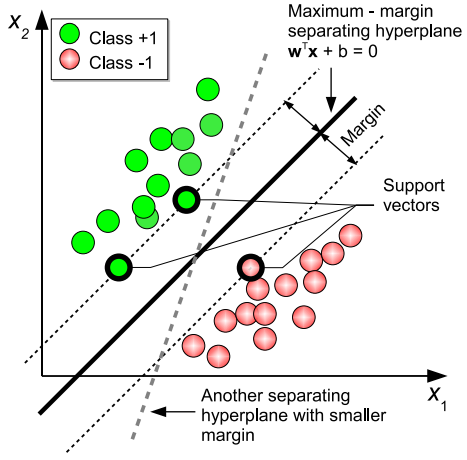

Fig. 9. Principle of support vector machine (SVM). A maximum-margin hyperplane that separates the positive (+1) and negative (−1) training examples is found by an optimization process. SVMs have excellent generalization performance.

Formally, the discriminant function of SVM is given by (Campbell et al., 2006a),

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i t_i K(\boldsymbol{x}, \boldsymbol{x}_i) + d. \quad (14)$$

Here $t_i \in \{+1, -1\}$ are the ideal output values, $\sum_{i=1}^{N} \alpha_i t_i = 0$ and $\alpha_i > 0$. The *support vectors* $\boldsymbol{x}_i$, their corresponding weights α_i and the bias term d, are determined from a training set using an optimization process. The *kernel function* $K(\cdot,\cdot)$ is designed so that it can be expressed as $K(\boldsymbol{x},\boldsymbol{y}) = \phi(\boldsymbol{x})^T \phi(\boldsymbol{y})$, where $\phi(\boldsymbol{x})$ is a mapping from the input space to kernel feature space of high dimensionality. The kernel function allows computing inner products of two vectors in the kernel feature space. In a high-dimensional space, the two classes are easier to separate with a hyperplane. Intuitively, linear hyperplane in the high-dimensional kernel feature space corresponds to a nonlinear decision boundary in the original input space (e.g. the MFCC space). For more information about SVM and kernels, refer to (Bishop, 2006; Müller et al., 2001).

4.4. Other models

Artificial neural networks (ANNs) have been used in various pattern classification problems, including speaker recognition (Farrell et al., 1994; Heck et al., 2000; Lapidot et al., 2002; Yegnanarayana and Kishore, 2002). A potential advantage of ANNs is that feature extraction and speaker modeling can be combined into a single network, enabling joint optimization of the (speaker-dependent) feature extractor and the speaker model (Heck et al., 2000). They are also handy in fusing different subsystems (Reynolds et al., 2003; Tong et al., 2006).

Speaker-specific mapping has been proposed in (Malayath et al., 2000; Misra et al., 2003). The idea is to extract two parallel feature streams with the same frame rate: a feature set representing purely phonetic information (speech content), and a feature set representing a mixture of phonetic and speaker-specific information. The speaker modeling is thus essentially to find a mapping from the "phonetic" spectrum to the "speaker-specific" spectrum by using subspace method (Malayath et al., 2000) or neural network (Misra et al., 2003).

Representing a speaker relative to other speakers is proposed in (Mami and Charlet, 2006; Sturim et al., 2001). Each speaker model is presented as a combination of some reference models known as the *anchor* models. The combination weights – coordinates in the anchor model space – compose the speaker model. The similarity score between the unknown speech sample and a target model is determined as the distance between their coordinate vectors.

4.5. Fusion

Like in other pattern classification tasks, combining information from multiple sources of evidence – a technique called *fusion* – has been widely applied in speaker recognition (Altincay and Demirekler, 2003; Hannani et al., 2004; Chen et al., 1997; Damper and Higgins, 2003; Farrell et al., 1998; Fredouille et al., 2000; Kinnunen et al., 2004; Mak et al., 2003; Moonasar and Venayagamoorthy, 2001; Ramachandran et al., 2002; Rodríguez-Liñares et al., 2003; Slomka et al., 1998). Typically, a number of different feature sets are first extracted from the speech signal; then an individual classifier is used for each feature set; following that the sub-scores or decisions are combined. This implies that each speaker has multiple speaker models stored in the database.

It is also possible to obtain fusion through modeling the *same* features using different classifier architectures, feature normalizations, or training sets (Brümmer et al., 2007; Farrell et al., 1998; Kinnunen et al., 2009; Moonasar and Venayagamoorthy, 2001). A general belief is that successful fusion system should combine as independent features as possible – low-level spectral features, prosodic features and high-level features. But improvement can also be obtained by fusion of different low-level spectral features (e.g. MFCCs and LPCCs) and different classifiers for them (Brümmer et al., 2007; Campbell et al., 2006a; Kinnunen et al., 2004). Fusing dependent (correlated) classifiers can enhance the robustness of the score due to variance reduction (Poh and Bengio, 2004).

Simplest form of fusion is combining the classifier output scores by weighted sum. That is, given the sub-scores s_k, where k indices the classifier, the fused match score is $s = \sum_{n=1}^{N_c} w_n s_n$. Here N_c is the number of classifiers and w_n is the relative contribution of the nth classifier. The fusion weights w_n can be optimized using a development set, or they can be set as equal ($w_n = 1/N_c$) which does not require weight optimization – but is likely to fail if the accuracies of the individual classifiers are diverse. In cases where the classifier outputs can be interpreted as posterior probability estimates, product can be used instead of sum. However, the sum rule is the preferred option since the product rule amplifies estimation errors (Kittler et al., 1998). A theoretically elegant technique for optimizing the fusion weights based on *logistic regression* has been proposed in (Brümmer et al., 2007; Brümmer and Preez, 2006). An implementation of the method is available in the Fusion and Calibration (FoCal) toolkit.[2] This method, being simple and robust at the same time, is usually the first choice in our own research.

By considering outputs from the different classifiers as another random variable, *score vector*, a backend classifier can be built on top of the individual classifiers. For instance, a support vector machine or a neural network can be trained to separate the genuine and impostor score vectors (e.g. Hatch et al., 2005; Reynolds et al., 2003; Tong et al., 2006; Ferrer et al., 2008b). Upon verifying a person, each of the individual classifiers gives an output score and

[2] http://www.dsp.sun.ac.za/~nbrummer/focal/.

these scores are in turn arranged into a vector. The vector is then presented to the SVM and the SVM output score is compared against the verification threshold.

Majority of fusion approaches in speaker recognition are based on trial-and-error and optimization on given datasets. The success of a particular combination depends on the performance of the individual systems, as well as their complementariness. Whether the combiner yields improvement on an unseen dataset depends on how the optimization set matches the new dataset (in terms of signal quality, gender distribution, lengths of the training and test material, etc.).

Recently, some improvements to fusion methodology have been achieved by integrating *auxiliary side information*, also known as *quality measures*, into the fusion process (Ferrer et al., 2008a; Garcia-Romero et al., 2004; Kryszczuk et al., 2007; Solewicz and Koppel, 2007). Unlike the traditional methods where the fusion system is trained on development data and kept fixed during run-time, the idea in side-information fusion is to adapt the fusion on each test case. Signal-to-noise ratio (SNR) (Kryszczuk et al., 2007) and nonnativeness score of the test segment (Ferrer et al., 2008a) have been used as the auxiliary side information, for instance. Another recent enhancement is to model the correlations between the scores of individual subsystems, since intuitively uncorrelated systems fuse better than correlated ones (Ferrer et al., 2008b). Both the auxiliary information and correlation modeling were demonstrated to improve accuracy and are certainly worth further studying.

5. Robust speaker recognition

As a carrier wave of phonetic information, affective attributes, speaker characteristics and transmission path information, the acoustic speech signal is subject to much variations, most of which are undesirable. It is well known that any mismatch between the training and testing conditions dramatically decreases the accuracy of speaker recognition. The main focus of speaker recognition research has been in tackling this mismatch. Normalization and adaptation methods have been applied to all the parts of speaker recognition systems.

5.1. Voice activity detection

Voice activity detector (VAD), as illustrated in Fig. 10, aims at locating the speech segments from a given audio signal (Benyassine et al., 1997). The problem is analogous to face detection from images: we wish to locate the objects of interest before any further processing. VAD is an important sub-component for any real-world recognition system. Even though a seemingly simple binary classification task, it is, in fact, rather challenging to implement a VAD that works consistently across different environments. Moreover, short-duration utterances (few seconds) require special care (Fauve et al., 2008).

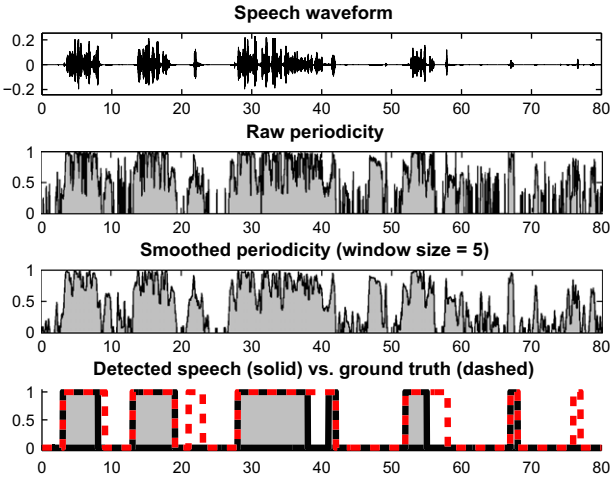

Fig. 10. Voice activity detector (VAD) based on periodicity (Hautamäki et al., 2007). It is known that voiced speech sounds (vowels, nasals) are more discriminative than fricative and stop sounds. By using periodicity rather than energy may lead to better performance in noisy environments.

A simple solution that works satisfactorily on typical telephone-quality speech data, uses signal energy to detect speech. As an example, we provide a Matlab code fragment in the following:

```
E=20*log10(std(Frames')+eps);%  Energies
maxl=max(E);%Maximum
I=(E>maxl-30) & (E>-55);%Indicator
```

Here `Frames` is a matrix that contains the short-term frames of the whole utterance as its row vectors (it is also assumed that the signal values are normalized to the range $[-1, 1]$). This VAD first computes the energies of all frames, selects the maximum, and then sets the detection threshold as 30 dB below the maximum. Another threshold (-55 dB) is needed for canceling frames with too low an absolute energy. The entire utterance (file) is required before the VAD detection is carried out. A real-time VAD, such as the *long-term spectral divergence* (LTSD) method Ramirez et al. (2004) is required in most real-world systems. Periodicity-based VAD (Fig. 10), an alternative to energy-based methods, was studied in (Hautamäki et al., 2007).

5.2. Feature normalization

In principle, it is possible to use generic noise suppression techniques to enhance the quality of the original time-domain signal prior to feature extraction. However, signal enhancement as an additional step in the entire recognition process will increase the computational load. It is more desirable to design a feature extractor which is itself robust (Mammone et al., 1996), or to normalize the features before feeding them onto the modeling or matching algorithms.

The simplest method of feature normalization is to subtract the mean value of each feature over the entire

utterance. With the MFCC and LPCC features, this is known as *cepstral mean subtraction* (CMS) or *cepstral mean normalization* (CMN) (Atal, 1974; Furui, 1981). In the log-spectral and cepstral domains, convolutive channel noise becomes additive. By subtracting the mean vector, the two feature sets obtained from different channels both become zero-mean and the effect of the channel is correspondingly reduced. Similarly, the variances of the features can be equalized by dividing each feature by its standard deviation. When VAD is used, the normalization statistics are usually computed from the detected speech frames only.

The utterance-level mean and variance normalization assume that channel effect is constant over the entire utterance. To relax this assumption, mean and variance estimates can be updated over a sliding window (Viikki and Laurila, 1998). The window should be long enough to allow good estimates for the mean and variance, yet short enough to capture time-varying properties of the channel. A typical window size is 3–5 s (Pelecanos and Sridharan, 2001; Xiang et al., 2002).

Feature warping (Pelecanos and Sridharan, 2001) and *short-term Gaussianization* (Xiang et al., 2002) aim at modifying the short-term feature distribution to follow a reference distribution. This is achieved by "warping" the cumulative distribution function of the features so that it matches the reference distribution function, for example a Gaussian. In (Pelecanos and Sridharan, 2001), each feature stream was warped independently. In (Xiang et al., 2002) the independence assumption was relaxed by applying a global linear transformation prior to warping, whose purpose was to achieve short-term decorrelation or independence of the features. Although Gaussianization was observed to improve accuracy over feature warping (Xiang et al., 2002), it is considerably more complex to implement.

RelAtive SpecTrAl (RASTA) filtering (Hermansky and Morgan, 1994; Malayath et al., 2000) applies a bandpass filter in the log-spectral or cepstral domain. The filter is applied along the temporal trajectory of each feature, and it suppresses modulation frequencies which are outside of typical speech signals. For instance, a slowly varying convolutive channel noise can be seen as a low-frequency part of the modulation spectrum. Note that the RASTA filter is signal-independent, whereas CMS and variance normalization are adaptive in the sense that they use statistics of the given signal. For useful discussions on data-driven temporal filters versus RASTA, refer to (Malayath et al., 2000).

Mean and variance normalization, Gaussianization, feature warping and RASTA filtering are unsupervised methods which do not explicitly use any channel information. *Feature mapping* (FM) (Reynolds, 2003) is a supervised normalization method which transforms the features obtained from different channel conditions into a channel-independent feature space so that channel variability is reduced. This is achieved with a set of channel-dependent GMMs adapted from a channel-independent root model. In the training or operational phase, the most likely channel (highest GMM likelihood) is detected, and the relationship between the root model and the channel-dependent model is used for mapping the vectors into channel-independent space. A generalization of the method which does not require detection of the top-1 Gaussian component was proposed in (Zhu et al., 2007).

Often different feature normalizations are used in combination. A typical robust front-end (Reynolds et al., 2005) consists of extracting MFCCs, followed by RASTA filtering, delta feature computation, voice activity detection, feature mapping and global mean/variance normalization in that order. Different orders of the normalization steps are possible; in (Burget et al., 2007) cepstral vectors were first processed through global mean removal, feature warping, and RASTA filtering, followed by adding first-, second-, and third-order delta features. Finally, voice activity detector and dimensionality reduction using heteroscedastic linear discriminant analysis (HLDA) were applied.

Graph-theoretic compensation method was proposed in (Hautamäki et al., 2008a). This method considered the training and test utterances as graphs where the graph nodes correspond to "feature points" in the feature space. The matching was then carried out by finding the corresponding feature point pairs from the two graphs based on graph isomorphism, and used for global transformation of the feature space, followed by conventional matching. The graph structure was motivated by invariance against the affine feature distortion model for cepstral features (e.g. Mak and Tsang, 2004; Mammone et al., 1996). The method requires further development to validate the assumptions of the feature distortion model and to improve computational efficiency.

5.3. Speaker model compensation

Model-domain compensation involves modifying the speaker model parameters instead of the feature vectors. One example is *speaker model synthesis* (SMS) (Teunen et al., 2000), which adapts the target GMM parameters into a new channel condition, if this condition has not been present in the enrollment phase. This is achieved with the help of transformations between a channel-independent background model and channel-dependent adapted models. Roughly, speaker model synthesis is a model-domain equivalent of feature mapping (FM) (Reynolds, 2003). Feature mapping can be considered more flexible since the mapped features can be used with any classifier and not only with the GMM.

Both SMS and FM require a labeled training set with training examples from a variety of different channel conditions. In (Mason et al., 2005), an unsupervised clustering of the channel types was proposed so that labeling would not be needed. The results indicate that feature mapping based on unsupervised channel labels achieves equal or better accuracy compared with supervised labeling. It should be noted, however, that state-of-the-art speaker modeling with supervectors use *continuous* intersession variability models and therefore extend the SMS and FM methods to handle with

unknown conditions. The continuous model compensation methods have almost completely surpassed the SMS and FM methods, and will be the focus of Section 6.

5.4. Score normalization

In *score normalization*, the "raw" match score is normalized relative to a set of other speaker models known as *cohort*. The main purpose of score normalization is to transform scores from different speakers into a similar range so that a common (speaker-independent) verification threshold can be used. Score normalization can correct some speaker-dependent score offsets not compensated by the feature and model domain methods.

A score normalization of the form

$$s' = \frac{s - \mu_I}{\sigma_I} \tag{15}$$

is commonly used. In (15), s' is the normalized score, s is the original score, and μ_I and σ_I are the estimated mean and standard deviation of the impostor score distribution, respectively. In *zero normalization* ("Z-norm"), the impostor statistics μ_I and σ_I are target speaker dependent and they are computed off-line in the speaker enrollment phase. This is done by matching a batch of non-target utterances against the target model, and obtaining the mean and standard deviation of those scores. In *test normalization* ("T-norm") (Auckenthaler et al., 2000), the parameters are test utterance dependent and they are computed "on the fly" in the verification phase. This is done by matching the unknown speaker's feature vectors against a set of impostor models and obtaining the statistics.

Usually the cohort models are common for all speakers, however, speaker-dependent cohort selection for T-norm has been studied in (Ramos-Castro et al., 2007; Sturim and Reynolds, 2005). Z-norm and T-norm can also be combined. According to (Vogt et al., 2005), Z-norm followed by T-norm does produce good results.

Score normalization can be improved by using side information such as channel type. Handset-dependent background models were used in (Heck and Weintraub, 1997). The handset type (carbon button or electret) through which the training utterance is channeled was automatically detected, and the corresponding background model was used for score normalization in the verification phase. In (Reynolds et al., 2000), handset-dependent mean and variance of the likelihood ratio were obtained for each target speaker. In the matching phase, the most likely handset was detected and the corresponding statistics were used to normalize the likelihood ratio. In essence, this approach is a handset-dependent version of Z-norm, which the authors call "H-norm". In a similar way, handset-dependent T-norm ("HT-norm") has been proposed (Dunn et al., 2001). Note that the handset-dependent normalization approaches (Dunn et al., 2001; Heck and Weintraub, 1997; Reynolds et al., 2000) require an automatic handset labeler which inevitable makes classification errors.

Although Z-norm and T-norm can be effective in reducing speaker verification error rates, they may seriously fail if the cohort utterances are badly selected, that is, if their acoustic and channel conditions differ too much from the typical enrollment and test utterances of the system. According to (Burget et al., 2007), score normalization may not be needed at all if the other components, most notable eigenchannel compensation of speaker models, are well-optimized. However, Z- and T-norms and their combinations seem to be an essential necessity for the more complete joint factor analysis model (Kenny et al., 2008). In summary, it remains partly a mystery when score normalization is useful, and would deserve more research.

6. Supervector methods: a recent research trend

6.1. What is a supervector?

One of the issues in speaker recognition is how to represent utterances that, in general, have a varying number of feature vectors. In early studies (Markel et al., 1977) speaker models were generated by time-averaging features so that each utterance could be represented as a single vector. The average vectors would then be compared using a distance measure (Kinnunen et al., 2006a), which is computationally very efficient but gives poor recognition accuracy. Since the 1980s, the predominant trend has been creating a model of the training utterance followed by "data-to-model" type of matching at run-time (e.g. likelihood of an utterance with respect to a GMM). This is computationally more demanding but gives good recognition accuracy.

Interestingly, the speaker recognition community has recently re-discovered a robust way to present utterances using a single vector, a so-called *supervector*. On one hand, these supervectors can be used as inputs to support vector machine (SVM) as illustrated in Fig. 11. This leads to *sequence kernel* SVMs, where the utterances with variable number of feature vectors are mapped to a fixed-length vector using the sequence kernel; for review and useful insights, refer to (Longworth and Gales, 2007; Wan and Renals, 2005). On the other hand, conventional adapted Gaussian mixture speaker model (Reynolds et al., 2000) can also be seen as a supervector. Combinations of generative models and SVM have also lead to good results (Campbell et al., 2006b).

Often "supervector" refers to combining many smaller-dimensional vectors into a higher-dimensional vector; for instance, by stacking the d-dimensional mean vectors of a K-component adapted GMM into a Kd-dimensional Gaussian supervector (Campbell et al., 2006b). In this paper, we understand supervector in a broader sense as any high- and fixed-dimensional representation of an utterance. It is important that the supervectors of different utterances arise from a "common coordinate system" such as being adapted from a universal background model, or being generated using a fixed polynomial basis (Campbell et al., 2006a). In this way the supervector elements are

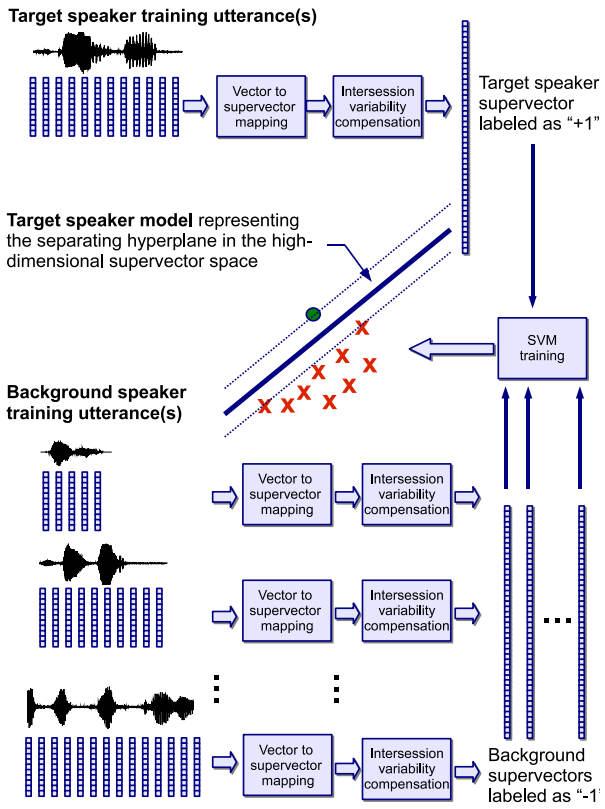

Fig. 11. The concept of modern sequence kernel SVM. Variable-length utterances are mapped into fixed-dimensional *supervectors*, followed by intersession variability compensation and SVM training.

meaningfully aligned and comparable when doing similarity computations in the supervector space. With SVMs, normalizing the dynamic ranges of the supervector elements is also crucial since SVMs are not scale invariant (Wan and Renals, 2005).

An important recent advance in speaker recognition has been the development of *explicit inter-session variability compensation* techniques (Burget et al., 2007; Kenny et al., 2008; Vogt and Sridharan, 2008). Since each utterance is now presented as a single point in the supervector space, it becomes possible to directly quantify and remove the unwanted variability from the supervectors. *Any* variation in different utterances of the same speaker, as characterized by their supervectors – be it due to different handsets, environments, or phonetic content – is harmful.

Does this mean that we will need several training utterances recorded through different microphones or environments when enrolling a speaker? Not necessarily. Rather, the intersession variability model is trained on an independent development data and then removed from the supervectors of a new speaker. The intersession variability model itself is continuous, which is in contrast with speaker model synthesis (SMS) (Teunen et al., 2000) and feature mapping (FM) (Reynolds, 2003) discussed in Section 5. Both SMS and FM assume a discrete collection of recording conditions (such as mobile/landline channels or carbon button/electric handsets). However, the explicit inter-session variability normalization techniques enable modeling channel conditions that "fall in between" some conditions that are not seen in training data.

Various authors have independently developed different session compensation methods for both GMM- and SVM-based speaker models. *Factor analysis* (FA) techniques Kenny (2006) are designed for the GMM-based recognizer and take explicit use of stochastic properties of the GMM, whereas the methods developed for SVM supervectors are often based on numerical linear algebra (Solomonoff et al., 2005). To sum up, two core design issues with the modern supervector based recognizers are (1) how to create the supervector of an utterance, (2) how to estimate and apply the session variability compensation in the supervector space. In addition, the question of how to compute the match score with the session-compensated models needs to be solved (Glembek et al., 2009).

6.2. GLDS kernel SVM

One of the simplest SVM supervectors is *generalized linear discriminant sequence* (GLDS) kernel (Campbell et al., 2006a). The GLDS method creates the supervector by explicit mapping into kernel feature space using a *polynomial expansion* (Campbell et al., 2002), denoted here as $\boldsymbol{b}(\boldsymbol{x})$. As an example, second-order polynomial expansion for a 2-dimensional vector $\boldsymbol{x} = (x_1, x_2)^T$ is given by $\boldsymbol{b}(\boldsymbol{x}) = (1, x_1, x_2, x_1^2, x_1x_2, x_2^2)^T$. During enrollment, all the background speaker and target speaker utterances $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_T\}$ are represented as average expanded feature vectors:

$$\boldsymbol{b}_{\text{avg}} = \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{b}(\boldsymbol{x}_t). \tag{16}$$

The averaged vectors are further variance-normalized using the background utterances, and assigned with the appropriate label for SVM training (+1 = target speaker vectors; −1 = background speaker vectors). The SVM optimization (using standard linear kernel) yields a set of support vectors $\boldsymbol{b}_i$, their corresponding weights α_i and a bias d. These are collapsed into a single model vector as,

$$\boldsymbol{w} = \sum_{i=1}^{L}\alpha_i t_i \boldsymbol{b}_i + \boldsymbol{d}, \tag{17}$$

where $\boldsymbol{d} = (d, 0, 0, \ldots, 0)^T$ and $t_i \in \{+1, -1\}$ are the ideal outputs (class labels of the support vectors), and L is the number of support vectors. In this way, the speaker model can be presented as a single supervector. The collapsed model vector $\boldsymbol{w}$ is also normalized using background utterances, and it serves as the model of the target speaker.

The match score in the GLDS method is computed as an inner product $s = \boldsymbol{w}_{\text{target}}^T \boldsymbol{b}_{\text{test}}$, where $\boldsymbol{w}_{\text{target}}$ denotes the normalized model vector of the target speaker and $\boldsymbol{b}_{\text{test}}$ denotes the normalized average expanded feature vector

of the test utterance. Since all the speaker models and the test utterance are represented as single vectors, the verification phase is computationally efficient. The main drawback of the GLDS method is that it is difficult to control the dimensionality of the supervectors; in practice, the polynomial expansion includes either second- or third-order monomials before the dimensionality gets infeasible.

6.3. Gaussian supervector SVM

Since the universal background model (UBM) is included as a part in most speaker recognition systems, it provides a natural way to create supervectors (Campbell et al., 2006b; Dehak and Chollet, 2006; Lee et al., 2008). This leads to *hybrid* classifier where the generative GMM–UBM model is used for creating "feature vectors" for the discriminative SVM.

In (Campbell et al., 2006b) the authors derive the *Gaussian supervector* (GSV) kernel by bounding the Kullback–Leibler (KL) divergence measure between GMMs. Suppose that we have the UBM, $\lambda_{\text{UBM}} = \{P_k, \boldsymbol{\mu}_k, \Sigma_k\}_{k=1}^{K}$, and two utterances a and b which are described by their MAP-adapted GMMs (Section 4.2). That is, $\lambda_a = \{P_k, \boldsymbol{\mu}_k^a, \Sigma_k\}_{k=1}^{K}$ and $\lambda_b = \{P_k, \boldsymbol{\mu}_k^b, \Sigma_k\}_{k=1}^{K}$ (note that the models differ only in their means). The KL divergence kernel is then defined as,

$$K(\lambda_a, \lambda_b) = \sum_{k=1}^{K} \left(\sqrt{P_k}\boldsymbol{\Sigma}_k^{-(1/2)}\boldsymbol{\mu}_k^a\right)^T \left(\sqrt{P_k}\boldsymbol{\Sigma}_k^{-(1/2)}\boldsymbol{\mu}_k^b\right). \tag{18}$$

From the implementation point of view, this just means that all the Gaussian means $\boldsymbol{\mu}_k$ need to be normalized with $\sqrt{P_k}\boldsymbol{\Sigma}_k^{-(1/2)}$ before feeding them into SVM training. Again, this is a form of variance normalization. Hence, even though only the mean vectors of the GMM are included in the supervector, the variance and weight information of the GMM is implicitly present in the role of normalizing the Gaussian supervector. It is also possible to normalize all the adapted GMM supervectors to have a constant distance from the UBM (Dehak et al., 2008). As in the GLDS kernel, the speaker model obtained via SVM optimization can be compacted as a single model supervector.

A recent extension to Gaussian supervectors is based on bounding the Bhattacharyya distance (You et al., 2009). This leads to a *GMM–UBM mean interval* (GUMI) kernel to be used in conjunction with SVM. The GUMI kernel exploits the speaker's information conveyed by the mean of GMM as well as those by the covariance matrices in an effective manner. Another alternative kernel known as *probabilistic sequence kernel* (PSK) (Lee et al., 2008; Lee et al., 2007) uses output values of the Gaussian functions rather than the Gaussian means to create a supervector. Since the individual Gaussians can be assumed to present phonetic classes (Reynolds and Rose, 1995), the PSK kernel can be interpreted as presenting high-level information related to phone occurrence probabilities.

6.4. MLLR supervector SVM

In (Karam and Campbell, 2007; Stolcke et al., 2007), the authors use *Maximum likelihood linear regression* (MLLR) transformation parameters as inputs to SVM. MLLR transforms the mean vectors of a speaker-independent model as $\boldsymbol{\mu}_k' = \boldsymbol{A}\boldsymbol{\mu}_k + \boldsymbol{b}$, where $\boldsymbol{\mu}_k'$ is the adapted mean vector, $\boldsymbol{\mu}_k$ is the world model mean vector and the parameters $\boldsymbol{A}$ and $\boldsymbol{b}$ define the linear transform. The parameters $\boldsymbol{A}$ and $\boldsymbol{b}$ are estimated by maximizing the likelihood of the training data with a modified EM algorithm (Leggetter and Woodland, 1995). Originally MLLR was developed for speaker adaptation in speech recognition (Leggetter and Woodland, 1995) and it has also been used in speaker recognition as an alternative to maximum a posterior (MAP) adaptation of the universal background model (UBM) (Mak et al., 2006).

The key differences between MLLR and Gaussian supervectors are in the underlying speech model – phonetic hidden Markov models versus GMMs, and the adaptation method employed – MLLR versus *maximum a posteriori* (MAP) adaptation. MLLR is motivated to benefit from more detailed speech model and the efficient use of data through transforms that are shared across Gaussians (Stolcke et al., 2007). Independent studies (Castaldo et al., 2007b; Lei and Mirghafori, 2007) have shown that detailed speech model improve the speaker characterization ability of supervectors.

A similar work to MLLR supervectors is to use *feature transformation* (FT) parameters as inputs to SVM (Zhu et al., 2008), where a flexible FT function clusters transformation matrices and bias vectors with different regression classes. The FT framework is based on GMM–UBM rather than hidden Markov model, therefore, does not require a phonetic acoustic system. The FT parameters are estimated with the MAP criteria that overcome possible numerical problems with insufficient training. A recent extension of this framework (Zhu et al., 2009) includes the joint MAP adaptation of FT and GMM parameters.

6.5. High-level supervector SVM

The GLDS-, GMM- and MLLR-supervectors are suitable for modeling short-term spectral features. For the prosodic and high-level features (Subsections 3.4 and 3.5), namely, features created using a tokenizer front-end, it is customary to create a supervector by concatenating the uni-, bi- and tri-gram ($N = 1, 2, 3$) frequencies into a vector or *bag-of-N-grams* (Campbell et al., 2004; Shriberg et al., 2005). The authors of Campbell et al. (2004) developed *term frequency log likelihood ratio* (TFLLR) kernel that normalizes the original N-gram frequency by $1/\sqrt{f_i}$, where f_i is the overall frequency of that N-gram. Thus the value of rare N-grams is increased and the value of frequent N-grams is decreased, thereby equalizing their contribution in kernel computations.

The high-level features created by a phone tokenizer, or by quantization of prosodic feature values by binning (Shriberg et al., 2005), are inherently noisy: tokenizer error (e.g. phone recognizer error) or small variation in the original feature value may cause the feature to fall into a wrong category (bin). To tackle this problem, the authors of Ferrer et al. (2007) proposed to use *soft* binning with the aid of Gaussian mixture model and use the weights of the Gaussians as the features for SVM supervector.

6.6. Normalizing SVM supervectors

Two forms of SVM supervector normalizations are necessary: normalizing the dynamic range of features and intersession variability compensation. The first one, normalizing the dynamic range, is related to the inherent property of the SVM model. SVM is not invariant to linear transformations in feature space and some form of variance normalization is required so that certain supervector dimensions do not dominate the inner product computations. Often variance normalization is included in the definition of the kernel function and specific to a given kernel as seen in the previous subsections. Kernel-independent *rank normalization* has also been successfully applied (Stolcke et al., 2008). Rank normalization replaces each feature by its relative position (rank) in the background data. For useful insights on normalization, refer to (Stolcke et al., 2008; Wan and Renals, 2005). Let us now turn our focus to the other necessary normalization, the intersession variability compensation.

Nuisance attribute projection (NAP) is a successful method for compensating SVM supervectors (Campbell et al., 2005; Solomonoff et al., 2005). It is not specific to some kernel, but can be applied to *any* kind of SVM supervectors. The NAP transformation removes the directions of undesired sessions variability from the supervectors before SVM training. The NAP transformation of a given supervector $\boldsymbol{s}$ is (Brümmer et al., 2007),

$$\boldsymbol{s}' = \boldsymbol{s} - \boldsymbol{U}(\boldsymbol{U}^T\boldsymbol{s}), \tag{19}$$

where $\boldsymbol{U}$ is the *eigenchannel* matrix. The eigenchannel matrix is trained using a development dataset with a large number of speakers, each having several training utterances (sessions). The training set is prepared by subtracting the mean of the supervectors within each speaker and pooling all the supervectors from different speakers together; this removes most of the speaker variability but leaves session variability. By performing eigen-analysis on this training set, one captures the principal directions of channel variability. The underlying assumption is that the session variability lies in a speaker-independent low-dimensional subspace; after training the projection matrix, the method can be applied for unseen data with different speakers. The Eq. (19) then just means subtracting the supervector that has been projected on the channel space. For practical details of NAP, refer to (Brümmer et al., 2007; Fauve et al., 2007).

Some of the dimensions removed by NAP may contain speaker-specific information (Vogt et al., 2008). Moreover, session compensation and SVM optimization processes are treated independently from each other. Motivated with these facts, discriminative variant of NAP has been studied in (Burget et al., 2009; Vogt et al., 2008). In (Vogt et al., 2008), *scatter difference analysis* (SDA), a similar method to linear discriminant analysis (LDA), was used for optimizing the NAP projection matrix, and in (Burget et al., 2009), the session variability model was directly integrated within the optimization criterion of the SVM; this leaves the decision about usefulness of the supervector dimensions for the SVM optimizer. This approach improved recognition accuracy over the NAP baseline in Burget et al. (2009), albeit introducing a new control parameter that controls the contribution of the nuisance subspace constraint. Nevertheless, discriminative session compensation is certainly an interesting new direction for future studies.

Within-class covariance normalization (WCCN), another SVM supervector compensation method similar to NAP, was proposed in (Hatch and Stolcke, 2006). The authors considered generalized linear kernels of the form $K(\boldsymbol{s}_1, \boldsymbol{s}_2) = \boldsymbol{s}_1\boldsymbol{R}\boldsymbol{s}_2$, where $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ are supervectors and $\boldsymbol{R}$ is a positive semidefinite matrix. With certain assumptions, a bound of a binary classification error metric can be minimized by choosing $\boldsymbol{R} = \boldsymbol{W}^{-1}$, where $\boldsymbol{W}$ is the expected within-class (within-speaker) covariance matrix. The WCCN was then combined with principal component analysis (PCA) in (Hatch et al., 2006) to attack the problem of estimating and inverting $\boldsymbol{W}$ to large data sets. The key difference between NAP and WCCN is the way how they weight the dimensions in the supervector space (Stolcke et al., 2007). The NAP method completely removes some of the dimensions by projecting the supervectors to a lower-dimensional space, whereas WCCN weights rather than completely removes the dimensions.

6.7. Factor analysis techniques

In the previous subsection we focused on compensating SVM supervectors. We will now discuss a different technique based on generative modeling, that is, Gaussian mixture model (GMM) with factor analysis (FA) technique. Recall that the MAP adaptation technique for GMMs (Reynolds et al., 2000), as described in Section 4.2, adapts the mean vectors of the universal background model (UBM) while the weights and covariances are shared between all speakers. Thus a speaker model is uniquely represented as the concatenation of the mean vectors, which can be interpreted as a supervector.

For a given speaker, the supervectors estimated from different training utterances may not be the same especially when these training samples come from different handsets. Channel compensation is therefore necessary to make sure that test data obtained from different channel (than that of the training data) can be properly scored against the speaker models. For channel compensation to be possible,

the channel variability has to be modeled explicitly. The technique of *joint factor analysis* (JFA) (Kenny, 2006) was proposed for this purpose.

The JFA model considers the variability of a Gaussian supervector as a linear combination of the speaker and channel components. Given a training sample, the speaker-dependent and channel-dependent supervector $\boldsymbol{M}$ is decomposed into two statistically independent components, as follows

$$\boldsymbol{M} = \boldsymbol{s} + \boldsymbol{c}, \tag{20}$$

where $\boldsymbol{s}$ and $\boldsymbol{c}$ are referred to as the *speaker* and *channel* supervectors, respectively. Let d be the dimension of the acoustic feature vectors and K be the number of mixtures in the UBM. The supervectors $\boldsymbol{M}$, $\boldsymbol{s}$ and $\boldsymbol{c}$ live in a Kd-dimensional parameter space. The channel variability is explicitly modeled by the channel model of the form,

$$\boldsymbol{c} = \boldsymbol{U}\boldsymbol{x}, \tag{21}$$

where $\boldsymbol{U}$ is a rectangular matrix and $\boldsymbol{x}$ are the *channel factors* estimated from a given speech sample. The columns of the matrix $\boldsymbol{U}$ are the *eigenchannels* estimated for a given dataset. During enrollment, the channel factors $\boldsymbol{x}$ are to be estimated jointly with the speaker factors y of the speaker model of the following form:

$$\boldsymbol{s} = \boldsymbol{m} + \boldsymbol{V}\boldsymbol{y} + \boldsymbol{D}\boldsymbol{z}. \tag{22}$$

In the above equation, $\boldsymbol{m}$ is the UBM supervector, $\boldsymbol{V}$ is a rectangular matrix with each of its columns referred to as the eigenvoices, $\boldsymbol{D}$ is $Kd \times Kd$ diagonal matrix and $\boldsymbol{z}$ is a $Kd \times 1$ column vector. In the special case $\boldsymbol{y} = \boldsymbol{0}$, $\boldsymbol{s} = \boldsymbol{m} + \boldsymbol{D}\boldsymbol{z}$ describes exactly the same adaptation process as the MAP adaptation technique (Section 4.2). Therefore, the speaker model in the JFA technique can be seen as an extension to the MAP technique with the eigenvoice model $\boldsymbol{V}\boldsymbol{y}$ included, which has been shown to be useful for short training samples.

The matrices $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{D}$ are called the *hyperparameters* of the JFA model. These matrices are estimated beforehand on large datasets. One possible way is to first estimate $\boldsymbol{V}$ followed by $\boldsymbol{U}$ and $\boldsymbol{D}$ (Kenny, 2006; Kenny et al., 2008). For a given training sample, the latent factors $\boldsymbol{x}$ and $\boldsymbol{y}$ are jointly estimated and followed by estimation of $\boldsymbol{z}$. Finally, the channel supervector $\boldsymbol{c}$ is discarded and the speaker supervector $\boldsymbol{s}$ is used as the speaker model. By doing so, channel compensation is accomplished via the explicit modeling of the channel component during training. For detailed account of estimation procedure the reader should refer to (Kenny, 2006; Kenny et al., 2008). For comparing various scoring methods, refer to (Glembek et al., 2009).

The JFA model dominated the latest NIST 2008 speaker recognition evaluation (SRE) (NIST, 2008) and it was pursued further in the Johns Hopkins University (JHU) summer 2008 workshop (Burget et al., 2009). Independent evaluations by different research groups have clearly indicated the potential of JFA. The method has a few practical deficiencies, however. One is sensitivity to training and test lengths (and their mismatch), especially for short utterances (10–20 s). The authors of Burget et al. (2009) hypothesized that this was caused by *within*-session variability (due to phonemic variability) rather than *inter*-session variability captured by the baseline JFA. The authors then extended the JFA model by explicitly adding a model of the within-session variability. Other choices to tackle the JFA dependency on utterance length were studied as well – namely, utilizing variable length development utterances to create *stacked* channel matrix. The extended JFA and the stacking approach both showed improvement over the baseline JFA when the training and test utterance lengths were not matched, hence improving the generalization of JFA for unknown utterance lengths. The within-session variability modeling, however, has a price: a phone recognizer was used for generating data for within-session modeling. It may be worthwhile to study simplified approach – segmenting the data into fixed-length chunks – as proposed in (Burget et al., 2009).

Given the demonstrated excellent performance of the JFA compensation and Gaussian supervector SVMs (Campbell et al., 2006b), it seems appropriate to ask how they compare with each other, and whether they could be combined? These questions were recently addressed in (Dehak et al., 2008; Dehak et al., 2009). In (Dehak et al., 2008) the authors compared JFA and SVM both with linear and nonlinear kernels, compensated with nuisance attribute projection (NAP). They concluded that JFA without speaker factors gives similar accuracy to SVM with Gaussian supervectors; however, JFA outperformed SVM when speaker factors were added. In (Dehak et al., 2009) the same authors used the speaker factors of the JFA model as inputs to SVM. Within-class covariance normalization (WCCN) (Stolcke et al., 2007) was used instead of NAP. The results indicated that using the speaker factors in SVM is effective but the accuracy was not improved over the JFA-compensated GMM. The combined JFA–SVM method, however, results in faster scoring.

6.8. Summary: which supervector method to use?

Given the multiple choices to create a supervector and to model intersession variability, which one to choose for practical use? It is somewhat difficult to compare the methods in literature due to differences in data set selections, parameter settings and other implementation details. However, there are some common practice that we can follow. To facilitate discussion, we present here the results of the latest NIST 2008 speaker recognition evaluation submission by the I4U consortium (Li et al., 2009). All the classifiers of I4U used short-term spectral features and the focus was in the supervectors classifiers. Three well-known methods – Gaussian mixture model–universal background model (GMM–UBM) (Reynolds et al., 2000), generalized linear discriminant sequence (GLDS) kernel SVM (Campbell et al., 2006a) and Gaussian supervector (GSV) kernel

Table 1
Performance of individual classifiers and their fusion of I4U system on I4U's telephone quality development dataset (Li et al., 2009). UNC = Uncompensated, EIG = Eigenchannel, JFA = Joint factor analysis, GLDS = Generalized linear discriminant sequence, GSV = Gaussian supervector, FT = Feature transformation, PSK = Probabilistic sequence kernel, BK = Bhattacharyya kernel. All the SVM-based systems use nuisance attribute projection (NAP) compensation.

	Tuning set EER (%)	Eval. set EER (%)
Gaussian mixture model		
1. GMM–UBM (UNC)	8.45	8.10
2. GMM–UBM (EIG) (Kenny et al., 2008)	5.47	5.22
3. GMM–UBM (JFA) (Kenny et al., 2008)	3.19	3.11
Support vector machine with different kernels		
4. GLSD–SVM (Campbell et al., 2006a)	4.30	4.44
5. GSV–SVM (Campbell et al., 2006b)	4.47	4.43
6. FT–SVM (Zhu et al., 2008)	4.20	3.66
7. PSK–SVM (Lee et al., 2008)	5.29	4.77
8. BK–SVM (You et al., 2009)	4.46	5.16
Fusing systems 2–8	2.49	2.05

SVM (GSV–SVM) (Campbell et al., 2006b) were studied. In addition, three novel SVM kernels were proposed: feature transformation kernel (FT-SVM) (Zhu et al., 2009), probabilistic sequence kernel (PSK–SVM) (Lee et al., 2008; Lee et al., 2007) and Bhattacharyya kernel (BK-SVM) (You et al., 2009).

Table 1 reports the performance of individual systems, together with the weighted summation fusion of the classifiers. The accuracy is measured in *equal error rate* (EER), a verification error measure that gives the accuracy at decision threshold for which the probabilities of false rejection (miss) and false acceptance (false alarm) are equal (see Section 7).

From the results in Table 1 it is clear that intersession compensation significantly improves the accuracy of the GMM–UBM system. It can also be seen that the best individual classifier is the GMM–UBM system with JFA compensation, and that JFA outperforms the eigenchannel method (which is a special case of JFA). Finally, fusing all the session-compensated classifiers improves accuracy as expected.

Even though JFA outperforms the SVM-based methods, for practitioners we recommend to start with the two simplest approaches at this moment: GLDS–SVM and GSV–SVM. The former does not require much optimization whereas the latter comes almost as a by-product when a GMM–UBM system is used. Furthermore, they do not require as many datasets as JFA does, are simple to implement and fast in computation. They should be augmented with nuisance attribute projection (NAP) (Brümmer et al., 2007) and test normalization (T-norm) (Auckenthaler et al., 2000).

7. Performance evaluation and software packages

7.1. Performance evaluation

Assessing the performance of new algorithms on a common dataset is essential to enable meaningful performance comparison. In early studies, corpora consisted of a few or at the most a few dozen speakers, and data was often self-collected. Recently, there has been significant effort directed towards standardizing the evaluation methodology in speaker verification.

The National Institute of Standards and Technology (NIST)[3] provides a common evaluation framework for text-independent speaker recognition methods (Przybocki et al., 2007). NIST evaluations include test trials under both matched conditions such as telephone only, and unmatched conditions such as language effects (matched languages vs unmatched languages), cross channel and two-speaker detection. NIST has conducted speaker recognition benchmarking on an annual basis since 1997, and registration is open to all parties interested in participating in this benchmarking activity. During the evaluation, NIST releases a set of speech files as the development data to the participants. At this initial phase, the participants do not have access to the "ground truth", that is, the speaker labels. Each participating group then runs their algorithms "blindly" on the given data and submits the recognition scores and verification decisions. NIST then evaluates the performances of the submissions and the results are discussed in a follow-up workshop. The use of "blind" evaluation data makes it possible to conduct an unbiased comparison of the various algorithms. These activities would be difficult without a common evaluation dataset or a standard evaluation protocol.

Visual inspections of the *detection error trade-off* (DET) curves (Martin et al., 1997) and *equal error rate* (EER) are commonly used evaluation tools in the speaker verification literature. An example of DET curve is shown in Fig. 12. The problem with EER is that it corresponds to an arbitrary detection threshold, which is not a likely choice in a real application where it is critical to maintain the balance between user convenience and security. NIST uses a *detection cost function* (DCF) as the primary evaluation metric to assess speaker verification performance:

$$\mathrm{DCF}(\Theta) = 0.1 \times P_{\mathrm{miss}}(\Theta) + 0.99 \times P_{\mathrm{fa}}(\Theta). \quad (23)$$

Here $P_{\mathrm{miss}}(\Theta)$ and $P_{\mathrm{fa}}(\Theta)$ are the probabilities of *miss* (i.e. rejection of a genuine speaker) and *false alarm* (i.e. acceptance of an impostor), respectively. Both of them are functions of a global (speaker-independent) verification threshold Θ.

Minimum DCF (MinDCF), defined as the DCF value at the threshold for which (23) is smallest, is the optimum cost. When the decision threshold is optimized on a

[3] http://nist.gov/.

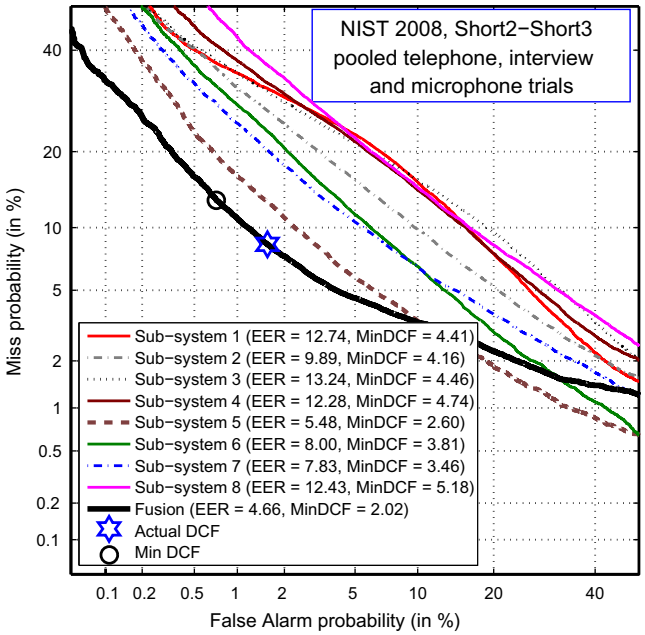

Fig. 12. Example of detection error trade-off (DET) plot presenting various subsystems and a combined system using score-level fusion.

development set and applied to the evaluation corpus, this produces *actual DCF*. Therefore, the difference between the minimum DCF and the actual DCF indicates how well the system is *calibrated* for a certain application and how robust is the threshold setting method. For an in-depth and thorough theoretical discussion as well as the alternative formulations of application-independent evaluation metrics, refer to (Brümmer and Preez, 2006).

While the NIST speaker recognition benchmarking considers mostly conversational text-independent speaker verification in English, there have been a few alternative evaluations, for instance the NFI-TNO evaluation[4] which considered authentic forensic samples (mostly in Dutch), including wiretap recordings. Another evaluation, specifically for Chinese, was organized in conjunction with the Fifth International Symposium on Chinese Spoken Language Processing (ISCSLP'06).[5] This evaluation included open-set speaker identification and text-dependent verification tasks in addition to text-independent verification.

Some of the factors affecting speaker recognition accuracy in the NIST and NFI-TNO evaluations have been analyzed in (Leeuwen et al., 2006). It is widely known that cross-channel training and testing display a much lower accuracy compared to that with same channel. Including different handsets in the training material also improves recognition accuracy. Another factor significant to performance is the duration of training and test utterances. The greater the amount of speech data used for training and/or testing, the better the accuracy. Training utterance duration seems to be more significant than test segment duration.

7.2. Software packages for speaker recognition

As can be seen throughout this article, the state-of-the-art speaker recognition methods are getting more and more advanced and they often combine several complementary techniques. Implementing a full system from scratch may not be meaningful. In this subsection we point out a few useful software packages that can be used for creating a state-of-the-art speaker recognition system.

Probably the most comprehensive and up-to-date software package is ALIZE toolkit[6], an open-source software developed at Université d'Avignon, France. For more details, the interested reader is referred to (Fauve et al., 2007).

For research purposes, it is possible to build up a complete speaker recognition system using various different software packages. The Matlab software by MathWorks Inc. is excellent especially for developing new feature extraction methods. Octave[7] is an open-source alternative to Matlab is, and there are a plenty of free toolboxes for both of them such as Statistical Pattern Recognition Toolbox[8] and NetLab.[9] Aside from Matlab/Octave, the Hidden Markov Model Toolkit (HTK)[10] is also popular in statistical modeling, whereas Torch[11] software represents state-of-the-art SVM implementation.

For score fusion of multiple sub-systems, we recommend the FoCal toolkit.[12] For evaluation purposes, such as plotting DET curves, we recommend the DETware toolbox (for Matlab) by NIST.[13] A similar tool but with more features is SRETools.[14]

8. Future horizons of speaker recognition

During the past 10 years, speaker recognition community has made significant advances in the technology. In summary, we have selected a few of the most influential techniques that have been proven to work in practice in independent studies, or shown significant promise in the past few NIST technology evaluation benchmarks:

- Universal background modeling (UBM) (Reynolds et al., 2000).
- Score normalization, calibration, fusion (Auckenthaler et al., 2000; Burget et al., 2007).

[4] http://speech.tm.tno.nl/aso/.
[5] http://www.iscslp2006.org/.
[6] Now under "Mistral" platform for biometrics authentication. Available at: http://mistral.univ-avignon.fr/en/.
[7] http://www.gnu.org/software/octave/.
[8] http://cmp.felk.cvut.cz/cmp/software/stprtool/.
[9] http://www.ncrg.aston.ac.uk/netlab/index.php.
[10] http://htk.eng.cam.ac.uk/.
[11] http://www.torch.ch/.
[12] http://niko.brummer.googlepages.com/focal.
[13] http://www.itl.nist.gov/iad/mig/tools/DETware_v2.1.targz.htm.
[14] http://sretools.googlepages.com/.

- Sequence kernel SVMs (Campbell et al., 2006a; Campbell et al., 2006b).
- Use of prosodics and high-level features with SVM (Campbell et al., 2004; Shriberg et al., 2005; Stolcke et al., 2007).
- Phonetic normalization using ASR (Castaldo et al., 2007b; Stolcke et al., 2007).
- Explicit session variability modeling and compensation (Brümmer et al., 2007; Castaldo et al., 2007b; Hatch et al., 2006; Kenny et al., 2008).

Even though effective, these methods are highly data-driven and massive amounts of data are needed for training the background models, cohort models for score normalization, and modeling session and speaker variabilities. The data sets need to be labeled and organized in a controlled manner requiring significant human efforts. It is not trivial to decide how to split the system development data for UBM training, session modeling, and score normalization. If the development data conditions do not match to those of the expected operation environment, the accuracy will drop significantly, sometimes to unusable level. It is clear that laborious design of data set splits cannot be expected, for instance, from forensic investigators who just want to use speaker recognition software in "turn-key" fashion.

For transferring the technology into practice, therefore, in future it will be important to focus on making the methods less sensitive to selection of the data sets. The methods also require computational simplifications before they can be used in real-world applications such as in smart cards or mobile phones, for instance. Finally, the current techniques require several *minutes* of training and test data to give satisfactory performance, that presents a challenge for applications where real-time decision is desired. For instance, the core evaluation condition in recent NIST benchmarkings uses about 2.5 min of speech data. New methods for short training and test utterances (less than 10 s) will be needed. The methods for long data do not readily generalize to short-duration tasks as indicated in (Bonastre et al., 2007; Burget et al., 2009; Fauve et al., 2008).

The NIST speaker recognition evaluations (Leeuwen et al., 2006; Przybocki et al., 2007) have systematized speaker recognition methodology development and constant positive progress has been observed in the past years. However, the NIST evaluations have mostly focused on combating *technical* error sources, most notably that of training/test channel mismatch (for instance, using different microphones in training and test material). There are also many other factors that have impacts on the speaker recognition performance. We should also address *human-related* error sources, such as the effects of emotions, vocal organ illness, aging, and level of attention. Furthermore, one of the most popular questions asked by laymen is "what if someone or some machine imitates me or just plays previously recorded signal back?". Before considering speaker recognition in large-scale commercial applications, the research community must answer such questions. These questions have been considered in some studies, mostly in the context of phonetic sciences, but always for a limited number of speakers and using non-public corpora. As voice transformation technique advances, low cost voice impersonation becomes possible (Bonastre et al., 2007; Pellom and Hansen, 1999). This opens up a new horizon to study attack and defense in voice biometrics.

Much of the recent progress in speaker recognition is attributed to the success in classifier design and session compensation, which largely rely on traditional short-term spectral features. These features were introduced nearly 30 years ago for speech recognition (Davis and Mermelstein, 1980). Despite there is a strong belief that temporal, prosodic and high-level features are salient speaker cues, we have not benefited much from them. So far, they are playing a secondary role complementary to short-term spectral features. This warrants further investigation, especially as to how temporal and prosodic features can capture high-level phenomena (robust) without using computationally intensive speech recognizer (practical). It remains a great challenge in the near future to understand what features to exactly look for in speech signal.

9. Summary

We have presented an overview of the classical and new methods of automatic text-independent speaker recognition. The recognition accuracy of current speaker recognition systems under controlled conditions is high. However, in practical situations many negative factors are encountered including mismatched handsets for training and testing, limited training data, unbalanced text, background noise and non-cooperative users. The techniques of robust feature extraction, feature normalization, model-domain compensation and score normalization methods are necessary. The technology advancement as represented by NIST evaluations in the recent years has addressed several technical challenges such as text/language dependency, channel effects, speech durations, and cross-talk speech. However, many research problems remain to be addressed, such as human-related error sources, real-time implementation, and forensic interpretation of speaker recognition scores.

Acknowledgements

The authors would like to thank Ms. Sharifah Mahani Aljunied for spell-checking an earlier version of the manuscript, and Dr. Kong-Aik Lee for providing insights into channel compensation of supervectors.

References

Adami, A., 2007. Modeling prosodic differences for speaker recognition. Speech Comm. 49 (4), 277–291.

Adami, A., Mihaescu, R., Reynolds, D., Godfrey, J., 2003. Modeling prosodic dynamics for speaker recognition. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2003), Hong Kong, China, April 2003, pp. 788–791.

Alexander, A., Botti, F., Dessimoz, D., Drygajlo, A., 2004. The effect of mismatched recording conditions on human and automatic speaker recognition in forensic applications. Forensic Science International 146S, December 2004, pp. 95–99.

Alku, P., Tiitinen, H., Näätänen, R., 1999. A method for generating natural-sounding speech stimuli for cognitive brain research. Clin. Neurophysiol. 110 (8), 1329–1333.

Altincay, H., Demirekler, M., 2003. Speaker identification by combining multiple classifiers using Dempster–Shafer theory of evidence. Speech Comm. 41 (4), 531–547.

Ambikairajah, E., 2007. Emerging features for speaker recognition. In: Proc. Sixth Internat. IEEE Conf. on Information, Communications & Signal Processing, Singapore, December 2007, pp. 1–7.

Andrews, W., Kohler, M., Campbell, J., 2001. Phonetic speaker recognition. In: Proc. Seventh European Conf. on Speech Communication and Technology (Eurospeech 2001), Aalborg, Denmark, September 2001, pp. 2517–2520.

Andrews, W., Kohler, M., Campbell, J., Godfrey, J., Hernandez-Cordero, J., 2002. Gender-dependent phonetic refraction for speaker recognition. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2002), Vol. 1, Orlando, Florida, USA, May 2002, pp. 149–152.

Arcienega, M., Drygajlo, A., 2001. Pitch-dependent GMMs for text-independent speaker recognition systems. In: Proc. Seventh European Conf. on Speech Communication and Technology (Eurospeech 2001), Aalborg, Denmark, September 2001, pp. 2821–2824.

Ashour, G., Gath, I., 1999. Characterization of speech during imitation. In: Proc. Sixth European Conf. on Speech Communication and Technology (Eurospeech 1999), Budapest, Hungary, September 1999, pp. 1187–1190.

Atal, B., 1972. Automatic speaker recognition based on pitch contours. J. Acoust. Soc. Amer. 52 (6), 1687–1697.

Atal, B., 1974. Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. J. Acoust. Soc. Amer. 55 (6), 1304–1312.

Atlas, L., Shamma, S., 2003. Joint acoustic and modulation frequency. EURASIP J. Appl. Signal Process. 7, 668–675.

Auckenthaler, R., Mason, J., 2001. Gaussian selection applied to text-independent speaker verification. In: Proc. Speaker Odyssey: the Speaker Recognition Workshop (Odyssey 2001), Crete, Greece, June 2001, pp. 83–88.

Auckenthaler, R., Carey, M., Lloyd-Thomas, H., 2000. Score normalization for text-independent speaker verification systems. Digital Signal Process. 10 (1–3), 42–54.

Bartkova, K., Gac, D.L., Charlet, D., Jouvet, D., 2002. Prosodic parameter for speaker identification. In: Proc. Internat. Conf. on Spoken Language Processing (ICSLP 2002), Denver, Colorado, USA, September 2002, pp. 1197–1200.

Benyassine, A., Schlomot, E., Su, H., 1997. ITU-T recommendation g729 annex b: a silence compression scheme for use with g729 optimized for v.70 digital simultaneous voice and data applications. IEEE Comm. Mag. 35, 64–73.

BenZeghiba, M., Bourland, H., 2003. On the combination of speech and speaker recognition. In: Proc. Eighth European Conf. on Speech Communication and Technology (Eurospeech 2003), Geneva, Switzerland, September 2003, pp. 1361–1364.

BenZeghiba, M., Bourland, H., 2006. User-customized password speaker verification using multiple reference and background models. Speech Comm. 48 (9), 1200–1213.

Besacier, L., Bonastre, J.-F., 2000. Subband architecture for automatic speaker recognition. Signal Process. 80, 1245–1259.

Besacier, L., Bonastre, J., Fredouille, C., 2000. Localization and selection of speaker-specific information with statistical modeling. Speech Comm. 31, 89–106.

Bimbot, F., Magrin-Chagnolleau, I., Mathan, L., 1995. Second-order statistical measures for text-independent speaker identification. Speech Comm. 17, 177–192.

Bimbot, F., Bonastre, J.-F., Fredouille, C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-Garcia, J., Petrovska-Delacretaz, D., Reynolds, D., 2004. A tutorial on text-independent speaker verification. EURASIP J. Appl. Signal Process. 4, 430–451.

Bishop, C., 2006. Pattern Recognition and Machine Learning. Springer Science+Business Media, LLC, New York.

Bocklet, T., Shriberg, E., 2009. Speaker recognition using syllable-based constraints for cepstral frame selection. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2009), Taipei, Taiwan, April 2009, pp. 4525–4528.

Boersma, P., Weenink, D., 2009. Praat: doing phonetics by computer [computer program]. WWW page, June 2009, <http://www.praat.org/>.

Bonastre, J.-F., Matrouf, D., Fredouille, C., 2007. Artificial impostor voice transformation effects on false acceptance rates. In: Proc. Interspeech 2007 (ICSLP), Antwerp, Belgium, August 2007, pp. 2053–2056.

Brümmer, N., Preez, J., 2006. Application-independent evaluation of speaker detection. Comput. Speech Lang. 20, 230–275.

Brümmer, N., Burget, L., Černocký, J., Glembek, O., Grézl, F., Karafiát, M., Leeuwen, D., Matějka, P., Schwartz, P., Strasheim, A., 2007. Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006. IEEE Trans. Audio, Speech Language Process. 15 (7), 2072–2084.

Burget, L., Matějka, P., Schwarz, P., Glembek, O., Černocký, J., 2007. Analysis of feature extraction and channel compensation in a GMM speaker recognition system. IEEE Trans. Audio, Speech Language Process. 15 (7), 1979–1986.

Burget, L., Brümmer, N., Reynolds, D., Kenny, P., Pelecanos, J., Vogt, R., Castaldo, F., Dehak, N., Dehak, R., Glembek, O., Karam, Z., Noecker, J., Na, E., Costin, C., Hubeika, V., Kajarekar, S., Scheffer, N., and Černocký, J. 2009. Robust speaker recognition over varying channels – report from JHU workshop 2008. Technical report, March 2009, (URL valid June 2009). <http://www.clsp.jhu.edu/workshops/ws08/documents/jhu_report_main.pdf>.

Burton, D., 1987. Text-dependent speaker verification using vector quantization source coding. IEEE Trans. Acoustics, Speech, Signal Process. 35 (2), 133–143.

Campbell, J., 1997. Speaker recognition: a tutorial. Proc. IEEE 85 (9), 1437–1462.

Campbell, W., Assaleh, K., Broun, C., 2002. Speaker recognition with polynomial classifiers. IEEE Trans. Speech Audio Process. 10 (4), 205–212.

Campbell, W., Campbell, J., Reynolds, D., Jones, D., Leek, T., 2004. Phonetic speaker recognition with support vector machines. In: Thrun, S., Saul, L., Schokopf, B. (Eds.), . In: Advances in Neural Information Processing Systems, Vol. 16. MIT Press, Cambridge, MA.

Campbell, W., Sturim, D., Reynolds, D., 2005. SVM based speaker verification using a GMM supervector kernel and NAP variability compensation. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2005), Philadelphia, USA, March 2005, pp. 637–640.

Campbell, W., Campbell, J., Reynolds, D., Singer, E., Torres-Carrasquillo, P., 2006a. Support vector machines for speaker and language recognition. Comput. Speech Lang. 20 (2–3), 210–229.

Campbell, W., Sturim, D., Reynolds, D., 2006b. Support vector machines using GMM supervectors for speaker verification. IEEE Signal Process. Lett. 13 (5), 308–311.

Carey, M., Parris, E., Lloyd-Thomas, H., Bennett, S., 1996. Robust prosodic features for speaker identification. In: Proc. Internat. Conf. on Spoken Language Processing (ICSLP 1996), Philadelphia, Pennsylvania, USA, 1996, pp. 1800–1803.

Castaldo, F., Colibro, D., Dalmasso, E., Laface, P., Vair, C., 2007a. Compensation of nuisance factors for speaker and language recognition. IEEE Trans. Audio, Speech Language Process. 15 (7), 1969–1978.

Castaldo, F., Colibro, D., Dalmasso, E., Laface, P., Vair, C., 2007b. Compensation of nuisance factors for speaker and language recognition. IEEE Trans. Audio, Speech Language Process. 15 (7), 1969–1978.

Chan, W., Zheng, N., Lee, T., 2007. Discrimination power of vocal source and vocal tract related features for speaker segmentation. IEEE Trans. Audio, Speech Language Process. 15 (6), 1884–1892.

Charbuillet, C., Gas, B., Chetouani, M., Zarader, J., 2006. Filter bank design for speaker diarization based on genetic algorithms. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2006), Vol. 1, Toulouse, France, May 2006, pp. 673–676.

Chaudhari, U., Navratil, J., Maes, S., 2003. Multigrained modeling with pattern specific maximum likelihood transformations for text-independent speaker recognition. IEEE Trans. Speech Audio Process. 11 (1), 61–69.

Chen, K., Wang, L., Chi, H., 1997. Methods of combining multiple classifiers with different features and their applications to text-independent speaker recognition. Internat. J. Pattern Recognition Artif. Intell. 11 (3), 417–445.

Chen, Z.-H., Liao, Y.-F., Juang, Y.-T., 2004. Eigen-prosody analysis for robust speaker recognition under mismatch handset environment. In: Proc. Internat. Conf. on Spoken Language Processing (ICSLP 2004), Jeju, South Korea, October 2004, pp. 1421–1424.

Chetouani, M., Faundez-Zanuy, M., Gas, B., Zarader, J., 2009. Investigation on LP-residual presentations for speaker identification. Pattern Recognition 42 (3), 487–494.

Cheveigné, A., Kawahara, H., 2001. Comparative evaluation of f_0 estimation algorithms. In: Proc. Seventh European Conf. on Speech Communication and Technology (Eurospeech 2001), Aalborg, Denmark, September 2001, pp. 2451–2454.

Damper, R., Higgins, J., 2003. Improving speaker identification in noise by subband processing and decision fusion. Pattern Recognition Lett. 24, 2167–2173.

Davis, S., Mermelstein, P., 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans. Acoustics, Speech, Signal Process. 28 (4), 357–366.

DeCheveigne, A., Kawahara, H., 2002. YIN, a fundamental frequency estimator for speech and music. J. Acoust. Soc. Amer. 111 (4), 1917–1930.

Dehak, N., Chollet, G., 2006. Support vector GMMs for speaker verification. In: Proc. IEEE Odyssey: the Speaker and Language Recognition Workshop (Odyssey 2006), San Juan, Puerto Rico, June 2006.

Dehak, N., Kenny, P., Dumouchel, P., 2007. Modeling prosodic features with joint factor analysis for speaker verification. IEEE Trans. Audio, Speech Language Process. 15 (7), 2095–2103.

Dehak, N., Dehak, R., Kenny, P., Dumouchel, P., 2008. Comparison between factor analysis and GMM support vector machines for speaker verification. In: The Speaker and Language Recognition Workshop (Odyssey 2008), Stellenbosch, South Africa, January 2008. Paper 009.

Dehak, N., Kenny, P., Dehak, R., Glembek, O., Dumouchel, P., Burget, L., Hubeika, V., Castaldo, F., 2009. Support vector machines and joint factor analysis for speaker verification. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2009), Taipei, Taiwan, April 2009, pp. 4237–4240.

Deller, J., Hansen, J., Proakis, J., 2000. Discrete-Time Processing of Speech Signals, second ed. IEEE Press, New York.

Doddington, G., 2001. Speaker recognition based on idiolectal differences between speakers. In: Proc. Seventh European Conf. on Speech Communication and Technology (Eurospeech 2001), Aalborg, Denmark, September 2001, pp. 2521–2524.

Dunn, R., Quatieri, T., Reynolds, D., Campbell, J. 2001. Speaker recognition from coded speech and the effects of score normalization. In: Proc. 35th Asilomar Conf. on Signals, Systems and Computers, Vol. 2, Pacific Grove, California, USA, November 2001, pp. 1562–1567.

Espy-Wilson, C., Manocha, S., Vishnubhotla, S., 2006. A new set of features for text-independent speaker identification. In: Proc. Interspeech 2006 (ICSLP), Pittsburgh, Pennsylvania, USA, September 2006, pp. 1475–1478.

Ezzaidi, H., Rouat, J., O'Shaughnessy, D., 2001. Towards combining pitch and MFCC for speaker identification systems. In: Proc. Seventh European Conf. on Speech Communication and Technology (Eurospeech 2001), Aalborg, Denmark, September 2001, pp. 2825–2828.

Faltlhauser, R., Ruske, G., 2001. Improving speaker recognition performance using phonetically structured gaussian mixture models. In: Proc. Seventh European Conf. on Speech Communication and Technology (Eurospeech 2001), Aalborg, Denmark, September 2001, pp. 751–754.

Farrell, K., Mammone, R., Assaleh, K., 1994. Speaker recognition using neural networks and conventional classifiers. IEEE Trans. Speech Audio Process. 2 (1), 194–205.

Farrell, K., Ramachandran, R., Mammone, R., 1998. An analysis of data fusion methods for speaker verification. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 1998), Vol. 2, Seattle, Washington, USA, pp. 1129–1132.

Fauve, B., Matrouf, D., Scheffer, N., Bonastre, J.-F., Mason, J., 2007. State-of-the-art performance in text-independent speaker verification through open-source software. IEEE Trans. Audio, Speech Language Process. 15 (7), 1960–1968.

Fauve, B., Evans, N., Mason, J., 2008. Improving the performance of text-independent short duration SVM- and GMM-based speaker verification. In: The Speaker and Language Recognition Workshop (Odyssey 2008), Stellenbosch, South Africa, January 2008. Paper 018.

Ferrer, L., Shriberg, E., Kajarekar, S., Sönmez, K., 2007. Parameterization of prosodic feature distributions for SVM modeling in speaker recognition. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2007), Vol. 4, Honolulu, Hawaii, USA, April 2007, pp. 233–236.

Ferrer, L., Graciarena, M., Zymnis, A., Shriberg, E., 2008. System combination using auxiliary information for speaker verification. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (*ICASSP 2008*), Las Vegas, Nevada, March–April 2008, pp. 4853–4856.

Ferrer, L., Sönmez, K., Shriberg, E., 2008. An anticorrelation kernel for improved system combination in speaker verification. In: The Speaker and Language Recognition Workshop (Odyssey 2008), Stellenbosch, South Africa, January 2008. Paper 022.

Fredouille, C., Bonastre, J.-F., Merlin, T., 2000. AMIRAL: a block-segmental multirecognizer architecture for automatic speaker recognition. Digital Signal Process. 10 (1–3), 172–197.

Furui, S., 1981. Cepstral analysis technique for automatic speaker verification. IEEE Trans. Acoustics, Speech Signal Process. 29 (2), 254–272.

Furui, S., 1997. Recent advances in speaker recognition. Pattern Recognition Lett. 18 (9), 859–872.

Garcia-Romero, D., Fierrez-Aguilar, J., Gonzalez-Rodriguez, J., Ortega-Garcia, J., 2004. On the use of quality measures for text-independent speaker recognition. In: Proc. Speaker Odyssey: the Speaker Recognition Workshop (Odyssey 2004), Vol. 4, Toledo, Spain, May 2004, pp. 105–110.

Gersho, A., Gray, R., 1991. Vector Quantization and Signal Compression. Kluwer Academic Publishers, Boston.

Glembek, O., Burget, L., Dehak, N., Br ummer, N., Kenny, P., 2009. Comparison of scoring methods used in speaker recognition with joint factor analysis. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2009), Taipei, Taiwan, April 2009, pp. 4057–4060.

Gong, W.-G., Yang, L.-P., Chen, D., 2008. Pitch synchronous based feature extraction for noise-robust speaker verification. In: Proc. Image and Signal Processing (CISP 2008), Vol. 5, (May 2008), pp. 295–298.

Gonzalez-Rodriguez, J., Garcia-Gomar, D. G.-R. M., Ramos-Castro, D., Ortega-Garcia, J., 2003. Robust likelihood ratio estimation in Bayes-

ian forensic speaker recognition. In: Proc. 8th European Conf. on Speech Communication and Technology (Eurospeech 2003), Geneva, Switzerland, September 2003, pp. 693–696.

Gopalan, K., Anderson, T., Cupples, E., 1999. A comparison of speaker identification results using features based on cepstrum and Fourier–Bessel expansion. IEEE Trans. Speech Audio Process. 7 (3), 289–294.

Gudnason, J., Brookes, M., 2008. Voice source cepstrum coefficients for speaker identification. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2008), Las Vegas, Nevada, March–April 2008, pp. 4821–4824.

Gupta, S., Savic, M., 1992. Text-independent speaker verification based on broad phonetic segmentation of speech. Digital Signal Process. 2 (2), 69–79.

Hannani, A., Petrovska-Delacrétaz, D., Chollet, G., 2004. Linear and non-linear fusion of ALISP-based and GMM systems for text-independent speaker verification. In Proc. Speaker Odyssey: the Speaker Recognition Workshop (Odyssey 2004), Toledo, Spain, May 2004, pp. 111–116.

Hansen, E., Slyh, R., Anderson, T., 2004. Speaker recognition using phoneme-specific GMMs. In: Proc. Speaker Odyssey: the Speaker Recognition Workshop (Odyssey 2004), Toledo, Spain, May 2004, pp. 179–184.

Harrington, J., Cassidy, S., 1999. Techniques in Speech Acoustics. Kluwer Academic Publishers, Dordrecht.

Harris, F., 1978. On the use of windows for harmonic analysis with the discrete fourier transform. Proc. IEEE 66 (1), 51–84.

Hatch, A., Stolcke, A., 2006. Generalized linear kernels for one-versus-all classification: application to speaker recognition. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2006), Toulouse, France, May 2006, pp. 585–588.

Hatch, A., Stolcke, A., Peskin, B., 2005. Combining feature sets with support vector machines: application to speaker recognition. In: The 2005 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), November 2005, pp. 75–79.

Hatch, A., Kajarekar, S., Stolcke, A., 2006. Within-class covariance normalization for SVM-based speaker recognition. In: Proc. Interspeech 2006 (ICSLP), Pittsburgh, Pennsylvania, USA, September 2006, pp. 1471–1474.

Hautamäki, V., Tuononen, M., Niemi-Laitinen, T., Fränti, P., 2007. Improving speaker verification by periodicity based voice activity detection. In: Proc. 12th Internat. Conf. on Speech and Computer (SPECOM 2007), Moscow, Russia, October 2007, pp. 645–650.

Hautamäki, V., Kinnunen, T., Fränti, P., 2008a. Text-independent speaker recognition using graph matching. Pattern Recognition Lett. 29 (9), 1427–1432.

Hautamäki, V., Kinnunen, T., Kärkkäinen, I., Tuononen, M., Saastamoinen, J., Fränti, P., 2008b. Maximum *a posteriori* estimation of the centroid model for speaker verification. IEEE Signal Process. Lett. 15, 162–165.

Hébert, M., 2008. Text-dependent speaker recognition. In: Benesty, J., Sondhi, M., Huang, Y. (Eds.), Springer Handbook of Speech Processing. Springer-Verlag, Heidelberg, pp. 743–762.

Hébert, M., Heck, L., 2003. Phonetic class-based speaker verification. In: Proc. Eighth European Conf. on Speech Communication and Technology (Eurospeech 2003), Geneva, Switzerland, September 2003, pp. 1665–1668.

Heck, L., Genoud, D., 2002. Combining speaker and speech recognition systems. In: Proc. Internat. Conf. on Spoken Language Processing (ICSLP 2002), Denver, Colorado, USA, September 2002, pp. 1369–1372.

Heck, L., and Weintraub, M. 1997. Handset-dependent background models for robust text-independent speaker recognition. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing* (*ICASSP 1997*) (Munich, Germany, April 1997), pp. 1071–1074.

Heck, L., Konig, Y., Sönmez, M., Weintraub, M., 2000. Robustness to telephone handset distortion in speaker recognition by discriminative feature design. Speech Comm. 31, 181–192.

Hedge, R., Murthy, H., Rao, G., 2004. Application of the modified group delay function to speaker identification and discrimination. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2004), Vol. 1, Montreal, Canada, May 2004, pp. 517–520.

He, J., Liu, L., Palm, G., 1999. A discriminative training algorithm for VQ-based speaker identification. IEEE Trans. Speech Audio Process. 7 (3), 353–356.

Hermansky, H., 1990. Perceptual linear prediction (PLP) analysis for speech. J. Acoust. Soc. Amer. 87, 1738–1752.

Hermansky, H., 1998. Should recognizers have ears?. Speech Comm. 25 (1–3) 3–27.

Hermansky, H., Morgan, N., 1994. RASTA processing of speech. IEEE Trans. Speech Audio Process. 2 (4), 578–589.

Hess, W., 1983. Pitch Determination of Speech Signals: Algorithms and Devices. Springer-Verlag, Berlin.

Higgins, A., Bahler, L., Porter, J., 1991. Speaker verification using randomized phrase prompting. Digital Signal Process. 1, 89–106.

Huang, X., Acero, A., Hon, H.-W., 2001. Spoken Language Processing: a Guide to Theory, Algorithm, and System Development. Prentice-Hall, New Jersey.

Imperl, B., Kacic, Z., Horvat, B., 1997. A study of harmonic features for the speaker recognition. Speech Comm. 22 (4), 385–402.

Jain, A., Duin, R., Mao, J., 2000. Statistical pattern recognition: a review. IEEE Trans. Pattern Anal. Machine Intell. 22 (1), 4–37.

Jang, G.-J., Lee, T.-W., Oh, Y.-H., 2002. Learning statistically efficient features for speaker recognition. Neurocomputing 49, 329–348.

Jin, Q., Schultz, T., Waibel, A., 2002. Speaker identification using multilingual phone strings. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2002), Vol. 1, Orlando, Florida, USA, May 2002, pp. 145–148.

Kajarekar, S., Hermansky, H., 2001. Speaker verification based on broad phonetic categories. In: Proc. Speaker Odyssey: the Speaker Recognition Workshop (Odyssey 2001), Crete, Greece, June 2001, pp. 201–206.

Karam, Z., Campbell, W., 2007. A new kernel for SVM MLLR based speaker recognition. In: Proc. Interspeech 2007 (ICSLP), Antwerp, Belgium, August 2007, pp. 290–293.

Karpov, E., Kinnunen, T., Fränti, P., 2004. Symmetric distortion measure for speaker recognition. In: Proc. Ninth Internat. Conf. on Speech and Computer (SPECOM 2004), St. Petersburg, Russia, September 2004, pp. 366–370.

Kenny, P., 2006. Joint factor analysis of speaker and session variability: theory and algorithms. Technical Report CRIM-06/08-14.

Kenny, P., Boulianne, G., Ouellet, P., Dumouchel, P., 2007. Speaker and session variability in GMM-based speaker verification. IEEE Trans. Audio, Speech Language Process. 15 (4), 1448–1460.

Kenny, P., Ouellet, P., Dehak, N., Gupta, V., Dumouchel, P., 2008. A study of inter-speaker variability in speaker verification. IEEE Trans. Audio, Speech Language Process. 16 (5), 980–988.

Kinnunen, T., 2002. Designing a speaker-discriminative adaptive filter bank for speaker recognition. In: Proc. Internat. Conf. on Spoken Language Processing (ICSLP 2002), Denver, Colorado, USA, September 2002, pp. 2325–2328.

Kinnunen, T., 2004. Spectral Features for Automatic Text-Independent Speaker Recognition. Licentiate's Thesis, University of Joensuu, Department of Computer Science, Joensuu, Finland.

Kinnunen, T., 2006. Joint acoustic-modulation frequency for speaker recognition. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2006), Vol. I, Toulouse, France, 2006, pp. 665–668.

Kinnunen, T., Alku, P., 2009. On separating glottal source and vocal tract information in telephony speaker verification. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2009), Taipei, Taiwan, April 2009, pp. 4545–4548.

Kinnunen, T., González-Hautamäki, R., 2005. Long-term f_0 modeling for text-independent speaker recognition. In: Proc. 10th Internat. Conf. on Speech and Computer (SPECOM'2005), Patras, Greece, October 2005, pp. 567–570.

Kinnunen, T., KilpelSinen, T., Fränti, P., 2000. Comparison of clustering algorithms in speaker identification. In: Proc. IASTED Internat. Conf. on Signal Processing and Communications (SPC 2000), Marbella, Spain, September 2000, pp. 222–227.

Kinnunen, T., Hautamäki, V., Fränti, P., 2004. Fusion of spectral feature sets for accurate speaker identification. In: Proc. Ninth Internat. Conf. on Speech and Computer (SPECOM 2004), St. Petersburg, Russia, September 2004, pp. 361–365.

Kinnunen, T., Hautamäki, V., Fränti, P., 2006. On the use of long-term average spectrum in automatic speaker recognition. In: 5th Internat. Symposium on Chinese Spoken Language Processing (ISCSLP'06), Singapore, December 2006, pp. 559–567.

Kinnunen, T., Karpov, E., Fränti, P., 2006b. Real-time speaker identification and verification. IEEE Trans. Audio, Speech Language Process. 14 (1), 277–288.

Kinnunen, T., Koh, C., Wang, L., Li, H., Chng, E., 2006. Temporal discrete cosine transform: Towards longer term temporal features for speaker verification. In: Proc. Fifth Internat. Symposium on Chinese Spoken Language Processing (ISCSLP 2006), Singapore, December 2006, pp. 547–558.

Kinnunen, T., Zhang, B., Zhu, J., Wang, Y., 2007. Speaker verification with adaptive spectral subband centroids. In: Proc. Internat. Conf. on Biometrics (ICB 2007), Seoul, Korea, August 2007, pp. 58–66.

Kinnunen, T., Lee, K.-A., Li, H. 2008. Dimension reduction of the modulation spectrogram for speaker verification. In: The Speaker and Language Recognition Workshop (Odyssey 2008), Stellenbosch, South Africa, January 2008.

Kinnunen, T., Saastamoinen, J., Hautamäki, V., Vinni, M., Fränti, P., 2009. Comparative evaluation of maximum *a posteriori* vector quantization and Gaussian mixture models in speaker verification. Pattern Recognition Lett. 30 (4), 341–347.

Kitamura, T., 2008. Acoustic analysis of imitated voice produced by a professional impersonator. In: Proc. Interspeech 2008, September 2008, pp. 813–816.

Kittler, J., Hatef, M., Duin, R., Matas, J., 1998. On combining classifiers. IEEE Trans. Pattern Anal. Machine Intell. 20 (3), 226–239.

Kolano, G., Regel-Brietzmann, P., 1999. Combination of vector quantization and Gaussian mixture models for speaker verification. In: Proc. Sixth European Conf. on Speech Communication and Technology (Eurospeech 1999), Budapest, Hungary, September 1999, pp. 1203–1206.

Kryszczuk, K., Richiardi, J., Prodanov, P., Drygajlo, A., 2007. Reliability-based decision fusion in multimodal biometric verification systems. EURASIP J. Adv. Signal Process. 1. Article ID 86572.

Lapidot, I., Guterman, H., Cohen, A., 2002. Unsupervised speaker recognition based on competition between self-organizing maps. IEEE Trans. Neural Networks 13, 877–887.

Laskowski, K., Jin, Q., 2009. Modeling instantaneous intonation for speaker identification using the fundamental frequency variation spectrum. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2009), Taipei, Taiwan, April 2009, pp. 4541–4544.

Lee, K.-A., You, C., Li, H., Kinnunen, T., 2007. A GMM-based probabilistic sequence kernel for speaker verification. In: Proc. Interspeech 2007 (ICSLP), Antwerp, Belgium, August 2007, pp. 294–297.

Lee, K., You, C., Li, H., Kinnunen, T., Zhu, D., 2008. Characterizing speech utterances for speaker verification with sequence kernel SVM. In: Proc. Ninth Interspeech (Interspeech 2008), Brisbane, Australia, September 2008, pp. 1397–1400.

Leeuwen, D., Martin, A., Przybocki, M., Bouten, J., 2006. NIST and NFI-TNO evaluations of automatic speaker recognition. Comput. Speech Lang. 20, 128–158.

Leggetter, C., Woodland, P., 1995. Maximum likelihood linear regression for speaker adaptation of continuous density HMMs. Comput. Speech Lang. 9, 171–185.

Lei, H., Mirghafori, N., 2007. Word-conditioned HMM supervectors for speaker recognition. In: Proc. Interspeech 2007 (ICSLP), Antwerp, Belgium, August 2007, pp. 746–749.

Leung, K., Mak, M., Siu, M., Kung, S., 2006. Adaptive articulatory feature-based conditional pronunciation modeling for speaker verification. Speech Comm. 48 (1), 71–84.

Li, K.-P., Porter, J., 1988. Normalizations and selection of speech segments for speaker recognition scoring. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 1988), New York, USA, April 1988, pp. 595–598.

Li, H., Ma, B., Lee, K.-A., Sun, H., Zhu, D., Sim, K., You, C., Tong, R., KärkkSinen, I., Huang, C.-L., Pervouchine, V., Guo, W., Li, Y., Dai, L., Nosratighods, M., Tharmarajah, T., Epps, J., Ambikairajah, E., Chng, E.-S., Schultz, T., Jin, Q., 2009. The I4U system in NIST 2008 speaker recognition evaluation. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2009), Taipei, Taiwan, April 2009, pp. 4201–4204.

Linde, Y., Buzo, A., Gray, R., 1980. An algorithm for vector quantizer design. IEEE Trans. Comm. 28 (1), 84–95.

Longworth, C., Gales, M., 2007. Combining derivative and parametric kernels for speaker verification. IEEE Trans. Audio, Speech Language Process. 6 (1), 1–10.

Louradour, J., Daoudi, K., 2005. SVM speaker verification using a new sequence kernel. In: Proc. 13th European Conf. on Signal Processing (EUSIPCO 2005), Antalya, Turkey, September 2005.

Louradour, J., Daoudi, K., André-Obrecht, R., 2005. Discriminative power of transient frames in speaker recognition. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2005), Vol. 1, Philadelphia, USA, 2005, pp. 613–616.

Lu, X., Dang, J., 2007. An investigation of dependencies between frequency components and speaker characteristics for text-independent speaker identification. Speech Comm. 50 (4), 312–322.

Ma, B., Zhu, D., Tong, R., 2006. Chinese dialect identification using tone features based on pitch flux. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2006), Vol. 1, Toulouse, France, May 2006, pp. 1029–1032.

Ma, B., Zhu, D., Tong, R., Li, H., 2006. Speaker cluster based GMM tokenization for speaker recognition. In: Proc. Interspeech 2006 (ICSLP), Pittsburgh, Pennsylvania, USA, September 2006, pp. 505–508.

Ma, B., Li, H., Tong, R., 2007. Spoken language recognition with ensemble classifiers. IEEE Trans. Audio, Speech Language Process. 15 (7), 2053–2062.

Magrin-Chagnolleau, I., Durou, G., Bimbot, F., 2002. Application of time–frequency principal component analysis to text-independent speaker identification. IEEE Trans. Speech Audio Process. 10 (6), 371–378.

Mak, M.-W., Tsang, C.-L., 2004. Stochastic feature transformation with divergence-based out-of-handset rejection for robust speaker verification. EURASIP J. Appl. Signal Process. 4, 452–465.

Mak, M.-W., Cheung, M., Kung, S., 2003. Robust speaker verification from GSM-transcoded speech based on decision fusion and feature transformation. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2003), Vol. 2, Hong Kong, China, April 2003, pp. 745–748.

Mak, M.-W., Hsiao, R., Mak, B., 2006. A comparison of various adaptation methods for speaker verification with limited enrollment data. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2006), Vol. 1, Toulouse, France, May 2006, pp. 929–932.

Makhoul, J., 1975. Linear prediction: a tutorial review. Proc. IEEE 64 (4), 561–580.

Malayath, N., Hermansky, H., Kajarekar, S., Yegnanarayana, B., 2000. Data-driven temporal filters and alternatives to GMM in speaker verification. Digital Signal Process. 10 (1–3), 55–74.

Mami, Y., Charlet, D., 2006. Speaker recognition by location in the space of reference speakers. Speech Comm. 48 (2), 127–411.

Mammone, R., Zhang, X., Ramachandran, R., 1996. Robust speaker recognition: a feature based approach. IEEE Signal Process. Mag. 13 (5), 58–71.

Mariéthoz, J., Bengio, S., 2002. A comparative study of adaptation methods for speaker verification. In: Proc. Internat. Conf. on Spoken Language Processing (ICSLP 2002), Denver, Colorado, USA, September 2002, pp. 581–584.

Markel, J., Oshika, B., Gray Jr., A.H., 1977. Long-term feature averaging for speaker recognition. IEEE Trans. Acoustics, Speech, Signal Process. 25 (4), 330–337.

Martin, A., Doddington, G., Kamm, T., Ordowski, M., Przybocki, M., 1997. The DET curve in assessment of detection task performance. In: Proc. Fifth European Conf. on Speech Communication and Technology (Eurospeech 1997), Rhodos, Greece, September 1997, pp. 1895–1898.

Mary, L., Yegnanarayana, B., 2006. Prosodic features for speaker verification. In: Proc. Interspeech 2006 (ICSLP), Pittsburgh, Pennsylvania, USA, September 2006, pp. 917–920.

Mary, L., Yegnanarayana, B., 2008. Extraction and representation of prosodic features for language and speaker recognition. Speech Comm. 50 (10), 782–796.

Mason, M., Vogt, R., Baker, B., Sridharan, S., 2005. Data-driven clustering for blind feature mapping in speaker verification. In: Proc. Interspeech 2005, Lisboa, Portugal, September 2005, pp. 3109–3112.

McLaughlin, J., Reynolds, D., Gleason, T., 1999. A study of computation speed-ups of the GMM–UBM speaker recognition system. In: Proc. Sixth European Conf. on Speech Communication and Technology (Eurospeech 1999), Budapest, Hungary, September 1999, pp. 1215–1218.

Misra, H., Ikbal, S., Yegnanarayana, B., 2003. Speaker-specific mapping for text-independent speaker recognition. Speech Comm. 39 (3–4), 301–310.

Miyajima, C., Watanabe, H., Tokuda, K., Kitamura, T., Katagiri, S., 2001. A new approach to designing a feature extractor in speaker identification based on discriminative feature extraction. Speech Comm. 35, 203–218.

Moonasar, V., Venayagamoorthy, G., 2001. A committee of neural networks for automatic speaker recognition (ASR) systems. In: Proc. Internat. Joint Conf. on Neural Networks (IJCNN 2001), Washington, DC, USA, July 2001, pp. 2936–2940.

Müller, C. (Ed.), 2007a. Speaker Classification I: Fundamentals, Features, and Methods. Lecture Notes in Computer Science, Vol. 4343. Springer.

Müller, C. (Ed.), 2007b. Speaker Classification II: Selected Projects. Lecture Notes in Computer Science, Vol. 4441. Springer.

Müller, K.-R., Mika, S., Rätsch, G., Tsuda, K., Schölkopf, B., 2001. An introduction to kernel-based learning algorithms. IEEE Trans. Neural Networks 12 (2), 181–201.

Murty, K., Yegnanarayana, B., 2006. Combining evidence from residual phase and MFCC features for speaker recognition. IEEE Signal Process. Lett. 13 (1), 52–55.

Naik, J., Netsch, L., and Doddington, G. 1989. Speaker verification over long distance telephone lines. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing* (*ICASSP 1989*) (Glasgow, May 1989), pp. 524–527.

Nakasone, H., Mimikopoulos, M., Beck, S., Mathur, S., 2004. Pitch synchronized speech processing (PSSP) for speaker recognition. In: Proc. Speaker Odyssey: the Speaker Recognition Workshop (Odyssey 2004), Toledo, Spain, May 2004, pp. 251–256.

Ney, H., Martin, S., Wessel, F., 1997. Statistical language modeling using leaving-one-out. In: Young, S., Bloothooft, G. (Eds.), Corpus-based Methods in Language and Speech Processing. Kluwer Academic Publishers, pp. 174–207.

Niemi-Laitinen, T., Saastamoinen, J., Kinnunen, T., Fränti, P., 2005. Applying MFCC-based automatic speaker recognition to GSM and forensic data. In: Proc. Second Baltic Conf. on Human Language Technologies (HLT'2005), Tallinn, Estonia, April 2005, pp. 317–322.

NIST 2008 SRE results page, September 2008. <http://www.nist.gov/speech/tests/sre/2008/official_results/index.html>.

Nolan, F., 1983. The Phonetic Bases of Speaker Recognition. Cambridge University Press, Cambridge.

Oppenheim, A., Schafer, R., Buck, J., 1999. Discrete-Time Signal Processing, second ed. Prentice-Hall, 1999.

Orman, D., Arslan, L., 2001. Frequency analysis of speaker identification. In: Proc. Speaker Odyssey: the Speaker Recognition Workshop (Odyssey 2001), Crete, Greece, June 2001, pp. 219–222.

Paliwal, K., Alsteris, L., 2003. Usefulness of phase spectrum in human speech perception. In: Proc. Eighth European Conf. on Speech Communication and Technology (Eurospeech 2003), Geneva, Switzerland, September 2003, pp. 2117–2120.

Park, A., Hazen, T., 2002. ASR dependent techniques for speaker identification. In: Proc. Internat. Conf. on Spoken Language Processing (ICSLP 2002), Denver, Colorado, USA, September 2002, pp. 1337–1340.

Pelecanos, J., Sridharan, S., 2001. Feature warping for robust speaker verification. In: Proc. Speaker Odyssey: the Speaker Recognition Workshop (Odyssey 2001), Crete, Greece, June 2001, pp. 213–218.

Pelecanos, J., Myers, S., Sridharan, S., Chandran, V., 2000. Vector quantization based Gaussian modeling for speaker verification. In: Proc. Internat. Conf. on Pattern Recognition (ICPR 2000), Barcelona, Spain, September 2000, pp. 3298–3301.

Pellom, B., Hansen, J., 1998. An efficient scoring algorithm for gaussian mixture model based speaker identification. IEEE Signal Process. Lett. 5 (11), 281–284.

Pellom, B.L., Hansen, J.H.L., 1999. An experimental study of speaker verification sensitivity to computer voice-altered imposters. In: Proc. of the IEEE 1999 Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 1999), Vol. 2, Phoenix, AZ, USA, March 1999, pp. 837–840.

Pfister, B., Beutler, R., 2003. Estimating the weight of evidence in forensic speaker verification. In: Proc. Eighth European Conf. on Speech Communication and Technology (Eurospeech 2003), Geneva, Switzerland, September 2003, pp. 701–704.

Plumpe, M., Quatieri, T., Reynolds, D., 1999. Modeling of the glottal flow derivative waveform with application to speaker identification. IEEE Trans. Speech Audio Process. 7 (5), 569–586.

Poh, N., Bengio, S., 2004. Why do multi-stream, multi-band and multi-modal approaches work on biometric user authentication tasks? In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2004), Vol. 5, Montreal, Canada, May 2004, pp. 893–896.

Prasanna, S., Gupta, C., Yegnanarayana, B., 2006. Extraction of speaker-specific excitation information from linear prediction residual of speech. Speech Comm. 48, 1243–1261.

Przybocki, M.A., Martin, A., Le, A., 2007. NIST speaker recognition evaluations utilizing the mixer corpora – 2004, 2005, 2006. IEEE Trans. Audio, Speech Language Process. 15 (7), 1951–1959.

Rabiner, L., Juang, B.-H., 1993. Fundamentals of Speech Recognition. Prentice-Hall, Englewood Cliffs, New Jersey.

Ramachandran, R., Farrell, K., Ramachandran, R., Mammone, R., 2002. Speaker recognition – general classifier approaches and data fusion methods. Pattern Recognition 35, 2801–2821.

Ramirez, J., Segura, J., Benítez, C., de la Torre, A., Rubio, A., 2004. Efficient voice activity detection algorithms using long-term speech information. Speech Comm. 42 (3–4), 271–287.

Ramos-Castro, D., Fierrez-Aguilar, J., Gonzalez-Rodriguez, J., Ortega-Garcia, J., 2007. Speaker verification using speaker- and test-dependent fast score normalization. Pattern Recognition Lett. 28 (1), 90–98.

Reynolds, D., 1995. Speaker identification and verification using Gaussian mixture speaker models. Speech Comm. 17, 91–108.

Reynolds, D., 2003. Channel robust speaker verification via feature mapping. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2003), Vol. 2, Hong Kong, China, April 2003, pp. 53–56.

Reynolds, D., Rose, R., 1995. Robust text-independent speaker identification using Gaussian mixture speaker models. IEEE Trans. Speech Audio Process. 3, 72–83.

Reynolds, D., Quatieri, T., Dunn, R., 2000. Speaker verification using adapted gaussian mixture models. Digital Signal Process. 10 (1), 19–41.

Reynolds, D., Andrews, W., Campbell, J., Navratil, J., Peskin, B., Adami, A., Jin, Q., Klusacek, D., Abramson, J., Mihaescu, R., Godfrey, J.,

Jones, D., Xiang, B., 2003. The SuperSID project: exploiting high-level information for high-accuracy speaker recognition. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2003). Hong Kong, China, April 2003, pp. 784–787.

Reynolds, D., Campbell, W., Gleason, T., Quillen, C., Sturim, D., Torres-Carrasquillo, P., Adami, A., 2005. The 2004 MIT Lincoln laboratory speaker recognition system. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2005), Vol. 1, Philadelphia, USA, 2005, pp. 177–180.

Roch, M., 2006. Gaussian-selection-based non-optimal search for speaker identification. Speech Commu. 48, 85–95.

Rodríguez-Liñares, L., García-Mateo, C., Alba-Castro, J., 2003. On combining classifiers for speaker authentication. Pattern Recognition 36 (2), 347–359.

Rose, P., 2002. Forensic Speaker Identification. Taylor & Francis, London.

Saastamoinen, J., Karpov, E., Hautamäki, V., Fränti, P., 2005. Accuracy of MFCC based speaker recognition in series 60 device. EURASIP J. Appl. Signal Process. 17, 2816–2827.

Saeidi, R., Mohammadi, H., Ganchev, T., Rodman, R.D., 2009. Particle swarm optimization for sorted adapted gaussian mixture models. IEEE Trans. Audio, Speech Language Process. 17 (2), 344–353.

Shriberg, E., Ferrer, L., Kajarekar, S., Venkataraman, A., Stolcke, A., 2005. Modeling prosodic feature sequences for speaker recognition. Speech Comm. 46 (3–4), 455–472.

Sivakumaran, P., Ariyaeeinia, A., Loomes, M., 2003a. Sub-band based text-dependent speaker verification. Speech Comm. 41, 485–509.

Sivakumaran, P., Fortuna, J., Ariyaeeinia, A., 2003. Score normalization applied to open-set, text-independent speaker identification. In: Proc. Eighth European Conf. on Speech Communication and Technology (Eurospeech 2003), Geneva, Switzerland, September 2003, pp. 2669–2672.

Slomka, S., Sridharan, S., Chandran, V., 1998. A comparison of fusion techniques in mel-cepstral based speaker identification. In: Proc. Internat. Conf. on Spoken Language Processing (ICSLP 1998), Sydney, Australia, November 1998, pp. 225–228.

Slyh, R., Hansen, E., Anderson, T., 2004. Glottal modeling and closed-phase analysis for speaker recognition. In: Proc. Speaker Odyssey: the Speaker Recognition Workshop (Odyssey 2004), Toledo, Spain, May 2004, pp. 315–322.

Solewicz, Y., Koppel, M., 2007. Using post-classifiers to enhance fusion of low- and high-level speaker recognition. IEEE Trans. Audio, Speech Language Process. 15 (7), 2063–2071.

Solomonoff, A., Campbell, W., Boardman, I., 2005. Advances in channel compensation for SVM speaker recognition. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2005), Philadelphia, USA, March 2005, pp. 629–632.

Sönmez, M., Heck, L., Weintraub, M., Shriberg, E., 1997. A lognormal tied mixture model of pitch for prosody-based speaker recognition. In: Proc. Fifth European Conf. on Speech Communication and Technology (Eurospeech 1997), Rhodos, Greece, September 1997, pp. 1391–1394.

Sönmez, K., Shriberg, E., Heck, L., Weintraub, M., 1998. Modeling dynamic prosodic variation for speaker verification. In: Proc. Internat. Conf. on Spoken Language Processing (ICSLP 1998), Sydney, Australia, November 1998, pp. 3189–3192.

Soong, F., Rosenberg, A., 1988. On the use of instantaneous and transitional spectral information in speaker recognition. IEEE Trans. Acoustics, Speech Signal Process. 36 (6), 871–879.

Soong, F.K., Rosenberg, A.E., Juang, B.-H., Rabiner, L.R., 1987. A vector quantization approach to speaker recognition. AT & T Technical J. 66, 14–26.

Stolcke, A., Kajarekar, S., Ferrer, L., Shriberg, E., 2007. Speaker recognition with session variability normalization based on MLLR adaptation transforms. IEEE Trans. Audio, Speech Language Process. 15 (7), 1987–1998.

Stolcke, A., Kajarekar, S., Ferrer, L., 2008. Nonparametric feature normalization for SVM-based speaker verification. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2008), Las Vegas, Nevada, April 2008, pp. 1577–1580.

Sturim, D., Reynolds, D., 2005. Speaker adaptive cohort selection for Tnorm in text-independent speaker verification. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2005), Vol. 1, Philadelphia, USA, March 2005, pp. 741–744.

Sturim, D., Reynolds, D., Singer, E., Campbell, J., 2001. Speaker indexing in large audio databases using anchor models. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2001), Vol. 1, Salt Lake City, Utah, USA, May 2001, pp. 429–432.

Teunen, R., Shahshahani, B., Heck, L., 2000. A model-based transformational approach to robust speaker recognition. In: Proc. Internat. Conf. on Spoken Language Processing (ICSLP 2000), Vol. 2, Beijing, China, October 2000, pp. 495–498.

Thévenaz, P., Hügli, H., 1995. Usefulness of the LPC-residue in text-independent speaker verification. Speech Comm. 17 (1–2), 145–157.

Thian, N., Sanderson, C., Bengio, S., 2004. Spectral subband centroids as complementary features for speaker authentication. In: Proc. First Internat. Conf. on Biometric Authentication (ICBA 2004), Hong Kong, China, July 2004, pp. 631–639.

Thiruvaran, T., Ambikairajah, E., Epps, J., 2008a. Extraction of FM components from speech signals using all-pole model. Electronics Lett. 44 (6).

Thiruvaran, T., Ambikairajah, E., Epps, J., 2008. FM features for automatic forensic speaker recognition. In: Proc. Interspeech 2008, Brisbane, Australia, September 2008, pp. 1497–1500.

Tong, R., Ma, B., Lee, K., You, C., Zhu, D., Kinnunen, T., Sun, H., Dong, M., Chng, E., Li, H., 2006. Fusion of acoustic and tokenization features for speaker recognition. In: Fifth Internat. Symposium on Chinese Spoken Language Processing (ISCSLP 2006), Singapore, December 2006, pp. 494–505.

Torres-Carrasquillo, P., Reynolds, D., Deller Jr., J.D., 2002. Language identification using Gaussian mixture model tokenization. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2002), Vol. 1, Orlando, Florida, USA, May 2002, pp. 757–760.

Tranter, S., Reynolds, D.A., 2006. An overview of automatic speaker diarization systems. IEEE Trans. Audio, Speech Language Process. 14 (5), 1557–1565.

Tydlitat, B., Navratil, J., Pelecanos, J., Ramaswamy, G., 2007. Text-independent speaker verification in embedded environments. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2007), Vol. 4, Honolulu, Hawaii, April 2007, pp. 293–296.

Viikki, O., Laurila, K., 1998. Cepstral domain segmental feature vector normalization for noise robust speech recognition. Speech Comm. 25, 133–147.

Vogt, R., Sridharan, S., 2008. Explicit modeling of session variability for speaker verification. Comput. Speech Lang. 22 (1), 17–38.

Vogt, R., Baker, B., Sridharan, S., 2005. Modelling session variability in text-independent speaker verification. In: Proc. Interspeech 2005, Lisboa, Portugal, September 2005, pp. 3117–3120.

Vogt, R., Kajarekar, S., Sridharan, S., 2008. Discriminant NAP for SVM speaker recognition. In: The Speaker and Language Recognition Workshop (Odyssey 2008), Stellenbosch, South Africa, January 2008. Paper 010.

Wan, V., Renals, S., 2005. Speaker verification using sequence discriminant support vector machines. IEEE Trans. Speech Audio Process. 13 (2), 203–210.

Wildermoth, B., and Paliwal, K. 2000. Use of voicing and pitch information for speaker recognition. In: Proc. Eighth Australian Internat. Conf. on Speech Science and Technology, Canberra, December 2000, pp. 324–328.

Wolf, J., 1972. Efficient acoustic parameters for speaker recognition. J. Acoust. Soc. Amer. 51 (6), 2044–2056 (Part 2).

Xiang, B., 2003. Text-independent speaker verification with dynamic trajectory model. IEEE Signal Process. Lett. 10, 141–143.

Xiang, B., Berger, T., 2003. Efficient text-independent speaker verification with structural gaussian mixture models and neural network. IEEE Trans. Speech Audio Process. 11, 447–456.

Xiang, B., Chaudhari, U., Navratil, J., Ramaswamy, G., Gopinath, R., 2002. Short-time Gaussianization for robust speaker verification. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2002), Vol. 1, Orlando, Florida, USA, May 2002, pp. 681–684.

Xiong, Z., Zheng, T., Song, Z., Soong, F., Wu, W., 2006. A tree-based kernel selection approach to efficient Gaussian mixture model-universal background model based speaker identification. Speech Comm. 48, 1273–1282.

Yegnanarayana, B., Kishore, S., 2002. AANN: an alternative to GMM for pattern recognition. Neural Networks 15, 459–469.

You, C., Lee, K., Li, H., 2009. An SVM kernel with GMM-supervector based on the Bhattacharyya distance for speaker recognition. IEEE Signal Process. Lett. 16 (1), 49–52.

Yuo, K.-H., Wang, H.-C., 1999. Joint estimation of feature transformation parameters and Gaussian mixture model for speaker identification. Speech Comm. 28 (3), 227–241.

Zheng, N., Lee, T., Ching, P., 2007. Integration of complementary acoustic features for speaker recognition. IEEE Signal Process. Lett. 14 (3), 181–184.

Zhu, D., Ma, B., Li, H., Huo, Q., 2007. A generalized feature transformation approach for channel robust speaker verification. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2007), Vol. 4, Honolulu, Hawaii, April 2007, pp. 61–64.

Zhu, D., Ma, B., Li, H., 2008. Using MAP estimation of feature transformation for speaker recognition. In: Proc. Interspeech 2008, Brisbane, Australia, September 2008.

Zhu, D., Ma, B., Li, H., 2009. Joint MAP adaptation of feature transformation and gaussian mixture model for speaker recognition. In: Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2009), Taipei, Taiwan, April 2009, pp. 4045–4048.

Zilca, R., 2002. Text-independent speaker verification using utterance level scoring and covariance modeling. IEEE Trans. Speech Audio Process. 10 (6), 363–370.

Zilca, R., Kingsbury, B., Navrátil, J., Ramaswamy, G., 2006. Pseudo pitch synchronous analysis of speech with applications to speaker recognition. IEEE Trans. Audio, Speech Language Process. 14 (2), 467–478.

Zissman, M., 1996. Comparison of four approaches to automatic language identification of telephone speech. IEEE Trans. Speech Audio Process. 4 (1), 31–44.