36. Overview of Speaker Recognition

A. E. Rosenberg, F. Bimbot, S. Parthasarathy

An introduction to automatic speaker recognition is presented in this chapter. The identifying characteristics of a person's voice that make it possible to automatically identify a speaker are discussed. Subtasks such as speaker identification, verification, and detection are described. An overview of the techniques used to build speaker models as well as issues related to system performance are presented. Finally, a few selected applications of speaker recognition are introduced to demonstrate the wide range of applications of speaker recognition technologies. Details of text-dependent and text-independent speaker recognition and their applications are covered in the following two chapters.

36.1 Speaker Recognition

36.1.1 Personal Identity Characteristics

Human beings have many characteristics that make it possible to distinguish one individual from another. Some individuating characteristics can be perceived very readily such as facial features and vocal qualities and behavior. Others, such as fingerprints, iris patterns, and DNA structure are not readily perceived and require measurements, often quite complex measurements, to capture distinguishing characteristics. In recent years biometrics has emerged as an applied scientific discipline with the objective of automatically capturing personal identifying characteristics and using the measurements for security, surveillance, and forensic applications [36.1]. Typical applications using biometrics secure transactions, information, and premises to authorized individuals. In surveillance applications, the goal is to detect and track a target individual among a set of nontarget individuals. In forensic applications a sample of biometric measurements is obtained from an unknown individual, the perpetrator. The task is to compare this sample with a database of similar measurements from known individuals to find a match.

Many personal identifying characteristics are based on physiological properties, others on behavior, and some combine physiological and behavioral properties. From the point of view of using personal identity characteristics as a biometric for security, physiological characteristics may offer more intrinsic security since they are not subject to the kinds of voluntary variations found in behavioral features. Voice is an example of a biometric that combines physiological and behav-

ioral characteristics. Voice is attractive as a biometric for many reasons. It can be captured non-intrusively and conveniently with simple transducers and recording devices. It is particularly useful for remote-access transactions over telecommunication networks. A drawback is that voice is subject to many sources of variability, including behavioral variability, both voluntary and involuntary. An example of involuntary variability is a speaker's inability to repeat utterances precisely the same way. Another example is the spectral changes that occur when speakers vary their vocal effort as background noise increases. Voluntary variability is an issue when speakers attempt to disguise their voices. Other sources of variability include physical voice variations due to respiratory infections and congestion. External sources of variability are especially problematic, including variations in background noise, and transmission and recording characteristics.

36.1.2 Speaker Recognition Definitions

Different tasks are defined under the general heading of speaker recognition. They differ mainly with respect to the kind of decision that is required for each task. In speaker identification a voice sample from an unknown speaker is compared with a set of labeled speaker models. When it is known that the set of speaker models includes all speakers of interest the task is referred to as *closed-set* identification. The label of the best matching speaker is taken to be the identified speaker. Most speaker identification applications are *open-set*, meaning that it is possible that the unknown speaker is not included in the set of speaker models. In this case, if no satisfactory match is obtained, a *no-match* decision is provided.

In a speaker verification trial an identity claim is provided or asserted along with the voice sample. In this case, the unknown voice sample is compared only with the speaker model whose label corresponds to the identity claim. If the quality of the comparison is satisfactory, the identity claim is accepted; otherwise the claim is rejected. Speaker verification is a special case of open-set speaker identification with a one-speaker target set. The speaker verification decision mode is intrinsic to most access control applications. In these applications, it is assumed that the claimant will respond to prompts cooperatively.

It can readily be seen that in the speaker identification task performance degrades as the number of speaker models and the number of comparisons increases. In a speaker verification trial only one comparison is required, so speaker verification performance is independent of the size of the speaker population.

A third speaker recognition task has been defined in recent years in National Institute of Standards and Technology (NIST) speaker recognition evaluations; it is generally referred to as speaker detection [36.2, 3]. The NIST task is an open-set identification decision associated exclusively with conversational speech. In this task an unknown voice sample is provided and the task is to determine whether or not one of a specified set of known speakers is present in the sample. A complicating factor for this task is that the unknown sample may contain speech from more than one speaker, such as in the summed two sides of a telephone conversation. In this case, an additional task called speaker tracking is defined, in which it is required to determine the intervals in the test sample during which the detected speaker is talking. In other applications where the speech samples are multispeaker, speaker tracking has also been referred to as speaker segmentation, speaker indexing, and speaker diarization [36.4–10]. It is possible to cast the speaker segmentation task as an acoustical change detection task without creating models. The time instants where a significant acoustic change occurs are assumed to be the boundaries between different speaker segments. In this case, in the absence of speaker models, speaker segmentation would not be considered a speaker recognition task. However, in most reported approaches to this task some sort of speaker modeling does take place. The task usually includes labeling the speaker segments. In this case the task falls unambiguously under the speaker recognition heading.

In addition to decision modes, speaker recognition tasks can be categorized by the kind of speech that is input. If the speaker is prompted or expected to provide a known text and if speaker models have been trained explicitly for this text, the input mode is said to be text dependent. If, on the other hand, the speaker cannot be expected to utter specified texts the input mode is text independent. In this case speaker models are not trained on explicit texts.

36.1.3 Bases for Speaker Recognition

The principal function associated with the transmission of a speech signal is to convey a message. However, along with the message, additional kinds of information are transmitted. These include information about the gender, identity, emotional state, health, etc. of the speaker. The source of all these kinds of information lie in both physiological and behavioral characteristics.

The physiological features are shown in Fig. 36.1 showing a cross-section of the human vocal tract. The shape of the vocal tract, determined by the position of articulators, the tongue, jaw, lips, teeth, and velum, creates a set of acoustic resonances in response to periodic puffs of air generated by the glottis for voiced sounds or aperiodic excitation caused by air passing through tight constrictions in the vocal tract. The spectral peaks associated with periodic resonances are referred to as speech formants. The locations in frequency and, to a lesser degree, the shapes of the resonances distinguish one speech sound from another. In addition, formant locations and bandwidths and spectral differences associated with the overall size of the vocal tract serve to distinguish the same sounds spoken by different speakers. The shape of the nasal tract, which determines the quality of nasal sounds, also varies significantly from speaker to speaker. The mass of the glottis is associated with the basic fundamental frequency for voiced speech sounds. The average basic fundamental frequency is approximately 100 Hz for adult males, 200 Hz for adult females, and 300 Hz for children. It also varies from individual to individual.

Speech signal events can be classified as segmental or suprasegmental. Generally, segmental refers to the features of individual sounds or segments, whereas suprasegmental refers to properties that extend over several speech sounds. Speaking behavior is associated with the individual's control of articulators for individual speech sounds or segments and also with suprasegmental characteristics governing how individual speech sounds are strung together to form words. Higher-level speaking behavior is associated with choices of words and syntactic units. Variations in fundamental frequency or pitch and rhythm are also higher-level features of the speech signal along with such qualities as breathiness, strength of vocal effort, etc. All of these vary significantly from speaker to speaker.

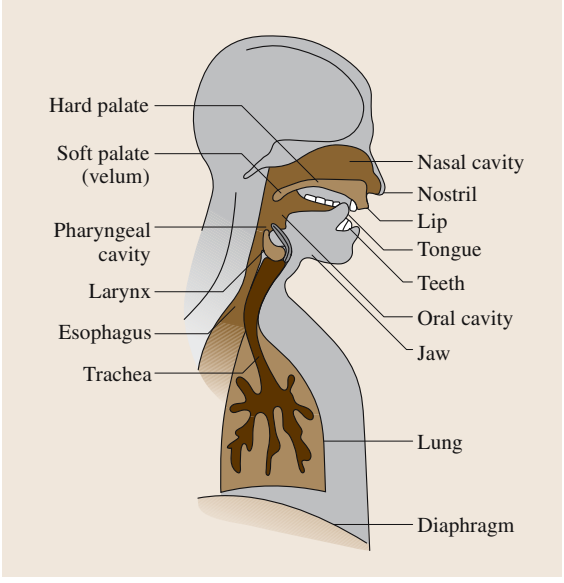

Fig. 36.1 Physiology of the human vocal tract (Reproduced with permission from L. H. *Jamieson* [36.11])

36.1.4 Extracting Speaker Characteristics from the Speech Signal

A perceptual view classifies speech as containing *low-level* and *high-level* kinds of information. Low-level features of speech are associated with the periphery in the brain's perception of speech and are relatively accessible from the speech signal. High-level features are associated with more-central locations in the perception mechanism. Generally speaking, low-level speaker features are easier to extract from the speech signal and model than high-level features. Many such features are associated with spectral correlates such as formant locations and bandwidths, pitch periodicity, and segmental timings. High-level features include the perception of words and their meaning, syntax, prosody, dialect, and idiolect.

It is not easy to extract stable and reliable formant features explicitly from the speech signal. In most instances it is easier to carry out short-term spectral amplitude measurements that capture low-level speaker characteristics implicitly. Short-term spectral measurements are typically carried out over 20–30 ms windows and advanced every 10 ms. Short speech sounds have durations less than 100 ms whereas stressed vowel sounds can last for 300 ms or more. Advancing the time window every 10 ms enables the temporal characteristics of individual speech sounds to be tracked and the 30 ms analysis window is usually sufficient to provide good spectral resolution of these sounds and at the same time short enough to resolve significant temporal characteristics. There are two principal methods of short-term spectral analysis, filter bank analysis and linear predictive coding (LPC) analysis. In filter bank analysis the speech signal is passed through a bank of bandpass filters covering a range of frequencies consistent with the transmission characteristics of the signal. The spacing of the filters can be uniform or, more likely, spaced nonuniformly, consistent with perceptual criteria such as the mel or bark scale [36.12], which provides a linear spacing in frequency below 1000 Hz

and logarithmic spacing above. The output of each filter is typically implemented as a windowed, short-term Fourier transform using fast Fourier transform (FFT) techniques. This output is subject to a nonlinearity and low-pass filter to provide an energy measurement. LPC-derived features almost always include regression measurements that capture the temporal evolution of these features from one speech segment to another. It is no accident that short-term spectral measurements are also the basis for speech recognizers. This is because an analysis that captures the differences between one speech sound and another can also capture the difference between the same speech sound uttered by different speakers, often with resolutions surpassing human perception.

Other measurements that are often carried out are correlated with prosody such as pitch and energy tracking. Pitch or periodicity measurements are relatively easy to make. However, periodicity measurement is meaningful only for voiced speech sounds so it is necessary also to have a detector that can discriminate voiced from unvoiced sounds. This complication often makes it difficult to obtain reliable pitch tracks over long-duration utterances.

Long-term average spectral and fundamental frequency measurements have been used in the past for speaker recognition, but since these measurements provide feature averages over long durations they are not capable of resolving detailed individual differences.

Although computational ease is an important consideration for selecting speaker-sensitive feature measurements, equally important considerations are the stability of the measurements, including whether they are subject to variability, noise, and distortions from one measurement of a speaker's utterances to another. One source of variability is the speaker himself. Features that are correlated with behavior such as pitch contours – pitch measured as a function of time over specified utterances – can be consciously varied from one token of an utterance to another. Conversely, cooperative speakers can control such variability. More difficult to deal with are the variability and distortion associated with recording environments, microphones, and transmission media. The most severe kinds of variability problems occur when utterances used to train models are recorded under one set of conditions and test utterances are recorded under another.

A block diagram of a speaker recognition is shown in Fig. 36.2, showing the basic elements discussed in this section. A sample of speech from an unknown speaker is input to the system. If the system is a speaker verification system, an identity claim or assertion is also input. The speech sample is recorded, digitized, and analyzed. The analysis is typically some sort of short-term spectral analysis that captures speaker-sensitive features as described earlier in this section. These features are compared with prototype features compiled into the models of known speakers. A matching process is invoked to compare the sample features and the model features. In the case of closed-set speaker identification, the match is assigned to the model with the best matching score. In the case of speaker verification, the matching score is compared with a predetermined threshold to decide whether to accept or reject the identity claim. For open-set identification, if the matching score for the best matching model does not pass a threshold test, a no-match decision is made.

36.1.5 Applications

As mentioned, the most widespread applications for automatic speaker recognition are for security. These are typically speaker verification applications intended to control access to privileged transactions or information remotely over a telecommunication network. These are usually configured in a text-dependent mode in which customers are prompted to speak personalized verification phrases such as personal identification numbers

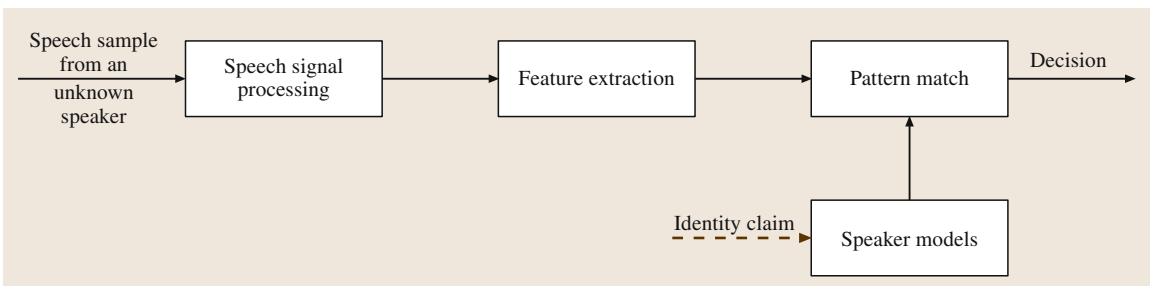

Fig. 36.2 Block diagram of a speaker recognition system

(PINs) spoken as a string of digits. Typically, PIN utterances are decoded using a speaker-independent speech recognizer to provide an identity claim. The utterances are then processed in a speaker recognition mode and compared with speaker models associated with the identity claim. Speaker models are trained by recording and processing prompted verification phrases in an enrollment session.

In addition to security applications, speaker verification may be used to offer personalized services to users. For example, once a speaker verification phrase is authenticated, the user may be given access to a personalized phone book for voice repertory dialing.

A forensic application is likely to be an open-set identification or verification task. A sample of speech exists from an unknown perpetrator. A suspect is required to speak utterances contained in the suspect speech sample in order to train a model. The suspect speech sample is compared both with the suspect and nonsuspect models to decide whether to accept or reject the hypothesis that the suspect and perpetrator voices are the same.

In surveillance applications the input speech mode is most likely to be text independent. Since the speaker may be unaware that his voice is being monitored, he cannot be expected to speak specified texts. The decision task is open-set identification or verification.

Large amounts of multimedia data, including speech, are being recorded and stored on digital media. The existence of such large amounts of data has created a need for efficient, versatile, and accurate data mining tools for extracting useful information content from the data. A typical need is to search or browse through the data, scanning for specified topics, words, phrase, or speakers. Most of this data is multispeaker data, collected from broadcasts, recorded meetings, telephone conversations, etc. The process of obtaining a list of speaker segments from such data is referred to as speaker indexing, segmentation, or diarization. A more-general task of annotating audio data from various audio sources including speakers has been referred to as audio diarization [36.10].

Still another speaker recognition application is to improve automatic speech recognition by adapting speaker-independent speech models to specified speakers. Many commercial speech recognizers do adapt their speech models to individual users, but this cannot be regarded as a speaker recognition application unless speaker models are constructed and speaker recognition is a part of the process. Speaker recognition can also be used to improve speech recognition for multispeaker data. In this situation speaker indexing can provide a table of speech segments assigned to individual speakers. The speech data in these segments can then be used to adapt speech models to each speaker. Speech recognition of multispeaker speech samples can be improved in another way. Errors and ambiguities in speech recognition transcripts can be corrected using the knowledge provided by speaker segmentation assigning the segments to the correct speakers.

36.2 Measuring Speaker Features

36.2.1 Acoustic Measurements

As mentioned in Sect. 36.1, low-level acoustic features such as short-time spectra are commonly used in speaker modeling. Such features are useful in authentication systems because speakers have less control over spectral details than higher-level features such as pitch.

Short-Time Spectrum

There are many ways of representing the short-time spectrum. A popular representation is the mel-frequency cepstral coefficients (MFCC), which were originally developed for speaker-independent speech recognition. The choice of center frequencies and bandwidths of the filter bank used in MFCC were motivated by the properties of the human auditory system. In particular, this representation provides limited spectral resolution above 2 kHz, which might be detrimental in speaker recognition. However, somewhat counterintuitively, MFCCs have been found to be quite effective in speaker recognition.

There are many minor variations in the definition of MFCC but the essential details are as follows. Let $\{S(k), 0 \le k < K\}$ be the discrete Fourier transform (DFT) coefficients of a windowed speech signal $\hat{s}(t)$. A set of triangular filters are defined such that

$$w_j(k) = \begin{cases} \frac{(k/K) f_s - f_{c_{j-1}}}{f_{c_j} - f_{c_{j-1}}}, & l_j \le k \le c_j, \\ f_{c_{j+1}} - \left(\frac{k}{K}\right) f_s / f_{c_{j+1}} - f_{c_j}, & c_j < k \le u_j, \\ 0, & \text{elsewhere}, \end{cases} \tag{36.1}$$

where $f_{c_{j-1}}$ and $f_{c_{j+1}}$ are the lower and upper limits of the pass band for filter j with $f_{c_0} = 0$ and $f_{c_j} < f_s/2$ for all j, and l_j, c_j and u_j are the DFT indices corresponding to the lower, center, and upper limits of the pass band for filter j. The log-energy at the outputs for the J filters are given by

$$e(j) = \ln\left[\frac{1}{\sum_{k=l_j}^{u_j} w_j(k)} \sum_{k=l_j}^{u_j} \|S(k)\|^2 w_j(k)\right] , \quad (36.2)$$

and the MFCC coefficients are the discrete cosine transform of the filter energies computed as

$$C(k) = \sum_{j=0}^{J} e(j) \cos\left[k\left(j - \frac{1}{2}\right)\frac{\pi}{J}\right] ,$$
$$k = 1, 2, \ldots, K . \quad (36.3)$$

The zeroth coefficient $C(0)$ is set to be the average log-energy of the windowed speech signal. Typical values of the various parameters involved in the MFCC computation are as follows. A cepstrum vector is calculated using a window length of 20 ms and updated every 10 ms. The center frequencies f_{c_j} are uniformly spaced from 0 to 1000 Hz and logarithmically spaced above 1000 Hz. The number of filter energies is typically 24 for telephone-band speech and the number of cepstrum coefficients used in modeling varies from 12 to 18 [36.13].

Cepstral coefficients based on short-time spectra estimated using linear predictive analysis and perceptual linear prediction are other popular representations [36.14].

Short-time spectral measurements are sensitive to channel and transducer variations. Cepstral mean subtraction (CMS) is a simple and effective method to compensate for convolutional distortions introduced by slowly varying channels. In this method, the cepstral vectors are transformed such that they have zero mean. The cepstral average over a sufficiently long speech signal approximates the estimate of a stationary channel [36.14]. Therefore, subtracting the mean from the original vectors is roughly equivalent to normalizing the effects of the channel, if we assume that the average of the clean speech signal is zero. Cepstral variance normalization, which results in feature vectors with unit variance, has also been shown to improve performance in text-independent speaker recognition when there is more than a minute of speech for enrollment. Other feature normalization methods, such as feature warping [36.15] and Gaussianization [36.16], map the observed feature distribution to a normal distribution over a sliding window, and have been shown to be useful in speaker recognition.

It has been long established that incorporating dynamic information is useful for speaker recognition and speech recognition [36.17]. The dynamic information is typically incorporated by extending the static cepstral vectors by their first and second derivatives computed as:

$$\Delta C_k = \frac{\sum_{t=-l}^{l} t c_{t+k}}{\sum_{t=-l}^{l} |t|} , \quad (36.4)$$

$$\Delta\Delta C_k = \frac{\sum_{t=-l}^{l} t^2 c_{t+k}}{\sum_{t=-l}^{l} t^2} . \quad (36.5)$$

Pitch

Voiced sounds are produced by a quasiperiodic opening and closing of the vocal folds in the larynx at a *fundamental frequency* that depends on the speaker. Pitch is a complex auditory attribute of sound that is closely related to this fundamental frequency. In this chapter, the term pitch is used simply to refer to the measure of periodicity observed in voiced speech.

Prosodic information represented by pitch and energy contours has been used successfully to improve the performance of speaker recognition systems [36.18]. There are a number of techniques for estimating pitch from the speech signal [36.19] and the performance of even simple pitch-estimation techniques is adequate for speaker recognition. The major failure modes occur during speech segments that are at the boundaries of voiced and unvoiced sounds and can be ignored for speaker recognition. A more-significant problem with using pitch information for speaker recognition is that speakers have a fair amount of control over it, which results in large intraspeaker variations and mismatch between enrollment and test utterances.

36.2.2 Linguistic Measurements

In traditional speaker authentication applications, the enrollment data is limited to a few repetitions of a password, and the same password is spoken to gain access to the system. In such cases, speaker models based on short-time spectra are very effective and it is difficult to

extract meaningful *high-level* or linguistic features. In applications such as indexing broadcasts by speaker and passive surveillance, a significant amount of enrollment data, perhaps several minutes, may be available. In such cases, the use of linguistic features has been shown to be beneficial [36.18].

Word Usage

Features such as vocabulary choices, function word frequencies, part-of-speech frequencies, etc., have been shown to be useful in speaker recognition [36.20]. In addition to words, spontaneous speech contains fillers and hesitations that can be characterized by statistical models and used for identifying speakers [36.20, 21]. There are a number of issues with speaker recognition systems based on lexical features: they are susceptible to errors introduced by large-vocabulary speech recognizers, a significant amount of enrollment data is needed to build robust models, and the speaker models are likely to characterize the topic of conversation as well as the speaker.

Phone Sequences and Lattices

Models of phone sequences output by speech recognizers using phonotactic grammars, typically phone unigrams, can be used to represent speaker characteristics [36.22]. It is assumed that these models capture speaker-specific pronunciations of frequently occurring words, choice of words, and also an implicit characterization of the acoustic space occupied by the speech signal from a given speaker. It turns out that there is an optimal tradeoff between the constraints used in the recognizer to produce the phone sequences and the robustness of the speaker models of phone sequences. For example, the use of lexical constraints in the automatic speech recognition (ASR) reproduces phone sequences found in a predetermined dictionary and prevents phone sequences that may be characteristic of a speaker but not represented in the dictionary.

The phone accuracy computed using one-best output phone strings generated by ASR systems without lexical constraints is typically not very high. On the other hand, the correct phone sequence can be found in a phone lattice output by an ASR with a high probability. It has been shown that it is advantageous to construct speaker models based on phone-lattice output rather than the one-best phone sequence [36.22]. Systems based on one-best phone sequences use the counts of a term such as a phone unigram or bigram in the decoded sequence. In the case of lattice outputs, these raw counts are replaced by the expected counts given by

$$\boldsymbol{E}[C(\tau|X)] = \sum_{Q} p(Q|X)C(\tau|Q)\,, \tag{36.6}$$

where Q is a path through the phone lattice for the utterance X with associated probability $p(Q|X)$, and $C(\tau|Q)$ is the count of the term τ in the path Q.

Other Linguistic Features

A number of other features that have been found to be useful for speaker modeling are (a) pronunciation modeling of carefully chosen words, and (b) prosodic statistics such as pitch and energy contours as well as durations of phones and pauses [36.23].

36.3 Constructing Speaker Models

A speaker recognition system provides the ability to construct a model λ_s for speaker s using enrollment utterances from that speaker, and a method for comparing the quality of match of a *test* utterance to the speaker model. The choice of models is determined by the application constraints. In applications in which the user is expected to say a fixed password each time, it is beneficial to develop models for words or phrases to capture the temporal characteristics of speech. In passive surveillance applications, the test utterance may contain phonemes or words not seen in the enrollment data. In such cases, less-detailed models that model the overall acoustic space of the user's utterances tend to be effective. A survey of general techniques that have been used in speaker modeling follows. The methods can be broadly classified as nonparametric or parametric. Nonparametric models make few structural assumptions and are effective when there is sufficient enrollment data that is matched to the test data. Parametric models allow a parsimonious representation of the structural constraints and can make effective use of the enrollment data if the constraints are appropriately chosen.

36.3.1 Nonparametric Approaches

Templates

This is the simplest form of speaker modeling and is appropriate for fixed-password speaker verification sys-

tems [36.24]. The enrollment data consists of a small number of repetitions of the password spoken by the target speaker. Each enrollment utterance $\boldsymbol{X}$ is a sequence of feature vectors $\{\boldsymbol{x}_t\}_{t=0}^{T-1}$ generated as described in Sect. 36.2, and serves as the *template* for the password as spoken by the target speaker. A test utterance $\boldsymbol{Y}$ consisting of vectors $\{\boldsymbol{y}_t\}_{t=0}^{T'-1}$, is compared to each of the enrollment utterances and the identity claim is accepted if the distance between the test and enrollment utterances is below a decision threshold. The comparison is done as follows. Associated with each pair of vectors, x_i and y_j, is a distance, $d(\boldsymbol{x}_i, \boldsymbol{y}_j)$. The feature vectors of $\boldsymbol{X}$ and $\boldsymbol{Y}$ are aligned using an algorithm referred to as *dynamic time warping* to minimize an overall distance defined as the average intervector distance $d(x_i, y_j)$ between the aligned vectors [36.12].

This approach is effective in simple fixed-password applications in which robustness to channel and transducer differences are not an issue. This technique is described here mostly for historical reasons and is rarely used in real applications today.

Nearest-Neighbor Modeling

Nearest-neighbor models have been popular in nonparametric classification [36.25]. This approach is often thought of as estimating the local density of each class by a Parzen estimate and assigning the test vector to the class with the maximum local density. The local density of a class (speaker) with enrollment data $\boldsymbol{X}$ at a test vector $\boldsymbol{y}$ is defined as

$$p_{\mathrm{nn}}(\boldsymbol{y}; \boldsymbol{X}) = \frac{1}{V[d_{\mathrm{nn}}(\boldsymbol{y}, \boldsymbol{X})]}\,, \tag{36.7}$$

where $d_{\mathrm{nn}}(\boldsymbol{y}, \boldsymbol{X}) = \min_{\boldsymbol{x}_j \in \boldsymbol{X}} \|\boldsymbol{y} - \boldsymbol{x}_j\|$ is the nearest-neighbor distance and $V(r)$ is the volume of a sphere of radius r in the D-dimensional feature space. Since $V(r)$ is proportional to r^D,

$$\ln[p_{\mathrm{nn}}(\boldsymbol{y}; \boldsymbol{X})] \approx -D \ln[d_{\mathrm{nn}}(\boldsymbol{y}, \boldsymbol{X})]\,. \tag{36.8}$$

The log-likelihood score of the test utterances $\boldsymbol{Y}$ with respect to a speaker specified by enrollment $\boldsymbol{X}$ is given by

$$s_{\mathrm{nn}}(\boldsymbol{Y}; \boldsymbol{X}) \approx -\sum_{\boldsymbol{y}_j \in \boldsymbol{Y}} \ln[d_{\mathrm{nn}}(\boldsymbol{y}, \boldsymbol{X})]\,, \tag{36.9}$$

and the speaker with the greatest $s(\boldsymbol{Y}; \boldsymbol{X})$ is identified.

A modified version of the nearest-neighbor model, motivated by the discussion above, has been successfully used in speaker identification [36.26]. It was found empirically that a score defined as

$$\begin{aligned} s'_{\mathrm{nn}}(\boldsymbol{Y}; \boldsymbol{X}) = {} & \frac{1}{N_y} \sum_{\boldsymbol{y}_j \in \boldsymbol{Y}} \min_{\boldsymbol{x}_i \in \boldsymbol{X}} \|\boldsymbol{y}_j - \boldsymbol{x}_i\|^2 \\ & + \frac{1}{N_x} \sum_{\boldsymbol{x}_j \in \boldsymbol{X}} \min_{\boldsymbol{y}_i \in \boldsymbol{Y}} \|\boldsymbol{y}_i - \boldsymbol{x}_j\|^2 \\ & - \frac{1}{N_y} \sum_{\boldsymbol{y}_j \in \boldsymbol{Y}} \min_{\boldsymbol{y}_i \in \boldsymbol{Y};\, j \neq i} \|\boldsymbol{y}_i - \boldsymbol{y}_j\|^2 \\ & - \frac{1}{N_x} \sum_{\boldsymbol{x}_j \in \boldsymbol{X}} \min_{\boldsymbol{x}_i \in \boldsymbol{X};\, j \neq i} \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 \end{aligned} \tag{36.10}$$

gives much better performance than $s_{\mathrm{nn}}(\boldsymbol{Y}; \boldsymbol{X})$.

36.3.2 Parametric Approaches

Vector Quantization Modeling

Vector quantization constructs a set of representative samples of the target speaker's enrollment utterances by clustering the feature vectors. Although a variety of clustering techniques exist, the most commonly used is k-means clustering [36.14]. This approach partitions N feature vectors into K disjoint subsets S_j to minimize an overall distance such as

$$D = \sum_{j=1}^{J} \sum_{\boldsymbol{x}_i \in S_j} (\boldsymbol{x}_i - \boldsymbol{\mu}_j)\,, \tag{36.11}$$

where $\boldsymbol{\mu}_j = (1/N_j) \sum_{\boldsymbol{x}_i \in S_j} \boldsymbol{x}_i$ is the centroid of the N_j samples in the j-th cluster. The algorithm proceeds in two steps:

1. Compute the centroid of each cluster using an initial assignment of the feature vectors to the clusters.
2. Reassign $\boldsymbol{x}_i$ to that cluster whose centroid is closest to it.

These steps are iterated until successive steps do not reassign samples.

This algorithm assumes that there exists an initial clustering of the samples into K clusters. It is difficult to obtain a good initialization of K clusters in one step. In fact, it may not even be possible to reliably estimate K clusters because of data sparsity. The Linde–Buzo–Gray (LBG) algorithm [36.27] provides a good solution for this problem. Given m centroids, the LBG algorithm produces additional centroids by perturbing one or more of the centroids using a heuristic. One common heuristic is to choose the $\boldsymbol{\mu}$ for the cluster with the largest variance and produce two centroids $\boldsymbol{\mu}$ and $\boldsymbol{\mu} + \boldsymbol{\epsilon}$. The enrollment feature vectors are assigned to the resulting $m+1$ centroids. The k-means algorithm described previously can

then be applied to refine the centroid estimates. This process can be repeated until $m = M$ or the cluster sizes fall below a threshold. The LBG algorithm is usually initialized with $m = 1$ and computes the centroid of all the enrollment data. There are many variations of this algorithm that differ in the heuristic used for perturbing the centroids, the termination criteria, and similar details. In general, this algorithm for generating VQ models has been shown to be quite effective. The choice of K is a function of the size of enrollment data set, the application, and other system considerations such as limits on computation and memory.

Once the VQ models are established for a target speaker, scoring consists of evaluating D in (36.11) for feature vectors in the test utterance. This approach is general and can be used for text-dependent and text-independent speaker recognition, and has been shown to be quite effective [36.28]. Vector quantization models can also be constructed on sequences of feature vectors, which are effective at modeling the temporal structure of speech. If distance functions and centroids are suitably redefined, the algorithms described in this section continue to be applicable.

Although VQ models are still useful in some situations, they have been superseded by models such as the Gaussian mixture models and hidden Markov models, which are described in the following sections.

Gaussian Mixture Models

In the case of text-independent speaker recognition (the subject of Chap. 38) where the system has no prior knowledge of the text of the speaker's utterance, Gaussian mixture models (GMMs) have proven to be very effective. This can be thought of as a refinement of the VQ model. Feature vectors of the enrollment utterances $\boldsymbol{X}$ are assumed to be drawn from a probability density function that is a mixture of Gaussians given by

$$p(\boldsymbol{x}|\lambda) = \sum_{k=1}^{K} w_k p_k(\boldsymbol{x}|\lambda_k) \,, \tag{36.12}$$

where $0 \le w_k \le 1$ for $1 \le k \le K$, $\sum_{k=1}^{K} w_k = 1$, and

$$p_k(\boldsymbol{x}|\lambda_k) = \frac{1}{(2\pi)^{D/2}|\Sigma_k|^{1/2}} \times \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_k)^{\mathrm{T}}\Sigma_k^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k)\right) \,, \tag{36.13}$$

λ represents the parameters $(\boldsymbol{\mu}_i, \Sigma_i, w_i)_{i=1}^{K}$ of the distribution. Since the size of the training data is often small, it is difficult to estimate full covariance matrices reliably. In practice, $\{\Sigma_k\}_{k=1}^{K}$ are assumed to be diagonal.

Given the enrollment data $\boldsymbol{X}$, the maximum-likelihood estimates of the λ can be obtained using the *expectation-maximization* (EM) algorithm [36.12]. The K-means algorithm can be used to initialize the parameters of the component densities. The posterior probability that $\boldsymbol{x}_t$ is drawn from the component $p_m(\boldsymbol{x}_t|\lambda_m)$ can be written

$$P(m|\boldsymbol{x}_t, \lambda) = \frac{w_m p_m(\boldsymbol{x}_t|\lambda_m)}{p(\boldsymbol{x}_t|\lambda)} \,. \tag{36.14}$$

The maximum-likelihood estimates of the parameters of λ in terms of $P(m|\boldsymbol{x}_t, \lambda)$ are

$$\boldsymbol{\mu}_m = \frac{\sum_{t=1}^{\mathrm{T}} P(m|\boldsymbol{x}_t, \lambda)\boldsymbol{x}_t}{\sum_{t=1}^{\mathrm{T}} P(m|\boldsymbol{x}_t, \lambda)} \,, \tag{36.15}$$

$$\Sigma_m = \frac{\sum_{t=1}^{\mathrm{T}} P(m|\boldsymbol{x}_t, \lambda)\boldsymbol{x}_t\boldsymbol{x}_t^{\mathrm{T}}}{\sum_{t=1}^{\mathrm{T}} P(m|\boldsymbol{x}_t, \lambda)} - \boldsymbol{\mu}_m\boldsymbol{\mu}_m^{\mathrm{T}} \,, \tag{36.16}$$

$$w_m = \frac{1}{T}\sum_{t=1}^{\mathrm{T}} P(m|\boldsymbol{x}_t, \lambda) \,. \tag{36.17}$$

The two steps of the EM algorithm consist of computing $P(m|\boldsymbol{x}_t, \lambda)$ given the current model, and updating the model using the equations above. These two steps are iterated until a convergence criteria is satisfied.

Test utterance scores are obtained as the average log-likelihood given by

$$s(\boldsymbol{Y}|\lambda) = \frac{1}{T}\sum_{t=1}^{\mathrm{T}} \log[p(\boldsymbol{y}_t|\lambda)] \,. \tag{36.18}$$

Speaker verification is often based on a likelihood-ratio test statistic of the form $p(\boldsymbol{Y}|\lambda)/p(\boldsymbol{Y}|\lambda_{\mathrm{bg}})$ where λ is the speaker model and λ_{bg} represents a background model [36.29]. For such systems, speaker models can also be trained by adapting λ_{bg}, which is generally trained on a large independent speech database [36.30]. There are many motivations for this approach. Generating a speaker model by adapting a well-trained background GMM may yield models that are more robust to channel differences, and other kinds of mismatch between enrollment and test conditions than models estimated using only limited enrollment data. Details of this procedure can be found in Chap. 38.

Speaker modeling using GMMs is attractive for text-independent speaker recognition because it is simple to implement and computationally inexpensive. The fact that this model does not model temporal aspects of speech is a disadvantage. However, it has been difficult to exploit temporal structure to improve speaker recognition performance when the linguistic content of test utterances does not overlap significantly with the linguistic content of enrollment utterances.

Hidden Markov Models

In applications where the system has prior knowledge of the text and there is significant overlap of what was said during enrollment and testing, text-dependent statistical models are much more effective than GMMs. An example of such applications is access control to personal information or bank accounts using a voice password. Hidden Markov models (HMMs) [36.12] for phones, words, or phrases, have been shown to be very effective [36.31, 32]. Passwords consisting of word sequences drawn from specialized vocabularies such as digits are commonly used. Each word can be characterized by an HMM with a small number of states, in which each state is represented by a Gaussian mixture density. The maximum-likelihood estimates of the parameters of the model can be obtained using a generalization of the EM algorithm [36.12].

The ML training aims to approximate the underlying distribution of the enrollment data for a speaker. The estimates deviate from the *true* distribution due to lack of sufficient training data and incorrect modeling assumptions. This leads to a suboptimal classifier design. Some limitations of ML training can be overcome using discriminative training of speaker models in which an attempt is made to minimize an overall cost function that depends on misclassification or detection errors [36.33–35]. Discriminative training approaches require examples from competing speakers in addition to examples from the target speaker. In the case of closed-set speaker identification, it is possible to construct a misclassification measure to evaluate how likely a test sample, spoken by a target speaker, is misclassified as any of the others. One example of such a measure is the *minimum classification error* (MCE) defined as follows. Consider the set of S discriminant functions $\{g_k(\boldsymbol{x}; \Lambda_s), 1 \le s \le S\}$, where $g_k(\boldsymbol{x}; \Lambda_s)$ is the log-likelihood of observation $\boldsymbol{x}$ given the models Λ_s for speaker s. A set of misclassification measures for each speaker can be defined as

$$d_s(\boldsymbol{x}; \Lambda) = -g_s(\boldsymbol{x}; \Lambda_s) + G_s(\boldsymbol{x}; \Lambda), \tag{36.19}$$

where Λ is the set of all speaker models and $G_s(\boldsymbol{x}; \Lambda)$ is the antidiscriminant function for speaker s. $G_s(\boldsymbol{x}; \Lambda)$ is defined so that $d_s(\boldsymbol{x}; \Lambda)$ is positive only if $\boldsymbol{x}$ is incorrectly classified. In speech recognition problems, $G_s(\boldsymbol{x}; \Lambda)$ is usually defined as a collective representation of all competing classes. In the speaker identification task, it is often advantageous to construct pairwise misclassification measures such as

$$d_{ss'}(\boldsymbol{x}; \Lambda) = -g_s(\boldsymbol{x}; \Lambda_s) + g'_s\left(\boldsymbol{x}; \Lambda'_s\right), \tag{36.20}$$

with respect to a set of competing speakers s', a subset of the S speakers. Each misclassification measure is embedded into a smooth empirical *loss function*

$$l_{ss'}(\boldsymbol{x}; \Lambda) = \frac{1}{1 + \exp(-\alpha d_{ss'}(\boldsymbol{x}; \Lambda))}, \tag{36.21}$$

which approximates a loss directly related to the number of classification errors, and α is a smoothness parameter. The loss functions can then be combined into an overall loss given by

$$l(\boldsymbol{x}; \Lambda) = \sum_s \sum_{s' \in \mathcal{S}_c} l_{ss'}(\boldsymbol{x}; \Lambda) \delta_s(\boldsymbol{x}), \tag{36.22}$$

where $\delta_s(\boldsymbol{x})$ is an indicator function which is equal to 1 when $\boldsymbol{x}$ is uttered by speaker s and 0 otherwise, and $\mathcal{S}_c$ is the set of competing speakers. The total loss, defined as the sum of $l(\boldsymbol{x}; \Lambda)$ over all training data, can be optimized with respect to all the model parameters using a gradient-descent algorithm. A similar algorithm has been developed for speaker verification in which samples from a large number of speakers in a development set is used to compute a minimum verification measure [36.36].

The algorithm described above is only to illustrate the basic principles of discriminative training for speaker identification. Many other approaches that differ in their choice of the loss function or the optimization method have been developed and shown to be effective [36.35, 37].

The use of HMMs in text-dependent speaker verification is discussed in detail in Chap. 37.

Support Vector Modeling

Traditional discriminative training approaches such as those based on MCE have a tendency to overtrain on the training set. The complexity and generalization ability of the models are usually controlled by testing on

a held-out development set. Support vector machines (SVMs) [36.38] provide a way for training classifiers using discriminative criteria and in which the model complexity that provides good generalization to test data is determined automatically from the training data. SVMs have been found to be useful in many classification tasks including speaker identification [36.39].

The original formulation of SVMs was for two-class problems. This seems appropriate for speaker verification in which the positive samples consist of the enrollment data from a target user and the negative samples are drawn from a large set of imposter speakers. Many extensions of SVMs to multiclass classification have also been developed and are appropriate for speaker identification. There are many issues with SVM modeling for speaker recognition, including the appropriate choice of features and the kernel. The use of SVMs for text-independent speaker recognition is the subject of Chap. 38.

Other Approaches

Most state-of-the-art speaker recognition systems use some combination of the modeling methods described in the previous sections. Many other interesting models have been proposed and have been shown to be useful in limited scenarios. Eigenvoice modeling is an approach in which the speaker models are confined to a low-dimensional linear subspace obtained using independent training data from a large set of speakers. This method has been shown to be effective for speaker modeling and speaker adaptation when the enrollment data is too limited for the effective use of other text-independent approaches such as GMMs [36.40]. Artificial neural networks [36.41] have also been shown to be useful in some situations, perhaps in combination with GMMs. When sufficient enrollment data is available, a method for speaker detection that involves comparing the test segment directly to similar segments in enrollment data has been shown to be effective [36.42].

36.4 Adaptation

In most speaker recognition scenarios, the speech data available for enrollment is too limited for training models that adequately characterize the range of test conditions in which the system needs to operate. For example, in fixed-password speaker authentication systems used in telephony services, enrollment data is typically collected in a single call. The enrollment and test conditions may be mismatched in a number of ways: the telephone handset that is used, the location of the call, which determines the kinds of background noises, and the channel over which speech is transmitted such as cellular or landline networks. In text-independent modeling, there are likely to be additional problems because of mismatch in the linguistic content. A very effective way to mitigate the effects of mismatch is model adaptation.

Models can be adapted in an unsupervised way using data from authenticated utterances. This is common in fixed-password systems and can reduce the error rate significantly. It is also necessary to update the decision thresholds when the models are adapted. Since the selection of data for model adaptation is not supervised, there is the possibility that models are adapted on imposter utterances. This can be disastrous. The details of unsupervised model and threshold adaptation and the various issues involved are explained in detail in Chap. 37.

Speaker recognition is often incorporated into other applications that involve a dialog with the user. Feedback from the dialog system can be used to supervise model adaptation. In addition, meta-information available from a dialog system such as the history of interactions can be combined with speaker recognition to design a flexible and secure authentication system [36.43].

36.5 Decision and Performance

36.5.1 Decision Rules

Whether they are used for speaker identification or verification, the various models and approaches presented in Sect. 36.3 provide a score $s(Y|\lambda)$ measuring the match between a given test utterance Y and a speaker model λ. Identification systems yield a set of such scores corresponding to each speaker in a target list. Verification systems output only one score using the speaker model of the claimed speaker. An

accept or reject decision has to be made using this score.

Decision in closed-set identification consists of choosing the identified speaker $\hat{S}$ as the one that corresponds to the maximum score:

$$\hat{S} = \arg\max_j s(Y|\lambda_j) \,, \tag{36.23}$$

where the index j ranges over the whole set of target speakers.

Decision in verification is obtained by comparing the score computed using the model for the claimed speaker S_i given by $s(Y|\lambda_i)$ to a predefined threshold θ. The claim is accepted if $s(Y|\lambda_i) \geq \theta$, and rejected otherwise.

Open-set identification relies on a step of closed-set identification eliciting the most likely identity, followed by a verification step to determine whether the hypothesized identity match is good enough.

36.5.2 Threshold Setting and Score Normalization

Efficiency and robustness require that the score $s(Y|\lambda)$ be quite readily exploited in a practical application. In particular, the threshold θ should be as insensitive as possible across users and application context.

When the score is obtained in a probabilistic framework or can be interpreted as a (log) likelihood ratio (LLR), Bayesian decision theory [36.44] states that an optimal threshold for verification can be theoretically set once the desired false acceptance c_{fa} and false rejection c_{fr}, and the a priori probability p_{imp} of an impostor trying to enter the system, are specified. The optimal choice of the threshold is given by:

$$\theta^* = \frac{c_{\text{fa}}}{c_{\text{fr}}} \frac{p_{\text{imp}}}{1 - p_{\text{imp}}} \,. \tag{36.24}$$

In practice, however, the score $s(Y|\lambda)$ does not behave as theory would predict since the statistical models are not ideal. Various normalization procedures have been proposed to alleviate this problem. Initial work by *Li* and *Porter* [36.45] has inspired a number of score normalization techniques that intend to make the statistical distribution of $s(Y|\lambda)$ as independent as possible across speakers, acoustic conditions, linguistic content, etc. This has lead to a number of threshold normalization schemes, such as the Z-norm, H-Norm, and T-norm, which use side information, the distance between models, and speech material from a development set to determine the normalization parameters. These normalization procedures are discussed in more detail in Chaps. 37, 38 and [36.46]. Even so, the optimal threshold for a given operating condition is generally estimated experimentally from development data that is appropriate for a given scenario.

36.5.3 Errors and DET Curves

The performance of an identification system is related to the probability of misclassification, which corresponds to cases when the identified speaker is not the actual one.

Verification systems are evaluated based on two types of errors: false acceptance, when an impostor speaker succeeds in being verified with an erroneous claimed identity, and false rejection, when a target user claiming his/her genuine identity is rejected. The *a posteriori* estimates of the probabilities p_{fa} and p_{fr} of these two types of errors vary in the opposite way from each other when the decision threshold θ is varied. The tradeoff between p_{fa} and p_{fr} (sometimes mapped to the probability of detection p_{d}, defined as $1 - p_{\text{fr}}$) is often displayed in the form of a receiver operating characteristic (ROC), a term commonly used in detection theory [36.44]. In speaker recognition systems a different representation of the same data, referred to as the detection error tradeoff (DET) curve, has become popular.

The DET curve [36.47] is the standard way to depict the system behavior in terms of hypotheses separability by plotting p_{fa} as a function of p_{fr}. Rather than the probabilities themselves, the normal deviates corresponding to the probabilities are plotted. For a particular threshold value, the corresponding error rates p_{fa} and p_{fr} appear as a specific point on this DET curve. A popular point is the one where $p_{\text{fa}} = p_{\text{fr}}$, which is called the equal error rate (EER). Plotting DET curves is a good way to compare the potential of two methods in a laboratory but it is not suited for predicting accurately the performance of a system when deployed in real-life conditions.

The decision threshold θ is often chosen to optimize a cost that is a function of the probability of false acceptance and false rejection as well as the prior probability of an imposter attack. One such function is called the detection cost function (DCF), defined as [36.48]

$$C = p_{\text{imp}} c_{\text{fa}} p_{\text{fa}} + (1 - p_{\text{imp}}) c_{\text{fr}} p_{\text{fr}} \,. \tag{36.25}$$

The DCF is indeed a way to evaluate a system under a particular operating condition and to summarize into a single figure its estimated performance in a given application scenario. It has been used as the primary figure of merit for the evaluation of systems participating in the yearly NIST speaker recognition evaluations [36.48].

36.6 Selected Applications for Automatic Speaker Recognition

Text-dependent and text-independent speaker recognition technology and their applications are discussed in detail in the following two Chaps. 37 and 38. A few interesting, but perhaps not primary, applications of speaker recognition technology are described in this section. These applications were chosen to demonstrate the wide range of applications of speaker recognition.

36.6.1 Indexing Multispeaker Data

Speaker indexing can be approached as either a supervised or unsupervised task. Supervised means that prior speaker models exist for the speakers of interest included in the data. The data can then be scanned and processed to determine the segments associated with each of these speakers. Unsupervised means that prior speaker models do not exist. The type of approach taken depends on the type and amount of prior knowledge available for particular applications. There may be knowledge of the identities of the participating speakers and there may even be independent labeled speech data available for constructing models for these speakers, such as in the case of some broadcast news applications [36.6, 49, 50]. In this situation the task is supervised and the techniques for speaker segmentation or indexing are basically the same as used for speaker detection [36.9, 50, 51].

A more-challenging task is unsupervised segmentation. An example application is the segmentation of the speakers in a two-person telephone conversation [36.4, 9, 52, 53]. The speaker identities may or may not be known but independent labelled speech data for constructing speaker models is generally not available. Following is a possible approach to the unsupervised segmentation problem. The first task is to construct unlabeled single-speaker models from the current data. An initial segmentation of the data is carried out with an acoustic change detector using a criterion such as the generalized likelihood ratio (GLR) [36.4, 5] or Bayesian information criterion (BIC) [36.8, 54, 55]. The hypothesis underlying this process is that each of the resulting segments will be single-speaker segments. These segments are then clustered using an agglomerative clustering algorithm with a criterion for measuring the pairwise similarity between segments [36.56–58]. Since in the cited application the number of speakers is known to be two, the clustering terminates when two clusters are obtained. If the acoustic change criterion and the matching criterion for the clustering perform well the two clusters of segments will each contain segments mostly from one speaker or the other. These segment clusters can then be used to construct protospeaker models, typically GMMs. Each of these models is then used to resegment the data to provide an improved segmentation which, in turn, will provide improved speaker models. The process can be iterated until no further significant improvement is obtained. It then remains to apply speaker labels to the models and segmentations. Some independent knowledge is required to accomplish this. As mentioned earlier, the speakers in the telephone conversation may be known, but some additional information is required to assign labels to the correct models and segmentations.

36.6.2 Forensics

The perspective of being able to identify a person on the basis of his or her voice has received significant interest in the context of law enforcement. In many situations, a voice recording is a key element, and sometimes the only one available, for proceeding with an investigation, identifying or clearing a suspect, and even supporting an accusation or defense in a court of law.

The public perception is that voice identification is a straightforward task, and that there exists a reliable *voiceprint* in much the same way as there are fingerprints or genetic (DNA) prints. This is not true in general because the voice of an individual has a strong behavioral component, and is only partly based on anatomical properties. Moreover, the conditions under which the test utterance is recorded are generally not known or controlled. The test voice sample might be from an anonymous call, wiretapping, etc. For these reasons, the use of voice recognition in the context of forensic applications must be approached with caution [36.59].

The four procedures that are generally followed in the forensic context are described below.

Nonexpert Speaker Recognition by Lay Listener(s)

This procedure is used in the context of a voice lineup when a victim or a witness has had the opportunity of hearing a voice sample and is asked to say whether he or she recognizes this voice, or to determine if this voice sample matches one of a set of utterances. Since it is difficult to set up such a test in a controlled way and calibrate to the matching criteria an individual subject may use, such procedures can be used only to suggest a possible course of action during an investigation.

Expert Speaker Recognition

Expert study of a voice sample might include one or more of aural–perceptual approaches, linguistic analysis, and spectrogram examination. In this context, the expert takes into account several levels of speaker characterization such as pitch, timbre, diction, style, idiolect, and other idiosyncracies, as well as a number of physical measurements including fundamental frequencies, segment durations, formants, and jitter. Experts provide a decision on a seven-level scale specified by the International Association for Identification (IAI) standard [36.60] on whether two voice samples (the disputed recording and a voice sample of the suspect) are more or less likely to have been produced by a the same person. Subjective heterogeneous approaches coexist between forensic practitioners and, although the technical invalidity of some methods has been clearly established, they are still used by some. The expert-based approach is therefore generally used with extreme caution.

Semiautomatic Methods

This category refers to systems for which a supervised selection of speech segments is conducted prior to a computer-based analysis of the selected material. Whereas a calibrated metric can be used to evaluate the similarity of specific types of segments such as words or phrases, these systems tend to suffer from a lack of standardization.

Automatic Methods

Fully automated methods using state-of-the-art techniques offer an attractive paradigm for forensic speaker verification. In particular, these automatic approaches can be run without any (subjective) human intervention, they offer a reproducible procedure, and they lend themselves to large-scale evaluation. Technological improvements over the years, as well as progress in the presentation, reporting, and interpretation of the results, have made such methods attractive. However, levels of performance remain highly sensitive to a number of external factors ranging from the quality and similarity of recording conditions, the cooperativeness of speakers, and the potential use of technologies to fake or disguise a voice.

Thanks to a number of initiatives and workshops (in particular the series of ISCA and IEEE Odyssey workshops), the past decade has seen some convergence in terms of formalism, interpretation, and methodology between forensic science and engineering communities. In particular, the interpretation of voice forensic evidence in terms of Bayesian decision theory and the growing awareness of the need for systematic evaluation have constituted significant contributions to these exchanges.

36.6.3 Customization: SCANmail

Customization of services and applications to the user is another class of applications of speaker recognition technology. An example of a customized messaging system is one where members of a family share a voice mailbox. Once the family members are enrolled in a speaker recognition system, there is no need for them to identify themselves when accessing their voice mail. A command such as *Get my messages* spoken by a user can be used to identify and authenticate the user, and provide only those messages left for that user. There are many such applications of speaker recognition technology. An interesting and successful application of *caller identification* to a voicemail browser is described in this section.

SCANMail is a system developed for the purpose of providing useful tools for managing and searching through voicemail messages [36.61]. It employs ASR to provide text transcriptions, information retrieval on the transcriptions to provide a weighted set of search terms, information extraction to obtain key information such as telephone numbers from transcription, as well as automatic speaker recognition to carry out caller identification by processing the incoming messages. A graphical user interface enables the user to exercise the features of the system. The caller identification function is described in more detail below.

Two types of processing requests are handled by the caller identification system (CIS). The first type of request is to assign a speaker label to an incoming message. When a new message arrives, ASR is used to produce a transcription. The transcription as well as the speech signal is transmitted to the CIS for caller identification. The CIS compares the processed speech signal with the model of each caller in the recipient's address book. The recipient's address book is populated with speaker models when the user adds a caller to the address book by providing a label to a received message. A matching score is obtained for each of the caller models and compared to a caller-dependent rejection threshold. If the matching score exceeds the threshold, the received message is assigned a speaker label. Otherwise, CIS assigns an *unknown* label to the message.

The second type of request originates with the user action of adding a caller to an address book as mentioned earlier. In the course of reviewing a received message, the user has the capability to supply a caller label to the

message. The enrollment module in the CIS attempts to construct a speaker model for a new user using that message. The acoustic models are trained using text-independent speaker modeling. Acoustic models can be augmented with models based on meta-information, which may include personal information such as the caller's name or contact information left in the message, or the calling history.

36.7 Summary

Identifying speakers by voice was originally investigated for applications in speaker authentication. Over the last decade, the field of speaker recognition has become much more diverse and has found numerous applications. An overview of the technology and sample applications were presented in this chapter.

The modeling techniques that are applicable, and the nature of the problems, vary depending on the application scenario. An important dichotomy is based on whether the content (text) of the speech during training and testing overlaps significantly and is known to the system. These two important cases are the subject of the next two chapters.

References

36.1 J.S. Dunn, F. Podio: Biometrics Consortium website, http://www.biometrics.org (2007)

36.2 M.A. Przybocki, A.F. Martin: The 1999 NIST speaker recognition evaluation, using summed two-channel telephone data for speaker detection and speaker tracking, Eurospeech 1999 Proceedings (1999) pp. 2215–2218, http://www.nist.gov/speech/publications/index.htm

36.3 M.A. Przybocki, A.F. Martin: Nist speaker recognition evaluation chronicles, Odyssey Workshop 2004 Proc. (2004) pp. 15–22

36.4 H. Gish, M.-H. Siu, R. Rohlicek: Segregation of speakers for speech recognition and speaker identification, Proc. ICASSP (1991) pp. 873–876

36.5 L. Wilcox, F. Chen, D. Kimber, V. Balasubramanian: Segmentation of speech using speaker identification, Proc. ICASSP (1994) pp. 161–164

36.6 J.-L. Gauvain, L. Lamel, G. Adda: Partitioning and transcription of broadcast news data, Proc. of ICSLP (1998) pp. 1335–1338

36.7 S.E. Johnson: Who spoke when? – automatic segmentation and clustering for determining speaker turns, Proc. Eurospeech (1999) pp. 2211–2214

36.8 P. Delacourt, C.J. Wellekens: Distbic: A speaker-based segmentation for audio data indexing, Speech Commun. **32**, 111–126 (2000)

36.9 R.B. Dunn, D.A. Reynolds, T.F. Quatieri: Approaches to speaker detection and tracking in conversational speech, Digital Signal Process. **10**, 93–112 (2000)

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

36.11 L.H. Jamieson: Course notes for speech processing by computer, http://cobweb.ecn.purdue.edu ee649/notes/ (2007) Chap. 1

36.12 L.R. Rabiner, B.-H. Juang: *Fundamentals of Speech Recognition* (Prentice-Hall, Englewood Cliffs 1993)

36.13 S. Davis, P. Mermelstein: Comparison of parametric representation for monosyllable word recognition in continuously spoken sentences, IEEE Trans. Acoust. Speech Signal Process. **28**, 357–366 (1980)

36.14 X. Huang, A. Acero, H.-W. Hon: *Spoken Language Processing: A Guide to Theory, Algorithm and System Development* (Prentice-Hall, Englewood Cliffs 2001)

36.15 J. Pelecanos, S. Sridharan: Feature warping for robust speaker verification, Proc. ISCA Workshop on Speaker Recognition – 2001: A Speaker Odyssey (2001)

36.16 B. Xiang, U. Chaudhari, J. Navratil, G. Ramaswamy, R. Gopinath: Short-time Gaussianization for robust speaker verification, Proc. ICASSP, Vol. 1 (2002) pp. 681–684

36.17 S. Furui: Comparison of speaker recognition methods using static features and dynamic features, IEEE Trans. Acoust. Speech Signal Process. **29**, 342–350 (1981)

36.18 J.P. Campbell, D.A. Reynolds, R.B. Dunn: Fusing high- and low-level features for speaker recognition, Proc. Eurospeech, Vol. 1 (2003)

36.19 W. Hess: *Pitch Determination of Speech Signals* (Springer, Berlin, Heidelberg 1983)

36.20 G. Doddington: Speaker recognition based on idiolectal differences between speakers, Proc. Eurospeech (2001) pp. 2521–2524

36.21 W.D. Andrews, M.A. Kohler, J.P. Campbell, J.J. Godfrey: Phonetic, idiolectal, and acoustic speaker recognition, Proceedings of Odyssey Workshop (2001)

36.22 A. Hatch, B. Peskin, A. Stolcke: Improved phonetic speaker recognition using lattice decoding, Proc. ICASSP, Vol. 1 (2005)

36.23 D. Reynolds, W. Andrews, J. Campbell, J. Navratil, B. Peskin, A. Adami, Q. Jin, D. Klusacek, J. Abramson, R. Mihaescu, J. Godfrey, D. Jones, B. Xiang: The SuperSID project: Exploiting high-level information for high-accuracy speaker recognition, Proc. ICASSP (2003) pp. 784–787

36.24 A.E. Rosenberg: Automatic speaker verification: A review, Proc. IEEE **64**, 475–487 (1976)

36.25 K. Fukunaga: *Introduction to Statistical Pattern Recognition*, 2nd edn. (Elsevier, New York 1990)

36.26 A.L. Higgins, L.G. Bahler, J.E. Porter: Voice identification using nearest-neighbor distance measure, Proc. ICASSP (1993) pp. 375–378

36.27 Y. Linde, A. Buzo, R.M. Gray: An algorithm for vector quantization, IEEE Trans. Commun. **28**, 94–95 (1980)

36.28 F.K. Soong, A.E. Rosenberg, L.R. Rabiner, B.H. Juang: A vector quantization approach to speaker recognition, Proc. IEEE ICASSP (1985) pp. 387–390

36.29 D.A. Reynolds, R.C. Rose: Robust text independent speaker identification using Gaussian mixture speaker models, IEEE Trans. Speech Audio Process. **3**, 72–83 (1995)

36.30 D.A. Reynolds, T.F. Quatieri, R.B. Dunn: Speaker verification using adapted Gaussian mixture models, Digital Signal Process. **10**, 19–41 (2000)

36.31 A.E. Rosenberg, S. Parthasarathy: Speaker background models for connected digit password speaker verification, Proc. ICASSP (1996) pp. 81–84

36.32 S. Parthasarathy, A.E. Rosenberg: General phrase speaker verification using sub-word background models and likelihood-ratio scoring, Proc. Int. Conf. Spoken Language Processing (1996) pp. 2403–2406

36.33 O. Siohan, A.E. Rosenberg, S. Parthasarathy: Speaker identification using minimum classification error training, Proc. ICASSP (1998) pp. 109–112

36.34 A.E. Rosenberg, O. Siohan, S. Parthasarathy: Small group speaker identification with common password phrases, Speech Commun. **31**, 131–140 (2000)

36.35 L. Heck, Y. Konig: Discriminative training of minimum cost speaker verification systems, Proc. RLA2C – Speaker Recognition Workshop (1998) pp. 93–96

36.36 A. Rosenberg, O. Siohan, S. Parthasarathy: Speaker verification using minimum verification error training, Proc. ICASSP (1998) pp. 105–108

36.37 J. Navratil, G. Ramaswamy: Detac – a discriminative criterion for speaker verification, Proc. Int. Conf. Spoken Language Processing (2002)

36.38 V.N. Vapnik: *The Nature of Statistical Learning Theory* (Springer, New York 1995)

36.39 W.M. Campbell, D.A. Reynolds, J.P. Campbell: Fusing discriminative and generative methods for speaker recognition: experiments on switchboard and NFI/TNO field data, Proc. ODYSSEY 2004 – The Speaker and Language Recognition Workshop (2004) pp. 41–44

36.40 O. Thyes, R. Kuhn, P. Nguyen, J.-C. Junqua: Speaker identification and verification using eigenvoices, Proc. ICASSP (2000) pp. 242–245

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

36.42 D. Gillick, S. Stafford, B. Peskin: Speaker detection without models, Proc. ICASSP (2005)

36.43 G.N. Ramaswamy, R.D. Zilca, O. Alecksandrovich: A programmable policy manager for conversational biometrics, Proc. Eurospeech (2003)

36.44 H.V. Poor: *An Introduction to Signal Detection and Estimation* (Springer, Berlin, Heidelberg 1994)

36.45 K.P. Li, J.E. Porter: Normalizations and selection of speech segments for speaker recognition scoring, Proc. IEEE ICASSP (1988) pp. 595–598

36.46 F. Bimbot: A tutorial on text-independent speaker verification, EURASIP J. Appl. Signal Process. **4**, 430–451 (2004)

36.47 A. Martin, G. Doddington, T. Kamm, M. Ordowski, M. Przybocki: The det curve in assessment of detection task performance, Proc. Eurospeech (1997) pp. 1895–1898

36.48 A. Martin, M. Przybocki: The NIST 1999 speaker recognition evaluation – an overview, Digital Signal Process. **10**, 1–18 (2000)

36.49 M.A. Siegler, U. Jain, B. Raj, R.M. Stern: Automatic segmentation, classification, and clustering of broadcast news data, Proc. DARPA Speech Recognition Workshop (1997) pp. 97–99

36.50 A.E. Rosenberg, I. Magrin-Chagnolleau, S. Parthasarathy, Q. Huang: Speaker detection in broadcast news databases, Proc. Int. Conf. on Spoken Lang. Processing (1998) pp. 1339–1342

36.51 J.-F. Bonastre, P. Delacourt, C. Fredouille, T. Merlin, C. Wellekens: A speaker tracking system based on speaker turn detection for nist evaluation, Proc. ICASSP (2000) pp. 1177–1180

36.52 A.G. Adami, S.S. Kajarekar, H. Hermansky: A new speaker change detection method for two-speaker segmentation, Proc. ICASSP (2002) pp. 3908–3911

36.53 A.E. Rosenberg, A. Gorin, Z. Liu, S. Parthasarathy: Unsupervised segmentation of telephone conversations, Proc. Int. Conf. on Spoken Lang. Processing (2002) pp. 565–568

36.54 S.S. Chen, P.S. Gopalakrishnan: Speaker, environment and channel change detection and

clustering via the bayesian information criterion, Proc. DARPA Broadcast News Transcription and Understanding Workshop (1998), http://www.nist.gov/speech/publications/darpa98/index.htm

36.55 A. Tritschler, R. Gopinath: Improved speaker segmentation and segments clustering using the bayesian information criterion, Proc. Eurospeech (1999)

36.56 A.D. Gordon: *Classification: Methods for the Exploratory Analysis of Multivariate Data* (Chapman Hall, Englewood Cliffs 1981)

36.57 F. Kubala, H. Jin, R. Schwartz: Automatic speaker clustering, Proc. DARPA Speech Recognition Workshop (1997) pp. 108–111

36.58 D. Liu, F. Kubala: Online speaker clustering, Proc. ICASSP (2003) pp. 572–575

36.59 J.-F. Bonastre, F. Bimbot, L.-J. Boë, J. Campbell, D. Reynolds, I. Magrin-Chagnolleau: Person authentication by voice: a need for caution, Proc. Eurospeech (2003) pp. 33–36

36.60 Voice Identification and Acoustic Analysis Subcommittee of the International Association for Identification: Voice comparison standards, J. Forensic Identif. **41**, 373–392 (1991)

36.61 A.E. Rosenberg, S. Parthasarathy, J. Hirschberg, S. Whittaker: Foldering voicemail messages by caller using text independent speaker recognition, Proc. Int. Conf. on Spoken Language Processing (2000)