Classification-Based Techniques for Digital Coding of Speech-plus-Noise

Khaled Helmi El-Maleh

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

January 2004

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

This thesis is dedicated to my beloved parents, wife, and kids.

Abstract

With the increasing demand for wireless voice services and limited bandwidth resources, it is critical to develop and implement coding techniques which use spectrum efficiently. One approach to increasing system capacity is to lower the bit rate of telephone speech. A typical telephone conversation contains approximately 40% speech and 60% silence or background acoustic noise. A reduction of the average coding rate can be achieved by using a Voice Activity Detection (VAD) unit to distinguish speech from silence or background noise. The VAD decision can be used to select different coding modes for speech and noise or to discontinue transmission during speech pauses.

The quality of a telephone conversation using a VAD-based coding system depends on three major modules: the speech coder, the noise coder, and the VAD. Existing schemes for reduced-rate coding of background noise produce a signal that sounds different from the noise at the transmitting side. The frequent changes of the noise character between that produced during talk spurts (noise coded along with the speech) and that produced during speech pauses (noise coded at a reduced rate) are noticeable and can be annoying to the user.

The objective of this thesis is to develop techniques that enhance the output quality of variable-rate and discontinuous-transmission speech coding systems operating in noisy acoustic environments during the pauses between speech bursts. We propose novel excitation models for natural-quality reduced-rate coding of background acoustic noise in voice communication systems. A better representation of the excitation signal in a noise-synthesis model is achieved by classifying the type of acoustic environment noise. Class-dependent residual substitution is used at the receive side to synthesize a background noise that sounds similar to the background noise at the transmit side. The improvement in the quality of synthesized noise during speech gaps helps in preserving noise continuity between talk spurts and speech pauses, and enhances the overall perceived quality of a conversation.

Sommaire

Avec la demande grandissante pour des services de transmission sans fil de la parole et la diminution de la largeur de bande disponible, il devient primordial de développer et d'implanter des techniques de codage qui utilisent le spectre de façon efficace. Une approche qui permet d'augmenter la capacité d'un système consiste à diminuer le débit binaire des signaux de parole téléphoniques. Une conversation téléphonique typique contient environ 40% de temps de parole et 60% de temps de silence ou de bruit ambiant acoustique. Il est possible de réduire le taux moyen d'encodage en utilisant une unité de détection d'activité de la parole (VAD) qui permet de distinguer la parole du silence ou du bruit ambiant. La décision provenant du VAD peut être utilisée afin de sélectionner un mode d'encodage différent pour la parole et le bruit ou pour interrompre la transmission durant les périodes de parole inactives.

La qualité d'une conversation téléphonique qui utilise un système d'encodage basé sur un VAD dépend de trois principaux modules: l'encodeur de parole, l'encodeur de bruit et le VAD. Les méthodes de codage de bruit à taux réduit existantes produisent un signal qui est perçu différemment de celui qui est transmit. Les changements fréquents des charactéristiques du bruit entre celui produit durant les périodes de parole actives et celui produit durant les périodes de silence (bruit encodé à taux réduit) sont facilement discernables et peuvent être dérangeants pour l'usager.

L'objectif de ce mémoire est de développer des techniques qui augmentent la qualité des systèmes d'encodage de la parole à taux variables et à transmission interrompue durant les périodes de silence et de parole en rafale, opérant en présence de bruit acoustique. Nous proposons un nouveau modèle d'excitation à taux réduit de codage de bruit ambiant acoustique de qualité "naturelle" pour des systèmes de communications de parole. Une meilleure représentation du signal d'excitation dans un modèle de synthèse de bruit est obtenue en classifiant le type de bruit acoustique environnant. Une substitution par classe du résiduel est utilisée au récepteur afin de synthétiser le bruit ambiant de façon à ce que le signal perçu soit semblable à ce qui a été transmis. L'amélioration de la qualité du bruit synthétisé durant les silences aide à préserver la continuité du bruit entre les périodes de parole soudaines et les pauses et améliore la qualité générale d'une conversation.

Acknowledgments

First, I would like to express my deepest gratitude to my supervisor Prof. Peter Kabal for his support, motivation and guidance during the course of my Ph.D. study at McGill University. The financial support provided by Prof. Kabal is gratefully acknowledged.

I would like to thank the Canadian Institute for Telecommunication Research (CITR) and Nortel Networks (Canada) who financially supported this work. The research was conducted in the Telecommunications and Signal Processing (TSP) laboratory at McGill University and I would like to acknowledge the use of their very good facilities.

I would like also to thank my Ph.D. defense committee members: Prof. B. Champagne, Prof. F. Labeau, Prof. M. R. Soleymani, Prof. J. Webb, and Prof. E. Goren for their valuable comments and feedback.

Special thanks to my fellow graduate students of the TSP lab for their friendly and supportive atmosphere. The help of Alexander Wyglinski (at the time of submitting the thesis, and at the time of finalizing it) is greatly appreciated. My gratitude goes to Benoît Pelletier for the French translation of the thesis abstract. I would like to extended my thanks to Mark Klein and Colm Elliott for sharing my interest in sound classification and for the fruitful discussions we had. During my Ph.D. years at McGill I have enjoyed the companionship of Dr. Nader Sheikholeslami Alagha, Dr. Hossein Najafzadeh-Azghandi, and Dr. Jacek Stachurski who were former Ph.D. students in the TSP lab.

I am deeply indebted to my parents, brothers and sisters for their love and support from my birth to date. I will never forget my father who has done everything possible to make me reach a joyful end in my graduate studies. Another important person in my life is my wife Manal. A wife of a Ph.D. candidate (like me) deserves all praises for her endless patience, sacrifice, and encouragement. I can not thank you enough Manal! My lovely kids (Leena, Omar, Muhammad and Mariam), now you can enjoy more time with your dad!

Contents

List of Figures

List of Tables

List of Acronyms

AMR	Adaptive Multirate
CELP	Code Excited Linear Prediction
CDMA	Code Division Multiple Access
CNG	Comfort Noise Generation
DLSF	Differential Line Spectral Frequency
DTX	Discontinuous Transmission
ETSI	European Telecommunication Standardization Institute
EVRC	Enhanced Variable Rate Coder
FCM	Fuzzy c-means
GSM	Global System for Mobile Communications
HOC	Higher-Order Crossing
HOS	Higher-Order Statistics
ITU	International Telecommunication Union
LP	Linear Prediction
LPC	Linear Prediction Coefficients
LSF	Line Spectral Frequency
NN	Nearest Neighbor
NS	Noise Suppression
PDF	Probability Distribution function
QGC	Quadratic Gaussian Classifier
SFM	Spectral Flatness Measure
SID	Silence Insertion Description
SMV	Selectable Mode Vocoder
SNR	Signal-to-Noise Ratio
TOS	Third-Order Statistics
WGN	White Gaussian Noise
VAD	Voice Activity Detection
VBR	Variable Bit Rate
ZCR	Zero Crossing Rate

Chapter 1

Introduction

We start this chapter by providing a brief discussion of some of the basics of speech coding technology. We then discuss the effect of background noise on the performance of speech coders. Next, we state the key reasons that have motivated us to consider our research problem, and then present our research objective and approach to achieve that goal. Section 3 summarizes the contributions of this thesis. Finally, an overview of the remaining chapters of this dissertation is presented.

1.1 Speech Coding: Some Basics

A block diagram of a general speech transmission/storage system is depicted in Figure 1.1. An input speech signal is digitized using an analog-to-digital (A/D) converter. For an 8 kHz sampling frequency and an 8-bit per sample quantizer, a bit rate of 64 kbps is required to transmit/store speech in digital form. A speech encoder receives the output of the A/D unit and produces an output bitstream with a much lower bit rate. The bitstream contains bits of quantized speech parameters. The bitstream is either transmitted through a communication channel or passed to a digital storage device. A decoder translates the received bitstream back to a digital speech signal. Typically a speech decoder performs the inverse operations of its own speech encoder. Finally, a digital-to-analog (D/A) converter transforms the digital samples back to speech [1].

A speech coding scheme can be characterized using the following attributes: coding bit rate, quality, algorithmic delay, robustness, and computational complexity [2] [3]. The first three dimensions are usually given as design requirements while the other two are tied to

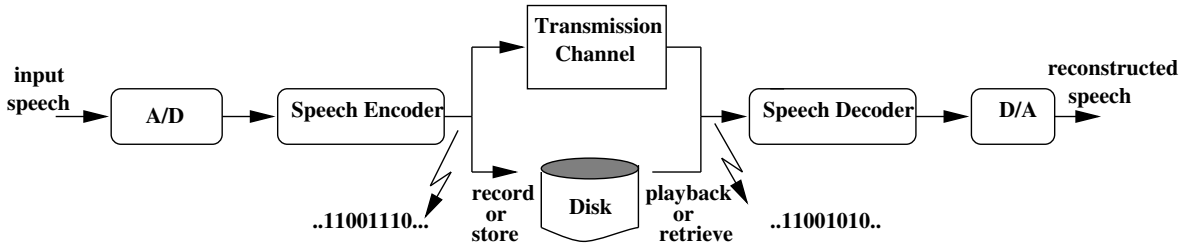

Fig. 1.1 A block diagram of a speech transmission/storage system.

the design of a speech coding algorithm.

Below we provide a brief discussion of each coder attribute:

- *Coding bit rate*
  This is the number of bits per second to represent the speech signal. It is desirable to lower the bit rate needed to digitally represent speech signals. However, lower coding bit rates can result in reduced perceptual quality. Thus, a key challenge in the design of low-bit-rate speech coders is to maintain high quality.

- *Speech quality*
  This refers to the perceived quality of coded speech. As mentioned above maintaining high speech quality is an important requirement for any speech compression system. High quality speech means the output of a speech decoder should sound natural and intelligible with no perceivable coding artifacts. Both subjective and objective measures are commonly used to gauge the performance of a given speech coder.

- *Algorithmic delay*
  For real-time speech communication, it is important to minimize the processing delay of speech coders. Speech coders process input speech on a frame-by-frame basis, where a frame is typically of duration of 5–30 ms. The algorithmic delay of a speech coder is calculated as the sum of the frame size and any delay due to look-ahead. For delay-sensitive speech applications, it is important to use short frames (i.e., 5–10 ms).

- *Robustness (to acoustic background noise and channel errors)*
  Another important feature of a speech coder is its ability to produce 'satisfactory' coded speech quality both in clean and noisy acoustical environments. This is known

as the *acoustical robustness*. Another type of robustness is the ability of speech coders to mitigate the effect of transmission errors on the speech quality after decoding.

- *Computational complexity*
  The computational complexity of speech coders is a function of the algorithm structure and its type. For example, waveform speech coders typically require less computation than parametric or hybrid speech coding algorithms. The amount of program and data memory required by the encoder/decoder is another important design consideration.

1.2 Speech-plus-Noise Coding

A major challenge to designing low-rate speech coders for wireless voice communication is the presence of background acoustic noise (car noise, street noise, office noises like typing or phones ringing, air conditioning noise, music in the background, etc.). The quality of low-bit-rate speech coders suffers when the input speech is mixed with noise [4]. Models of speech production are utilized by low-rate speech coders to achieve compression. These models are not general enough to allow for modelling speech combined with other sounds. For example, a speech-plus-white noise can produce harmonic sounds, and other harmonically rich background noises can cause wavering, squawks, squeaks, chirps, clicks, pops or warbling in the synthesized speech [5]. Such artifacts are annoying to the end user.

Recently, several studies have reported that low-bit-rate model-based speech coders reproduce some structured background noise with annoying artifacts [6] [7] [8] [9]. To remove such artifacts, either special coding modes designed for non-speech inputs [10] [11] or special noise post-processing schemes are used [12] [13] [14] [15].

To reduce the effect of background acoustic noise on the quality of coded speech, it is common to include a noise suppression (NS) unit in speech encoders [16]. Even though noise suppression helps to remedy some of the limitations of model-based coding, NS distortions (timbre changes and loss of signal components, phase distortions and temporal smearing) and artifacts (residual noise such as musical noise) can exacerbate auditory impression rather than enhancing it [17]. Such distortions are especially critical at low signal-to-noise ratios (SNRs). Another side effect of suppressing noise is that users at the far end can lose the awareness of the context of the conversation. Moreover, the conversation will sound

'static'. A recent study by Gierlich and Kettler [18] has shown that the transmission quality of background noise plays a major role for the naturalness of a voice conversation and its overall quality perceived by the end user.

1.3 Research Motivation

In wireless communication, signals travel through the air from a transmitter to the receiver via radio channels. The radio spectrum is a limited, natural resource with radio channels allocated for different communication applications. In the last few years, the demand for wireless communications services has increased tremendously. However, there is no proportionate increase in the bandwidth allocated. The increasing number of cellular telephony users and the limited radio spectrum motivate the need to develop new techniques to enhance the spectral capacity of wireless systems.

Digital speech coding technology utilizes knowledge of both speech waveform redundancy and human perception of sounds to represent voice signals in a compact form. The gain in the reduction of the speech transmission bit rate allows for the accommodation of more users in a given bandwidth.

In addition to speech coding, other techniques for enhanced spectral efficiency have recently been proposed or have been implemented in wireless communications systems. Some examples include: frequency reuse using smaller cells, efficient modulation schemes [19], frequency hopping [20] [21], and smart antennas [22] [23]. In this thesis, our focus will be on capacity enhancement of wireless systems using speech coding.

A limiting factor to the efficient use of the radio spectrum is the co-channel interference resulting from other signals in the same service area. In wireless communications systems, co-channel interference reduces the efficiency of the system by degrading both the quality of service and the use of bandwidth. One factor that can help in reducing the effect of interference is to lower the power of transmitted information. In general, reducing the transmission bit rate translates into a reduction of the required power level. Thus, reducing the coding bit rate of speech can help in minimizing the effect of co-channel interference.

1.3.1 Speech Pauses and Spectral Efficiency

Telephone speech generally takes the form of a two-way conversation between two parties. In a typical conversation, each user talks for about 40% of the time. The other 60%

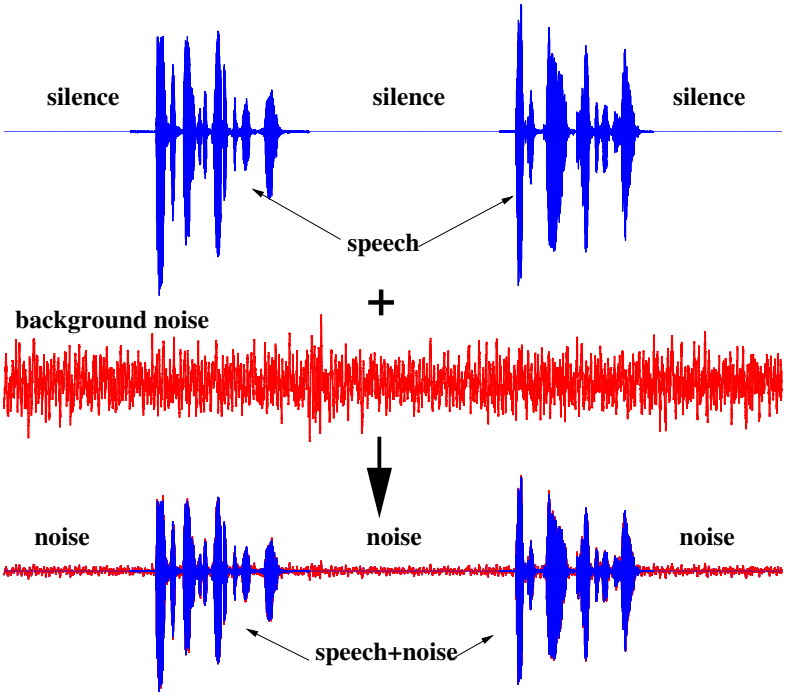

Fig. 1.2 Illustration of the on-off patterns of conversational speech.

of the time includes listening and speaking pauses [24]. Between the speech bursts, the silence is filled with environment acoustic noise, such as those in a car, street, restaurant, and office. Figure 1.2 shows the on-off patterns of conversational speech[1]. During the speech pauses, the transmitter is simply being used to send background noise information to the receiver. Background acoustic noise carries less perceptual information than speech and thus using the full speech coding bit-rate to transmit background noise is a waste of bandwidth resources. Substantial savings in average bit rate could be achieved if speech pauses could be detected and coded at a much lower rate than used for coding speech.

The first voice transmission system to exploit speech pauses was the Time Assignment Speech Interpolation (TASI), proposed in the 1950s to increase the capacity of the transatlantic telephone cables [25]. A digital version of the system, known as Digital Speech Interpolation (DSI) was later developed for band-limited satellite systems [26]. In a DSI system, a user is connected to a channel only for the duration of a speech burst, rather than

[1]The noise signal in the figure has been scaled down and then added to the speech signal to give a 15 dB speech-plus-noise signal.

for the whole conversation. Using statistical multiplexing techniques, DSI-based telephony allows dynamic time sharing of resources among multiple simultaneous users [27].

A wireless communication system can benefit from the low voice activity to increase the spectral efficiency and to prolong the battery life of mobile terminals. The first cellular system to use a discontinuous transmission (DTX) mode during the silent periods was the Global System for Mobile Communications (GSM) [28] [29]. Switching the transmitter off for more than 50% of a telephone call reduces the power consumption of portable units, and approximately doubles the life time of the battery. Moreover, freeing the radio channel from transmitted signals during the absence of speech reduces the co-channel interference [30].

Other wireless systems require a continuous mode of transmission for system synchronization and channel monitoring. During absence of speech, a lower rate coding mode is used to encode background noise. An example is a Code Division Multiple Access (CDMA) wireless communication system [31] [32]. In CDMA-based systems, speech is transmitted using variable bit rate (VBR) coding strategy with high rates for coding speech and a reduced coding rate (below 1 kbps) to transmit background noise [33] [34]. The reduced transmission power during the inactive periods enhances CDMA system capacity by allowing for reduced power and hence reduced interference [35].

Multimedia communications systems can also use the speech pauses for the digital simultaneous transmission of voice and data (DSVD). During the silent periods, the channel resources are reallocated to transmit data such as text, fax, images, and video. Recently, the International Telecommunication Union (ITU) has standardized a silence compression scheme known as G.729 Annex B for DSVD applications [36]. Another standard G.723.1 has been proposed for low-bit-rate multimedia applications such as audio conferencing and video-telephony [37] [38].

Figure 1.3 shows a general voice transmission system with voice activity detection (VAD). Two major issues have to be considered in the design of a voice communications system exploiting the silence portion of speech: the accurate detection of the speech activity, and means to deal with the background noise during silent periods. A well-designed VAD enables the accurate detection of speech bursts and avoids misclassification of noise as speech. Chapter 2 will be devoted to the voice activity detection problem with a detailed discussion of the various aspects of VAD design.

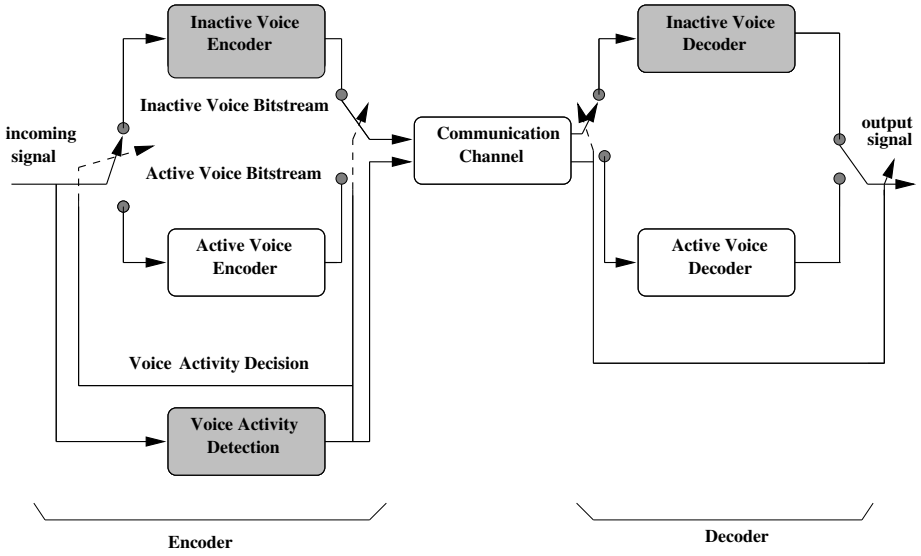

Fig. 1.3 Voice transmission system with voice activity detection.

1.3.2 Background Noise Coding

Removing the background noise between speech bursts has an undesirable effect on the perceived quality. Background noise transmitted during speech activity disappears during the interruption of transmission. This results in on-off switching artifacts that can be unpleasant and disconcerting to the listener. A common solution to alleviate the annoyance and discomfort to the listener is to generate a synthetic noise signal at the receiving end to fill the gaps between the speech bursts. This is known as *comfort noise* [39]. Different approaches have been proposed to design a comfort noise generator (CNG) [40] [41] [42] [43]. A common approach is to transmit a periodic update of the noise statistics using what is known as silence insertion descriptor (SID) frames [44].

In the present generation of background noise coding and comfort noise insertion systems, a simple excitation-filter model is used for noise synthesis. A signal is modelled as the output signal of a filter excited by a source signal [45] [46]. Existing schemes for background noise coding and comfort noise generators fail to regenerate background noise with natural quality during speech inactivity [47] [48]. However, when speech is present and being coded at the full rate, incidental noise will be coded along with the speech. The change in the character of the noise between speech activity and speech pauses is noticeable and can be annoying.

Not much attention has been given to the design of the noise coding mode in existing VBR speech coders [48]. Most of the efforts have been focused on designing coding modes for the different phonetic classes of speech [24] [49] [50] [51]. In this thesis, our focus will be on the design of enhanced-quality noise coding mode for VBR speech coders.

1.4 Research Objective

The main objective of this thesis is to develop techniques that can enhance the output quality of variable-rate and discontinuous-transmission speech coding systems operating in noisy acoustic environments. The focus of our work will be on designing enhanced background noise coding schemes at very low bit rates (below 1 kbps).

Our approach to achieve our objective is summarized below:

- Investigate the limitations of the existing noise synthesis models, and focus on low-bit-rate modelling of the linear prediction (LP) residual of background noise.
- Propose enhanced noise synthesis models with only minor changes to existing noise coding schemes, with a possibility of changes only at the decoder.
- Use classification techniques as a means to capture important perceptual information of background noise sounds.
- Develop noise coding schemes for VBR and DTX-based speech coders that have the same bit-rate requirements as existing solutions.

1.5 Research Contribution

This thesis explores novel excitation models to encode background noise signals with natural quality at very low bit rates. The major research contributions are summarized below:

- Assessment of VAD schemes for use in systems which have separate modes for speech and background noise.
- A novel scheme called *class-dependent residual substitution* is proposed to enhance the synthesis of background noise using very low bit rates. This scheme can be implemented with only minor changes to the encoder/decoder of existing noise coders (Section 4.1, [52], [53], [54], and [55]).

- A noise mixture excitation model is developed as a generalization of the class-dependent residual substitution model. Soft-decision classification techniques are used to estimate the mixing weights of the mixture model (Section 4.2, Section 5.9, and [53]).

- Noise classification is used as a major tool in our novel excitation models ([56] [57]). Classification is used as a means to transmit vital 'perceptual' excitation information to the noise synthesis model at the receiver. Low-complexity robust noise classification schemes are presented in Chapter 5 using both hard-decision and soft-decision classification techniques.

- Line spectral frequencies (LSFs) are shown to be a robust feature set in distinguishing various types of background noise, in distinguishing speech from noise, and for speech/music discrimination (Section 5.7, Section 5.12, [56], and [58]).

- In Chapter 3, different statistical tools (kurtosis, higher-order crossings, quantile-quantile plot, and spectral flatness measure) are used to study the Gaussianity and whiteness properties of the linear prediction residual of some environmental background noises.

- In Section 5.12, a low-complexity frame-level speech/music discrimination system will be presented that requires only a frame delay of 20 ms.

1.6 Dissertation Outline

This dissertation is organized as follows:

In Chapter 2, we present an overview of the design issues crucial to the efficient use of the speech pauses in voice communications systems. Various models of the statistical nature of conversational speech are reviewed. A large portion of the chapter is devoted to discuss the voice activity detection problem. Finally, we discuss the results of our comparative performance study of two recently-standardized VAD algorithms under various noise conditions.

In this work, linear prediction analysis and synthesis models have been used for reduced-rate coding of background acoustic noise. Chapter 3 reviews the basics of the linear prediction coding paradigm. We present different strategies to improve the modelling of the linear

prediction residual of background acoustic noise. This chapter will serve as the background material for the remaining chapters of the dissertation.

In Chapter 4, a formulation of our proposed noise coding scheme is presented. Class-dependent residual substitution is presented with a discussion of the basic concepts, and the concept-validation experiments. A general excitation model is presented based on a noise mixture assumption.

Noise classification is an important tool in our proposed noise coding scheme. In Chapter 5, we dedicate the chapter to discuss in details the steps required to design a noise classification system. Experimental classification results are presented using different signal features and different classification algorithms. In addition to hard-decision classification, we discuss the noise mixture classification problem and we propose novel methods to apply soft-decision classification approach to a residual mixture substitution model. At the end of the chapter we present our work in designing a low-complexity frame-level speech/music discrimination system that requires only a frame delay of 20 ms.

Finally, in Chapter 6 we draw conclusions from our work, summarize our contributions, and then discuss future related work items.

Chapter 2

Speech Pause Detection

Speech is a sequence of alternating short intervals of speech energy (called *talkspurts*) and silence gaps. In conversational speech, there are two major kinds of pauses: speaking pauses and listening pauses. The speaking pauses occur while a person is talking and are between words and syllables (*short* speaking pauses), or between phrases and sentences (*long* speaking pauses). The time duration of speaking pauses is generally shorter than listening pauses which occur when the speaker is listening to the other party.

Accurate modelling of the on-off patterns of conversational speech is essential for the design and analysis of systems exploiting speech pauses. Moreover, automatic discrimination between speech and silence is a major issue for the efficient use of the speech pauses. This chapter presents an overview of the design issues that are important in the efficient use of speech pauses in voice communications systems. It starts by reviewing major speech activity models. A discussion of several important parameters that characterize conversational speech is presented followed by a method to generate artificial dialog speech. A good portion of the chapter will be devoted to describing voice activity detection (VAD) design issues and performance evaluation techniques. Finally, we present the results of a comparative study of the performance of two state-of-the-art VAD algorithms under various noisy conditions.

2.1 Modelling the On-Off Patterns of Conversational Speech

2.1.1 Brady's Speech Activity Model

In 1969, Brady proposed a six-state model to describe the on-off patterns of conversational speech [59]. In his experiments, he used a simple threshold-based speech detector to discriminate talkspurts from pauses. The model is shown in Figure 2.1. It describes the dynamics of the interaction of speakers A and B engaged in a conversation. The six states are divided equally among the three major scenarios: one speaking-one listening, mutual silence, and double talk. Only the state transitions shown in the figure are allowed. For example, the transition from the state (B talks, A silent) to the state (mutual silence, A spoke last) is not allowed. In [59], empirical state transition values, for the model, were measured using a large database of conversational speech. In his model, Brady did not consider the effects of silence gaps shorter than 200 ms and talkspurts shorter than 15 ms. Recently, Stern *et al.* [60] proposed modifications to Brady's model to include the effects of the short speaking pauses while preserving the effects of the longer silence and the dynamics of the interaction between speaking parties. The modified model has the 6 states of Brady's model in addition to two new states to represent the short syllabic speaking pauses. The modified model provides a tool for more accurately assessing the performance of new-generation wireless communications systems.

2.1.2 Characteristics of Talkspurts and Pauses

Several studies have been performed to characterize the statistics of talkspurts and pauses. Brady modelled the probability density function (PDF) of talkspurts by an exponential distribution and that of pauses by a constant-plus-exponential distribution [59]. Using measurement from monologue speech, Gruber [61] modelled the PDF of talkspurts durations by a geometric PDF, and that of silence durations by two weighted geometric PDFs. Using a database of 50 minutes of telephone speech, Lee and Un [62] modelled the PDFs of silence and talk by two weighted geometric functions. In the sequel, we focus our discussion on the Lee-Un model as it is used in the International Telecommunication Union (ITU) recommendation (P.59) for the generation of artificial conversational speech [63]. In their work, PDFs were estimated using a speech detector with no hangover frames[1].

[1]Hangover frames are used to prevent a pre-mature transition from active speech to silence.

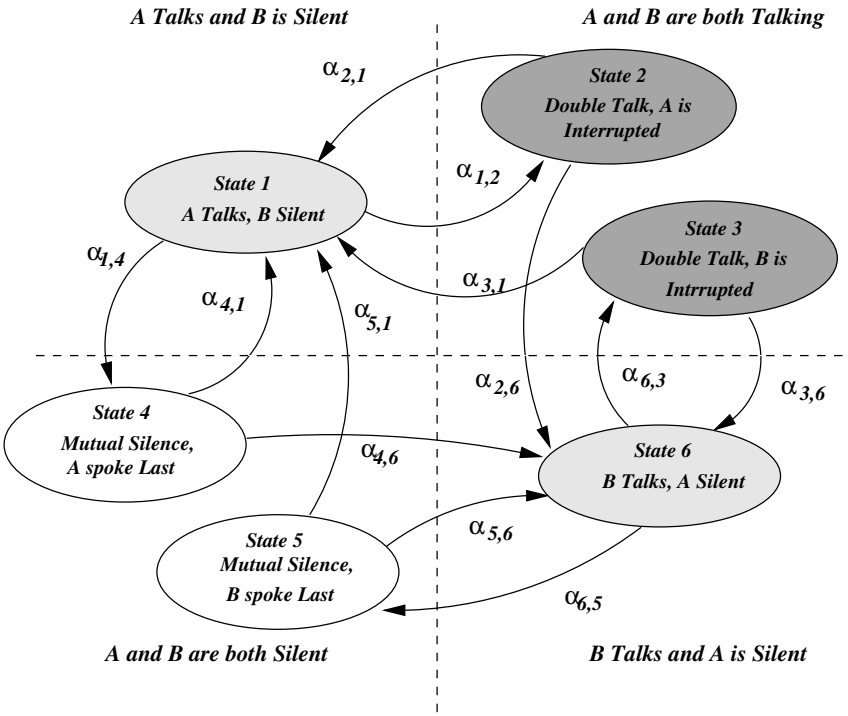

Fig. 2.1 The Brady six-state model for the on-off characteristics of conversational speech [59].

The measured talkspurt PDF is given by:

$$f_T(k) = C_1(1-u_1)u_1^{k-1} + C_2(1-u_2)u_2^{k-1}, \qquad k = 1, 2, \ldots, \tag{2.1}$$

where $C_1 = 0.60278$, $C_2 = 0.39817$, $u_1 = 0.92446$, and $u_2 = 0.98916$. The PDF for silence durations was modelled as a sum of two weighted geometric PDFs:

$$f_S(k) = K_1(1-w_1)w_1^{k-1} + K_2(1-w_2)w_2^{k-1}, \qquad k = 1, 2, \ldots, \tag{2.2}$$

where $K_1 = 0.76693$, $K_2 = 0.23307$, $w_1 = 0.89700$, and $w_2 = 0.99791$.

In the above equations, each increment of the variable k represents a 5 ms frame. For the two PDFs shown in Figure 2.2, the mean talkspurt duration is 227 ms, and the average pause duration is 596 ms. Using these mean values, the long-term speech activity factor (SAF) is 27.6%. Other studies have shown that a typical SAF for conversational speech is between 27% and 40%. The SAF depends on the sensitivity of the speech detector

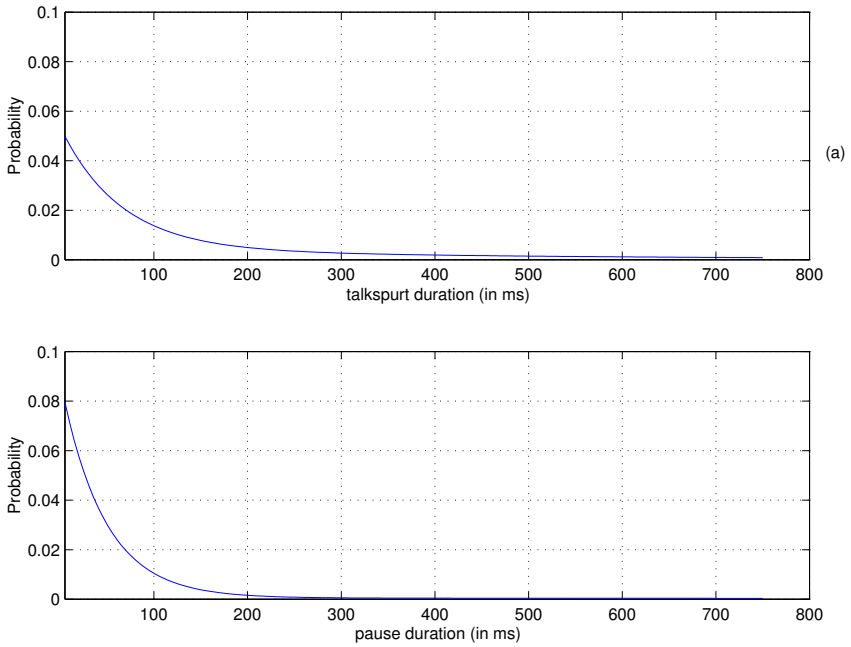

Fig. 2.2 A plot of the PDFs of the durations of talkspurt and pauses in conversational speech [62].

algorithm, and whether hangover frames are used or not [64]. Another parameter that characterizes conversational speech is the *average talkspurt rate*. It is defined as the inverse of the sum of mean talkspurt and the mean silence durations. The mean talkspurt rate also varies depending on the used voice detection device and takes values from 14 to 72 talkspurts/minute for each end of a conversation [62]. Averaged parameters for conversational speech from 3 major languages are shown in Table 2.1 [63].

2.1.3 Generating Artificial Conversational Speech

During the design phase of speech processing systems with speech pause detectors, it is necessary to have a large database of telephone-speech conversations. This requires the collection of speech signals that can cover important operating conditions of the system. For example, both clean and noisy telephone-speech recordings are needed to evaluate the robustness of a VAD algorithm.

It is not always easy to gather all these data and thus it is desirable to generate artificial conversational speech for simulation purposes. The ITU Recommendation P.59 presents

Table 2.1 Temporal parameters in conversational speech (average for English, Italian, and Japanese) [63].

Parameter	Duration (sec.)	Rate (%)
Talkspurt	1.004	38.53
Double talk	0.228	6.59
Pause	1.587	61.47
Mutual silence	0.508	22.48

a method for generating artificial telephony speech signals [63]. These generated signals include characteristics of human conversational speech such as the duration of the talkspurt, pause, double talk and mutual silence. Talkspurts and pauses are generated using a 4-state transition model shown in Figure 2.3. This model is a simpler version of Brady's model presented in Figure 2.1. In this model, three probabilities $P_1 = 40\%$, $P_2 = 50\%$, and $P_3 = 50\%$ are needed to represent the transition statistics between the 4 states. The durations of single talk (T_{ST}), double talk (T_{DT}), and mutual silence (T_{MS}) are varied using the following equations:

$$T_{ST} = -0.854 \ln(1 - x_1), \tag{2.3}$$

$$T_{DT} = -0.226 \ln(1 - x_2), \tag{2.4}$$

$$T_{MS} = -0.456 \ln(1 - x_3), \tag{2.5}$$

where x_1, x_2, and x_3 are random variables with uniform distribution in [0,1].

To generate artificial speech to fill the 'talk' periods, the ITU Recommendation P.50 [65] can be used. In this method, a time-varying spectral shaping filter is excited by either a periodic or random noise excitation depending on a voiced/unvoiced decision [66]. The frequency response of the spectral shaping filter simulates the transmission characteristics of the vocal tract. This model for generating artificial voices is similar to the Linear Predictive (LP) vocoder [67].

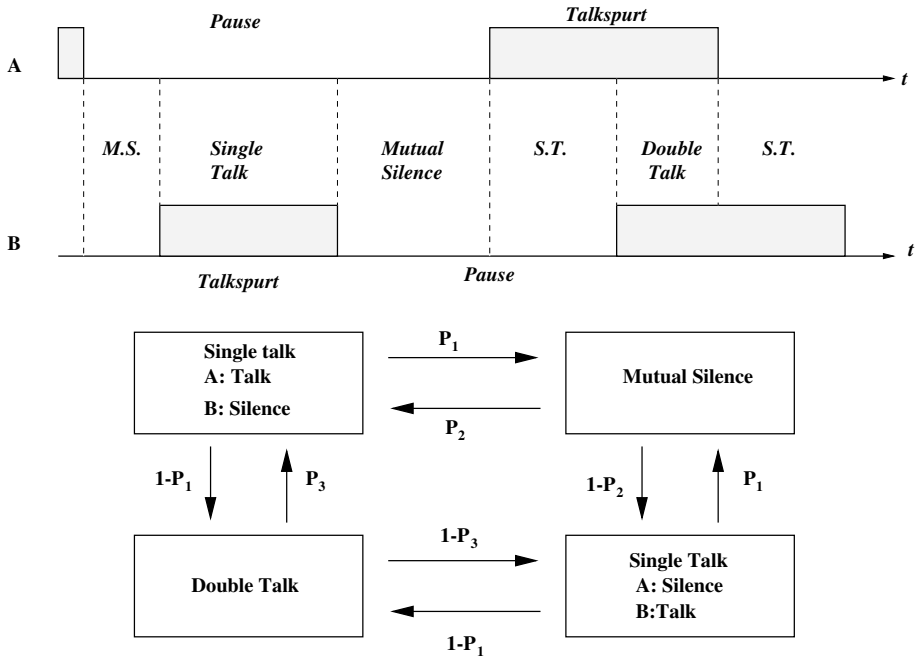

Fig. 2.3 State transition model for two-way voice conversation.

2.1.4 Measuring Speech Activity

The accurate measurement of the level of speech signals is an important step in many speech processing applications. For example, in assessing the performance of speech coders in noisy acoustical environments under different noise levels, clean speech signals are digitally mixed with different types and levels of acoustic noises. For accurate measurement of the active level of speech signals, silence gaps have to be excluded from the computation of the energy of the signal.

The ITU has recommended an adaptive algorithm to decide whether a speech segment is active or inactive. This is known as ITU Recommendation P.56 [68] [69]. The algorithm calculates an envelope waveform such that pauses shorter than 100 ms are included in the calculation, and pauses longer than 350 ms are excluded. Short gaps between speech bursts are considered part of the active signal. A threshold level of 15 dB below the root-mean-square level of the signal is used to separate active speech from noise. An implementation of the P.56 algorithm is defined as a *speech voltmeter* tool in the ITU-T Software Tool Library [70].

In noisy environments, the speech voltmeter falsely considers noise segments as speech

activity. Thus, the P.56 algorithm should be used only for signals with a high signal-to-noise ratio (SNR). To demonstrate this, we show in Table 2.2 the measured speech activity for simulated one-way conversational speech signals in clean and noisy environments[2]. A comparison is made between a hand-labelled speech activity of 41.3% and the speech-activity measurements using P.56, and the two VADs of the Selectable Mode Vocoder (SMV) [71] [72]. Several observations can be made from the data. First, speech activity values in the clean speech condition are close to the 41.3% value, but lower by a few percent. The main reason for this difference is that practical VADs do not consider short pauses between syllables words as active speech. In noisy conditions, the values computed by P.56 are not accurate. The VADs vary in their speech detection capability as shown in the table[3]. A more detailed analysis of the performance of the two VADs in noisy conditions will be given in Section 2.3.

Table 2.2 A comparison between speech activity measurements (%) using P.56 and SMV VADs

Input Signal	Speech Activity P.56	Speech Activity SMV VAD-A	Speech Activity SMV VAD-B
clean speech	35.1	38.6	32.3
speech-plus-car noise	92.7	43.5	31.3
speech-plus-babble noise	68.2	53.2	40.7
speech-plus-street noise	45.0	39.4	26.8

2.2 Voice Activity Detection

2.2.1 Problem Formulation

Conversational speech is a sequence of consecutive segments of silence and speech. In noisy environments, background acoustic noise contaminates the speech signal resulting in either speech-plus-noise or noise-only. In many speech processing applications, it is important to discriminate speech from noise. This process is called *voice activity detection* (VAD). The VAD operation can be viewed as a decision problem in which the detector decides

[2]For the noisy speech signals, a 15 dB level was used.
[3]Babble noise results from a large number of simultaneous talkers.

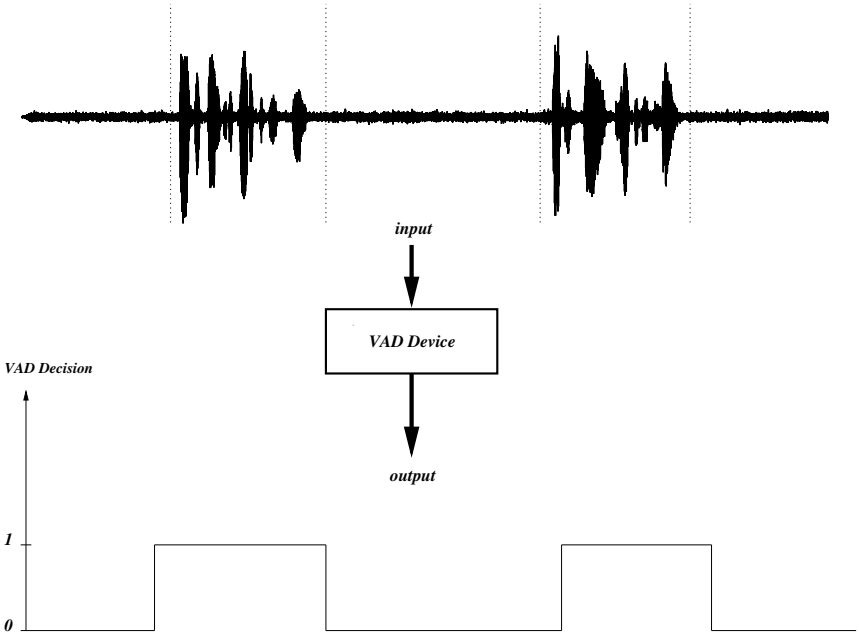

Fig. 2.4 An example of an ideal VAD operation.

between noise-only or speech-plus-noise. This is a challenging problem in noisy acoustical environments.

The basic principle of a VAD device is that it extracts measured features or quantities from the input signal and then compares these values with thresholds, usually extracted from noise-only periods. Voice activity (VAD=1) is declared if the measured values exceed the thresholds. Otherwise, no speech activity or noise (VAD=0) is present. VAD design involves selecting the features, and the way the thresholds are updated. Most VAD algorithms output a binary decision on a frame-by-frame basis where a "frame" of the input signal is a short unit of time such as 5–40 ms. Accuracy, robustness to noise conditions, simplicity, adaptation, and real-time processing are some of the required features of a good VAD. Figure 2.4 shows an example of an ideal VAD operation.

A general block diagram of a VAD design is shown in Figure 2.5. An important step in the design is the selection of a 'good' set of decision features. In the early VAD algorithms, short-time energy, zero crossing rate, and linear prediction coefficients were among the common features used in the detection process [73]. Cepstral coefficients [74], spectral entropy [75], a least-square periodicity measure [76], wavelet transform coefficients [77] are examples of recently proposed VAD features.

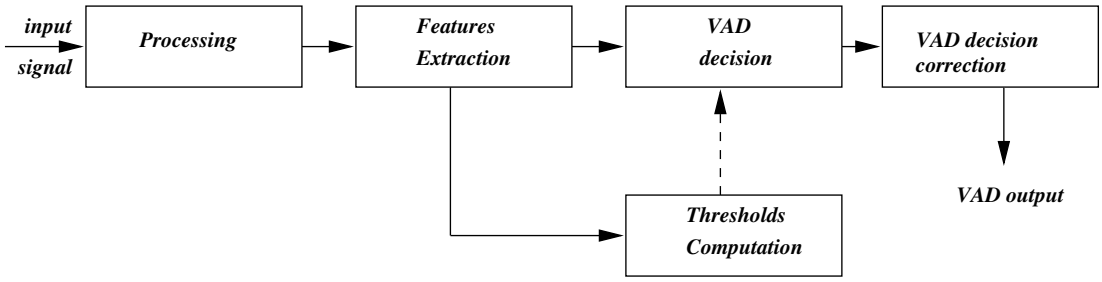

Fig. 2.5 A block diagram of a basic VAD design.

The accuracy and reliability of a VAD algorithm depends heavily on the decision thresholds. Adaptation of thresholds values helps to track time-varying changes in the acoustic environments, and hence gives a more reliable voice detection result. A VAD decision can be improved by using post-decision correction schemes. In [78] we presented a scheme that effectively corrected isolated VAD errors.

2.2.2 VAD Algorithms: A Literature Review

VAD devices are used in many speech processing systems. Background acoustic noise is a major source of performance degradation of such systems. Accurate detection of noise-only frames in a noisy speech signal is indispensable for reducing the effect of noise in many speech applications such as speech enhancement, speech recognition, and speech coding [24].

An important use of voice activity detection is in silence compression schemes for multimedia and wireless voice applications [37]. In a device with a silence compression mode, speech pauses are detected using a VAD unit and then either the bit rate is reduced or the channel is reallocated to other concurrent applications (i.e., data, video, images etc.) [38]. In Chapter 1, we have emphasized the importance of VAD algorithms to enhance spectral capacity of wireless voice transmission systems.

For reliable operation of single-microphone speech enhancement (i.e., noise reduction) systems in non-stationary noisy environments, it is critical to update the noise spectrum estimate [79] [80]. Two approaches have been proposed in the literature to track the noise spectrum. In the first approach, the noise spectrum is only updated during noise-only periods. This requires using a noise detection (i.e., VAD) algorithm. The other approach avoids the use of a VAD unit by continuously updating the noise statistics. Spectral

subtraction is a well-known noise reduction technique that typically uses a VAD algorithm for its operation. Recently, Fischer and Stahl [81] examined operating spectral subtraction without a VAD by continuously updating the noise estimate. They observed that the noise estimate was not reliable and this resulted in performance degradation of the reduction algorithm. They concluded that voice activity detection plays an important role in noise reduction systems.

Another important application of VAD algorithms is speech recognition. It has been reported that a major cause of errors in automatic speech recognition is the inaccurate detection of the end-points of speech bursts [82] [83]. Also, VAD is used to disable speech recognition for non-speech noise input signals.

The literature has seen many proposals of new VAD designs. Maintaining a reliable voice activity detection in harsh (i.e., SNRs below 10 dB) acoustic noise environments is still a major design challenge. We can classify existing VAD algorithms into four major categories: energy-based VADs, pattern recognition VADs, statistical model-based VADs, and higher-order statistics VADs. In each category, different decision rules are used, combined with different sets of VAD features.

A promising VAD category is statistical model-based VAD algorithms. Assuming that each spectral component of speech and noise signals have complex Gaussian distribution, Sohn *et al.* [84] [85] proposed a VAD based on the likelihood ratio test. A novel part of this scheme is its soft-decision based noise spectrum estimation. Cho and Kondoz [86] analyzed this VAD algorithm and proposed improvements to minimize the number of detection errors in the transitional regions of speech. In a recent paper [87], a soft-decision model-based VAD has been proposed.

A new class of VAD algorithms is the ones using higher-order statistics (HOS). Rangoussi *et al.* exploited the different properties of the third-order statistics (TOS) of both speech and noise signals to design a robust VAD algorithm for speech recognition [88] [89]. Also, TOS was used in [90] to design an improved speech endpoint detection system in noisy environments. Recently, Nemer *et al.* [91] proposed an algorithm that combined HOS and energy features for speech/noise detection.

In the category of pattern-recognition VADs, Beritelli *et al.* [92] [93] presented a VAD using fuzzy logic. Supervised learning was used to design the VAD fuzzy decision rules. The performance of this fuzzy VAD has been shown to outperform G.729 and GSM VADs (described below) for low SNR conditions.

In recent years, several VAD algorithms have been standardized by major international bodies for wireless multimedia applications. The European Telecommunication Standardization Institute (ETSI) has standardized a VAD for GSM voice communication. This is known as the GSM VAD [94] and is considered to be the first standardized VAD for cellular telephony. This VAD accompanies GSM speech coders to enhance spectral efficiency of GSM systems by using discontinuous transmission. Recently, ETSI has also standardized two VAD options for the Adaptive Multirate (AMR) speech coder [95]. All the VADs for the GSM system belong to the energy-based VAD category [96].

For CDMA cellular systems, an energy-based VAD was developed in 1996 for the Enhanced Variable Rate Codec (EVRC) (known as EVRC VAD) [46]. Recently, two new VAD options have been selected for SMV (the new speech coder for CDMA systems) [71]. In Section 2.3 we will present a detailed comparison study of SMV VADs.

The ITU-T has also standardized a VAD algorithm for the 8 kbps G.729 speech coder. This is known as the G.729 VAD [36]. It uses statistical pattern recognition decision rules. This VAD is commonly used for voice over IP (VoIP) applications and often is considered as a reference VAD when comparing new VAD designs [97].

In addition to the aforementioned main VAD categories, other methods have been proposed in the literature. Doukas *et al.* [98] presented a VAD algorithm using source separation techniques. Spectral entropy has been proposed in [75] to determine regions of voice activity in noisy speech signals. Neural-network recognition capability has also been exploited to detect speech from noise [99] [100]. To improve the robustness of VAD algorithms for non-stationary noises and at very low SNRs (below 0 dB), decision fusion methods have been proposed in [101] to combine the output decisions of two different VAD algorithms.

2.2.3 VAD Performance Evaluation Methods

A desirable goal in designing a VAD algorithm is to minimize the probability of decision errors under a variety of noise conditions. A VAD error can take the form of speech clipping (detecting speech as noise) or false alarm (detecting noise as speech). The VAD clipping errors affect the quality of reconstructed speech signal, while the noise-detection errors reduce the utilization of the speech pauses by increasing the speech activity factor.

During a VAD evaluation phase, it is common to define a vector of "ideal" VAD flag

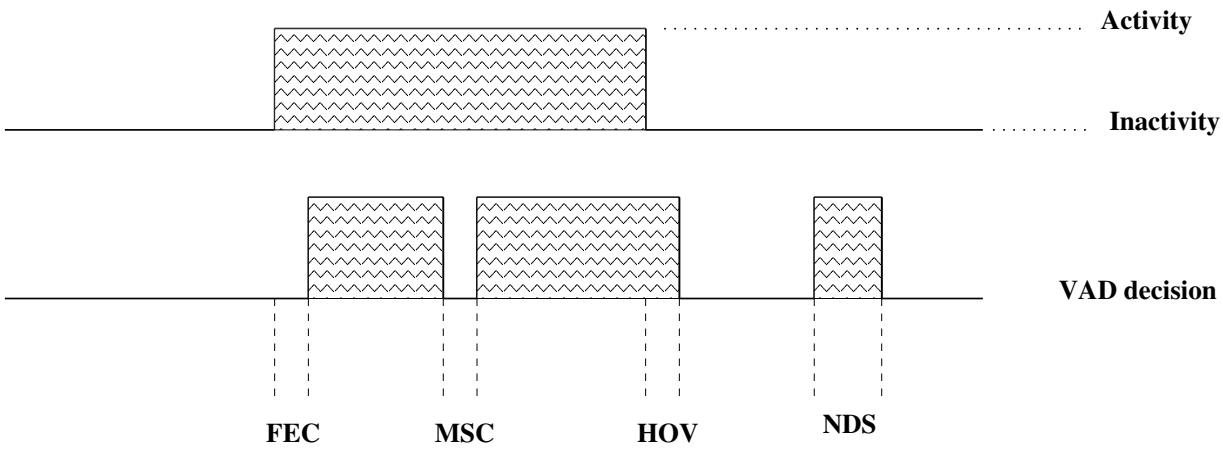

Fig. 2.6 VAD errors.

for each test signal. To create an ideal VAD sequence, a simulation of a one-way conversational speech signal is generated by using the method described in Section 2.1.3 or by concatenating noise-free speech bursts with silence gaps. Then, this signal is hand-labelled (i.e., VAD=1 for speech, and VAD=0 for silence). To evaluate the effect of noise conditions on the robustness and reliability of the speech detection algorithm, acoustic noise signals are added to the clean speech signal with different signal-to-noise ratios [102]. The ITU Software Tool Library [70] provides several tools for a proper calibration of speech and noise levels. A reasonable strategy to gauge the performance of a VAD in noisy conditions is to take the VAD decision of the clean speech signal as a reference[4]. Then, for a noisy test condition, the probability of detection (P_d), and probability of false alarm (P_f) for each test are computed by comparing the VAD decisions (clean vs. noise). These two parameters give an initial figure-of-merit that can be used for further 'tuning' of the VAD parameters. In [94], four objective measures were defined to evaluate the performance of a given VAD. These parameters are shown in Figure 2.6 and are outlined below:

- *Front End Clipping* (FEC): clipping introduced in passing from noise to speech activity,
- *Mid-speech Clipping* (MSC): clipping due to speech misclassified as noise,

[4]It is our experience that most VADs are capable of almost 'ideal' detection of speech in high SNR conditions.

- *Hangover* (HOV): errors due to the VAD flag remains active in passing from speech activity to noise,

- *Noise Detected as Speech* (NDS): noise interpreted as speech within a silence period.

The noise errors (NDS+HOV) quantify the accuracy of a VAD algorithm while speech errors (FEC+MSC) relate directly to the subjective quality of a VAD. Clipping errors during a conversation can severely impair the speech intelligibility. Additional subjective tests are commonly used to assess the effect of VAD errors on quality. Special attention is usually given to the audibility of clipping and the overall quality of the reconstructed signal.

Recently, Beritelli *et al.* [103] proposed new performance evaluation criteria for VAD algorithms. They defined three types of speech clipping errors based on the phonetic class of speech (voiced, unvoiced or mixed voicing). These clipping sub-classes are: VDN (voiced speech detected as noise), MDN (mixed voiced detected as noise), and UDN (unvoiced speech detected as noise). The degradation effect of speech clipping errors is more perceivable if the VAD errors occur in bursts. Thus, the duration of clipping errors should be taken into account during VAD evaluation. In [104], a new psychoacoustic parameter (*Activity Burst Corruption*) was defined to capture the perceptual effect of VAD errors using loudness measures. It was reported that this measure provides a good correlation between the objective and subjective VAD evaluation results.

2.3 A Performance Study of SMV VAD Algorithms

In an earlier study of VAD algorithms we evaluated the performance of three recent VAD algorithms under various acoustical background noise conditions [78]. We considered the VAD used in the GSM cellular system [94] [105], the EVRC VAD used in the North American CDMA-based cellular systems [46], and a third-order statistics (TOS)-based VAD [89]. The results of this study have shown a consistent superiority of both the EVRC and the TOS VADs when compared with the GSM VAD [78].

In a recent study, Beritelli *et al.*, [92] [93] compared the performance of the two AMR VADs, the ITU G.729 VAD and their own fuzzy logic VAD. In their work, both subjective and objective performance measures were used to compare the VADs under different noise conditions, and with different speech languages. The results of their work showed that the

AMR VADs outperformed the G.729 and the fuzzy VADs under the various test conditions. The G.729 VAD had the worst error performance and its errors resulted in more quality degradation.

In this section we present the results of a study that we performed to compare the two VADs of SMV [71]. These two VADs are considered state-of-the-art and thus we want to assess their performance in various noise conditions. The effect of noise suppression (as a pre-encoding processing unit) on the performance of the two VADs will be also assessed.

SMV is a multi-mode variable rate speech coder [71]. An important part of this coder is the voice activity detection unit. During the standardization phase of SMV, two VADs have been provided as options: SMV VAD-A and SMV VAD-B. SMV VAD-A uses a set of ad-hoc decision rules that use both spectral and periodicity features to distinguish speech from noise [71]. SMV VAD-B divides the spectrum of the input signal into two bands and the energy in each band is compared against two thresholds. Speech is detected if the energy in each band is greater than the corresponding lowest threshold. The thresholds are scaled versions of estimated sub-band noise energies from previous frames[5].

2.3.1 Experimental Set-up

To prepare the test speech material for our evaluation study we followed the Third Generation Partner Project 2 (3GPP2) speech-processing test plan for the SMV selection phase [106]. A clean-speech sequence of 8 male and female talkers (of duration of 8 minutes) was used with silence gaps between speech sentences. The speech activity factor for this speech file is around 75%. Three different signal power levels were used: -16 dBov[6] (high level), -26 dBov (nominal level), and -36 dBov (low level). Three commonly encountered acoustic noises were considered: car, street and babble. Background noise was digitally added to clean speech with SNR values of 15 and 20 dBs[7].

2.3.2 Simulations Results

To measure the performance of each VAD, we define its VAD decision in clean nominal condition as the reference VAD (VAD_{ref}) and we consider the VAD decisions in other

[5]This VAD is an enhanced version of the EVRC VAD.
[6]Level relative to overload.
[7]These SNR values are commonly used during the standardization of wireless speech codecs.

conditions (low level, high level, and noisy conditions) as the test VAD (VAD_{test}). For each case, we calculate the 4 probabilities defined below[8]:

- $P_{s|s} = \text{Prob}(\text{VAD}_{\text{test}} = 1 | \text{VAD}_{\text{ref}} = 1)$
- $P_{n|s} = \text{Prob}(\text{VAD}_{\text{test}} = 0 | \text{VAD}_{\text{ref}} = 1)$
- $P_{n|n} = \text{Prob}(\text{VAD}_{\text{test}} = 0 | \text{VAD}_{\text{ref}} = 0)$
- $P_{s|n} = \text{Prob}(\text{VAD}_{\text{test}} = 1 | \text{VAD}_{\text{ref}} = 0)$

Table 2.3 Performance of SMV VAD-A in different test conditions

VAD (%) Probability	low level	high level	car 15 dB	street 15 dB	babble 20 dB
$P_{s\|s}$	93.8	99.9	90.8	86.9	94.9
$P_{n\|s}$	6.2	0.1	9.2	13.1	5.1
$P_{n\|n}$	99.9	89.6	87.9	92.2	77.7
$P_{s\|n}$	0.1	10.4	12.1	7.8	22.3
Speech Activity	67.8	75.0	68.9	64.9	74.7

We show in Table 2.3, the performance results of SMV VAD-A and in Table 2.4 the results of SMV VAD-B. Several observations can be made from these tables:

- The level of the VAD-input signal is directly related to the nature of VAD errors. For instance, low-level input signal causes clipping of some low-energy speech sounds while high-level input creates more false alarm. In both cases, VAD-B has less dependency on the input level.
- The performance of the two VADs depends on the type of the background noise and its level (SNR). Overall, VAD-B shows better performance across all noise conditions. For example, VAD-A has 8% more false alarm rate than VAD-B for babble noise. Also, VAD-B has lower speech clipping rate for car and street noises at 15 dB.

Another way to compare the performance of the two VADs in noisy conditions is to study the correlation of VAD-A decision with VAD-B decision. A high correlation rate is

[8]Note that $P_{s|s} = 1 - P_{n|s}$, and $P_{n|n} = 1 - P_{s|n}$. $P_{n|s}$ gives the speech clipping error rate while $P_{s|n}$ represents the false alarm rate.

Table 2.4 Performance of SMV VAD-B in different test conditions

VAD (%) Probability	low level	high level	car 15 dB	street 15 dB	babble 20 dB
$P_{s\|s}$	97.5	99.9	93.9	89.9	94.9
$P_{n\|s}$	2.5	0.1	6.1	10.1	5.1
$P_{n\|n}$	99.9	94.3	88.0	91.3	85.9
$P_{s\|n}$	0.1	5.7	12.0	8.7	14.1
Speech Activity	71.6	74.9	72.2	68.4	73.4

expected for both VADs especially in the clean speech case. Table 2.5 shows VAD-decision correlation between the two VADs in both the clean nominal condition and the three noise conditions. In this test, VAD-A was considered as the reference VAD and VAD-B as the test VAD. Since both VADs use the same encoder pre-processing steps in SMV and the same input source signals, the VAD decisions are aligned for each frame.

Table 2.5 VAD-decision correlation between the two SMV VADs in different noisy conditions

VAD correlation	nominal level	car 15 dB	street 15 dB	babble 20 dB
$P_{s\|s}$	96.5	93.1	91.8	95.9
$P_{n\|s}$	3.5	6.9	8.2	4.1
$P_{n\|n}$	95.1	94.0	93.3	84.1
$P_{s\|n}$	4.9	6.0	6.7	15.9
Speech Activity (VAD A)	72.2	68.9	64.9	74.7
Speech Activity (VAD B)	73.5	72.2	68.4	73.4

As expected in the clean condition case, a high correlation rate (more than 95%) is observed for both speech and noise frames. The decision mismatch in other frames depends on the sensitivity of each VAD and its hangover mechanism. A correlation rate of higher than 90% is also observed for car and street noises. However, this is not the case for babble noise. Around 16% decision mismatch happens for noise-only sections of the input test signal. This agrees with the observation we have made that VAD-A has more noise mis-detection for babble noise.

The results presented in Table 2.3 and Table 2.4 were generated using the default SMV set-up that assumes the noise suppression (NS) is enabled and it precedes the VAD unit. In some applications, it is desirable to turn off the NS and thus it is important to study a VAD performance in such condition. In Table 2.6 and Table 2.7 we present the effect of turning on/off the noise suppression (of SMV encoder) on the performance of the two VADs. When turning the NS on, the output signal of the NS and thus the input signal to the VAD will have a higher SNR. In general, a VAD will have a better performance (lower error rates) with higher SNR.

Table 2.6 Effect of noise suppression (on/off) on the performance of SMV VAD-A in different noisy conditions

VAD	car 15dB		street 15dB		babble 20dB	
error (%)	NS-ON	NS-OFF	NS-ON	NS-OFF	NS-ON	NS-OFF
$P_{n\|s}$	9.2	12.4	13.1	16.9	5.1	4.2
$P_{s\|n}$	12.1	9.7	7.8	5.9	22.3	14.1

Several points can be made from Table 2.6 and Table 2.7:

- VAD-A has fewer false alarm errors when the NS is turned off. This is notable for babble noise in which noise errors were reduced by around 8%. However, VAD-B has the opposite behavior (i.e., has more false alarm rate when the NS is turned off).
- Both VADs have a slight degradation in detecting speech frames. The speech clipping rate is less for babble noise for both VADs when the NS is off.
- In general, the results show that VAD-A is better when noise suppression is off (i.e., with higher noise levels) and thus has more robustness to SNR variations.

2.4 Summary

We started this chapter by reviewing models characterizing the on-off patterns of conversational speech. A review of statistical modelling of talkspurts and silence durations was presented. A good portion of the chapter was devoted to review and study various schemes

Table 2.7 Effect of noise suppression (on/off) on the performance of SMV VAD-B in different noisy conditions

VAD	car 15dB		street 15dB		babble 20dB	
error %	NS-ON	NS-OFF	NS-ON	NS-OFF	NS-ON	NS-OFF
$P_{n\|s}$	6.1	7.9	10.1	11.9	5.1	4.5
$P_{s\|n}$	12.0	14.6	8.7	12.6	14.1	20.2

for voice activity detection. Finally, we discussed the results of our comparative performance study of two recently-standardized VAD algorithms. The VADs studied are needed for the noise coding schemes considered in Chapters 4 and 5.

Chapter 3

Linear Prediction-based Noise Coding

A major step in the design of signal compression systems is the selection of a modelling technique to represent the input signal. In the last few decades, linear prediction (LP) has become an integral part of speech processing systems (i.e., speech coding and speech recognition). The well-developed theory of linear prediction and the fast computational algorithms available to compute its parameters make it an attractive solution. In this dissertation, coding and classification of background noise are done using the linear prediction paradigm. In this chapter, we start by discussing some of the basic concepts and equations of LP analysis and synthesis. Later sections focus on the important issue of modelling the LP residual. We present a spectral excitation model for low-bit-rate coding of the noise LP residual. Finally, we discuss the importance of the excitation phase information for natural synthesis of some background noises.

3.1 Innovations Representation of a Random Process

Given a wide-sense stationary (WSS)[1] random process $x(n)$ represented as the output of a linear minimum-phase system[2] $H(z)$ excited by a white noise process $w(n)$. Conversely, the process $x(n)$ can be transformed to a whitened process $w(n)$ by passing $x(n)$ through

[1]WSS random process: its mean is constant and its correlation function depends only on the time difference.

[2]All its poles and zeros are inside the unit circle.

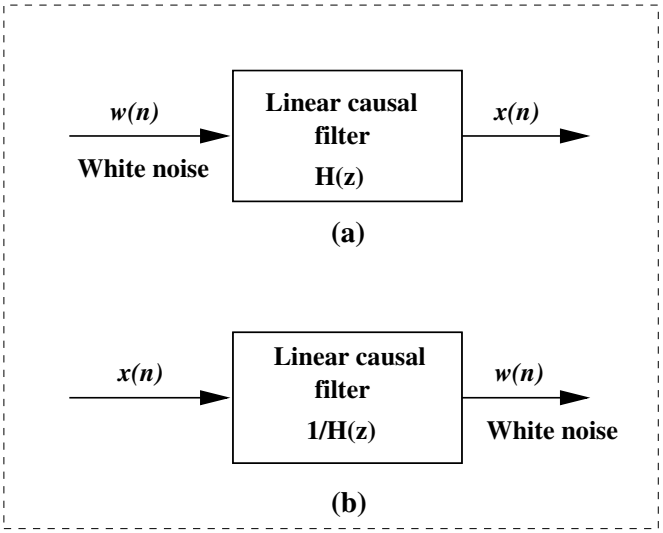

Fig. 3.1 Innovations representation of a random process. (a) Signal model. (b) Inverse filter.

the inverse filter $1/H(z)$. This inverse filter is known as the *noise whitening filter*, and its output $w(n)$ is called the *innovations process* associated with the random process $x(n)$. This excitation-filter representation, shown in Figure 3.1, is known as the innovations representation of a random process or the *Wold representation* [107].

There is a direct relation between the predictability of a given process and its innovations representation. The output of the whitening filter represents the difference between the given random process and its optimal prediction using past values of the process. The output of the inverse filter $w(n)$ carries only the *new* information present in the process; this is the basis of the term "innovations" [108]. In the next section, it will be shown that the excitation-filter model in a linear prediction system is conceptually similar to the Wold representation.

3.2 Basics of Linear Prediction

A good starting point to derive the basic equations in LP analysis-synthesis systems is the time-domain interpretation of linear prediction. For a signal $s(n)$, its sample at a given

time instant can be predicted using a linear combination of the preceding P past samples of the same signal. This relation is given as

$$s_p(n) = \sum_{k=1}^{P} a_k s(n-k). \tag{3.1}$$

An important parameter in this equation is P, the predictor order. The weighting parameters a_k's are known as the LP coefficients. In speech coding application, typically a 10^{th} order LP is used for a signal with an 8 kHz sampling frequency. In general, an LP order of (a signal sampling frequency in kHz +2) gives a good spectral estimation. In our study, we used a 10^{th} order LP analysis for both speech and noise signals.

It is desirable to select the weighting coefficients such that the predicted sample is a good approximation of the original sample. This optimization is done by minimizing the energy of the prediction error. The error signal $r(n)$ is defined as:

$$r(n) = s(n) - s_p(n) = s(n) - \sum_{k=1}^{P} a_k s(n-k). \tag{3.2}$$

The energy of the error signal, E_p, can be expressed as:

$$E_p = \sum_{n} r^2(n) \tag{3.3}$$

$$= \sum_{n} \left(s(n) - \sum_{k=1}^{P} a_k s(n-k) \right)^2. \tag{3.4}$$

Taking the partial derivative of E_p with respect to each LP parameter a_k and set it to zero results in a set of P linear equations with P unknown variables $a_1, \ldots, a_P$. The set of linear equations that results from this optimization depends on the method used. The autocorrelation and covariance methods are two of the most common and efficient LP spectral estimation techniques. In the autocorrelation method, the summation limits in Eq. (3.4) are $\pm\infty$. Also, the signal $s(n)$ in the sum is replaced by a windowed version $s_w(n)$ given as:

$$s_w(n) = s(n)w(n), \tag{3.5}$$

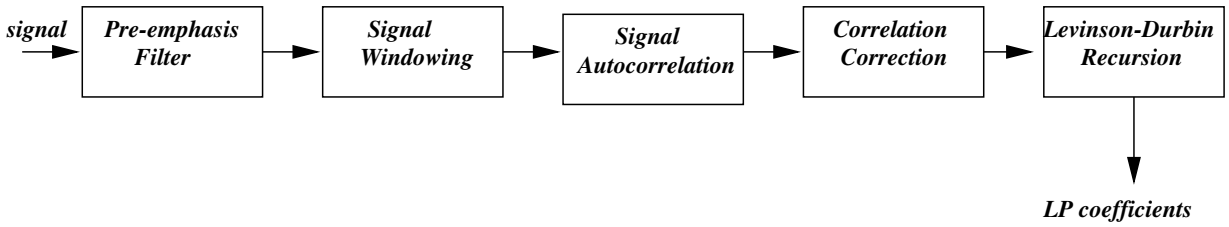

Fig. 3.2 A block diagram of the typical steps taken to compute LP coefficients.

where $w(n)$ is a window such as the Hamming window. The equations that result from the autocorrelation method are known as the *Yule-Walker equations* and they are described in matrix notation by $\mathbf{R_s a} = \mathbf{r_s}$. The matrix $\mathbf{R_s}$ and the vector $\mathbf{r_s}$ are populated with the first $P+1$ correlation values of the windowed input signal, $s_w(n)$. The vector of LP parameters $\mathbf{a}$ is computed by solving the P equations using efficient computational methods such as the *Levinson-Durbin algorithm* [109]. This numerical algorithm exploits the Toeplitz[3] symmetric structure of the autocorrelation matrix $\mathbf{R_s}$ to speed the computation of the LP coefficients. Another major advantage of the autocorrelation method is that it guarantees the stability of the LP synthesis filter (all the poles of the filter are inside the unit circle).

The covariance formulation for the computation of the LP coefficients also uses Eq. (3.4) but limits the summation to a finite window duration of N samples. Also, the original signal $s(n)$ is used in the equation without any windowing. A similar matrix equation results from the optimization of the prediction error but the matrix $\mathbf{R_s}$ and the vector $\mathbf{r_s}$ are populated with the first $(P+1)$ covariance values of the input signal, $s(n)$. The *Cholesky decomposition* method is the preferred algorithm for solving the covariance equations. A major problem with the covariance method is that it does not guarantee the stability of the LP synthesis filter. However, it gives better prediction gain than the correlation method (i.e., better short-term prediction).

In this work we have used the autocorrelation method for computing the LP coefficients. We show in Figure 3.2 a block diagram of the typical steps taken to compute the LP parameters using the autocorrelation method. Before windowing the samples of the input signal, a pre-emphasis (highpass) filtering is performed. A first-order pre-emphasis filter $(1 - \alpha z^{-1})$ helps to compress the dynamic range of voiced spectrum to prevent numerical

[3] A Topletiz matrix has equal elements along any diagonal.

ill-conditions. A typical value for the pre-emphasis coefficient (α) is 0.94 but can also be made adaptive. At the decoder, a corresponding de-emphasis operation ($1/(1-\alpha z^{-1})$) has to be implemented. After windowing and calculating the autocorrelation values of the windowed samples, a correction of the correlation values is done to widen the spectral peaks to guarantee stable LP filters. This operation is known as *bandwidth expansion*. Finally, the *Levinson-Durbin Recursion algorithm* is executed to generate the LP coefficients. For a proper estimation of the LP parameters, the input signal is assumed to be stationary over a short-time window (10–30 ms).

In general, the analysis part of linear prediction involves two operations: computing the LP parameters and then using them to generate the LP residual. In the sequel we will derive the excitation-filter relation between the LP residual signal and the LP analysis and synthesis filters. Taking the z-transform of both sides of Eq. (3.2) we get

$$R(z) = S(z)(1 - \sum_{k=1}^{P} a_k z^{-k}). \tag{3.6}$$

We can write the LP analysis equation as

$$R(z) = S(z)A(z), \tag{3.7}$$

where $A(z)$ is the LP analysis filter, and it is given by $A(z) = (1 - \sum_{k=1}^{P} a_k z^{-k})$. Similarly, the LP synthesis equation can be written as

$$S(z) = R(z)\frac{1}{A(z)} = R(z)H(z), \tag{3.8}$$

where $H(z)$ is the LP synthesis filter.

In Figure 3.3 we present a block diagram of a typical LP analysis-synthesis system. The LP residual $r(n)$ is computed by linear filtering the input signal $s(n)$. The filter coefficients are adapted for each frame. Feeding $r(n)$ as input to the LP synthesis filter produces the original signal $s(n)$. For signal coding using the LP model, the LP coefficient and the residual signal $r(n)$ need to be quantized and transmitted to the decoder to reproduce an output signal that is either waveform-similar or perceptually-similar to the original signal $s(n)$.

Scalar and vector quantization techniques have been proposed to encode the LP pa-

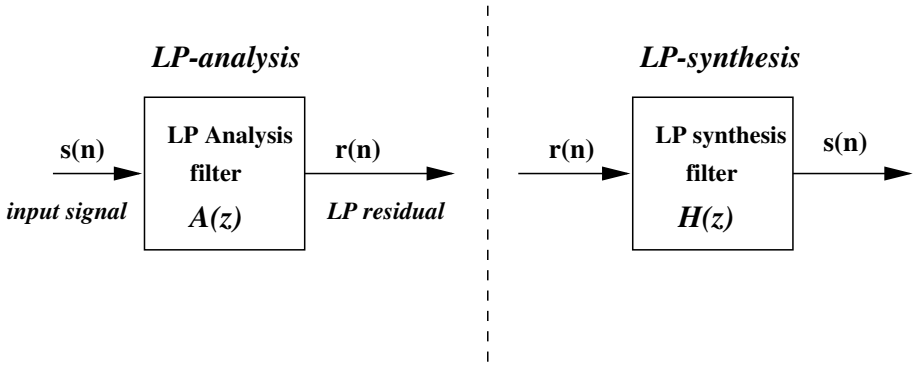

Fig. 3.3 LP analysis-synthesis system.

rameters (known also as LP spectral quantization). Several LP coefficient transformations (such as the line spectral frequencies[4]) have been shown to have better quantization properties than the LP coefficients. Paliwal and Atal [110] reported that around 24 bits per frame is sufficient for a transparent-quality quantization of a 10^{th}-order LP filter.

A major task in linear prediction-based speech coding is the modelling of the LP residual. For low bit rate coding it is sufficient to capture the key perceptual information embedded in the residual signal. In the next section, a brief overview of the excitation modelling problem is given followed by experimental results done to verify the Gaussianity and whiteness assumptions of the LP residual for background noise.

3.3 Modelling of the LP Residual

An assumption in linear prediction analysis-synthesis systems is that (in general) the output of the LP analysis filter is a Gaussian signal with a white spectrum. Extensive research has been done over the last two decades to efficiently represent the speech residual signal. In this section we will give a short overview of the major excitation models that have been proposed for representing the speech LP residual.

A speech signal is composed of a sequence of energy 'events' that vary in structure and spectral content. A two-way classification of speech divides the signal into voiced (or quasi-periodic) and unvoiced (random) segments. An excitation model has been used in LP vocoders in which a period-dependent pulse-train is used for voiced frames and a random

[4]The line spectral frequencies (LSFs) are defined in Chapter 5.

excitation for unvoiced ones. This simple model requires the output of a voicing decision algorithm. The model can not produce high-quality speech as it fails to adequately represent other classes of speech such as onsets and non-stationary voiced sounds. Moreover, errors in the voicing decision degrade the output speech. Mixed-excitation models of speech LP residual have a better representation of the various phonetic classes of speech [111]. Mixtures of periodic and random excitation components are used with mixing weights that vary over time and frequency, and depend on the speech type. A multiband excitation (MBE) model [112] does the mixing in the spectral domain by analyzing a set of frequency bands and deciding of the type of the excitation spectrum (harmonic spectrum or random spectrum). Multipulse (MPE) [113] and code-excited linear prediction (CELP) [114] excitations models perform the mixing and selection of the excitation components in the time domain. An *analysis-by-synthesis* procedure is used in the encoder side to optimize the contribution of each type of excitation using waveform-matching criterion.

Some of these models are speech-specific and require a large number of coding bits, thus they are not feasible for low-bit-rate coding of background noise. In existing noise coding and comfort noise systems, only the spectral parameters and the residual energy are quantized. The residual waveform is not encoded, instead a random Gaussian noise excitation, matching the noise residual energy, is used at the receiver to excite the LP synthesis filter. We have studied the LP residual of different background noise signals. We have observed that substituting the LP residual with a WGN excitation retains naturalness for some noises (i.e., car, computer fan), but it fails to reproduce with natural-quality other "structured" background noises such as babble, street and office noises. We have analyzed the noise LP residuals to understand why using a WGN excitation is not sufficient. The results of this study are presented next.

3.3.1 The Gaussianity Assumption of the LP Residual

We have examined the statistical distribution of the prediction error signal (the LP residual) to check the validity of the Gaussianity assumption. Several techniques exist for testing the Gaussianity of a given signal [115]. One direct way to examine this assumption is to verify if the probability distribution function (PDF) of the signal samples is bell-shaped.

One of the tools that we have used is the quantile-quantile plot (the Q-Q plot). It is a graphical technique for checking Gaussianity (Normality) of a given random signal [116].

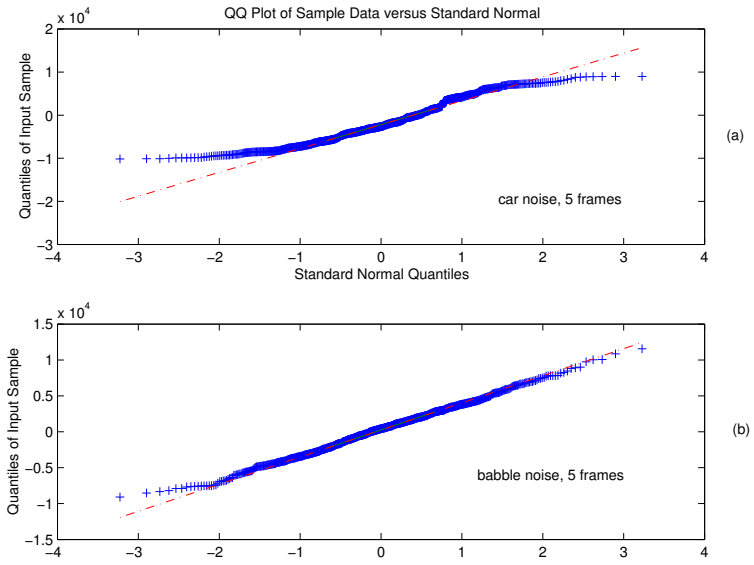

Fig. 3.4 Q-Q plot of $N = 5$ frames of a background noise and a standard Gaussian signal: (a) car noise, (b) babble noise.

The Q-Q plot will be a straight line through the origin with slope 1 if the data have a Gaussian distribution. Deviation from a straight line is a sign of non-Gaussianity. The shape of the plot can also be used to reveal the type of the non-normality. A straggler at the beginning or the end of the plot, is an indication of an outlier. Curvature at the ends of the plot is an indication of a PDF with heavier tails than a Normal PDF, while convexity or concavity in the graph is suggestive of a lack of symmetry.

Using the Q-Q plot[5], we examined the "Gaussianity" of car and babble noises. We show in Figure 3.4 the plot for the two noises using 5 frames for each noise (800 samples) and in Figure 3.5 the plot using 1-second segments (8000 samples) of each noise. The x-axis shows the quantiles of a standard Normal distribution while the y-axis depicts the quantiles of the input test data.

From the results, car noise samples seem to deviate from being Gaussian while babble noise PDF is close to being Gaussian distributed. We show in Figure 3.6 and Figure 3.7 the plots for the LP residual of car and babble noises. Both LP residuals have a good fit

[5]We used the Matlab function *qqplot* for this test. Quantiles are computed by first ranking the samples and then computing their percentiles.

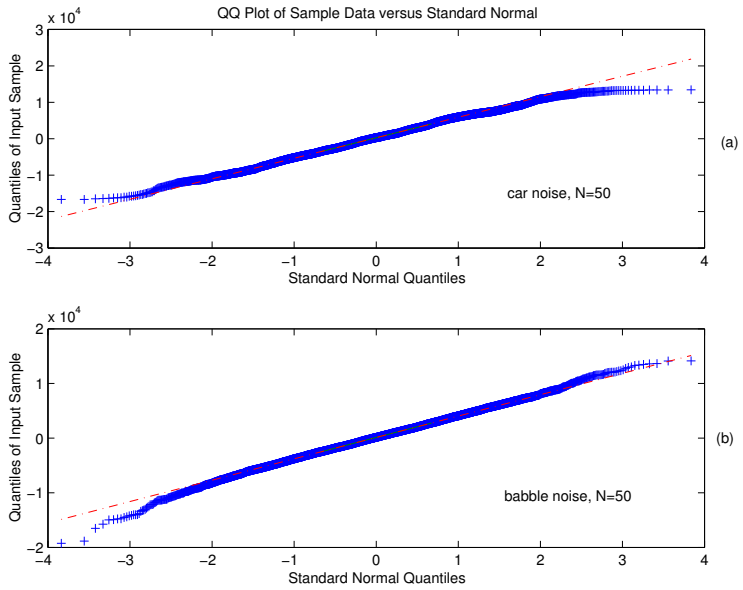

Fig. 3.5 Q-Q plot of $N = 50$ frames of a background noise and a standard Gaussian signal: (a) car noise, (b) babble noise.

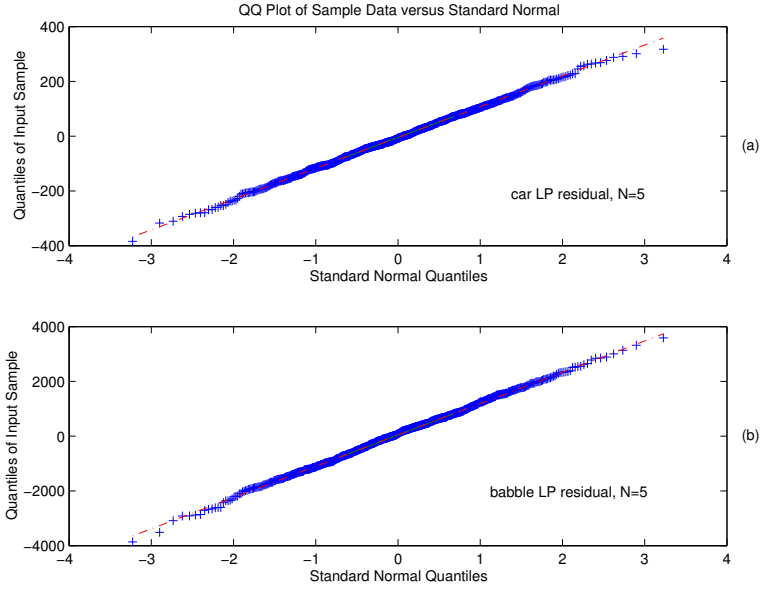

Fig. 3.6 Q-Q plot of $N = 5$ frames of the LP residual of a background noise and a standard Gaussian signal: (a) car noise, (b) babble noise.

to the Gaussian distribution except for some outliers. Using the 1 second segments, the Gaussianity assumption is more accurate, especially for the car noise LP residual.

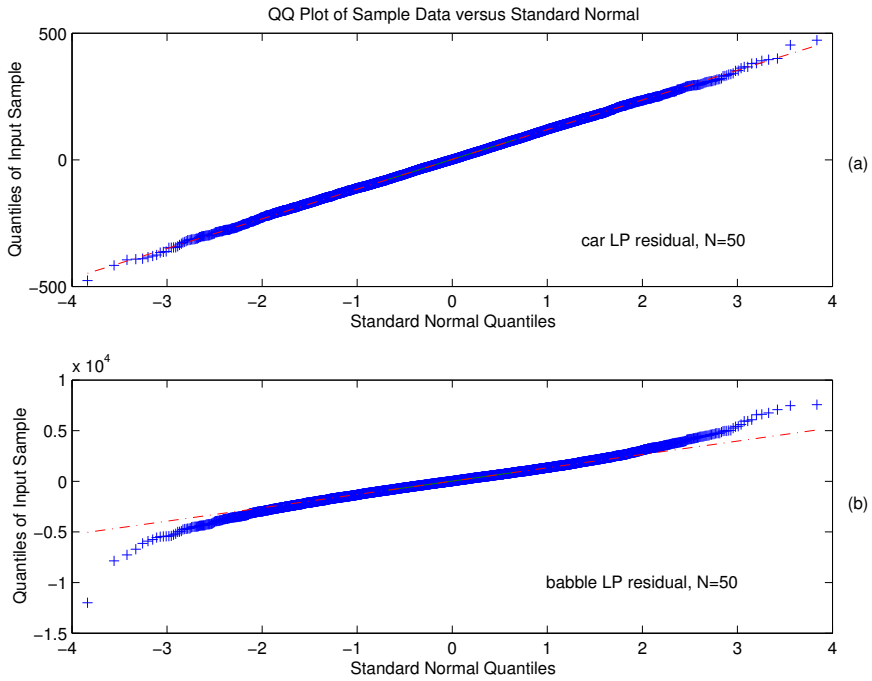

Fig. 3.7 Q-Q plot of $N = 50$ frames of the LP residual of a background noise and a standard Gaussian signal: (a) car noise, (b) babble noise.

Higher-order statistics (HOS) provide a good insight of the nature of random signals. The *kurtosis* of a signal is based on the fourth moment about the mean. For a normal distribution this value is 3 [117]. A kurtosis index measures kurtosis relative to a normal distribution. Positive kurtosis index values (kurtosis values larger than 3) indicate longer thicker tails than a normal distribution while negative values (kurtosis values less than 3) indicate shorter thinner tails. A positive kurtosis must be accompanied by a property of *peakedness* in the distribution which indicates that there is an excess of frequency of occurrence in the center of distribution.

We have measured the kurtosis of car and babble noises and their LP residuals to assess their Gaussianity. The results are shown in Table 3.1 for the noise LP residuals and in Table 3.2 for the 3 noises. Using a single frame does not give a kurtosis value close to 3 even for the Gaussian signal we used. Increasing the length to 5 frames we get values close to 3 for the noises. However, for longer segments (N larger than 5 frames) we notice that

Table 3.1 A comparison between the kurtosis of the LP residuals of car noise, babble noise and WGN as a function of the number of frames

Number of frames	Car LP residual	Babble LP residual	WGN
1	2.49	2.59	2.36
5	2.97	2.98	2.99
10	3.03	3.60	3.11
25	2.90	4.84	3.05
50	3.03	4.77	3.04
100	3.06	4.93	3.01
200	3.04	4.35	3.03

the LP residual of babble has kurtosis values greater than 4 which indicates that its PDF has higher peaks than a Gaussian PDF. This is not the case for the LP residual of car noise as it has kurtosis values close to that of a Gaussian signal. We can conclude that using a Gaussian excitation is more appropriate for car noise than for babble noise. Table 3.2 shows that kurtosis values for car noise are less than 3 even for longer segments.

Table 3.2 A comparison between the kurtosis of car noise, babble noise and WGN as a function of the number of frames

Number of frames	Car	Babble	WGN
1	2.12	2.36	2.36
5	2.13	2.81	2.99
10	2.41	2.89	3.11
25	2.55	3.17	3.05
50	2.67	3.40	3.04
100	2.55	3.67	3.01
200	2.71	3.53	3.03

Kubin *et al.* [118] examined the effect of replacing the LP residual of speech with a WGN excitation. They have shown that replacing the LP residual waveform of the unvoiced speech with a WGN, it is possible to reproduce unvoiced speech with high perceptual quality. However, for voiced speech, a WGN does not model the fine details embedded in

the voiced residual. A further filtering of the LP residual to remove long-term correlation between samples[6] can produce voiced residual with Gaussian distribution [119].

Recently, Kubin [120] [121] used an information-theoretic measure to evaluate the degree of Gaussianity of a given signal. He observed that the output of the LP analysis filter always has high degree of Gaussianity. His study has confirmed that least-squares linear prediction increases the Gaussianity of its output (the LP residual) relative to its input signal. Another result from this study is that the data-dependent linear prediction filtering does not represent a linear system operation. As an example, voiced speech is known to deviate from the Gaussianity assumption but it still can be synthesized using a Gaussian noise exciting time-varying linear filters[7].

3.3.2 The Whiteness Assumption of the LP Residual

One of the properties of the LP analysis filter is that its output signal (the LP residual) has less correlation between its samples than the input signal. This is known as the *whitening property* of the LP analysis filter. The LP synthesis filter represents the spectral envelope of the input signal. It was shown in Eq. (3.8) that the LP analysis filter has the inverse transfer function of the synthesis filter and thus it has the inverse spectral envelope of the signal. Ideally, the spectrum of the LP residual should be flat (i.e, white). There is a direct relationship between the filter order and the degree of flatness of the LP residual. For instance using an LP filter order of less than 5 will produce output residual that is less white than using a larger order.

It was shown in [122] that the criterion for minimizing the energy of the output of the inverse filter is equivalent to choosing the inverse filter that maximizes the spectral flatness of its output. We have investigated the degree of flatness of the magnitude spectrum of the LP residual using two methods. Our objective is to assess if we need to extract further spectral information from the LP residual for noise synthesis at the speech decoder.

Markel and Gray [67] proposed a spectral flatness measure (SFM) for quantifying the flatness of a signal spectrum. It is the ratio of the geometric mean to the arithmetic mean of a signal power spectrum. For very peaky signals, the SFM takes values close to zero, and for a constant spectrum the SFM is 1. The inverse of the SFM is a measure of

[6] This is known as pitch prediction filtering in the speech coding literature.
[7] Both pitch-prediction and LP synthesis filters.

signal predictability and typically has values between 3 and 16 for long-term power spectral densities of speech signals [123].

Given a zero-mean signal $x(n)$ with discrete power spectral density $S_x(k)$, the SFM (γ^2) can be expresses as:

$$\gamma^2 = \frac{[\prod_{k=1}^{M} S_x(k)]^{\frac{1}{M}}}{\frac{1}{M}\sum_{k=1}^{M} S_x(k)}, \tag{3.9}$$

where M is the number of spectral samples of $S_x(k)$.

Using this measure we evaluated the flatness (and predictability) of car and babble noise LP residuals. We used a WGN signal in this test as a reference. The results are presented in Table 3.3 for both 20-ms frames ($N = 1$) and for 1-second frames ($N = 50$). For each noise we show the SFM values for the input and output of the LP analysis filter. For a signal with a flat spectrum, the SFM should be close to 1. For signals with "colored" spectra, their SFM shall have lower values. The results indicate that the SFM can not give values close to 1 using short frames even for a white noise. However, for longer frames SFM values close to 1 are obtained for flat spectra. For long segments ($N = 50$) both LP residuals of car and babble noise have SFM close to 1 which suggest that they have flat spectra.

The signal predictability measure shown in the table indicates that car noise is a highly predictable signal. In Chapter 5 we will show that the car noise we used in our study has a lowpass spectral content with very correlated samples. Babble noise has a much lower predictability than car noise but still its samples have more correlation than a white noise.

Kedem [124] proposed a whiteness test using higher-order crossings (HOC). These are defined as zero-crossing counts (ZCC) of filtered versions of the input signal. Different filters can be used to generate the HOC signals. Kedem derived a closed form expression for the upper and lower limits for a sequence of HOC values for a white Gaussian noise signal. These limits are given by [124]:

$$(N-1)\{\frac{1}{2}+\frac{1}{\pi}\sin^{-1}(\frac{k-1}{k})\} \pm 1.96(N-1)^{\frac{1}{2}}\{\frac{1}{4}-[\frac{1}{\pi}\sin^{-1}(\frac{k-1}{k})]^2\}^{\frac{1}{2}}, k = 1, 2, \ldots, \tag{3.10}$$

where the plus sign is for the upper limit D_{max}, and the minus sign is for the lower limit

Table 3.3 Spectral flatness and predictability measures (Predict.) as a function of the signal length (N) for car, babble and white Gaussian noises

Signal	SFM (N=1)	SFM (N=50)	Predict. (N=1)	Predict. (N=50)
car	0.002	0.002	597.1	614.9
LP residual	0.503	0.909	1.99	1.1
babble	0.073	0.164	13.79	6.1
LP residual	0.472	0.909	2.12	1.1
WGN	0.552	0.980	1.81	1.02

D_{min}. In this equation, N is the length of a signal (in samples), and k is the HOC number. A signal behaves as a Gaussian white noise if its HOC values fall within these two limits. This whiteness test rejects the hypothesis of white noise for a signal if at least one of its HOC values fall outside the limits.

We extended our study of the whiteness of the LP residuals of car and babble noises using the HOC-based whiteness test. In our experiments[8], we used high-pass differentiator filters and measured the ZCC of the filter-output signals. Filtering was limited to six stages as the HOC values become the same with successive differencing. The first HOC value gives the ZCC of the input signal. We show the results of this test in Figure 3.8. The x-axis shows the HOC sequence numbers and the y-axis shows the range of values for the ZCC in a frame of 160 samples (20 ms frames).

The test confirms that both noises have LP residuals with flat spectra. In the same figure we show the HOC values for a test WGN signal. These values are shown as crossed circles. The LP residual of babble noise behaves more like WGN than is the case for car noise. Figure 3.9 shows that signals with non-flat spectra (such as car and babble noises) fall below the HOC limits of a white signal.

To summarize, both methods that were used for testing the whiteness of the LP residuals show that car and babble noises have LP residuals with flat spectra using a 10^{th} order LP filter.

[8]We reported in [58] that we used successfully the HOC as features for speech/music discrimination.

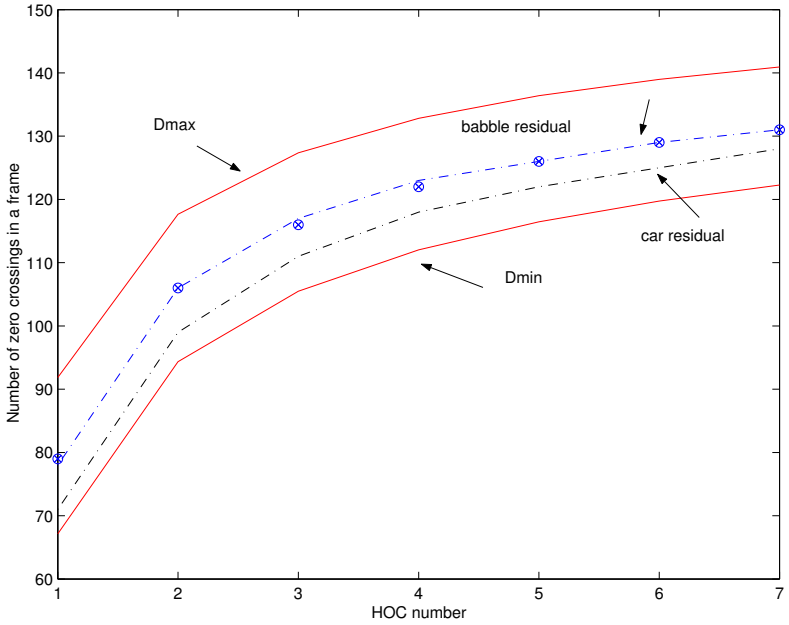

Fig. 3.8 HOC-based whiteness test for the LP residual of car and babble noises.

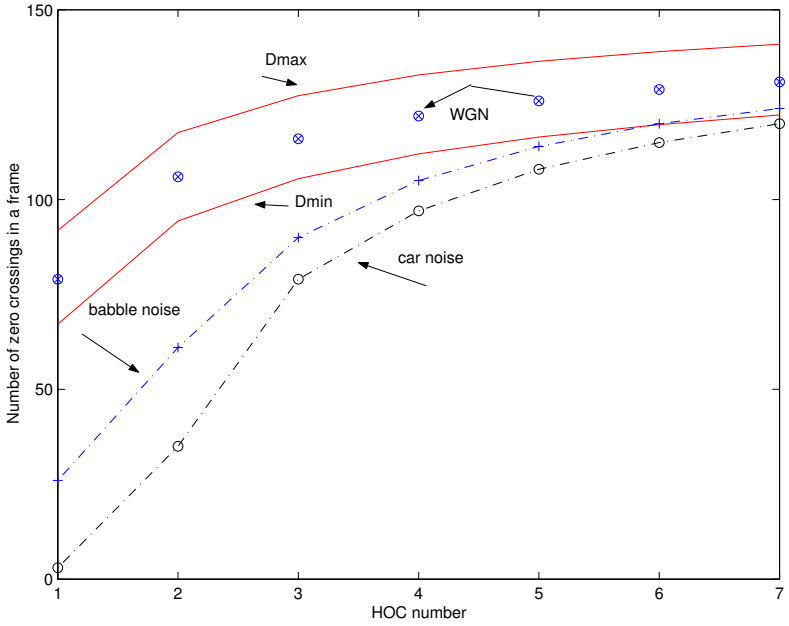

Fig. 3.9 HOC-based whiteness test for car and babble noises.

3.4 Spectral Excitation Models

We have shown that the LP residuals of car and babble noises can be modelled as white Gaussian. However, when we replace the residual of babble noise with a WGN excitation, the synthesized noise does not sound natural. This observation was also true for other structured noises in streets, restaurants, and buses. For car and other Gaussian-like noises such as fan noise, WGN is a good excitation model. This insufficiency of WGN as an excitation signal for some noises was reported by other related noise coding studies [47] [48].

To improve the quality of a synthesized background noise we need to have better excitation models that can capture important perceptual information in noise LP residuals. With a limited bit-budget for coding noises, the models have to be simple and require only few extra bits. Kroon and Recchione proposed encoding a time envelope of the noise residual to modulate the WGN excitation. This extra information was coded using 4 bits [47]. We have tried this idea and only a slight improvement in quality was obtained.

In a recent patent application [41], the spectral information in the LP residual was captured using a second LP analysis filter with a 5^{th} order. The output of the first LP filter (the LP residual) is used as input to the second LP filter. The residual LP parameters are transmitted to the decoder using a small number of quantization bits. Instead of applying this cascade LP approach, a higher order (i.e., 15) can be used for a single LP analysis.

In this section we report a different method to represent the spectral content of noise residuals. The discrete Fourier spectrum of the LP residual, $r(n)$, will be expressed using the magnitude-phase decomposition:

$$R(k) = |R(k)| \exp(j \angle R(k)), \tag{3.11}$$

where $|R(k)|$ is the magnitude spectrum and $\angle R(k)$ is the phase spectrum of $r(n)$.

In the sequel we will denote the magnitude spectrum of $r(n)$ as $M_r(k)$ and the phase spectrum as $\phi_r(k)$. Only a few bits are available to transmit both the magnitude and phase information. It is common in signal compression to assume that the human ear is insensitive to the phase information and thus it need not be transmitted. In that case, only the residual magnitude spectrum information needs coding. It is reported in noise perception studies that the human auditory system is not sensitive to the fine details of the magnitude spectrum [125]. The perceptual quality of a noise is determined mainly by the

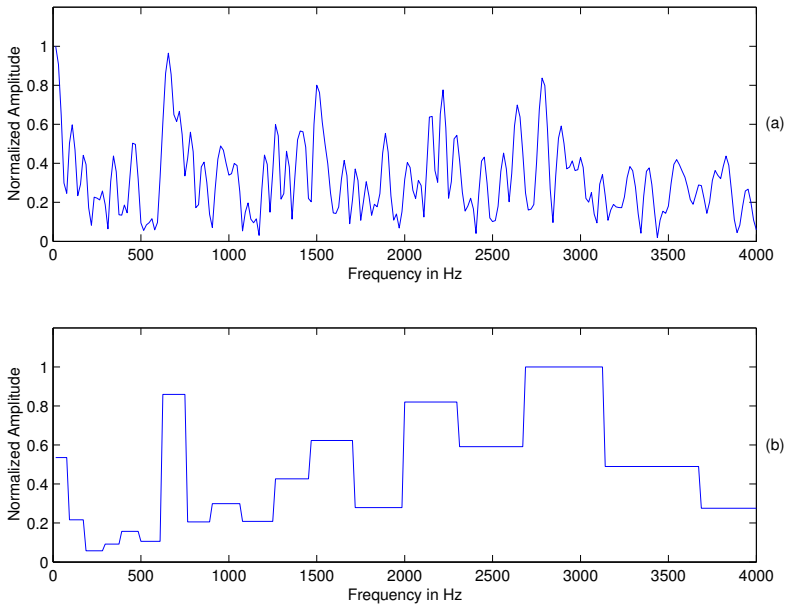

Fig. 3.10 A plot of (a) amplitude spectrum of a frame of the LP residual of car noise, and its (b) piecewise constant gain-spectrum.

total energy in each frequency band. Thus, it is sufficient to represent the noise magnitude spectrum with a set of energies computed in M frequency bands.

The energy in the i^{th} band, E_i, can be computed from the residual magnitude spectrum as:

$$E_i = \frac{1}{L_i} \sum_{k \in B_i} M_r(k), \tag{3.12}$$

where L_i is the number of frequency samples in the i^{th} band, and B_i is the range of frequencies for this band. A gain value for each band is simply the square root of the band-energy.

We have used this compact representation of the magnitude spectrum of the LP residual to enhance the noise excitation modelling. To encode the M gain values for each frame, we can use vector quantization techniques. In our experiments, we used a vector of 18 gains that are computed using the non-uniform critical-band division of the frequency axis. A few energies from a selected critical bands can be transmitted to reduce the required bits. However, as the LP residual has a broadband spectrum we preferred using the full band information.

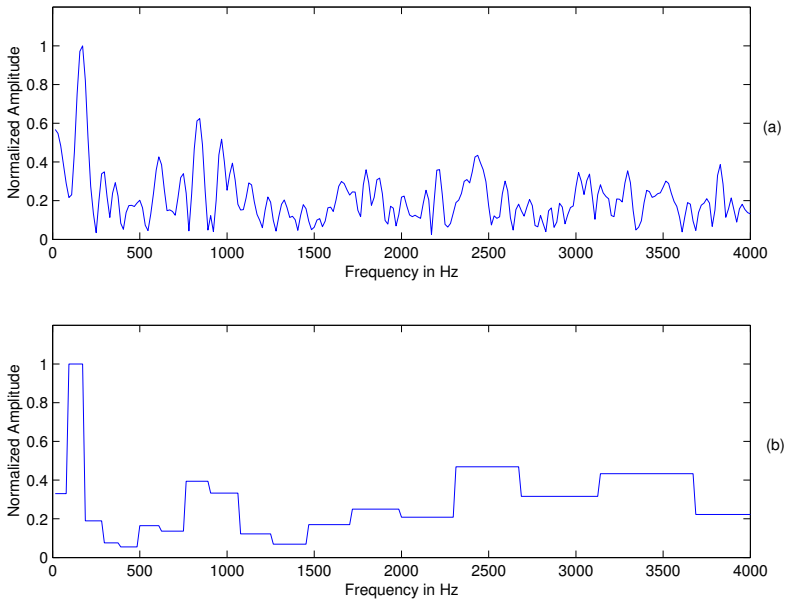

Fig. 3.11 A plot of (a) amplitude spectrum of a frame of the LP residual of babble noise, and its (b) piecewise constant gain-spectrum.

Using the spectral gain vector, there are several ways to construct a magnitude spectrum that replaces the residual spectrum $M_r(k)$. One simple way is to use a piecewise spectrum that uses a constant gain in each band. Another way is to modulate the magnitude spectrum of a WGN noise using the gain of each band. We show in Figure 3.10 the amplitude spectrum of a frame of the residual of car noise, and its piecewise constant gain-spectrum, and the same for babble noise in Figure 3.11.

Before quantizing the gain vector, we evaluated the quality of the synthesized noise using this spectral excitation model of the LP residual. In our experiment, we replaced ϕ_r with a random phase uniformly distributed in the interval $[-\pi, \pi]$. For some structured noises that we tested (babble and bus noises), the new excitation model slightly improved the quality. However, the noises produced were still muffled and far from being natural. The next step we tried was to use the 'true' LP residual phase with this 'crude' representation of the magnitude spectrum. Using this modified excitation produced natural-sounding noises. Further, we replaced the magnitude spectrum of the residual with a constant spectrum and used the residual phase ϕ_r. Surprisingly, the quality of the synthesized noise is natural. From these observations, it was concluded that the residual Fourier phase

contributes significantly to preserving naturalness in the synthesized noises. In the next section we relate our finding of the relationship between the phase spectrum of the LP residual and perception to other studies of the importance of phase.

3.5 Fourier-Phase of the LP Residual

It was shown in the previous section that representing the LP residual with its spectral magnitude-phase decomposition is useful for separately modelling and coding each component. Moreover, with this representation it is possible to assess the relative importance of phase and magnitude to the synthetic quality of a signal. Oppenheim and Lim [126] studied the perceptual importance of the Fourier phase of speech signals. It was shown that combining the original phase of a speech signal with a constant magnitude spectrum retains the intelligibility of speech. They explained that phase-only synthesis of speech preserves correlation between signal components. In our work we are mainly interested in the perceptual importance of the phase spectrum of the LP residual and in the sequel we present a summary of related studies in the literature.

For speech sounds, unvoiced excitation is random and has temporal and spectral statistics similar to a white Gaussian noise. Thus, the phase of its residual can be substituted with a uniform-distribution random phase. Thus, most of the perceptual studies of the Fourier phase and its modelling is focused on voiced and onset speech [127] [128].

Atal and David [129] [130] examined the effect of phase and magnitude of the LP excitation in preserving naturalness of synthesized speech. They modified the spectral amplitude and phase of the LP residual of voiced speech and evaluated the resulting change in speech quality. The original phase was replaced with zero phase, constant phase, and a pitch-dependent phase. Also, the spectral amplitude of the residual was modified and tested with the 3 'artificial' phases. In their study, they concluded that phase distortions produce less degradation in quality when compared with a distorted amplitude spectrum. Ma and O'Shaughnessy also performed a perceptual study of the Fourier amplitude and phase of the LP residual of vowel sounds [131]. They concluded that the relative importance of phase and magnitude spectrum of the LP residual is strongly dependent on the pitch frequency of vowel sounds. They also reported that the conclusion made by Atal and David, that the amplitude spectrum of the LP residual contributes more to the overall speech quality than the residual phase can not be generalized to all types of speech.

Gauthreot *et al.* [132] also investigated the perceptual content of the LP residual and proposed a low-bit rate coding of the residual phase. They confirmed that for speech signals, the residual spectral phase is perceptually important especially in the low frequency band below 600 Hz.

From the above studies and from our experimental results, we can conclude that the spectral phase of the LP residual of speech and other acoustic signals carries important perceptual information for natural-quality sound reproduction using the LP synthesis model. The next issue will be how to send this phase information to the decoder using low bit rate.

Trancoso *et al.* [133], Hedelin [134], and Cheetham *et al.* [135] proposed the use of an adaptive all-pass filter to parameterize the short-time phase of the LP residual. The objective of this filter is to compensate for the phase mismatch between the original LP residual and the reconstructed LP excitation at the decoder. In [136], the excitation search complexity in the CELP coder analysis-by-synthesis loop was reduced by comparing the phase of the codebook vectors with the residual phase spectrum. Recently, a low bit-rate representation of the short-time phase of speech was proposed in [137].

For low bit-rate noise coding, it is not feasible to encode the LP residual phase using the aforementioned schemes. Moreover, with the variety of noise sources, modelling the noise residual phase is not an easy task. In the next chapter we will present novel excitation models to "capture" vital perceptual information of the LP residual of background noise.

3.6 Summary

In this dissertation, coding and classification of background noise were performed using the linear prediction signal modelling framework. We started this chapter by presenting important LP equations and concepts that will be used in this thesis. A special attention was given to the excitation modelling of the noise LP residual. As the conventional white Gaussian excitation model fail to replace the LP residual of structured noises, we reported our results of using improved excitation models. A study of the LP residual spectrum has revealed that Fourier phase of the LP residual carries important perceptual information that is essential for natural-quality synthesis of background noises. We present in the next chapter a novel noise excitation model that will be shown to maintain the character of background noises.

Chapter 4

Class-Dependent Residual Substitution

In this chapter we present a novel excitation model for LP-based coding of background noise at very low bit rates. Noise classification is used to select an excitation signal that is perceptually similar to the LP residual signal at the transmitter. First, we present the basic idea of residual substitution and then we discuss the major units of this new scheme. We conclude the chapter by presenting the results of concept-validation experiments that were performed to study the class-dependent excitation model.

4.1 Residual Substitution: Basic Idea

Chapter 3 examined the modelling of the LP residual of background noise as white Gaussian noise (WGN) excitation. However, from our experiments, we observed that replacing the LP residual waveform of structured noises (i.e., babble, street, office) with WGN excitation is not sufficient to reproduce such noise types with realism. It was also concluded that the phase information of the LP residual carries important perceptual cues that preserve the character of background noise. Encoding the phase information requires a large number of bits and thus it is not feasible for very low-bit-rate coding of acoustic noise. Moreover, with the variety of noise sources, modelling the phase of the noise residual is not an easy task.

In listening to long segments of different types of acoustic noise signals, we have observed that there is a large amount of perceptual redundancy over time. This suggests that

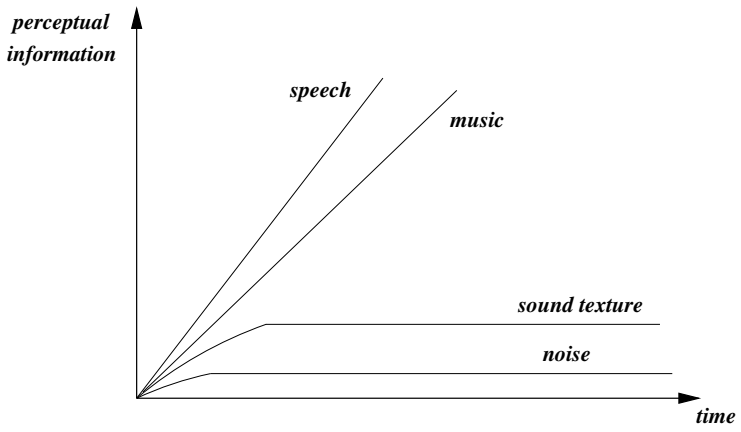

Fig. 4.1 Constant long-term characteristics of sound textures and noise [138].

segments of a given noise type have similar perceptual character. Not every acoustic signal has this property and those sounds with temporal perceptual redundancy belong to a class of sounds known as *sound texture*. Saint-Arnaud *et al.* [138] define a sound texture as a signal that exhibit similar long-term characteristics over time. Babble noise, traffic noise, and machine noises are examples of sound textures. We show in Figure 4.1 that sound textures have constant long-term characteristics when compared with speech and music signals.

We have exploited this observation to propose a new excitation model for the LP residual of background noise. This scheme will be referred as the class-dependent residual substitution excitation model. In our approach, the LP residual of the background noise during speech gaps is replaced at the receiver by an excitation signal that maintains the perceptual character. This is achieved by using a noise classification module that identifies the type of the background noise. At the receiver, the excitation selection module uses the noise classification decision to output an excitation signal from the identified noise class. For each noise class, a prototype segment of the LP residual of this class is stored in a residual codebook at the receiver side.

We have observed that if a stored LP residual waveform of an appropriate type is used, the character of the noise is well preserved. An example of a class of noise is babble noise. For example, if we save the residual from one instance of babble noise and use it for another, the output of the LP synthesis filter is perceptually similar to the original. A different

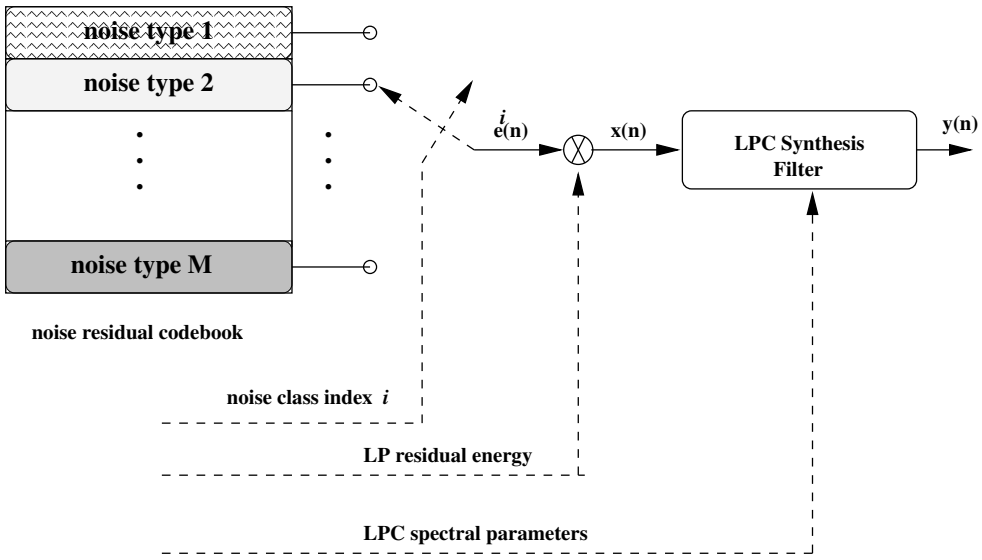

Fig. 4.2 Class-dependent Residual Substitution.

stored residual would be used if street noise is encountered. Our experimental results have confirmed that class-dependent residual substitution can produce natural quality for a number of different noise types common in mobile environments (i.e., car, street, bus, and restaurant). In Figure 4.2 we show a block diagram of an LP synthesis model with the proposed class-dependent residual substitution. To implement the class-dependent residual substitution model, two major operations are done: noise classification and residual substitution. Noise classification is preferably done at the encoder side and the residual substitution at the decoder side. The effectiveness of this excitation model depends mainly on the proper design of the noise residual codebook and the noise classification unit. In the next sections, we will discuss these major design issues and highlight several options to configure the class-dependent excitation model.

4.2 Residual Noise Mixture Model

In an acoustic noise environment, the background noise is often a mixture of acoustic signals from different noise sources. For example, in a public bus environment, the background noise consists of engine, babble, traffic and other ambient noises. The excitation model presented in Figure 4.2 assumes a single-source noise. Thus, in Figure 4.3 we present a general model for the class-dependent residual substitution scheme. The LP residual of the

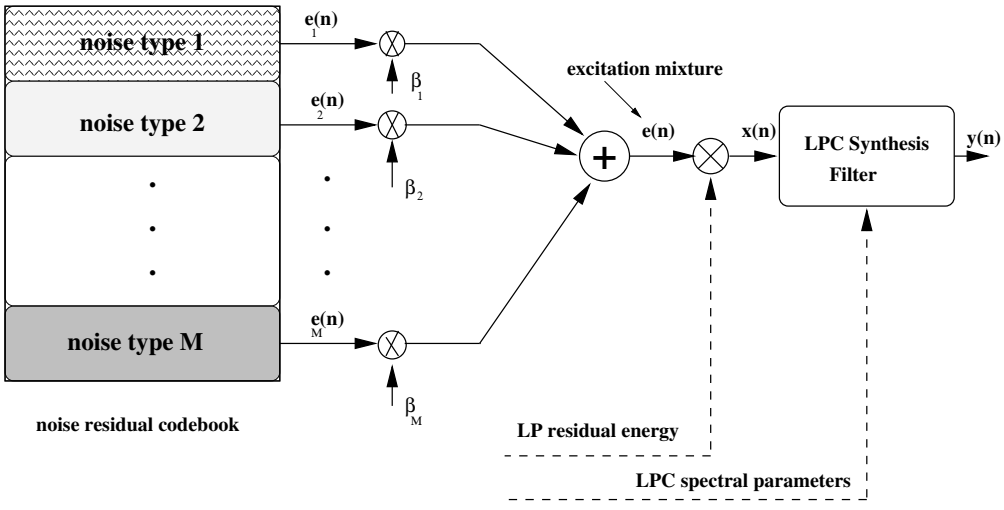

Fig. 4.3 Residual Mixture Substitution.

background noise at the transmit side is replaced at the receiver by an excitation mixture signal $e(n)$. The LP excitation signal $e(n)$ is modelled as a linear mixture of M excitation signals from the M noise classes, given as:

$$e(n) = \sum_{i=1}^{M} \beta_i e_i(n), \tag{4.1}$$

where $e_i(n)$ is an excitation signal from the i^{th} noise class, and $\beta_i(n)$ is the i^{th} mixing coefficient, taking a value between 0 and 1, with $\sum_{i=1}^{M} \beta_i = 1$.

The mixing coefficients quantify the contribution of the excitation of each noise class to the excitation mixture. A key issue in using the excitation mixture model is the estimation of the coefficients of the mixing model. These mixing weights can be either sent to the receiver or determined at the receive side. A soft-decision classification module can be used to output M decision values between 0 and 1. These soft-decision values can be transmitted to the receiver and used directly as mixing coefficients or can be mapped to other weighting values. In Chapter 5, we will revisit this mixture excitation model and we will present novel methods to estimate the mixing weights using soft-decision classification techniques.

The excitation model of Figure 4.2 is a special case of the mixture excitation model. For example, if the mixing vector is zero except for the j^{th} component, i.e., $\beta = [0 \dots 1 \dots 0]$, then we get an excitation signal from the j^{th} noise class, $e_j(n)$.

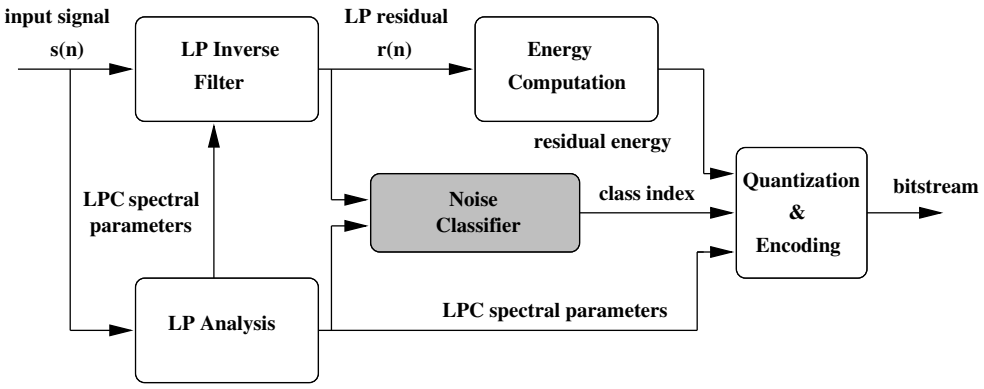

Fig. 4.4 Noise classification at the transmit side.

The excitation mixture model can also be used as a means to reduce the effect of classification errors with hard-decision noise classification. For instance, during a transition from one class to another, a graceful blending of the excitation signals from the two noise classes can be done.

4.3 Classification of Background Noise

In this section, we will give a short overview of noise classification and in Chapter 5 the details of noise classification system design will be given along with results from our evaluation experiments.

The first step in designing an M-class noise classifier is to define the M noise classes of interest. Then, a set of signal features is specified that in combination with a selected classification algorithm give good classification results. Training data from each noise class, in the form of labelled feature vectors, are used to train the classification algorithm. In the test phase, the classification rule maps an input feature vector to the closest class. A set of noise features is used as input to the noise classifier. The classifier outputs a noise class index i that is transmitted to the receiver for class-dependent excitation selection. Classification at the transmitter can use any set of features from the input signal that can discriminate between noise classes. Figure 4.4 shows the encoder part of an LP-based noise coder with a noise classification module.

To represent the noise type index for an M-class classifier, we need $\log_2(M)$ bits for each frame. For example, 2 bits are needed per frame for a 4-class case. It is also possible to

Table 4.1 Classification matrix: Gaussian classifier

	Babble %	Car %	Bus %	Factory %	Street %
Babble	**79.8**	0.0	12.8	2.0	5.4
Car	0.0	**99.6**	0.2	0.2	0.0
Bus	8.8	0.0	**85.2**	2.2	3.8
Factory	1.0	0.0	5.6	**93.2**	0.2
Street	1.8	0.0	24.8	2.0	**71.4**

perform noise classification at the receiver side using quantized signal parameters. This will save the extra bits and allow this scheme to be applied to existing noise coding schemes as it requires only changes to the noise decoder. However, doing classification at the encoder allows the use of a larger set of (unquantized) signal features, and thus can eliminate the effect of quantization on the accuracy of classification.

We have experimented with classifying the background noise into a number of canonical types. A decision is made once every 20 ms to select the noise type. Classification accuracies of about 89% were obtained, with the accuracy depending on the noise class[1]. Good classification results were obtained using a quadratic Gaussian classifier with the line spectral frequencies as features. A sample of the results is shown in Table 4.1 in the form of classification matrix [56].

4.4 Noise Residual Codebook

The noise residual codebook is populated with *prototype* LP residual waveforms from the M noise classes. The residual codebook has a size of $M \times L$, where M is the number of noise types, and L is the length of stored LP residual for each noise type. The residual waveforms are stored with unit energy. The noise residual codebook is only needed at the receiver.

The stored residual should be long enough to prevent any perceived repetition. For example, let us assume we have defined 4 noise types and we store 25 frames (each frame is 160 samples) for each noise type ($L = 25 \times 160 = 4000$ samples). Thus, a total of 16000 samples are stored. However, this new excitation model requires only few extra bits (2 bits

[1]Such an accuracy is sufficient for our application.

for $M = 4$) to transmit classification information to enhance the quality of existing noise synthesis models.

The use of noise residual codebook is similar in concept to using a fixed stochastic codebook in code excited linear prediction (CELP) speech coders [139]. In addition to the large memory requirements of stochastic codebooks (40,960 samples for a codebook of size 1024 and dimension of 40 samples per subframe), this codebook requires a large number of bits (40 bits for a frame of 160 samples) to convey the selected codebook entries to the receiver. The codebook entries in CELP are selected using computationally-intensive *analysis-by-synthesis* search procedure to capture phase-information of speech LP residual [1]. In our technique, we use open-loop noise classification to identify the codebook entry that corresponds to the class of the input noise. The complexity of noise classification is much smaller than that used in CELP.

To preserve the perceptual texture of the reconstructed noise, the excitation signal is constructed from sequential residual samples. An excitation counter is used to keep track of the location within the excitation codevector. Once the noise class index has been received at the decoding unit, the frame counter of this noise class is used to copy a segment (of a frame length duration) from this class excitation vector in the residual codebook. Logical tests are done to check if the end of the vector has been reached and there is a need to go back to the start of the excitation vector.

The noise residual codebook content can be either designed offline and kept fixed during operation or it can be updated dynamically. One way to update the content of the noise residual codebook at the receiver, is to use the excitation signal of the hangover frames. A hangover period of few frames (3–10) is commonly used in VAD algorithms to prevent any premature transition from speech to silence [105] [140]. In most cases, the hangover frames contain background noise. The hangover frames are commonly encoded with the full-rate of the speech coder, with a good reproduction of the LP residual at the transmit side. After classifying a hangover frame to one of the M noise classes, its excitation signal can be used to update the excitation codevector of the corresponding noise class.

4.5 Concept-Validation Experiments

We have performed several concept-validation experiments to assess the improvement in quality using the proposed class-dependent residual substitution scheme.

Our experimental setup consists of a conventional linear prediction analysis-synthesis system. A 10^{th} order LP analysis is performed every 20 ms using the autocorrelation method. A Hamming window of length 240 samples is used. The LP coefficients are calculated using the Levinson-Durbin algorithm and then bandwidth expanded using a radial scaling factor of 0.994 applied to the pole locations. The input noise signal is filtered through the LP inverse filter, controlled by the LP spectral parameters, to produce the LP residual signal. The residual waveform is replaced by an LP residual from a similar noise class, with the same energy content. The *new* LP residual excites the unquantized LP synthesis filter to produce a reconstructed noise signal. Listening tests confirm that substituting the residual of one noise class with *an appropriate* residual, preserves the perceptual texture of the input background noise.

To illustrate the benefits of our scheme, we have modified the noise coding mode of the CDMA enhanced variable rate codec (EVRC) to include the proposed class-dependent noise excitation model [46]. We have replaced the pseudo-random noise generator with a codebook containing stored LP residual from M noise types. For our implementation, we have selected M to be 4 noise classes (babble, car, street, and others[2]). Evaluation tests have shown that we have improved the overall quality with the proposed noise coding scheme without an increase in bit rate, other than for the classification bits. More details about our evaluation results for the EVRC coder are in Section 5.7.6.

In the GSM discontinuous transmission system, in a cycle of 24 noise frames, the first frame is transmitted using the full-rate coder, with zero bits for the remaining frames [28]. At the receiver side, interpolation is used to substitute the parameters of the untransmitted frames. A randomly-generated excitation is used to replace the residual for all the frames in the cycle [44]. The comfort noise generated using this approach sounds different from the background noise at the transmit side. The difference in quality is caused by discarding the residual waveform, and the infrequent transmission of spectral parameters.

We have simulated the GSM discontinuous transmission mode using a "controlled" frame loss model. In a cycle of K noise frames, we keep the spectral and the energy parameters of the first frame and discard the next $K-1$ frames. The LP residual of all the frames in the cycle are substituted with an LP residual from a similar noise class. The spectral and energy parameters are interpolated using,

[2]The noise class 'others' includes other types of background noise.

$$p(n+i) = (1 - \frac{i}{K})p(n-K) + \frac{i}{K}p(n), \tag{4.2}$$

where $p(n+i)$ is the parameter of frame $n+i$ (for $i = 0, 1, \ldots, K-1$), $p(n)$ is the parameter of the first frame in the current cycle, and $p(n-K)$ is the parameter for the first frame in the second latest cycle.

We have experimented with different choices of the cycle length K. Listening quality tests confirm that using class-dependent residual substitution and interpolated spectral envelope reproduce background noise with natural quality, even for a large frame loss rate (i.e., $K = 50$ frames). Class-dependent comfort noise insertion schemes, using the proposed noise excitation model, can enhance the quality of voice communication using GSM-based wireless systems.

4.6 Summary

We have introduced in this chapter our new class-dependent residual substitution model that can faithfully synthesize background noise with very low bit rate requirements. Both single-source noise and noise mixture excitation models have been presented, and various design issues have been addressed. The results of concept-validation experiments of the new scheme have been presented. A major unit of the residual substitution technique is noise classification and thus the next chapter is devoted to it.

Chapter 5

Classification of Background Noise

Noise classification is one of the essential parts of our proposed method for coding background acoustic noise at very low bit rates. In this major (and longest) chapter of the dissertation, we present a detailed study of the noise classification problem. First, we review a study describing the internal processing steps inside our auditory recognition system. We then present a literature review of other applications of noise classification. Next, the classification problem is realized as a pattern recognition system. Various design issues such as features definition and extraction, classification algorithms, and performance evaluation methods are explored. A good portion of the chapter will be dedicated to the discussion of our classification results (for both noise and speech) using various features and classification techniques. In the last portion of the chapter, we propose novel methods for an efficient implementation of the mixture residual substitution model presented in Chapter 4.

5.1 Auditory Sound Recognition and Classification

Humans have a remarkable ability to recognize different types of sounds. Little has been confirmed about the internal processing steps inside our auditory recognition system. However, research in experimental psychology provides some hypothesis that seem to agree with other findings in human auditory perception. McAdams [141] examined aspects of human auditory perception in the recognition and classification of sound sources and events. In this section, we will summarize the main steps of auditory sound recognition as they will be related in Section 5.3 to the general steps in designing an automatic sound recognition system.

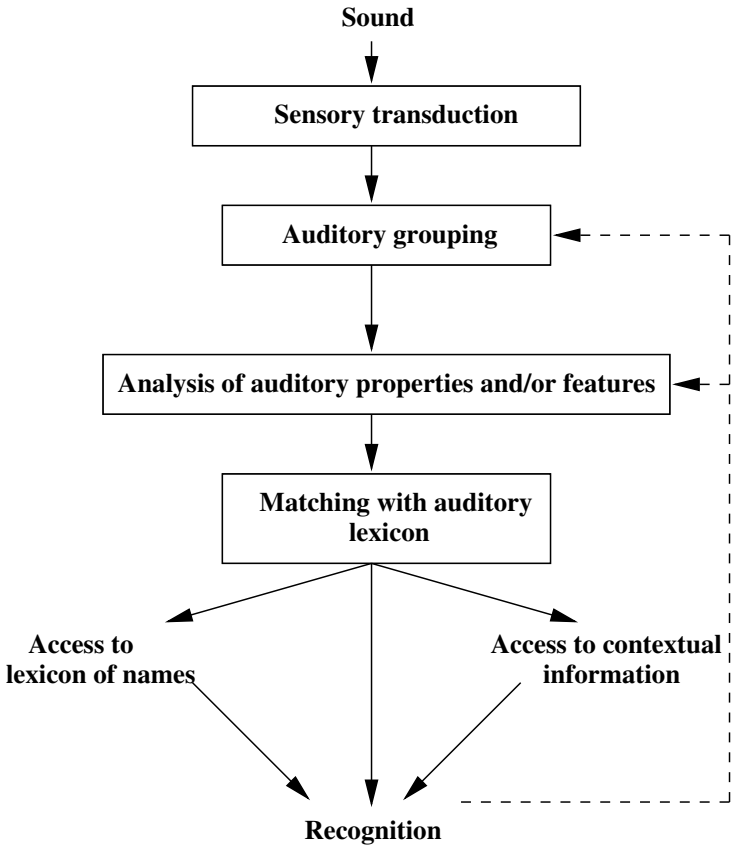

Fig. 5.1 Schematic diagram of the hypothesized stages of auditory processing involved in recognition and classification of sound [141].

The recognition (classification) process may be conceived as involving several hypothetical stages of auditory information processing. Figure 5.1 shows a schematic diagram of the multi-stage process. The first stage is the representation of the acoustic signal in the peripheral auditory nervous system (sensory transduction). The cochlea receives the sound vibration information, which excites different parts of the basilar membrane depending on the frequency content. The movement of the basilar membrane at each point is transduced into neural impulses that are transmitted through nervous fibers composing the auditory nerve of the brain. The degree of activity present in each nerve fiber represents in part the spectral content of the input sound. The detailed timing of neural impulses carries the temporal characteristics of the sound. The sound acoustic content is essentially mapped on to the basilar membrane and encoded in the array of auditory nerve fibers.

The second stage is an auditory grouping process in which the array of components of the input sound are integrated as a group and segregated from other sound events. This process of auditory grouping is one of the key principles of the relatively new field of Experimental Psychology *Auditory Scene Analysis* [142].

The perceptual properties of the sound are analyzed at the third stage. Both local (few milliseconds) and global (a fraction of a second to a few seconds) features of the sound are analyzed to extract information about the identity of the sound. The local *micro-temporal* features characterize simple sound events (i.e., a tone) and are determined by the resonance properties of the sound source. These features are believed to help in the detection of the structural invariants of a sound source. Rhythmic and textual aspects of a sound are better captured in the global *macro-temporal* features. These long-time properties can integrate information about the dynamical changes in the acoustic environment.

In the next stage, the auditory features are then matched with a repertoire of memory representations of sound classes and events (*matching with an auditory lexicon*). Two auditory matching processes have been described by researchers. The first one is called the *process of comparison* whereby the auditory features are compared with a stored memory representation and the one with the closest match is selected. The other matching process involves a direct activation of memory representations of sources and events that are excited by the input auditory features. Thus, recognition of a sound event is achieved by the selection of the memory template that scores the highest degree of activation. If none of the memory representations exceeds the threshold of activation or if too many sound events are matched then no decision is made (i.e., the "do not know" case). The last stage of the recognition process is the retrieval of the stored information about the identified sound event from the listener memory such as names, concepts, and meanings associated with the perceived sound.

One of the debatable issues in auditory sound recognition is the way the various stages interact to reach to the final recognition of input sounds. For further details about human audition we refer the reader to the monograph (*Thinking in Sound: The Cognitive Psychology of Human Audition*) [141].

5.2 Noise Classification: Literature Review

A good part of the acoustic signals that reach our ears is environmental noises. The noise-generating sources can be humans (footsteps, applause etc.), machines (engine, traffic, fan, etc.) and nature (wind, water, for example). A human has a special built-in capability to deal with acoustic noises in different conditions. For example, in a noisy environment a person often speaks louder to compact interfering noise. However, environmental noises are more problematic for computers and speech processing systems. These noise signals result in performance degradation of those systems. For example, the accuracy of a speech recognition device might severely be affected if the level of noise is high and there is a mismatch between training and operating conditions. In speech coding, background noises can be coded with annoying artifacts. By modifying the processing according to the type of background noise, the performance can be enhanced. This requires noise classification. A survey of the literature reveals that automatic noise classification has been used as a useful tool in speech processing systems (speech recognition [143], speech enhancement and coding) and in some other noise-processing systems. In the sequel, we review existing applications of noise classification.

Treurniet and Gong [144] used recognition of the noise type to design a noise-independent speech recognition system. Nicol and his colleagues reported in [145] the use of a vector quantization technique to discriminate between different classes of noise for robust speech recognition in adverse environments. In [146], noise classification was used to selectively enable or disable a modification of the internal processing of a noise suppression system. Recently, Kumar [147] used fuzzy classification of background noise to automatically control the volume of a mobile handset to guarantee a quality voice service in noisy environments. The effect of environmental noises on the quality of voice communication systems has been studied within the ITU-T (International Telecommunication Union) Study Group 12, Question 17 ("Noise Aspects in Evolving Networks"). Noise classification is one of the major parts of this study [148].

In programmable hearing-aid devices, the electro-acoustic response depends on the noise environment. Noise classification can be used to improve hearing-aid performance by automatically adjusting the response to the listening noise conditions. Kates [149] proposed a noise classification system for hearing-aid applications.

Another application that has used noise classification is noise monitoring systems. These

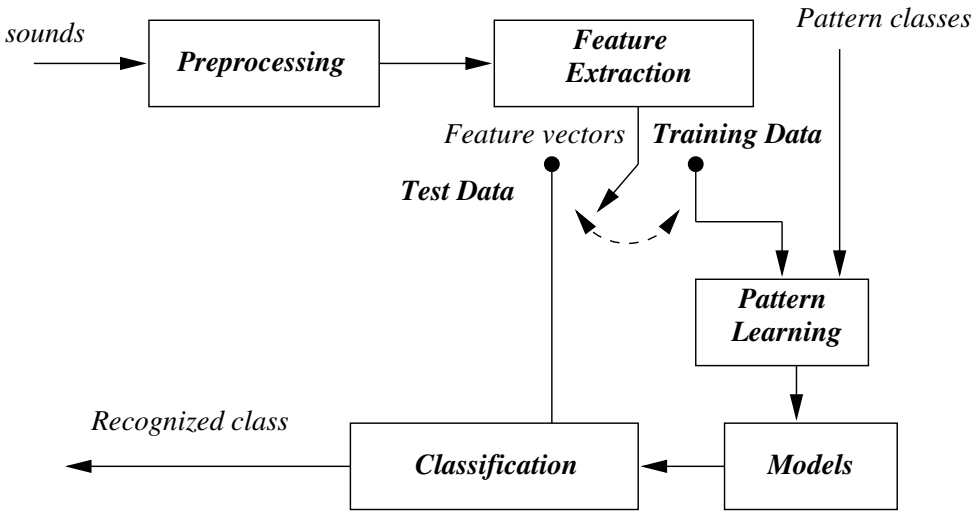

Fig. 5.2 A general block diagram of a sound pattern recognition system.

devices are used to record and analyze environmental sounds to identify noise pollution sources. Automatic discrimination of the various environment sources that are present in acoustic environments can be an effective tool to be used with conventional noise control methods [150]. Wearable computing devices can also benefit from identifying the surrounding noise scene. Clarkson [151] investigated the use of pattern recognition techniques to adjust the mode of operation according to the noise type and conditions.

In our work, we have used noise classification in the design of natural-quality multimode comfort noise coding algorithms for variable rate and DTX-based speech coders [56]. Recently, Beritelli *et al.* have also proposed the use of noise classification for speech coding applications [152] [153].

5.3 Noise Classification: Major Design Issues

The first step in designing an M-class noise classification system is the selection of M noise types. The choice of the noise classes depends on the intended applications. Noise classes can be defined in one of several ways. A class label can either indicate the noise source or the noise environment.

After the definition of the M noise classes, the next step will be the collection of acoustic signatures from each noise type. This data will be used for the design process of the recognition system. The collected data are divided into two distinct groups: one

for training (the *training data*) and one for testing and evaluation (the *test data*). It is important to gather a large training data sufficient enough to cover a large space of the M noise classes.

In Figure 5.2 we show a general block diagram of a sound recognition system. The design process starts by measuring some signal parameters that are believed to differentiate each noise type from the other noises. These are known as the *classification features* or feature vectors. In Section 5.4 we will discuss the choice of features and we will define the features that we have used in our noise classification system.

Another important step in the design phase is to select a classification method. A pattern recognition system designer will be overwhelmed with the large variety of classification algorithms. Once a classification method has been selected, then labelled features are then used to induce the classification algorithm via a supervised learning procedure that minimizes the probability of classification error. The final stage of the design process is to make sure that the classifier is meeting the design target by running performance evaluation test procedures to estimate the empirical error rate and compare it with a reference performance measure. Several design iterations might be required if the test results are not satisfactory. This might require the addition of new features, or changing the classification algorithm [154].

5.4 Classification Features

The choice of signal features is usually based on *a priori* knowledge of the nature of the signals to be classified. Features that capture the temporal and spectral structure of the input signal are often used. Examples of such features are zero crossing rate, root-mean-square energy, critical bands energies, and correlation coefficients. Features can be estimated using a short segment of the input signal (*short-time features*) or can be estimated using longer segments (*long-term features*).

Linear Prediction (LP) analysis is a major part of many modern speech-processing systems. Transformations of linear prediction coefficients (i.e., cepstral, log-area ratio coefficients) have been used successfully in many pattern-recognition problems (i.e., speech recognition, speaker recognition).

We have experimented with different sets of features derived from both the LP coefficients and the LP residual (i.e., residual critical band energies, zero crossing rate). In this

section we will define these classification features with more emphasis to the line spectral frequencies (LSFs) as they have been used as the core feature set for classifying different types of acoustic noises, and for classification of noise from speech.

5.4.1 Line Spectral Frequencies

The LSF representation, also known as line spectrum pairs (LSPs), was first introduced by Itakura in 1975 [155] as an alternative transformation of the linear prediction coefficients. Since their introduction, the LSFs have become the dominant parameters for representing the spectral envelope in LP-based speech coders. One of the contributions of our work is the use of the LSFs as the major feature set for noise and speech classification. In this section, we will first give a mathematical definition of the LSFs and show their relationship with the LP coefficients. Then, we present the salient properties of the LSFs that have made them popular in both spectral quantization and pattern recognition applications. We conclude by studying the statistical properties of the LSFs for both speech and different types of background acoustic noise.

- **Definition of LSFs**

  The derivation of the LSFs starts from the linear prediction inverse filter $A(z)$

  $$A(z) = 1 - \sum_{k=1}^{N} a_k z^{-k}, \tag{5.1}$$

  where N is the predictor order, and the a_k's are the predictor coefficients.

  Two new polynomials $P(z)$, and $Q(z)$ are then formed from $A(z)$ and its time-reversed system function $A(z^{-1})$

  $$\begin{aligned} P(z) &= A(z) + z^{-(N+1)} A(z^{-1}), \\ Q(z) &= A(z) - z^{-(N+1)} A(z^{-1}). \end{aligned} \tag{5.2}$$

  Using these two polynomials, the zeros of $A(z)$ are mapped onto the unit circle if $A(z)$ is a minimum-phase system (i.e., all its roots are inside the unit circle) [156]. The LSFs are defined as the angular frequencies of the roots of the LSF polynomials

$P(z)$ and $Q(z)$. Since the roots occur in complex conjugate pairs, the LSFs need only be computed on the upper semicircle of the z-plane.

Let us define

$$\begin{aligned} P(z) &= \sum_{k=0}^{N+1} p_k z^{-k}, \\ Q(z) &= \sum_{k=0}^{N+1} q_k z^{-k}, \end{aligned} \tag{5.3}$$

with $p_0 = 1$, $p_{N+1} = 1$ and $q_0 = 1$, $q_{N+1} = -1$.

From Eqs. (5.2) and (5.3), it can be shown that the LP coefficients are related to the coefficients of the LSF polynomials by the following relations [157]:

$$\begin{aligned} p_k &= a_k + a_{N+1-k}, \quad k \in \{1, 2, \ldots, N\}, \\ q_k &= a_k - a_{N+1-k}, \quad k \in \{1, 2, \ldots, N\}. \end{aligned} \tag{5.4}$$

The LP coefficients can be derived from the coefficients p_k and q_k as $a_k = (p_k + q_k)/2$.

Hereafter, the LSFs are denoted as the angular frequencies $\{\omega_1, \omega_2, \ldots, \omega_N\}$. The odd-suffixed LSFs correspond to the roots of $P(z)$ while the even-suffixed LSFs are the roots of $Q(z)$. The LSFs are ordered on the unit circle as follows

$$0 < \omega_1 < \omega_2 < ..... < \omega_N < \pi \tag{5.5}$$

In the sequel we will use the notation LSFi to mean the i^{th} LSF. The LSFs can also be expressed as frequencies in Hz.

Computation of the LSFs requires finding the roots of the polynomials $P(z)$ and $Q(z)$. Using numerical root-finding methods such as Newton-Raphson is computationally expensive, especially for real-time applications. Various methods have been proposed for the efficient computation of the LSFs. The most widely used method is the one developed by Kabal and Ramachandran [158]. They proposed a computationally efficient algorithm for computing the LSFs from the LP coefficients and vice versa. In

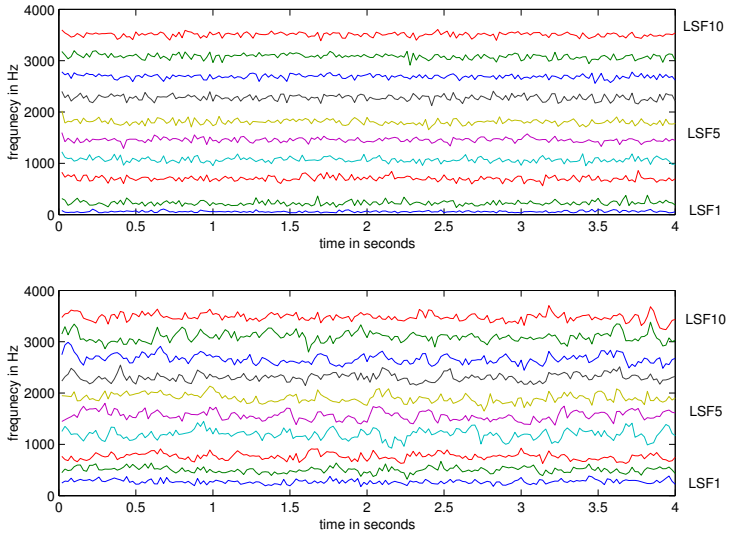

Fig. 5.3 Time evolution of the LSFs of background noise (a) car, (b) babble.

their algorithm, the use of trigonometric functions is obviated by using Chebychev polynomials to expand $P(z)$ and $Q(z)$. For a comparison between the different algorithms for the computation of the LSFs, see the recent study by Gracci [159]. In this work, we have used Kabal-Ramachandran method to compute the LSFs from the LP parameters.

- **LSFs properties**

  The LSFs have several useful properties that make them amenable to efficient quantization for low-bit-rate signal coding. The ordering property of the LSFs provides an easy and natural way to check the stability of the LP synthesis filters after quantization. Another useful property is the localized spectral sensitivity of each LSF parameter. That is a perturbation in one of the LSFs results in a change in the LP spectrum in the neighborhood of this LSF frequency. In Figure 5.3 we show the time evolution of the LSFs of two kinds of background noise (car and babble). We can observe that there is a strong correlation between the LSFs of successive spectra, even though babble noise shows more variability than car noise.

  As the LSFs have direct relationship with the roots of the LP filter, they have a close relationship to the peaks of the spectral envelope. The LSFs cluster around the

peaks of the spectral envelope. The distances between consecutive LSFs determine the bandwidth of the peaks in the spectrum.

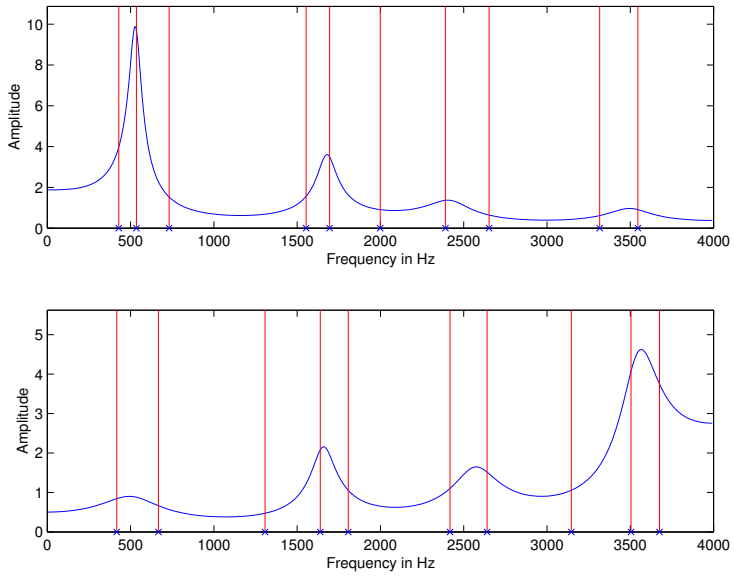

Fig. 5.4 Spectral envelope of a 20 ms speech frame with the 10 LSFs (in Hz) superimposed as vertical lines (a) voiced speech (b) unvoiced speech.

We show in Figure 5.4 the frequency locations of 10 LSFs (for an LP order of 10) superimposed on the spectral envelope of a 20-ms frame of speech. The spectral envelope and thus the locations of the LSFs are different depending on the phonetic character of a speech frame (i.e., voiced, unvoiced, onset). Unvoiced speech is highpass in nature and thus the last few LSFs are more perceptually important and they cluster around its peaks in the high frequency region. On the other hand, voiced speech is generally lowpass and the first few LSFs are more significant.

Environmental noises are generated from different sound-producing sources such as engines, people, machines, and nature. Each noise is characterized by different spectral and temporal contents. In Figures 5.5–5.6, we show a sample spectral envelope for some of the noises we have considered in this work: babble, car, factory, and street. It is clear from these figures that the noises are different in their spectral structure. Some noises are lowpass in nature such as car and factory while the others are more broadband especially babble noise. The 10 LSFs are positioned in the unit circle (i.e., in the frequency axis) depending on the "topology" of the spectrum. We have

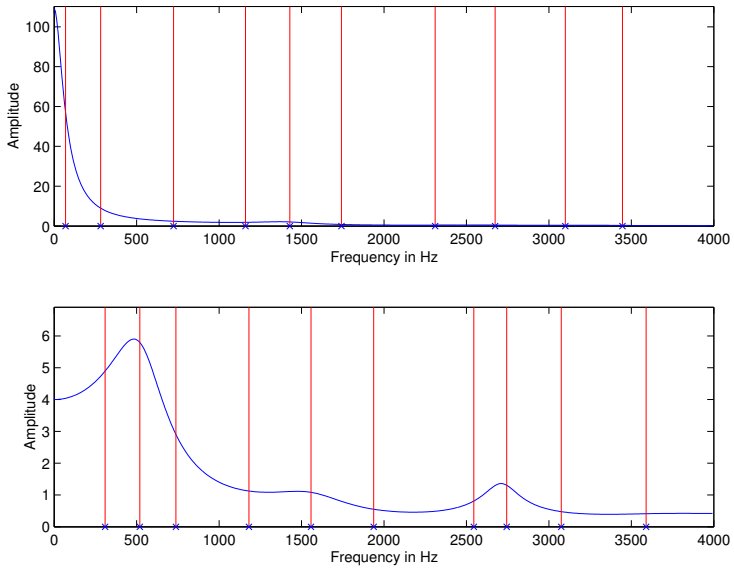

Fig. 5.5 Spectral envelope of a 20-ms frame of noise with the 10 LSFs (in Hz) superimposed as vertical lines (a) car, (b) babble.

exploited these differences in the LSFs spectral distribution to classify the different noises using only short segments of each noise (20 ms).

Recently, several researchers have studied the statistical properties of the LSFs. For a stationary autoregressive process, the LSFs are uncorrelated [160]. In [157], Tourneret has proposed a recursive method for computing the probability density function (PDF) of the LSFs as a function of the PDF of the LP parameters. As the LSFs and the LP parameters are related by a non-linear relationship they can not both be Gaussian. In Tourneret's work, the deviation of the PDF of the LSFs from its asymptotic Gaussian form was studied experimentally. He concluded that the LSFs have an approximately Gaussian distribution.

We have examined the statistical distribution of the LSFs for some typical noises using histograms of each LSF parameter. We show in Figure 5.7 the estimated histograms of the first two LSFs for 4 noise types and in Figure 5.8 the LSF7 and LSF8 histograms for the same noises. We can observe that the inter-class separability between the 4 noise types is stronger using the first 2 LSFs than with the higher LSFs. The 4 noises have almost overlapped PDFs for the higher LSFs.

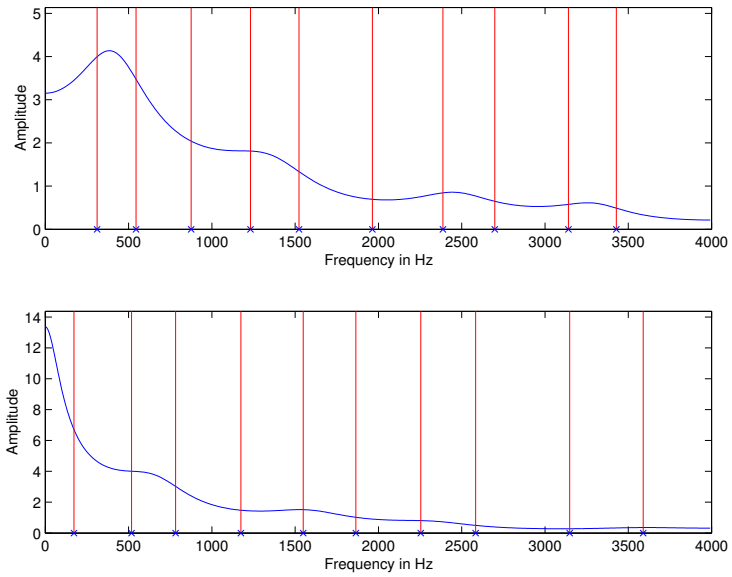

Fig. 5.6 Spectral envelope of a 20-ms frame of noise with the 10 LSFs (in Hz) superimposed as vertical lines (a) street, (b) factory.

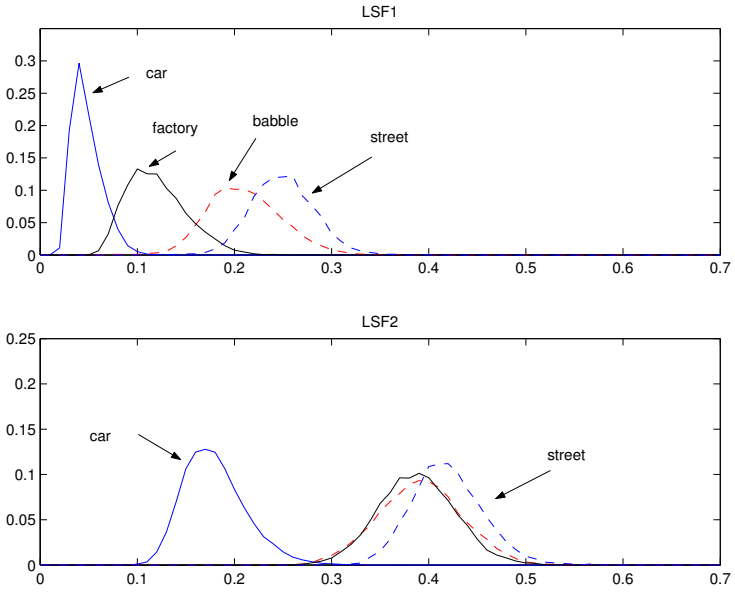

Fig. 5.7 Estimated histograms of the first 2 LSFs of 4 noises (car, babble, factory, street) (a) LSF1, (b) LSF2.

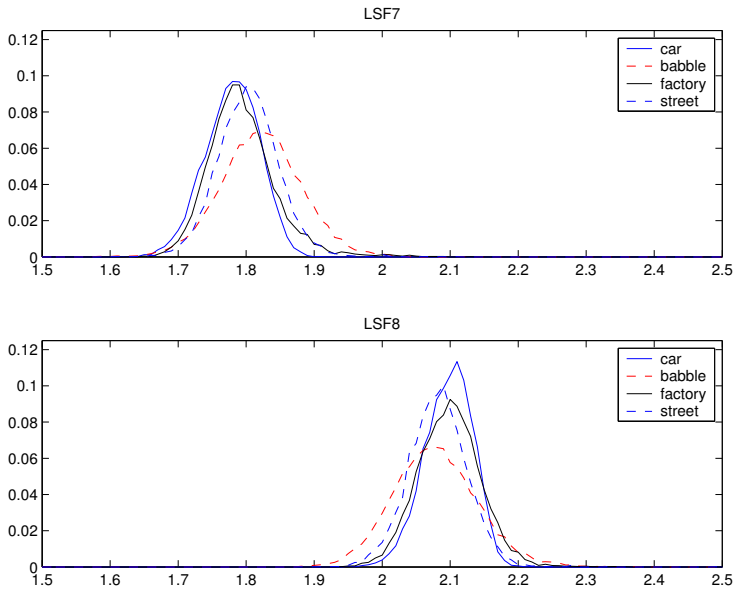

Fig. 5.8 Estimated histograms of LSF7 and LSF8 of 4 noises (car, babble, factory, street) (a) LSF7, (b) LSF8.

- **LSFs in pattern recognition**

  A commonly used LP-based feature for speech recognition is the cepstral coefficients defined in Section 5.4.2. Paliwal [161] [162], Liu and Lin [163] and Gurgen *et al.* [164] have studied the use of the LSFs as an alternative feature set for speech and speaker recognition. Campbell [165] has shown that for speaker recognition, the LSFs provide the best performance compared to other LP-based features.

  In [166] and [167], the LSFs were used as features for the classification of speech into its main phonetic units (i.e., voiced and unvoiced). Parry *et al.* [168] investigated the use of the phonetic structure of the LSF spectrum in the design of low rate spectral quantizers. In [169], a helicopter identification system was proposed comparing both the LSFs and the cepstral coefficients as classification features.

  In the above studies, the potential of the LSFs as a classification feature has been reported. In this chapter, we present our experimental results in using the LSFs as classification feature for discriminating between different types of environmental noises, between noise and speech, and between speech and different types of music.

5.4.2 Cepstral Coefficients

The cepstrum of a signal is the inverse Fourier transform of the logarithmic power spectrum:

$$\log\left[\frac{1}{|A(e^{j\omega})|^2}\right] = \sum_{n=-\infty}^{\infty} c_n e^{-jn\omega}, \tag{5.6}$$

where $c_n = c_{-n}$ and $c_0 = 0$, are labelled as *cepstral coefficients*. An infinite number of cepstral coefficients can be computed from the prediction coefficients $a_n's$ using

$$c_n = a_n + \sum_{k=1}^{n-1} \frac{k}{n} a_{n-k} c_k. \tag{5.7}$$

Recently, Kim *et al.* [170] derived a relationship between the cepstral coefficients $\{c_n\}$ and the LSFs $\{w_i\}$ as follows:

$$c_n = \frac{1}{2n}[1 + (-1)^n] + \frac{1}{n}\sum_{i=1}^{M} \cos nw_i + R(n), \qquad n = 1, 2, 3, \ldots, \tag{5.8}$$

where $R(n)$ is a term that provides the magnitude information about the inverse filter $A(z)$.

We have included the cepstral coefficients among our set of classification features and compared their performance with the LSFs.

5.4.3 Other Features

Both the LSFs and cepstral coefficients are spectral features characterizing the spectral content of acoustic signals. A commonly used feature is the zero crossing rate (ZCR). It is the zero-crossing count of a waveform over a defined time period. The ZCR also characterizes the frequency content of signals. For example, unvoiced speech has a much higher ZCR than voiced speech. This is in line with unvoiced speech being a rapidly changing signal and voiced speech consists of a more slowly varying waveform [73]. In our work, we have investigated the use of higher order crossings (HOC)[1] for sound classification.

We propose a new classification feature: the linear prediction ZCR (LP-ZCR). It is defined as the ratio of the ZCR of the input signal to the LP analysis filter and the ZCR of the output signal. As mentioned in Chapter 3, the output signal of an LP analysis filter

[1] HOCs have been defined in Section 3.3.2.

(the LP residual) is a decorrelated signal with almost flat spectrum. Thus, the ZCR of the output signal is always higher than the input signal. The LP-ZCR can quantify the correlation structure of the input sound. For example, a highly correlated sound such as voiced speech will have a low LP-ZCR, while unvoiced speech will have a value close to 1. The LP-ZCR for a white Gaussian noise is ideally 1.

In addition to the spectral features, we have experimented with using additional features from the LP residual signal. We computed several normalized energies from several bands of the LP residual spectrum and used them as features. These energies did not show promising results in differentiating between the different types of noise. Even when combined with the LSFs, the improvement in classification accuracy was small and thus this additional feature set was abandoned.

It should be clear that during the design stage of a pattern recognition system, it is natural to experience the tedious process of trial-and-error of different signal parameters until a feature set shows a promising discrimination power. A short cut to the feature selection process can be achieved using some *a priori* knowledge of the nature of signals to be classified and the amount of signal information that will be available to the classifier.

5.5 Classification Algorithms

5.5.1 Introduction

A critical decision that faces the designer of a pattern classification system is the selection of a classification technique. Various factors can help in choosing a *good* classifier. The literature of pattern recognition provides different categories of classification methods that vary in complexity, and the required training time to optimize their performance for a given problem.

In this work, our goal is to select a computationally simple, yet robust and efficient classifier architecture that can be integrated in mobile handsets without burdening the available computational and memory resources. As part of our learning process, we have experimented with different classification algorithms to gain understanding of the relationship between the performance and the architecture of each classifier. As we will show in this chapter, we have managed to identify simple classification techniques that will be suitable for mobile devices such as telephone handsets.

In this section we will give a detailed discussion of one of the major classical, yet important, classification framework (Bayesian classification) and present a brief overview of some other algorithms that we have evaluated for noise classification.

5.5.2 Bayesian Classification

An elegant way to represent an M-class classifier is in terms of a set of *discriminant functions* $g_i(\mathbf{x})$, $i = 1, 2, ..., M$. The classification decision rule assigns the class label ω_i to an input feature vector $\mathbf{x}$ if

$$g_i(\mathbf{x}) > g_j(\mathbf{x}),\ \forall\, j \neq i, \tag{5.9}$$

or,

$$\hat{\omega}(\mathbf{x}) = \omega_i,\ \ i = \arg\max_j g_j(\mathbf{x}). \tag{5.10}$$

The decision rule basically divides the feature space into M disjoint *decision regions* separated by *decision surfaces* defined by the equalities $g_i(\mathbf{x}) = g_j(\mathbf{x})$.

The classification of an input vector reduces to its assignment to a class based on its location in the feature space. Generally, if the features are well chosen, vectors belonging to the same class group together into *clusters*. Finding the decision rule can be viewed as finding the decision surfaces that will best separate the clusters in the feature space.

The target of a classification system is to minimize the probability of misclassification. It has been shown [171] that, for an input feature vector $\mathbf{x}$, choosing the class with the maximum *a posteriori* probability is the optimal decision rule, in the sense of the minimum probability of classification error. This is the Bayes classification decision rule defined as:

$$\omega^*(\mathbf{x}) = \arg\max_{j=1,2,...,M} P(\omega_j|\mathbf{x}). \tag{5.11}$$

Comparing Eqs. (5.10) and (5.11) we can observe that the discriminant functions for the Bayes classifier correspond to the *a posteriori* probabilities. The class-conditional probabilities $p(\mathbf{x}|\omega_i)$ are usually easier to compute than the *a posteriori* probabilities. The Bayes rule links the two probabilities as

$$P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)P(\omega_i)}{p(\mathbf{x})}, \tag{5.12}$$

where $P(\omega_i)$ is the *a priori* probability of each class, and $p(\mathbf{x})$ is given by

$$p(\mathbf{x}) = \sum_{j=1}^{M} p(\mathbf{x}|\omega_j)P(\omega_j). \tag{5.13}$$

Taking the natural logarithm of both sides of Eq. (5.12) and dropping the terms independent of class label, the discriminant functions for the Bayes classifier can be expressed as

$$g_i(\mathbf{x}) = \ln p(\mathbf{x}|\omega_i) + \ln P(\omega_i), \text{ for i=1,2, ..,M.} \tag{5.14}$$

To realize a Bayes classifier we need to know the M class-conditional probabilities. These can be estimated from the training data using either non-parametric density estimation techniques such as histograms, kernel density estimators, k-nearest neighbor methods etc. [171], or by assuming a parametric model and estimating its parameters.

A common parametric model is the multivariate Gaussian distribution. The class-conditional PDF for the ω_i class is given as

$$p(\mathbf{x}|\omega_i) = \frac{1}{(2\pi)^{d/2}|\mathbf{\Sigma}_i|^{1/2}} \exp[-\frac{1}{2}(\mathbf{x}-\mu_i)^T\Sigma_i^{-1}(\mathbf{x}-\mu_i)], \tag{5.15}$$

where $\mathbf{x}$ is a d-dimensional feature vector, μ_i is the d-dimensional mean vector of class ω_i, and $\mathbf{\Sigma}_i$ is the $d \times d$ covariance matrix of class ω_i.

A common method of estimation the mean and covariance parameters of the Gaussian classifier is the *maximum likelihood* (ML) method. Using the ML method it can be shown [154] that the estimated class mean vector, from a training data in the feature domain χ_i, is given as

$$\hat{\mu}_i = \frac{1}{N_i} \sum_{\mathbf{x_j} \in \chi_\mathbf{i}} \mathbf{x}_j, \tag{5.16}$$

and the class covariance matrix is estimated as

$$\hat{\mathbf{\Sigma}}_i = \frac{1}{N_i} \sum_{\mathbf{x_j} \in \chi_\mathbf{i}} (\mathbf{x}_j - \hat{\mu}_i)(\mathbf{x}_j - \hat{\mu}_i)^T. \tag{5.17}$$

If the training data size is very large or when there are not enough training samples, the mean and covariance can be estimated recursively using Eqs. (5.18) and (5.19). Let us

denote $\hat{\mu_{\mathbf{N}}}$ to be an estimate of the mean vector using N samples, and $\hat{\mu_{\mathbf{N+1}}}$ as the mean vector estimate using $N+1$ samples.

$$\hat{\mu}_{N+1} = \frac{1}{N+1}\sum_{k=1}^{N+1}\mathbf{x}_k = \hat{\mu}_N + \frac{1}{N+1}(\mathbf{x}_{N+1} - \hat{\mu}_N), \tag{5.18}$$

and the covariance matrix recursion is given by [165]:

$$\hat{\mathbf{\Sigma}}_{N+1} = \frac{1}{N}\sum_{k=1}^{N+1}(\mathbf{x}_k - \hat{\mu}_{N+1})(\mathbf{x}_k - \hat{\mu}_{N+1})^T = \frac{N-1}{N}\hat{\mathbf{\Sigma}}_N + \frac{1}{N+1}.(\mathbf{x}_{N+1} - \hat{\mu}_N)(\mathbf{x}_{N+1} - \hat{\mu}_N)^T. \tag{5.19}$$

These recursive estimates can be also useful to make the Gaussian models adapt to the conditions of the system by updating the model of each noise class using knowledge gained during operation.

It can be shown that the Gaussian classifier has a quadratic discriminant function given as

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \mu_i)^T\Sigma_i^{-1}(\mathbf{x} - \mu_i) - \frac{1}{2}\ln|\Sigma_i| + 2\ln P(w_i). \tag{5.20}$$

Thus, a Gaussian classifier is characterized by a set of parameter pairs (mean vector, and covariance matrix) for each one of the M classes. The decision criterion using this classifier will be simply computing the distance metric g_i for each class and picking the class that has the largest value. The Gaussian classifier is optimal if the feature vectors follow a Gaussian PDF. Otherwise, the mismatch in the PDF modelling can harm the efficacy of the classifier. Hereafter, the quadratic Gaussian classifier will be denoted as (QGC).

5.5.3 Nearest Neighbor Classification

In nearest-neighbor (NN) classification, for each input feature vector, a search is done to find the label of the vector in the dictionary of stored training vectors with the minimum distance [172]. Euclidean distance is commonly used as the metric to measure neighborhood. The nearest neighbor classifier is a non-parametric classifier as no assumption is made on the form of the PDFs of the training data.

A more general form of the NN decision rule is the k-NN classifier. The input feature vector is assigned the label most frequently represented among the k nearest patterns in

the training dictionary. A k-NN classifier generally improves over the performance of a 1-NN classifier at the cost of more computations.

One of the major disadvantages of NN classifiers is the need to store a large number of training vectors. As a remedy to this problem, only prototype vectors from the training data are computed and stored. This is known as prototype nearest neighbor classification (PNN) [173]. Several techniques have been proposed in the literature for the computation and the selection of a set of prototypes that define the NN dictionary. For example, Decaestecker [174] used both gradient descent and deterministic annealing to find prototypes for the NN classifier.

Learning vector quantization (LVQ)[2] is an example of a prototype nearest-neighbor classification. A set of L vectors (prototypes) is computed from the labelled training data to minimize the misclassification errors using nearest-neighbor decision rule. An initial set of L vectors is chosen from the training set. An iterative update rule is used to modify the vectors in such a way that achieves a better classification of the training set by the 1-NN rule based on the selected vectors. The final set of the L vectors defines the LVQ codebook to be used in the testing mode. The size of the codebook (L) and the distribution of the vectors amongst the classes are two free parameters to choose in the design process. For more details about LVQ-based classification see the book by Kohonen [175].

In Section 5.11.2, we will propose a reduced-memory PNN classifier for noise classification and show that we can still get good classification results using a reduced storage requirements (one prototype for each class).

5.5.4 Other Algorithms

In this section we will briefly describe other pattern classification schemes that we have tested at earlier stages of our work.

A decision tree classifier (DTC)[3] belongs to the family of machine learning techniques. During the training phase, a set of production rules are generated from the labelled data in the form of a decision tree. The decision tree is then used to classify unlabelled test vectors. The inductive tool used in our simulation is an implementation of the C4.5 programs developed by Quinlan [176]. Inductive learning produces decision trees that use the most

[2]LVQ was used in an early stage of our study and the classification results were reported in [57].

[3]We reported the use of DTC for noise-speech classification in [56]. This was part of a joint work with University of Wollongong, Australia.

discriminative features. In [177], a decision tree-based system was proposed for phoneme classification. For more details about decision tree-based classification see [176].

Another parametric classification algorithm is the linear classifier. It has a simple linear discriminant function given as

$$g_i(\mathbf{x}) = \mathbf{w}_i^T \mathbf{x} + w_{i,0},\ \text{i=1,2, } \ldots, \text{ M}, \tag{5.21}$$

where $\mathbf{w}_i = \{w_{i,1}, ..., w_{i,d}\}$ is called the *weight vector* and $w_{i,0}$ is the *threshold weight*. Designing a linear classifier reduces to finding the weight vectors $\mathbf{w}_i$ and the threshold weights using the training data. Different algorithms have been proposed for this purpose. In this work, we have adopted a least-squares approach to design linear noise classifier.

Neural network (NNet) classifiers have emerged in the last decade as a step towards emulating the internal pattern recognition process of the human brain. One of the well-studied NNets is the Multi-layer Perceptron (MLP) neural network[4] [179]. Three major layers characterize MLP NNet: input layer, hidden layer (at least one) and the output layer. The number of nodes in the input layer is related to the dimensionality of the input feature vector while the number of the output nodes is related to the form of the output. The number of hidden layers and their structure (number of nodes) and their inter-connections are free design parameters that depend on the intended application.

5.6 Performance Evaluation

An important step in the process of designing a sound classification system is the evaluation of its performance by estimating its probability of classification error. Error counting methods [171] are often used to estimate empirical error rate of a classification system. A cross-validation evaluation method uses the training data to train the classifier, and the test data to estimate the resulting error rate. If no independent test data is available during the design process, the common practice is to split the available data into two subsets: one for training and the other for testing. Different splitting strategies have been proposed in the literature and in the sequel we will briefly review them.

In the *Resubstitution* method, the entire data set is used for both training and testing

[4]The results of using MLP NNet for noise classification were presented in [178]. This was a student project that the author co-supervised.

the classifier. This technique can give biased results that do not measure the robustness of the system. However, it gives a lower bound on the empirical error rate. In the *Hold-out* method, a percentage of the available data is used as test vectors. For example, 30% of the vectors can be used for testing, and the remaining data for training the classifier. The test vectors can be selected randomly from the available data set to increase the variability of the feature set. The empirical error rate is computed by averaging the error rates of K iterations. Another validation method is the *leave-one-out* method. The classifier is trained using all the vectors except one vector that is used for testing. The procedure is then repeated for all the vectors. The error estimate is computed by counting the frequency of errors. We have experimented with all the aforementioned cross-validation methods and have selected the Hold-out method for computing the empirical error rate for each classifier.

An important figure of merit in pattern recognition is the Bayes error rate P_{Bayes} [171]. It measures the discriminating power of the features independent from the classification algorithm. It gives a lower bound on the performance of any classifier designed for a given problem. The Bayes error rate can be used as a reference to assess the loss in performance due to the choice of a particular classification algorithm. If the empirical rate of a decision algorithm is much higher than the Bayes rate this indicates that other classification architectures should be tried.

To compute the Bayes error rate we need to have the *a posteriori* or likelihood probabilities. An alternative way is to use lower bound formulas on the Bayes rate. In [172], a lower bound on the Bayes rate is a function of the asymptotic error rate of the nearest-neighbor decision rule P_{NN} given as

$$P_{Bayes} = \frac{M-1}{M}(1 - \sqrt{1 - \frac{M}{M-1}P_{NN}}). \tag{5.22}$$

where M is the number of classes.

5.7 Classification Results

In this section we will present our experimental results for classifying different types of background noise and for classifying noise from speech. The Bayes error rate and empirical error estimation were used to gauge the performance of the proposed features and classifiers prior to independent testing. A detailed presentation of the classification results for each

class is given in the form of a classification matrix.

In the sequel we will present the classification results using the LSF feature set with different classification algorithms. More emphasis will be given to the quadratic Gaussian classifier due to its promising performance and its suitability for real-time implementation.

5.7.1 Noise-only Classification

One of the important steps in designing a noise classifier is the definition of the M noise classes. In our case, the output of a noise classifier will be used to select an *appropriate* excitation signal corresponding to the selected class for the class-dependent residual excitation model.

One way to select the noise classes is based on the environment of the noise. For example, a street noise means an acoustic noise measured in a street environment. Similarly, a car noise indicates a background noise signal recorded inside a car. As our main target is to design noise coding schemes for wireless telephony devices, we have selected 5 commonly encountered noise environments (car, street, babble[5], bus, and factory) to specify 5 noise types or classes for the design of our noise classification system.

Table 5.1 gives the empirical error rate evaluated with the hold-out procedure for the various classifiers. Using Eq. (5.22) and the empirical error rate of the 1-NN classifier (19.8%), the Bayes error rate was estimated at 10.6%. This means that independent of the classifier structure, the best frame-level error rate for the 5 selected noises, and with the 10 LSFs as features is 10.6%. From the table, both the decision tree classifier and the quadratic Gaussian classifier approach that error rate with 11.9% and 13.6% respectively. The other tested classifiers are less accurate.

To test the Gaussian classifier, we used 500 test vectors from each class. In Table 5.2 we show the classification matrix of the Gaussian classifier for the 5 noise classes. It is clear that the classification accuracy is different for each class. For example, accuracy ranging from 90–100% were obtained for car noise, and factory noise. Street, babble, and bus noises are more often misclassified with accuracy rates ranging from 65–80%. In this test, the average accuracy rate for the 5 noises was 85.8%. Street noise was confused with bus noise for 25% of the input frames. Also, babble noise was detected as bus noise for around 13% of the frames. We show in Figure 5.9 a sample of a sequence of decisions for

[5]Babble noise is a representative of restaurant noise environment.

Table 5.1 Empirical error rate for the different classifiers (noise-only)

Classifier	Error Rate %
Optimal Bayes	10.6
Decision Tree	11.9
Quadratic Gaussian	13.6
Neural Network (MLP)	15.8
3-Nearest Neighbor	17.5
Learning Vector Quantization	19.2
1-Nearest Neighbor	19.8
Linear (least-squares method)	21.9

babble noise. In most cases, false decisions occur in isolation which suggests that we can use post-decision correction schemes to improve the accuracy rate.

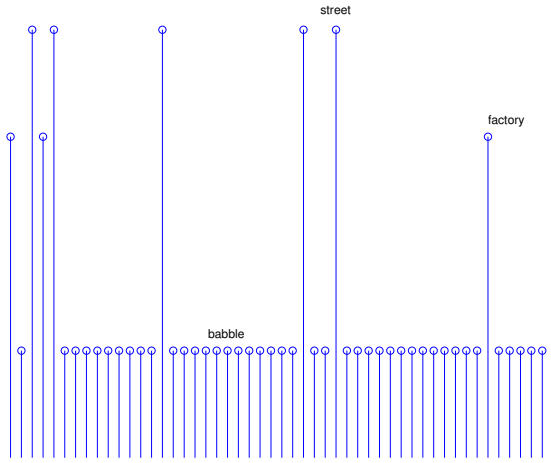

Fig. 5.9 A sequence of noise decision for a segment of babble noise.

To get some insight into the relationship between inter-class decision errors and the LSF distribution of each noise class, we show in Figures 5.10 and 5.11 scatter diagrams of the first 2 LSFs (LSF1 and LSF2) for the babble, car, and street noise types. From Figure 5.10, it is clear that car, factory and babble noises are well-separated in this 2-dimensional LSF space. This might explain why the confusion rate among these classes is minimal. However, this situation is different for the other noise classes with more false decisions: street, bus and babble noises. In Figure 5.11, the first two LSFs of these 3 noises

Table 5.2 Classification matrix: Gaussian classifier (noise-only)

	Babble %	Bus %	Car %	Factory %	Street %
Babble	**79.8**	12.8	0.0	2.0	5.4
Bus	8.8	**85.2**	0.0	2.2	3.8
Car	0.0	0.2	**99.6**	0.2	0.0
Factory	1.0	5.6	0.0	**93.2**	0.2
Street	1.8	24.8	0.0	2.0	**71.4**

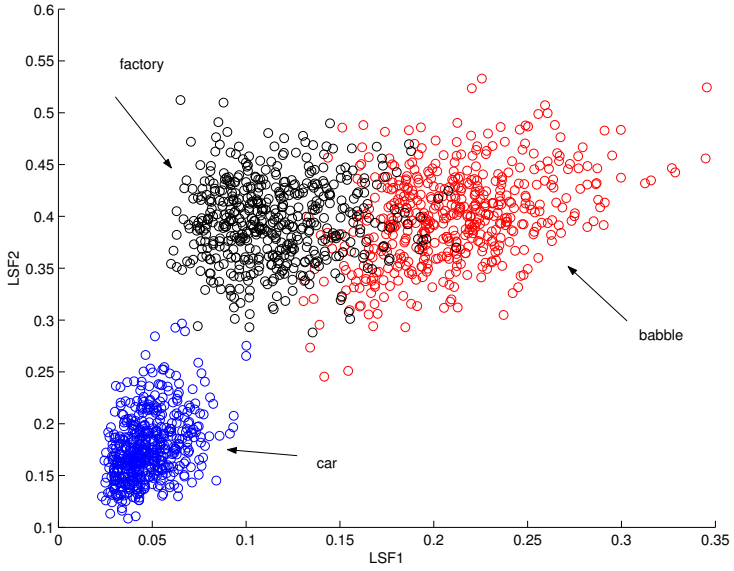

Fig. 5.10 Scatter diagram of the first two LSFs (LSF*1* and LSF*2*) of car, babble, and factory noises.

show large overlapping that agrees with the classification matrix in Table 5.2.

Recently, Beritelli *et al.* [153] proposed a fuzzy pattern classification system for background noise in mobile environments. We show in Table 5.3 the 11 features used to design the fuzzy classifier. The feature set includes mainly differential and norm of LP-based parameters such as cepstral and log area ratios (LAR) coefficients.

Beritelli *et al.* compared the results of their fuzzy noise classifier with our noise classification results presented in [56]. We show the classification matrix of the fuzzy classifier in Table 5.4. The same 5 noise types that we have used in our experimentation were used for

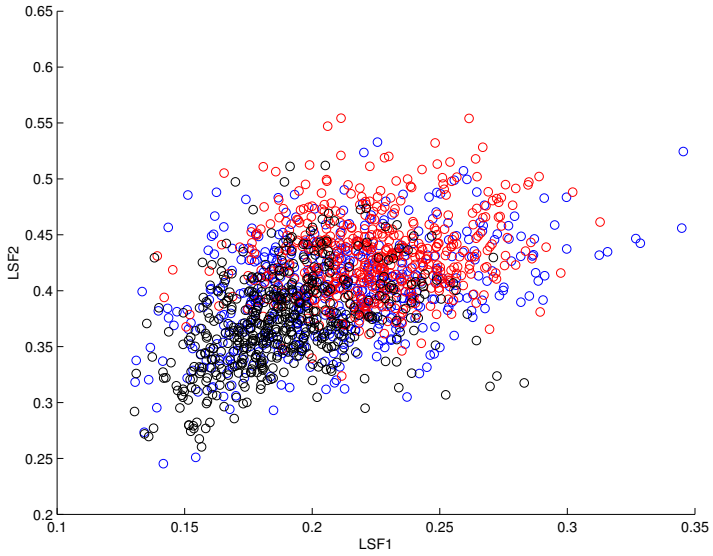

Fig. 5.11 Scatter diagram of the first two LSFs (LSF*1* and LSF*2*) of babble, bus, and street noises.

Table 5.3 A list of the classification features used in the Fuzzy classifier

Feature
Regularity
Right variance
Norm of cepstrum
Norm of LAR coefficients
Norm of LP cepstrum
Norm of cepstral coefficients
Differential power
Differential variance
Differential left variance
Differential prediction gain
Inverse of prediction gain
First LP coefficients
First cepstral coefficients
First normalized reflection coefficients
Normalized energy in the $0-900$ Hz band

their classifier. Even though bus noise was mostly confused as babble noise in our classifier, the bus noise that was used in Beritelli's classifier was confused more with car noise. It should be pointed out that both studies used different noise databases during both the design and testing of the noise classifiers.

We can conclude from the results presented from the two independent noise classification implementations that it is important to define a set of commonly associated noise events with each noise class. In our case, the bus noise recording is a rich mixture of engine noise, babble noise, background music from the bus radio, and some traffic noise. This indicates why the tested bus noise frames were confused with babble noise and street noise. We suppose that the bus noise signal used to train the fuzzy classifier is dominant with bus engine noise and thus was confused with car noise.

Table 5.4 Classification matrix: Fuzzy classifier

	Babble %	Bus %	Car %	Factory %	Street %
Babble	**84.0**	2.7	5.1	3.1	5.1
Bus	0.1	**90.4**	9.5	0.0	0.0
Car	0.0	5.7	**94.3**	0.0	0.0
Factory	1.6	0.0	0.0	**93.5**	4.9
Street	10.8	6.5	0.0	3.4	**79.3**

In the second stage of our study, we have decided to eliminate bus noise as one of the classes and replace it with white Gaussian noise (WGN). This noise is a good representation of several background noises characterized with Gaussian sample-distribution. Thus, our new set of 5 classes of noise are: babble, car, factory, street and WGN. Hereafter, all the results that we will present will be for these 5 noises.

To demonstrate the effect of the selection of noise classes on the error rate of a trained classifier with the same feature set, we show in Table 5.5 the accuracy rate using the noise class set with and without bus noise. The error rate for the Gaussian classifier has been reduced from 13.6% to 4.8%. The mean reason for this difference is that we have removed the noise class that causes the most overlapping in the LSF feature space between most of the included noises (i.e., babble, street, and bus). Also, we have added WGN which has a special LSF structure that is different from the other noise classes. Also, a Bayes error rate of 4% illustrates the discriminating power of the LSFs as a feature set for noise

Table 5.5 Empirical error rate for the different classifiers (noise-only)

Classifier	Error Rate % new noise set (with WGN)	Error Rate % old noise set (with bus noise)
Optimal Bayes	4.0	10.6
Quadratic Gaussian	4.8	13.6
1-Nearest Neighbor	7.8	19.8

classification.

A good practice during the training phase of pattern recognition systems is to experiment with several signal parameters. In addition to the LSFs we have tried ZCR, differential LSFs (DLSFs)[6], and cepstral coefficients. The test results of the Gaussian classifier with the different feature sets are shown in Table 5.6. DLSFs give information about the bandwidth of the major peaks and can be useful in characterizing noises with narrow resonant frequencies. As shown in the table we did not gain from using either the 9 DLSFs alone or from combining 3 ZCR features[7] with the 10 LSFs. The cepstral coefficients have shown performance close to the LSFs. The classification matrix for the LSFs is shown in Table 5.7 and for the cepstral set in Table 5.8. The LSFs outperformed the cepstral features by 3.0% as can be seen from the tables. The 92.2% accuracy for the cepstral coefficients is still sufficient for our application. For speech coding application the LSFs are already available, thus the LSFs will be the features that we will use for our classification system.

Table 5.6 Test results using the QGC with different feature sets

Feature Set	Accuracy Rate %
LSFs (10)	95.2
Cepstral coefficients (10)	92.2
DLSFs (9)	88.1
LSFs and ZCR (13)	83.5

We have studied the effect of LP order on the performance of the Gaussian classifier and we show the results in Table 5.9. As expected, lower order LP models are less accurate

[6]DLSFs are defined by taking successive differences of the LSFs.

[7]The 3 ZCR features are the ZCR of the input signal, the ZCR of its LP residual, and the ratio of the two ZCRs (LP-ZCR).

Table 5.7 Classification matrix (LSFs): QGC (95.2% accuracy)

	Babble %	Car %	Factory %	Street %	WGN %
Babble	**88.7**	0.0	3.3	8.0	0.0
Car	0.0	**99.8**	0.2	0.0	0.0
Factory	3.8	0.0	**94.3**	1.9	0.0
Street	5.8	0.0	1.1	**93.1**	0.0
WGN	0.0	0.0	0.0	0.0	**100.0**

Table 5.8 Classification matrix (cepstral coefficients): QGC (92.2% accuracy)

	Babble %	Car %	Factory %	Street %	WGN %
Babble	**80.2**	0.0	7.4	12.4	0.0
Car	0.0	**99.2**	0.8	0.0	0.0
Factory	3.0	0.0	**95.6**	1.4	0.0
Street	6.0	0.0	7.8	**86.2**	0.0
WGN	0.0	0.0	0.0	0.0	**100.0**

than higher order models and a minor improvement in accuracy (less than 2%) is gained by doubling the LP order from 10 to 20. Thus, we will stick with a 10^{th} order LP model as this is the most commonly used LP order in standardized narrowband speech coders.

Table 5.9 LP order and accuracy rate

LP order	Accuracy %
2	84.2
5	90.9
10	95.2
20	96.9

5.7.2 Classification of New Noises

In practical applications of noise classification, the input noise signals are not constrained to belong to one of the 5 pre-selected noise classes. Thus, a 'good' noise classifier should have the ability to map an input feature vector from a new noise class to the closest pre-selected classes. We tested the Gaussian classifier with 6 new noise signals (bus, restaurant, shopping mall, sports, subway, and traffic). The results are presented in Table 5.10. It is interesting to observe that the classifier maps the new noises to the noise classes with the same noise events. As an example, a restaurant noise is a mixture of babble noise, background music, and other ambient noises. In our test, restaurant noise was classified 89.4% as babble noise, which is a dominant background noise in such an environment. Another example is the bus noise that we used early as one of the selected classes. It was mapped 65.0% to babble noise and the remaining frames were mapped to street and factory classes. This matches the content of our bus noise recording as it has both babble-speech and traffic-like noises.

Table 5.10 Classification matrix of new noises

Noise	Babble %	Car %	Factory %	Street %	WGN %
Bus	65.0	0.0	15.6	18.8	0.0
Restaurant	89.4	0.0	6.6	4.0	0.0
Shop. Mall	53.8	0.0	0.0	46.0	0.2
Sports	37.2	0.2	6.5	56.1	0.0
Subway	68.0	0.0	28.9	3.1	0.0
Traffic	5.0	0.0	0.0	95.0	0.0

5.7.3 Identification of the Noise Type from Noisy Speech Signals

In speech-processing systems, the speech signal is often contaminated by different types of background acoustic noise. Identifying the type of the noise and its energy level from the noisy speech signal can be essential to mitigate its effect on system performance. In the previous sections, a noise classifier was used to distinguish between the M defined noise classes or to map new noises to one of the M noise types. An interesting experiment is to test the noise classifier with LSFs from noisy speech signals, and then evaluate if

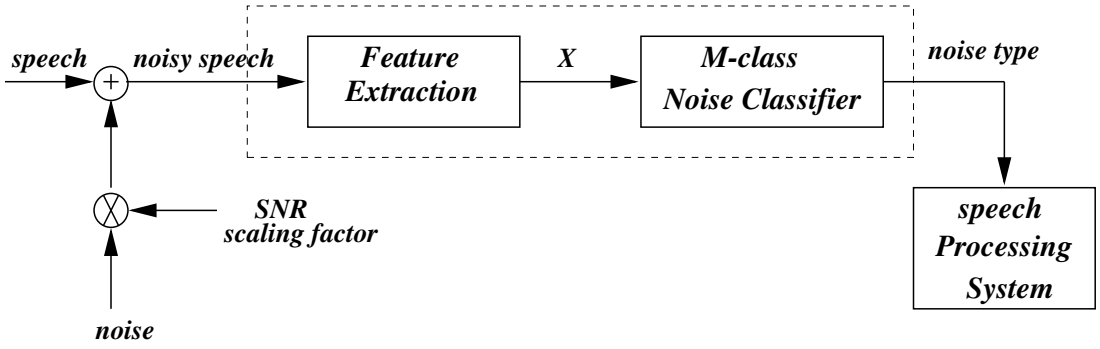

Fig. 5.12 Identification of the noise type from a noisy speech signal.

the selected noise class matches the actual noise contaminating the speech. We show in Figure 5.12 the experimental set-up for this study. A noisy speech signal is processed by the feature extraction unit to provide the input features to the classifier. The output of the classifier is a noise class label that can be used by a speech processing system to adapt its operation to the acoustic noise environment.

We show in Table 5.11 the noise-type identification results using 5 test noisy speech signals, each one with one type of the 5 noise classes. For this task, we used the Gaussian classifier with a 10-LSF feature vector. The results presented are averaged across 2500 test frames. For each noisy speech signal, the 'right' noise type was identified with accuracies ranging from 76% to 100%.

Table 5.11 Identification of the noise type from a noisy speech signal

Noise Input signal	Babble %	Car %	Factory %	Street %	WGN %
speech with babble	**89.8**	0.0	2.4	7.8	0.0
speech with car	21.2	**76.0**	2.8	0.0	0.0
speech with factory	15.0	0.0	**83.6**	1.4	0.0
speech with street	14.6	0.0	2.6	**82.8**	0.0
speech with WGN	0.0	0.0	0.0	0.0	**100.0**

5.7.4 Noise-and-Speech Classification

In this work, we have included speech as one additional class to the 5 noise types (babble, car, factory, street, and WGN) and designed a noise-and-speech classifier. One objective of

this work was to examine if the Gaussian classifier (using the LSFs) can distinguish speech from the rest of the noises. The training speech samples were from a clean-speech database.

In Table 5.12, we present the estimated Bayes rate and the estimated error rates for the noise-and-speech case. The classification matrix is shown in Table 5.13. Speech signal is around 86% accurately discriminated from the noises, and falsely detected as babble noise most of the time. This matches our human perception as babble noise is the only speech-like noise. The addition of speech as one class did not affect the results of most noises except babble noise, as expected.

We show in Table 5.14 that we can improve the accuracy results for the noise-and-speech case by combining the 10 LSFs with 9 DLSFs, and then selecting the 10 best features out of the 19[8]. However, this increase is only 1.2% and using only the 10 LSFs is sufficient.

Table 5.12 Empirical error rate for the different classifiers (noise-and-speech)

Classifier	Error Rate %
Optimal Bayes	4.4
Quadratic Gaussian	6.3
1-Nearest Neighbor	8.6

Table 5.13 Classification matrix: QGC (noise-and-speech)

	Speech %	Babble %	Car %	Factory %	Street %	WGN %
Speech	**86.8**	8.8	0.2	1.0	0.4	2.8
Babble	5.8	**79.2**	0.0	7.6	7.4	0.0
Car	0.2	0.0	**99.8**	0.0	0.0	0.0
Factory	0.2	3.6	0.0	**93.0**	3.2	0.0
Street	0.4	3.8	0.0	3.8	**92.0**	0.0
WGN	0.2	0.0	0.0	0.0	0.0	**99.8**

[8]Using features ranking techniques (principal component analysis and Fisher test) [171].

Table 5.14 Test results using the QGC with different feature sets (noise-and-speech)

Feature Set	Accuracy Rate %
9 DLSFs	85.2
10 LSFs	91.8
10 best LSFs and DLSFs	93.0

5.7.5 Classification of Human Speech-Like Noise

Human speech-like noise (HSLN) is a kind of babble noise generated by superimposing independent speech signals. HSLN of various number of superpositions (N) (1, 2, 4, ..., 1024, 4096) were used in [180] to investigate perceptual discrimination of speech from noise. For low number of superpositions (below 10), the resulting signal is perceived as speech-like. For N between 10 and 200 superpositions, the noise is perceived as speech-babble. As N increases further, the noise starts to sound like stationary Gaussian noise. We have used this set of signals (150 frames each) to test our noise-and-speech Gaussian classifier. The classification results are shown in Table 5.15. For $N = 1$, the signal is classified as speech. On the other hand, for 128 superpositions, the HSLN signal is classified 94% as babble noise, and 5.3% as speech. As N increases, the HSLN signal is classified more as babble than speech. It is worth noting that the speech we used in the training of the Gaussian classifier was from an English speech database while the HSLN speech signals are from a Japanese speech database. These classification results clearly illustrate the robustness of the Gaussian classifier.

5.7.6 Noise Classification: Application in a Variable Rate Speech Coder

To test noise classification in a practical system, we selected the Enhanced Variable Rate Codec (EVRC) of CDMA systems. EVRC is a variable rate speech coder with 3 coding rates. The higher rates (8.5 and 4.0 kbps) are used for coding speech segments while the lowest rate (800 bps) is for coding silence and background noise [46]. For noise frames, a total of 16 bits are used to code the frame. Table 5.16 shows the distribution of the 16 bits for each noise frame. The LP residual waveform is not encoded.

Table 5.15 Classification of HSLN signals: QGC (noise-and-speech)

N	Speech %	Babble %	Car %	Factory %	Street %	WGN %
1	100.0	0.0	0.0	0.0	0.0	0.0
2	95.4	4.6	0.0	0.0	0.0	0.0
4	82.0	17.3	0.0	0.0	0.7	0.0
8	76.2	19.2	0.0	0.0	4.6	0.0
16	51.0	45.0	0.0	0.0	4.0	0.0
32	30.0	68.0	0.0	0.0	2.0	0.0
128	5.3	94.0	0.0	0.0	0.7	0.0
512	0.7	96.0	0.0	0.0	3.3	0.0
1024	1.3	93.3	0.0	0.0	5.4	0.0
4096	0.7	93.4	0.0	0.0	5.9	0.0

Table 5.16 Bit allocation for a 20 ms noise frame of the EVRC coder

Parameter	EVRC
LSF coefficients	8
Residual energy	8
Residual waveform	-
Total	16

EVRC uses a 10^{th}-order linear predictor to represent each frame spectral envelope and quantize the 10 LSFs using split-vector quantization techniques. We have used the 10 unquantized LSFs from this coder to train a 5-class noise Gaussian classifier. Using new test data we evaluated the accuracy and robustness of the new classifier. In this experiment, we used the same 5 noises that we used in previous results: babble, car, factory, street, and WGN. The accuracy rate of the EVRC noise classifier was 88.8%, which is 6% lower than what we reported in Section 5.7.1. This reduction in accuracy can be attributed to the different pre-processing operations that precede the computation of the LSFs, and the different training data used.

One of the issues that we need to examine is the effect of noise suppression (NS) on the performance of the noise classifier. EVRC and most VBR speech coders use noise reduction scheme to enhance the operation of the coders in noisy environments. In collecting the training data for the EVRC noise classifier, we turned off the NS. This is to maintain

the character of each noise type without the influence of the NS unit. Table 5.17 shows an improvement of around 3% on the classification accuracy when we tested the classifier with LSFs calculated when the noise suppression was on. Thus, noise suppression helps classification.

Table 5.17 The effect of noise suppression on the accuracy rate of noise classification

Condition	Accuracy %
NS is on	88.8
NS is off	85.3

In the EVRC encoder, a highpass filter (with an 120 Hz cutoff frequency) precedes the noise suppressor. In all the results we reported above for EVRC, the highpass filter was on. When we turned it off, an increase of about 4.3% in classification accuracy was gained as shown in Table 5.18. This filtering operation eliminates important lowpass frequency information that is a key to discriminate between lowpass noises such as car noise from the other classes.

Table 5.18 The effect of pre-processing on the accuracy rate of noise classification

Condition	Accuracy %
With high-pass filter	88.8
Without high-pass filter	93.1

Table 5.19 Classification matrix (EVRC unquantized LSFs)

	Babble %	Car %	Street %	Other %
Babble	**99.4**	0.2	0.0	0.4
Car	1.5	**97.9**	0.1	0.5
Street	2.9	0.2	**95.9**	1.0
Others	8.3	0.4	0.9	**90.4**

As we have a small number of bits to encode background noise, we will just define 4 noise classes (babble, car, street, and others). The first 3 noises are commonly encountered noises in mobile environments. Other noises in the acoustic environment will be classified as "others". Using these 4 classes, we can observe that the accuracy rate is 7% higher than what we got from the 5-noise case. The classification matrix is shown Table 5.19. The results are much better for the 3 major noises than what we reported earlier in Table 5.7. These extra 2 bits for noise classification can fit easily within the 16-bit budget of the EVRC noise coding mode by using 6 bits for gain quantization.

The effect of LSF quantization on the accuracy rate of noise classification is shown in Table 5.20. Using a noise classifier trained using the EVRC unquantized LSFs and tested with quantized LSFs, around 3.7% drop in accuracy was measured.

Table 5.20 The effect of LSF quantization on the accuracy rate of noise classification

Condition	Accuracy %
Unquantized LSFs	88.8
Quantized LSFs	85.1

5.8 Noise Mixture Classification

In most noisy acoustic environments, a background noise is a mixture of sound events from different sound-producing objects such as people, engines, nature, and machines. It is realistic to assume the model shown in Figure 5.13 when dealing with noise signals in mobile environments.

In the previous sections, we assumed the input noise signal is associated with one noise class whether this is a noise environment (car, street, factory) or a single noise source (such as engine, people, car etc.). For this case, hard-decision classification was considered. Noise mixtures classification is more difficult and is a different problem.

Recently several researchers have addressed this problem using different approaches. Couvreur [150] [181] formulated noise mixture classification as a mixture decomposition problem. He assumed stationarity of noise spectral representation. Modelling each noise source as an autoregressive moving average process, noise mixture recognition becomes

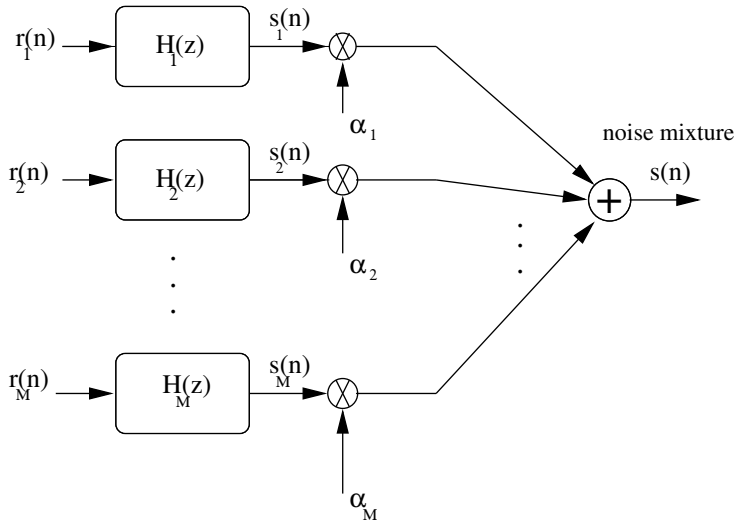

Fig. 5.13 Noise mixture model.

a problem of decomposition of linear mixtures of Gaussian processes. Using minimum description length criterion, he proposed a numerical solution to estimate the contribution of each active noise source. In his study, a comparison between the machine classifier and human recognition of noise mixtures revealed that the classifier generally outperformed human listeners for the identification of short 1-second mixture of noises.

Another study of the recognition of acoustic noise mixtures was proposed in [182]. Spectral pattern recognition and knowledge-based prediction of sound models were used for mixture classification. The authors reported that it is not easy to recognize a sound mixture when its components occupy common frequency bands, and one component is significantly louder than the other components. They concluded that spectral pattern recognition is not sufficient alone for robust recognition of sound mixtures.

In our work we have followed a different approach to the noise-mixture classification. As our goal is to estimate the relative contribution of each noise class present in the noise mixture, we have used soft-decision classification techniques to correlate the relative energy of each noise source with its probability of presence.

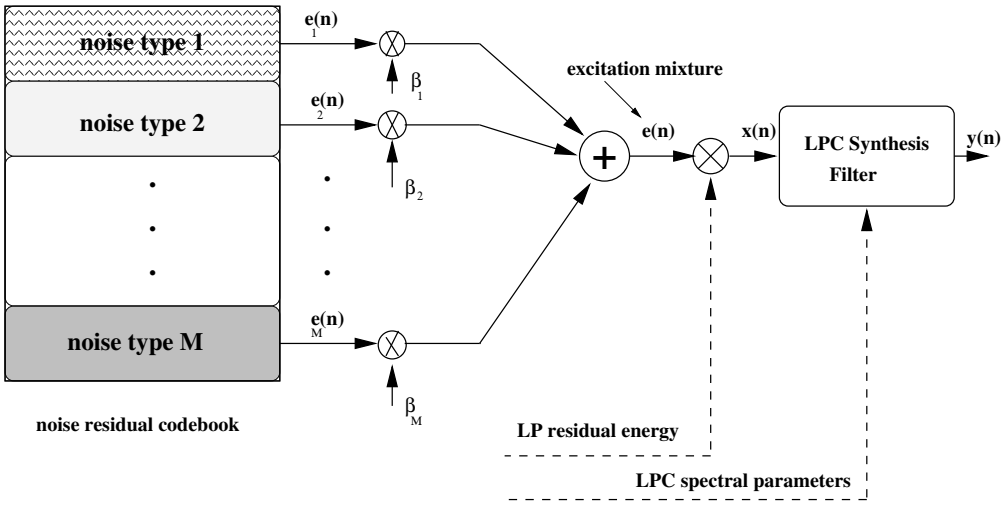

Fig. 5.14 Residual mixture substitution.

5.9 Residual Mixture Substitution

We have introduced in Chapter 4, the residual mixture model as a general model for our proposed class-dependent residual substitution technique. In this section we address the major issue of this mixture model which is the estimation of the mixture weights. As shown in Figure 5.14, the LP excitation is modelled as a linear mixture of M excitation signals from the M noise classes, given as:

$$e(n) = \sum_{i=1}^{M} \beta_i e_i(n), \tag{5.23}$$

where $e_i(n)$ is an excitation signal from the i^{th} noise class, and β_i is the i^{th} mixing coefficient, taking a value between 0 and 1, with $\sum_{i=1}^{M} \beta_i = 1$. The mixing coefficients quantify the contribution of the excitation of each noise class to the excitation mixture.

Estimating the weighting coefficient of each contributing signal in a mixture of signals is a classical signal processing problem. For example, given a noisy speech signal it is desirable to estimate the signal-to-noise ratio which is in essence the ratio of the signal power to the noise power. Also, in source separation problems, the objective is not only to estimate the mixing weights of each signal in a mixture but to estimate the waveform of each signal component [183].

For our application, we do not use computationally intensive algorithms for estimating

the mixing weights due to the limited resources in mobile devices. We have approached this mixture-weights estimation problem using a *probabilistic* solution. Instead of estimating the energy contribution of each noise source to the mixture we estimate the probability that this source exists in the mixture. Thus, for each frame of the input noise we estimate M *a posteriori* probabilities, $\{\alpha_i\}_{i=1}^{M}$. These M probability values will be mapped to M mixture weights using a novel scheme that will be proposed in this section.

Several techniques can be used to estimate the *a posteriori* probabilities of a classification system. In our work, soft-decision classification is the main tool that has been used for this task. We have experimented with different soft-decision methods and in the next section we will elaborate more in two of the techniques that we have used.

5.9.1 Estimation of the Mixing Weights

We show in Figure 5.15 the set-up for computing an estimate of the probabilities of M noise classes using a generic soft-decision classifier at the encoder side. The classifier outputs M values that are *soft labels* or *decisions* that reflect the degree of membership of the input feature vector to each noise class. For example, for the case of 4 noise classes, the output of the classifier will be $\underline{\alpha} = [\alpha_1, \alpha_2, \alpha_3, \alpha_4]$. These 4 values should sum to 1 and each takes values between 0 and 1.

Most classifiers that output a single class label for each input feature vector (i.e., hard-decision) employ a final operation that favors that class over the other classes. Examples of such operations include $max(.)$, and $min(.)$. The maximum likelihood (ML), and the maximum *a posteriori* (MAP) classifiers are known examples. If we remove this final selection stage we can transform a hard-decision classifier into a soft-decision classifier.

5.9.2 Encoding and Transmission of the Mixing Weights

Given a vector of soft labels $\underline{\alpha}$, we need to transmit this information to the receiver. One way is to use vector quantization (VQ) techniques to transmit this M-dimensional vector. This requires collecting training data for all possible combinations of soft labels and to design a codebook of size N and dimension M. For example, for the 4-class soft classifier the output vector can be $[0.5, 0.3, 0.1, 0.1]$. Using the trained codebook, the quantization process will search for the nearest vector in the N stored vectors. The encoder will send the index of the selected vector to the decoder, at which the mixing weights will be the

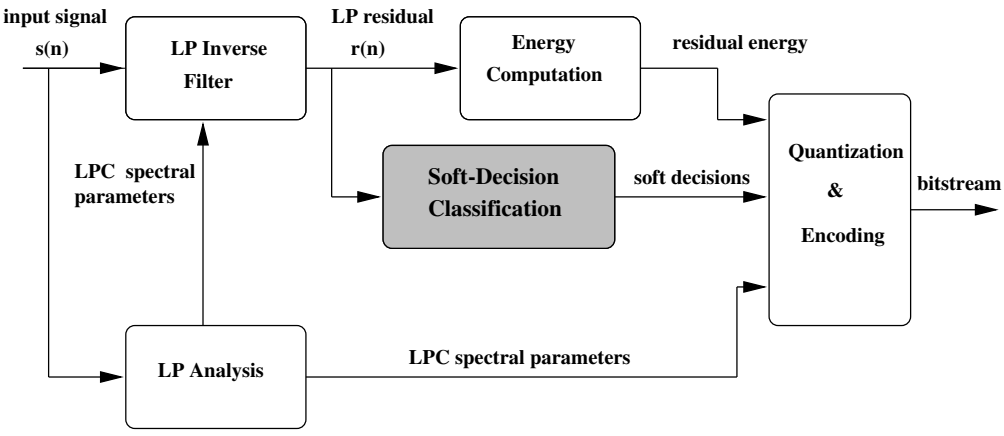

Fig. 5.15 Soft-decision classification at the encoder.

selected codevector from the codebook .

We propose in this work an alternative scheme to compute the mixture weights using the soft-decision values without the need of training VQ codebook. Assuming that we have a vector $\underline{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_M]$ of M soft labels, we want to transmit this information to the receiver to help the decoder use an *appropriate* mixing weights vector $\underline{\beta} = [\beta_1, \beta_2, \ldots, \beta_M]$. The steps of the algorithm are outlined below:

1. Rank the M soft labels in descending order,
2. If we want to transmit only $M/2$ indices, we select the indices of the $M/2$ largest values,
3. Send the $M/2$ indices sequentially in the following order: i_1 then $i_2, ..., i_{M/2}$.

Let us give an illustrative example from a 4-class classifier. We want to send only 2 indices that is 4 bits for each input frame. Given $\underline{\alpha} = [\alpha_1, \alpha_2, \alpha_3, \alpha_4]$, we find $i_1 = \arg\max_{i=1,2,3,4} \alpha_i$ and $i_2 = \arg\max_{i=1,2,3,4; i \neq i_1} \alpha_i$. The transmitted indices will be the pair (i_1, i_2), with i_1 transmitted first.

For the case of 4 noise sources and 2 transmitted indices there are 12 cases of ordered pairs as shown in Table 5.21. We show in the table examples of mixing weights that the decoder can use if the mixture excitation will be formed only from the 2 dominant noise sources.

Using the same idea of sending only 2 indices, we can still create mixture excitation from all the available noise sources. By sending 2 indices, the receiver knows the remaining 2

Table 5.21 Mixture weighting matrix: 2 dominant noise sources

Class Pair	β_1	β_2	β_3	β_4
(1,2)	0.75	0.25	0.0	0.0
(1,3)	0.75	0.0	0.25	0.0
(1,4)	0.75	0.0	0.0	0.25
(2,1)	0.25	0.75	0.0	0.0
(3,1)	0.25	0.0	0.75	0.0
(4,1)	0.25	0.0	0.0	0.75
(2,3)	0.0	0.75	0.25	0.0
(3,2)	0.0	0.25	0.75	0.0
(2,4)	0.0	0.75	0.0	0.25
(4,2)	0.0	0.25	0.0	0.75
(3,4)	0.0	0.0	0.75	0.25
(4,3)	0.0	0.0	0.25	0.75

classes that are weak or not present in the mixture. For these weak noise classes a minimal contribution from each source can be used as shown in Table 5.22, which is a more general form of Table 5.21. Other combinations of the mixing weights are possible for this case.

Table 5.22 Mixture weighting matrix: all 4 noise sources

Class Pair	β_1	β_2	β_3	β_4
(1,2)	0.65	0.25	0.05	0.05
(1,3)	0.65	0.05	0.25	0.05
(1,4)	0.65	0.05	0.05	0.25
(2,1)	0.25	0.65	0.05	0.05
(3,1)	0.25	0.05	0.65	0.05
(4,1)	0.25	0.05	0.05	0.65
(2,3)	0.05	0.65	0.25	0.05
(3,2)	0.05	0.25	0.65	0.05
(2,4)	0.05	0.65	0.05	0.25
(4,2)	0.05	0.25	0.05	0.65
(3,4)	0.05	0.05	0.65	0.25
(4,3)	0.05	0.05	0.25	0.65

One way to improve the proposed model is to do the following. If the maximum degree of membership is greater than 0.50 for one noise class, then the transmitter assumes it is a

single noise source case and will send this class index twice (i.e., (i_1, i_1)). If the indices are the same, then the LP excitation will be formed from stored excitation of this class only. If the indices are different, the multiple-source mixing procedure is used.

If the bit budget of the noise coding mode has enough bits to transmit the 4 indices (or generally the M indices), then a fixed weighting vector can be used depending on the order of the received indices. For example, for each frame a sequence of 4 indices arrived as i_1, i_2, i_3, and i_4. The first index is always the index with the largest membership value and then the next index etc. For example, the mixing weight vector can be always $[0.55, 0.25, 0.1, 0.1]$, where 0.55 is used as the mixing coefficient for the i_1 noise class, and 0.25 is used for the i_2 noise class etc. Other weight vectors can be used.

5.10 Soft-Decision Noise Classification

In Section 5.5.2 we have defined the Gaussian classifier and we have shown that its decision rule is one example of minimum distance classifiers. It is characterized by a set of M distinct discriminant functions, that is a function of the input feature vector and the mean and covariance of each noise class.

To derive a soft-decision version of the Gaussian classifier (soft-QGC) we rewrite Eq. 5.20 here as:

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \mu_i)^T \Sigma_i^{-1}(\mathbf{x} - \mu_i) - \frac{1}{2}\ln|\Sigma_i| + 2\ln P(w_i), \qquad i = 1, 2, \ldots, M. \tag{5.24}$$

The above decision functions for a given feature vector $\mathbf{x}$ are not qualified to be used directly as soft labels as they do not sum to 1 and they are not between 0 and 1. Thus we need to normalize each g_i to describe them as soft labels. In Eq. 5.25 we present one transformation that changes the Gaussian distance values into soft labels.

$$\alpha_i = \frac{g_i^{-2}}{\sum_{j=1}^{M} g_j^{-2}}, \qquad i = 1, \ldots, M, \tag{5.25}$$

where α_i is the membership value of the i^{th} noise class, and g_i is the distance measure for this class.

In essence, this soft-QGC is the same as the Gaussian classifier used in previous sections

(same means, and same covariances) but the main difference is in the form of the output. For each input feature vector of 10 LSFs and 5 noise classes, the classifier outputs 5 soft labels. For each class α_i represents the degree of membership of the input vector to this class. We present in Table 5.23 the average membership of each noise class obtained from the soft-QGC tested on the same 2500 noise test vectors. For example, for car noise, on average, LSF vectors from car noise 90% belong to the 'car' class, 3.9% belong to the 'babble' class, and with a weaker degree of membership to the other 2 classes. On the other hand, for street noise, the average membership is only 56.4% to the 'street' class and with higher membership values to both 'babble' and 'factory' noise classes. These results match our previous observation that these 3 noises classes overlap in the LSF feature space more than car and WGN.

Table 5.23 Membership classification matrix

	Babble	Car	Factory	Street	WGN
Babble	0.527	0.011	0.198	0.259	0.005
Car	0.039	0.899	0.044	0.016	0.002
Factory	0.192	0.028	0.653	0.125	0.002
Street	0.261	0.004	0.168	0.564	0.003
WGN	0.019	0.001	0.009	0.014	0.957

Soft-decision classification is more useful when dealing with new noises that are not one of the pre-defined noise classes. This is especially true if we use soft decisions to select the excitations from the stored noise residual codebook. For instance, for bus noise, if we use a hard-decision Gaussian classifier with 'bus' as not one of its classes, it will be forced to choose the closest class (say babble or street). Thus, the LP excitation of bus noise will be always a sequence of excitation samples from the 5 noises of the classifier. Using the soft decisions, for each frame, we can create a mixture of excitation samples with different contributions from the other noise classes. Thus, soft classification allows a richer generation of excitation signals and will not be limited to the size of the residual codebook. From Table 5.24, on average, the LP excitation of bus noise will have 40% contribution from babble, 30% from street, and around 28% from factory noise excitations.

The membership matrix for the noise-and-speech classification is shown in Table 5.25. Speech LSFs have 96% membership to its own class. This shows that speech has distinct LSFs than the other noises. WGN LSFs have around 11% membership to the speech class

Table 5.24 Membership classification matrix of new noises

Noise	Babble	Car	Factory	Street	WGN
Bus	0.400	0.017	0.277	0.304	0.002
Restaurant	0.552	0.015	0.193	0.230	0.010
Shop. Mall	0.441	0.003	0.083	0.442	0.031
Sports	0.392	0.013	0.144	0.435	0.016
Subway	0.440	0.006	0.182	0.363	0.009
Traffic	0.353	0.004	0.096	0.521	0.026

Table 5.25 Membership classification matrix: (noise-and-speech)

	Speech	Babble	Car	Factory	Street	WGN
Speech	0.960	0.021	0.001	0.005	0.010	0.003
Babble	0.225	0.404	0.008	0.155	0.204	0.004
Car	0.093	0.034	0.820	0.038	0.014	0.001
Factory	0.131	0.166	0.023	0.571	0.107	0.002
Street	0.132	0.225	0.003	0.146	0.492	0.002
WGN	0.113	0.017	0.001	0.007	0.012	0.850

as some speech LSFs can represent a flat spectral envelope. Other noises have different degrees of membership to the other classes.

One of the advantages of using soft decision classification is that it allows a "no-decision" or a reject option. For an input feature vector that has membership equally distributed between the classes, the classifier declares a no-decision. Depending on the application, a no-decision output might not be acceptable and it has to be replaced by another class label. For example, the decision of the previous frame can be used. Soft-decision classification can also provide a mechanism to ensure that the selected class is correct with a high probability, and also to correct a 'possible' error. To achieve this, the maximum membership is compared against an *ambiguity threshold*. If this value does not exceed the threshold then the classifier assumes no decision and outputs the class of the previous frame. We present in Figure 5.16 a soft-decision classifier with a reject option. The accuracy rates using different values of the ambiguity threshold are shown in Table 5.26. In all cases we show that we gain by using the ambiguity threshold to correct decisions for frames that have weak membership to all classes. We found that a threshold value of 0.45 gives the best

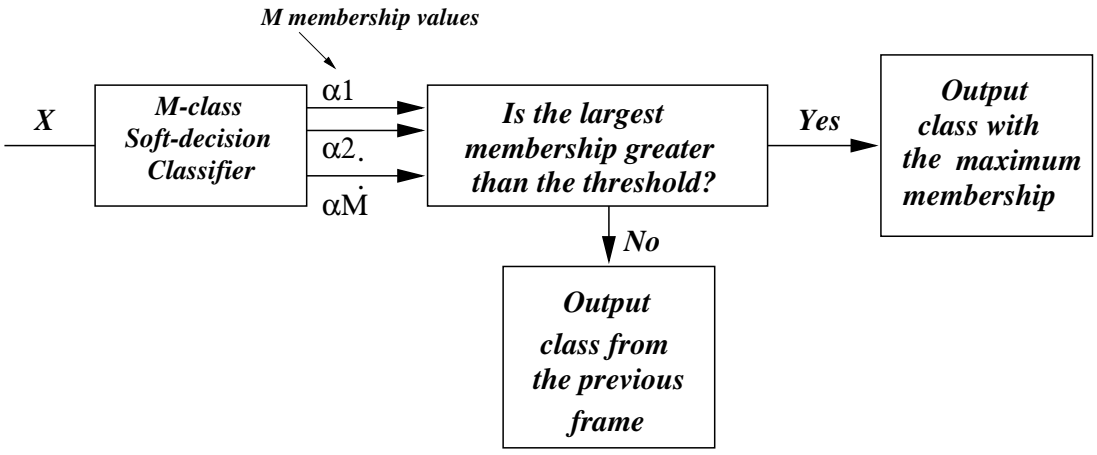

Fig. 5.16 A soft-decision classifier with a reject option.

performance with an accuracy of 94.7%. Increasing the threshold above 0.45 reduces the efficiency of the proposed technique. Using the previous frame decision seems suitable in our case due to the high correlation between consecutive frames as they come from the same noise source. Other ideas can be used to replace the no-decision output of a current frame. For example, we can use a replacement decision that is defined as the majority decision of the last few frames. This dose not require any extra delay.

Table 5.26 Test results using the QGC with different ambiguity thresholds

Threshold	Accuracy Rate %
None	91.8
0.30	91.9
0.35	93.1
0.40	94.1
0.45	94.7
0.50	92.6

5.11 Fuzzy Classification

Another important category of soft-decision classification is based on the theory of fuzzy sets. In fuzzy classification, each input feature vector is given a *membership grade* for each

class. Membership grades range in value from 0 to 1, and provide a measure of the degree to which an input feature vector belongs to or resembles the specified class. In this work, we have used fuzzy clustering to explore any embedded similarities between the LSFs of the 5 noises and speech. In this section we will first present the *fuzzy c-means clustering* (FCM) algorithm and then report the results of applying this algorithm to noise classification and clustering. In Section 5.11.2 we present the *centroid classifier*: a new simple, yet efficient noise classifier using the FCM algorithm.

5.11.1 The Fuzzy c-Means Clustering Algorithm

Clustering, also known as *unsupervised learning* or *self-organizing*, is a process of finding natural structure within training data. The similar data samples are grouped together into *clusters* or classes. If it is desirable to separate the classes into disjoint groups then traditional clustering techniques can be used such as the well-known c-means clustering algorithm. Otherwise, fuzzy clustering methods can be used if there are indications that there are overlaps between the classes. FCM is an iterative data clustering technique that was originally introduced by Bezdek in 1981 [184] [185]. Given a training data (collected from various sources and without any class labelling), the algorithm returns c centroid vectors, one centroid for each cluster. The number of the clusters c is specified by the user before running the algorithm. Starting from an initial guess of the c centroids, the iterative process enhances the estimation of centroids by minimizing an objective function such as the Euclidean distance.

In addition to the c centroid vectors, the FCM algorithm outputs a membership grade vector (of dimension c) for each vector in the training data. This information can be used to build fuzzy classification rules or to gain insight on the nature of the training data.

We have applied the FCM algorithm[9] to examine if there exist any natural grouping of the LSF vectors of the 5 noises (car, babble, street, factory, and WGN). Using the same noise training data that we used in this work, the FCM algorithm grouped the LSF data into 2, 3, 4, or 5 clusters. For each noise type, we calculated the average membership vector corresponding to each centroid. We show in Tables 5.27–5.29 the noise-centroid vectors for each case, and in Tables 5.30–5.32 the average membership values for the 5 noises[10]. It can

[9]The Matlab *fcm* function was used to generate the results.

[10]We do not show the results for the case of 5 clusters as they are similar to the 4-cluster case.

Table 5.27 LSFs fuzzy clustering results: 2 clusters case (LSFs are in radians)

	LSF1	LSF2	LSF3	LSF4	LSF5	LSF6	LSF7	LSF8	LSF9	LSF10
c1	0.2822	0.5589	0.8455	1.1302	1.4168	1.7009	1.9898	2.2736	2.5648	2.8472
c2	0.1531	0.3418	0.6138	0.8997	1.1831	1.4641	1.7972	2.0881	2.4331	2.7371

Table 5.28 LSFs fuzzy clustering results: 3 clusters case (LSFs are in radians)

	LSF1	LSF2	LSF3	LSF4	LSF5	LSF6	LSF7	LSF8	LSF9	LSF10
c1	0.2860	0.5669	0.8543	1.1384	1.4253	1.7097	1.9968	2.2809	2.5698	2.8517
c2	0.0698	0.2165	0.5528	0.8420	1.1458	1.4305	1.7843	2.0956	2.4223	2.7475
c3	0.1989	0.4025	0.6483	0.9301	1.2024	1.4807	1.8046	2.0852	2.4376	2.7296

be observed that increasing the number of centroids to more than 3 did not return extra new LSF centroids.

5.11.2 The Centroid Classifier

Using the training data of each noise type and the FCM algorithm, we get one prototype LSF vector for each class. These 5 prototype LSF vectors will be called hereafter the noise *centroid* filters. The spectral envelopes of car and babble noises are shown in Figure 5.17, and for factory and street noises in Figure 5.18. Similar to our observation throughout this study, car and factory noises are lowpass signals, while babble and street noises have higher frequencies content. We show in Figure 5.19 the centroid filters for speech and WGN. Speech signals have a much larger variation in their LSF distribution. This makes a single centroid filter not a good model of the speech LSF training data (for example, unvoiced speech is highpass in nature). The LSFs of a WGN signal (or any signal with a flat spectrum) are uniformly distributed in the unit circle. For an N^{th}-order LP filter, the WGN LSFs ω_i are defined as:

$$\omega_i = i\,\frac{\pi}{(N+1)}, \qquad i = 1, 2, \ldots, N. \tag{5.26}$$

Table 5.29 LSFs fuzzy clustering results: 4 clusters case (LSFs are in radians)

	LSF1	LSF2	LSF3	LSF4	LSF5	LSF6	LSF7	LSF8	LSF9	LSF10
c1	0.2869	0.5687	0.8566	1.1405	1.4274	1.7119	1.9986	2.2829	2.5710	2.8529
c2	0.1937	0.3984	0.6440	0.9280	1.2019	1.4809	1.8052	2.0861	2.4387	2.7322
c3	0.0556	0.1914	0.5425	0.8312	1.1380	1.4231	1.7812	2.0969	2.4192	2.7478
c4	0.1934	0.3983	0.6438	0.9279	1.2019	1.4809	1.8051	2.0862	2.4387	2.7323

Table 5.30 Average membership grade for each noise LSF training data: 2 clusters case

noise	c1	c2
Babble	0.1251	0.8749
Car	0.0892	0.9108
Factory	0.0672	0.9328
Street	0.1257	0.8743
WGN	0.9708	0.0292

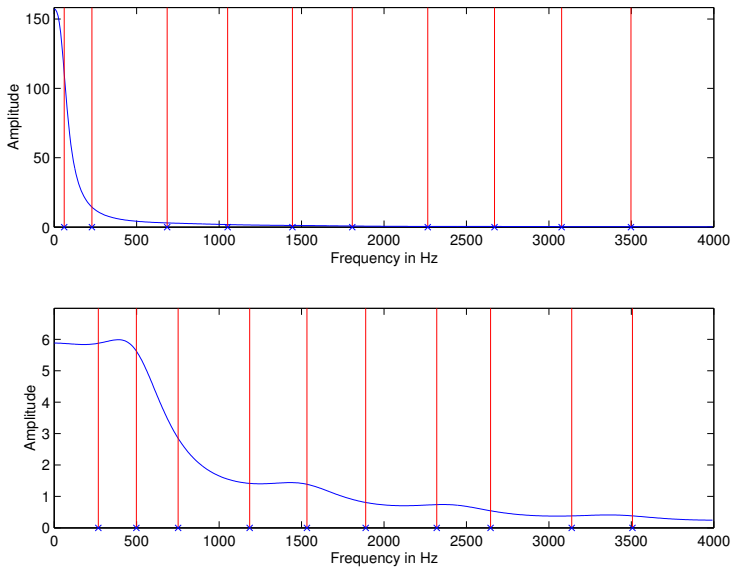

Fig. 5.17 Spectral envelope of the centroid filters: (a) car noise, (b) babble noise.

Table 5.31 Average membership grade for each noise LSF training data: 3 clusters case

noise	c1	c2	c3
Babble	0.0768	0.2601	0.6631
Car	0.0183	0.8736	0.1081
Factory	0.0478	0.2563	0.6959
Street	0.0585	0.1396	0.8018
WGN	0.9488	0.0181	0.0331

Table 5.32 Average membership grade for each noise LSF training data: 4 clusters case

noise	c1	c2	c3	c4
Babble	0.0471	0.4080	0.1369	0.4081
Car	0.0140	0.0888	0.8081	0.0890
Factory	0.0281	0.4239	0.1231	0.4249
Street	0.0351	0.4479	0.0704	0.4466
WGN	0.9208	0.0315	0.0162	0.0315

Using these 5 centroid LSF vectors as prototypes we define a special case of the prototype nearest-neighbor classifier that we call the *centroid classifier*. The 5 centroid LSF vectors for the 5 noise classes are shown in Table 5.33. It is obvious that the first LSF is the key parameter in distinguishing the different types of noises. In the higher portion of the spectrum, the higher LSFs have similar values.

Using the same noise test data (2500 frames) that we used to test the Gaussian classifier, we evaluated the accuracy of the centroid classifier. An accuracy of around 91% was obtained using this simple classifier compared to a 95% with the Gaussian classifier. Table 5.34 shows the detailed classification result for each noise class. These results are similar to the classification matrix of the Gaussian classifier (Table 5.7) except for babble and street noises. This degradation is expected as babble and street have more spectral variations over time. Using more than one prototype for such noises can help to reduce the decision errors.

If we compare the storage requirements of the centroid classifier and the Gaussian clas-

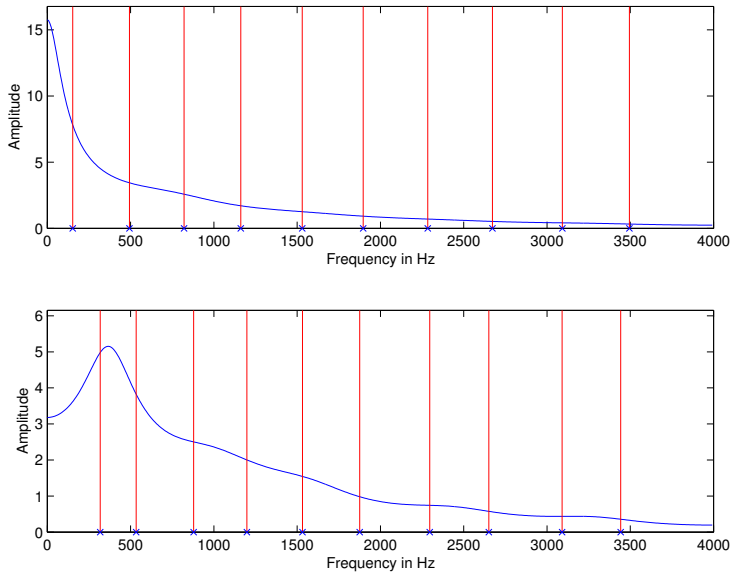

Fig. 5.18 Spectral envelope of the centroid filters: (a) factory noise, (b) street noise.

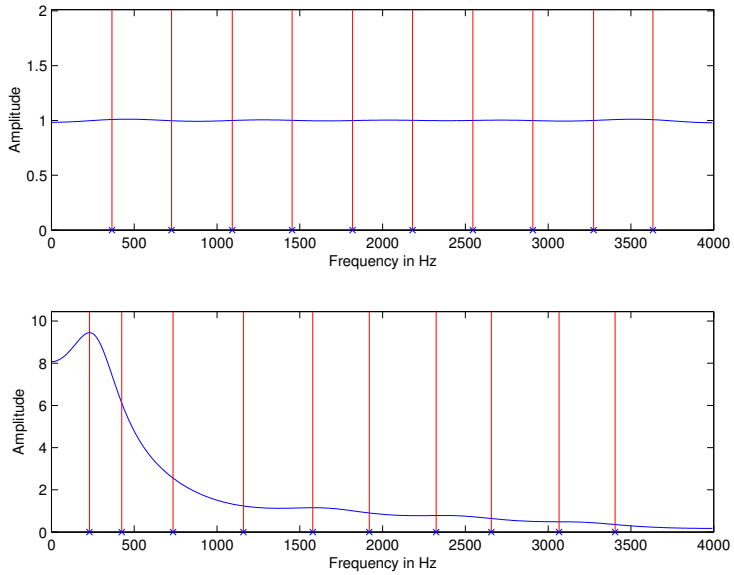

Fig. 5.19 Spectral envelope of the centroid filters: (a) WGN, (b) speech.

sifier, we find that the former requires much less memory. For a 5-class Gaussian classifier and a 10-dimensional feature vector, we need to store 5 mean vectors (10 values each) and

Table 5.33 LSFs (in Hz) of the noise centroid filters

Noise	LSF1	LSF2	LSF3	LSF4	LSF5	LSF6	LSF7	LSF8	LSF9	LSF10
Car	61.1	228.9	685.4	1052.1	1444.7	1807.7	2265.7	2670.5	3077.4	3497.6
Factory	152.8	493.6	821.2	1162.0	1530.3	1896.6	2284.2	2672.1	3092.4	3494.5
Babble	267.9	498.2	751.3	1185.4	1532.6	1889.4	2320.1	2646.4	3138.3	3506.4
Street	318.1	533.9	878.7	1198.4	1532.0	1875.7	2296.0	2650.8	3090.4	3442.7
WGN	363.6	727.3	1090.9	1454.5	1818.2	2181.8	2544.5	2909.1	3272.7	3636.4

5 covariance matrices (each with 55 elements[11]). Thus, the total storage requirement will be ($55 \times 5 + 5 \times 10 = 325$ values). However, for the centroid classifier we need only to store 50 values. This is a memory saving of 85% and with a slight drop in performance.

Table 5.34 Classification matrix: Centroid NN (noise-only)

	Babble %	Car %	Factory %	Street %	WGN %
Babble	**72.8**	0.4	9.0	17.8	0.0
Car	0.0	**99.8**	0.2	0.0	0.0
Factory	2.2	0.0	**96.4**	1.4	0.0
Street	10.8	0.0	3.6	**85.6**	0.0
WGN	0.0	0.0	0.0	0.0	**100.0**

The results presented in Table 5.34 were generated using a decision window (frame) of 20 ms. We show in Figure 5.20 the effect of increasing the number of decision frames on the accuracy rate for each noise class. Using the same 5 centroid LSF vectors, we calculate a score metric S_i for each noise class given by:

$$S_i = \sum_{k=1}^{L} d(x_k, \bar{x}_i), \qquad i = 1, 2, \ldots, M, \tag{5.27}$$

where x_k is the input feature vector (the 10 LSFs), $\bar{x}_i$ is the centroid vector for the i^{th} noise class, $d(x, y)$ is a distance metric (for example, the Euclidean distance), M is the number of classes, and L is the number of consecutive decision frames (decision window).

[11]The covariance matrix is symmetric and thus we need only to store the diagonal elements and the upper entries of the matrix.

The decision class label will be the one with the minimum score value over the decision window.

When L is 1, we get the same results we obtained for the frame-level classification. Using this technique, for babble and street noises we can achieve a 100% classification accuracy if we use a decision window of 25 consecutive frames (i.e., half a second for an 8 kHz sampling frequency). Less than 5 frames are needed to accurately classify car and factory noises. The centroid classifier, with its lower memory requirement and good performance, is recommended for noise classification for mobile devices and portable hearing-aids.

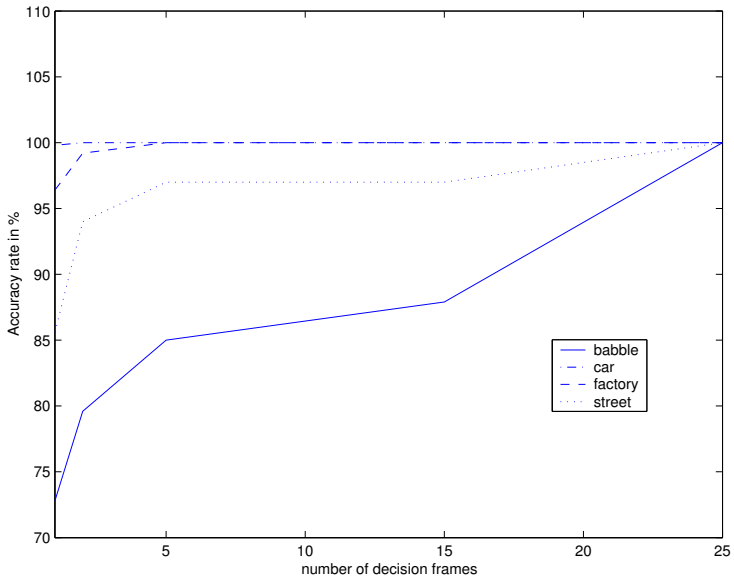

Fig. 5.20 The relationship between the number of decision frames and the accuracy rate of the centroid classifier.

In the next section we report our work in extending the use of pattern recognition techniques to the important problem of discrimination of speech from music.

5.12 Speech/Music Discrimination

A human listener can discriminate easily between speech and music signals by listening to a short segment (i.e., few seconds) of an audio signal. In recent years, different systems have been proposed for the automatic discrimination of speech signals and music signals. Saunders [186] proposed a real-time speech/music discriminator to be used in radio receivers

for the automatic monitoring of the audio content of FM radio channels. In automatic speech recognition (ASR) of broadcast news, it is important to disable the speech recognizer during the non-speech portion of the audio stream. Recently, Scheirer and Slaney [187] and Williams and Ellis [188] developed and evaluated different speech/music discrimination systems for ASR of audio sound tracks.

Another application that can benefit from distinguishing speech from music is low bit-rate audio coding. Traditionally, separate codec designs are used to digitally encode speech and music signals. Generally, speech coders do better on speech, and audio coders do better on music [189]. In many emerging multimedia applications such as the Internet, the sound stream carries both speech and music. Designing a universal coder to reproduce well both speech and music is the best approach—however, this is not a trivial problem. An alternative approach is to design a multi-mode coder that can accommodate different signals. The appropriate module is selected using the output of a speech/music classifier [190]. This approach has been already employed in the parametric coder of the Moving Picture Experts Group (MPEG)-4 standard [191] and recently in the multi-mode audio coder proposed by Ramprashad [189], and in [192] for mixed wideband speech and music coding.

An emerging multimedia application is a content-based audio and video retrieval. Sound classification is an important part of such systems. Automatic classification would remove the subjectivity inherent in the classification process and ultimately speed up the retrieval process. Zhang and Kuo [193] developed a content-based audio retrieval system that classifies audio signals as speech or music or noise. Minami *et al.* [194] proposed an audio-based approach to video indexing. A speech/music detector is used to help users to browse a video database.

5.12.1 Classification Features

Existing speech/music classification systems use long-term features such as variances and time-averages of spectral parameters [186], [187]. Tonality and pitch have also been combined into several designs [195]. Typically, these features are estimated over audio segments of 0.5–5 seconds. While these classifiers show high accuracy in distinguishing speech and music, they are not suitable for delay-sensitive applications such as interactive communications.

In our work, we have selected the LSFs as the core feature set for speech/music classification. This was motivated by our good results of using the LSFs in classifying different types of background acoustic noises and speech (Section 5.7). To enhance the performance, additional features (DSLFs, HOC, and LP-ZCR) were combined with the LSFs.

5.12.2 Evaluation Experiments

For this study we have selected two different classification algorithms: the quadratic Gaussian classifier and a nearest neighbor classifier. This selection of classification algorithms will enable us to compare our results with existing speech/music classifiers and will highlight the effect of the classification algorithm on the classification results.

The training data consisted of both music and speech audio recordings with 8 kHz sampling frequency. The speech data originated from ten speakers, five males and five females. Music was selected from various categories including *classical, instrumental, opera, rock, dance, rap and pop*. The training vectors correspond to 28 000 frames (i.e., 9.3 minutes) for speech and 32 000 frames (i.e., 10.7 minutes) for music. Additional music and speech samples were set aside for independent testing. The test music vectors were chosen from the same categories as the training set and the test speech vectors were taken from an *InGroup* which were speakers used in the training set and from an *OutGroup*, speakers who were not.

The Bayes error rate and empirical error estimation were used to gauge the performance of the features and classifiers prior to independent testing.

5.12.3 Classification Results

Four feature sets were used for experimentation: LSF, DLSF, LSF with HOC (LSF-HOC), and LSF with LP-ZCR (LSF-ZCR). Table 5.35 contains the Bayes error rate for the aforementioned feature sets and error estimations using the k-NN and the QGC.

Several observations can be made from the results in Table 5.35. First, the LSFs when used alone, have a low Bayes error rate showing that they have the potential to effectively discriminate music from speech. Second, the error rates from using a Gaussian classifier and NN classifiers demonstrate the effect of classification algorithms on the results. The QGC classifier has error rate that is around 16% more than the Bayes rate. This can be explained by our observation that the LSFs features of speech and music deviate from the Gaussianity

Table 5.35 Error estimation for the classification features

Features	Bayes Rate (%)	Error (%)		
		1-NN	3-NN	QGC
LSF	4.6	8.8	9.6	21.1
LSF-ZCR	5.9	11.2	9.9	18.0
LSF-HOC	6.7	12.6	11.4	18.7
DLSF	7.3	13.5	13.9	23.3

assumption. Combining zero-crossings features with the LSFs slightly improves the error rate for the QGC by reducing the overlapping of the feature spaces of music and speech.

The nearest neighbor classifiers have a superior holdout rate, approaching the Bayes error rate. This would indicate that they are a better choice than the Gaussian classifier. However, hold-out estimations for k-NN classifiers tend to be biased due to the large frame-to-frame correlation within the training set. Only independent testing gives a true measure of a classifier's performance.

Table 5.36 Accuracy (%) testing results for different music types (QGC)

music	LSF	DLSF	LSF-ZCR	LSF-HOC
Classical	93.6	93.2	96.2	89.5
Instrumental	79.3	80.3	92.8	90.1
Opera	77.3	76.3	73.7	53.9
Rock	72.1	69.4	87.6	84.4
Dance	68.2	59.5	86.6	83.9
Rap	60.7	54.6	80.9	77.1
Pop	57.8	57.3	84.3	82.4
Average	72.7	70.1	86.0	80.2

- **Music Test Results**

Table 5.36 shows the QGC results of independent testing on different categories of music. We can observe that the accuracy rate depends on the music type. For example, using the LSFs features alone, *Classical* music is 93.6% detected as music while *Rap* music is detected as music 60.7% of the time. This could be attributed to the speech-music content of each music type. *Classical* music tends to be devoid of any speech content while *Rap* is

dominated by rhythmic speech. Clearly, a large speech content will result in music being classified as speech. Combining additional features with the LSFs improved the decision accuracy with the troublesome categories. For instance, a gain of 20% in accuracy has been scored for the *Rap* music by combining the LP-ZCR feature with the LSFs. More than a 13% average increase in accuracy has been obtained for all types of music by using the LSF-ZCR feature set.

Table 5.37 Accuracy (%) results for music using the LSF features

music	QGC	3-NN
Classical	93.6	92.3
Instrumental	79.3	76.6
Opera	77.3	94.6
Rock	72.1	78.3
Dance	68.2	87.2
Rap	60.7	52.7
Pop	57.8	72.9
Average	72.7	79.2

In Table 5.37 we compare the music testing results from the Gaussian and the 3-NN classifiers. In general, the 3-NN has a better discrimination of music than the QGC. For example, *Opera* music was 94.6% accurately identified as music using the 3-NN, compared to 77.3% using the QGC. This shows that the estimated parameters of the Gaussian classifier are not capable of completely covering the large variations in music feature space.

- **Speech Test Results**

The results of independent testing on speech using the Gaussian classifier are shown in Table 5.38. *InGroup* speech was classified with slightly better accuracy than *OutGroup* speech. This could be attributed to the LSFs tendency to model the vocal tract of the speaker. Generally, *InGroup* speech will always have a higher probability of being classified correctly. Table 5.39 shows that the 3-NN also outperforms the QGC by about 8% in distinguishing speech using the LSFs features.

Table 5.38 Accuracy (%) testing results for speech (QGC)

speech	LSF	DLSF	LSF-ZCR	LSF-HOC
InGroup	75.6	71.8	74.0	78.3
OutGroup	72.9	68.8	70.6	73.9
Average	74.3	70.3	72.3	76.1

Table 5.39 Accuracy (%) results for speech using the LSF features

speech	QGC	3-NN
InGroup	75.6	84.3
OutGroup	72.9	80.6
Average	74.3	82.5

5.12.4 Segment-level Classification

To make a fair comparison with previous speech/music classifiers, the QGC was modified to make decisions over 50 frames (1 second). This was done by first making decisions for the individual frames. A global decision was then made for the entire block by choosing the class that appeared most frequently. By incorporating 50 frames of information into one decision, rather than one frame per decision, the accuracy of the classifier rises noticeably.

Table 5.40 Accuracy (%) results with decisions made over 50 frames (QGC)

input	LSF	DLSF	LSF-ZCR	LSF-HOC
Speech	93.8	96.8	95.2	100.0
Music	87.5	80.0	94.4	91.9
Average	90.7	88.4	94.8	95.9

As depicted in Tables 5.40 and 5.41, the performance over 50 frames compares favorably with the accuracy rates of Scheirer and Slaney's speech/music classifier [187] and the discriminator described in the MPEG-4 standard [191]. The accuracy rating for the Scheirer and Slaney classifier was obtained from their original testing using all of their 13 proposed features. The performance of the discriminator described in the MPEG-4 standard was measured through independent testing. The testing set for the LSF-based classifier was

reused for the MPEG-4 testing. The reference software provided by the MPEG committee was used to classify the testing set.

Table 5.41 Accuracy results for two other speech/music discriminators

Classifier	Accuracy (%)
MPEG-4	97.6
Scheirer and Slaney	93.2

5.13 Summary

In this chapter, we presented the design of new techniques for the classification of different types of background acoustic noises. One contribution of this work is that we have shown that we can classify noises using short signal frames (20 ms). We presented our results using different classification features and algorithms. A major application of this noise classification work was for the class-dependent residual substitution excitation models we presented in Chapter 4. At the end of the chapter we have shown that the line spectral frequencies is a potential feature set to discriminate the spectral structure of speech and music signals.

Chapter 6

Summary and Conclusions

This chapter starts by giving a summary of the work reported in this dissertation. An overview of the key points of each chapter is given. Conclusions and suggestions for further related research ideas are then discussed. Finally, our main contributions to the literature are outlined.

6.1 Summary of Our Work

The introduction chapter started by providing some background material about speech coding. The main motivation for our work is that existing schemes for background noise coding at very low bit rates (below 1 kbps) and comfort noise generators fail to regenerate background noise with natural quality during speech inactivity. The change in the character of the noise during speech activity and speech pauses is noticeable and can be annoying. This results in a degradation of the perceived quality of voice. It was stated that the main objective of this work is to enhance the perceived quality of voice communications in noisy environments.

We started Chapter 2 by reviewing models characterizing the on-off patterns of conversational speech. A review of statistical modelling of talkspurts and silence durations was presented. A good portion of the chapter was devoted to review and study various schemes for voice activity detection. Finally, we discussed the results of our comparative performance study of two recently-standardized VAD algorithms under various noise conditions.

In this dissertation, we have used the linear prediction (LP) signal modelling framework

for coding and classification of background noise. Chapter 3 started by discussing some of the basic concepts and equations of LP analysis and synthesis. A special attention was given to the excitation modelling of the noise LP residual. To study the Gaussianity and whiteness properties of the LP residual of background noise, different statistical tools (kurtosis, higher-order crossings, the quantile-quantile plot, and spectral flatness measure) were used. The tests confirm that the LP residuals of car and babble noises follow Gaussian distribution with flat spectra using a 10^{th} order LP filter. However, as the conventional white Gaussian excitation model fails to replace the LP residual of structured noises, we reported our results of using improved excitation models. A low-bit-rate spectral excitation model has been proposed to improve the coding of noise LP residuals. A study of the LP residual spectrum has revealed that Fourier phase of the LP residual carries important perceptual information that is essential for natural-quality synthesis of background noises.

In Chapter 4, we presented our new class-dependent residual substitution model and we have shown that it can faithfully synthesize background noise at very low bit rates. Both single-source and mixture excitation models were proposed and various design issues were addressed. The various options for the design of the noise residual codebook were discussed. The results of concept-validation experiments of the new scheme were discussed.

In Chapter 5, a detailed study of the noise classification problem was presented. One contribution of this work is that we have shown that we can classify background noise using short signal frames (20 ms). We have presented our results using different classification features and algorithms. The quadratic Gaussian classifier was shown to outperform other classification methods we tested. Noise classification was used to select the type of excitation source in the residual substitution excitation model. Novel methods have been proposed for an efficient implementation of the mixture residual substitution model. Soft-decision classification techniques were applied to estimate the mixture weights of the excitation model. In Section 5.11.2 the fuzzy c-means clustering algorithm was used to design the centroid classifier. At the end of the chapter we have shown that the line spectral frequencies are a robust feature set for discrimination of speech from music.

6.2 Conclusions

This thesis has examined a number of different approaches to coding background noise. For many applications, a simple Gaussian Quadratic classifier based on the LSFs is quite

adequate for the task. Any isolated misclassifications mean that for one frame, another noise class is used. These single substitutions are generally imperceptible in terms of the noise texture generated. The mixture arrangement has advantages over pure classification in that it is able to accommodate new types of background noise better. We have also outlined some receiver only schemes that can benefit the generation of comfort noise, and which do not require any modifications to the transmitter or the bit stream.

6.3 Future Work

We summarize below few suggestions for future work related to the topic of this dissertation.

It has been mentioned in Chapter 4 that the storage of the noise residual codebook is a major implementation cost of the new class-dependent residual substitution model. One way to minimize this requirement is to design "engines" that can generate excitation signals that match the statistical and perceptual properties of each noise class. For example, to capture the long-term phase information of noise LP residual we need to develop random number generators that can match the higher-order statistics of each noise signal. By doing this, the noise residual codebook can be replaced with *class-dependent excitation engines* that do not require memory storage.

Recently, new methods have been proposed to generate sound textures using wavelets and statistical learning techniques. Dubnov *et al.* [196] have proposed a wavelet tree learning method to synthesize new random instances of a sound texture given an example of such texture as input. Another related work is the proposal of wavelet-based models for the real-time synthesis of perceptually convincing environmental sounds [197]. These methods are worth investigation as means to generate class-dependent excitation vectors at the receiver.

Another important issue to the realization of the noise mixture excitation model is the estimation of the mixture weights. We proposed in Section 5.9 a *probabilistic* approach for estimating the mixing coefficients. Other estimation techniques can also be investigated using ideas from blind signal separation algorithms [198].

In our work, we have modelled the LSFs of noise, speech and music using a single Gaussian cluster (PDF) for each class. The classifiers can (possibly) be improved by using Gaussian Mixture Models (GMM) for each class to better match the PDFs.

In recent years, a strong interest has emerged to adopt and develop wideband speech

coders for wireless voice communication. For example, the Third Generation Partner Project (3GPP) has standardized the Wideband Adaptive Multirate (AMR-WB) [199] speech coder for GSM and WCDMA systems and 3GPP2 is in the process of finalizing a wideband variable rate coder for CDMA systems. In our work, our focus was on the design of noise coding excitation models for narrowband voice communication. Our work can also be extended to wideband variable-rate and DTX-based speech coders.

In this thesis, we have presented a narrowband frame-level speech/music discrimination system. In our work, we have examined two-way classification (speech and music). To better accommodate mixed signals and spoken words with music (*Rap* music, for example), a three-way classifier (speech, music-only, and music with speech) can be developed.

6.4 Our Contribution to the Literature

The main contribution of this thesis work is the proposal of novel excitation models to encode background noise signals with natural quality at very low bit rates. In Section 1.5 we outlined the major research contributions of this thesis. A list of our publications and filed patent applications from the results of this research work are shown below:

- **Publications**

  1. K. El-Maleh, M. Klein, G. Petruci, and P. Kabal, "Speech/music discrimination for multimedia applications," *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing* (Istanbul), June 2000, pp. 2445–2448.
  2. K. El-Maleh and P. Kabal, "Natural-quality background noise coding using residual substitution," *Proc. 6th European Conf. Speech Communication and Technology* (Budapest), September 1999, pp. 2359–2362.
  3. K. El-Maleh and P. Kabal, "An improved background noise coding mode for variable rate speech coders," *Proc. IEEE Speech Coding Workshop* (Porno, Finland), June 1999, pp. 135–137.
  4. K. El-Maleh, A. Samouelian, and P. Kabal, "Frame-level noise classification in mobile environments," *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing* (Phoenix, AZ), March 1999, pp. 237–240.

5. K. El-Maleh and P. Kabal, "Frame-level noise classification in mobile environments," Document TAD 15-E (WP3/12), ITU-T Study Group 12, Question 17 ("Noise Aspects in Evolving Networks"), Nov. 1998.
6. K. El-Maleh and P. Kabal, "Comparison of voice activity detection algorithms for wireless personal communications systems," *Proc. IEEE Canadian Conf. Electrical and Computer Engineering* (St. John's, NFL), May 1997, pp. 470–473.

- **Patent Applications**

1. K. El-Maleh and P. Kabal, "Method and apparatus for providing background acoustic noise during a discontinued/reduced rate transmission mode of a voice transmission system", *Canadian Patent Application No. CA 2275832, and US Patent Application No. 60/139751*, filed on June 18, 1999.

We show below a list of recent papers that have referenced our published papers in the three areas: voice activity detection, noise classification, and speech/music discrimination.

- **Voice activity detection**

1. F.-H. Liu and M. A. Picheny, "Model-based voice activity detection system and method using a log-likelihood ratio and pitch," *US Patent No. 6615170*, September 2, 2003.
2. C. H. Chiranth *et al.*, "Comparison of voice activity detection algorithms for VoIP", *Proc. Seventh Int. Symp. on Computers and Communications*, July 2002, pp. 530–535.
3. S. Kumar, "Smart acoustic volume controller for mobile phones," *112th Convention of the Audio Engineering Society* (Munich, Germany), May 2002.
4. B. Kollmeier and M. Marzinzik, "Speech pause detection for noise spectrum estimation by tracking power envelope dynamics," *IEEE Trans. on Speech and Audio Processing*, vol 10, no. 2, Feburary 2002, pp. 109–118.
5. H. Ozer, and S. G. Tanyer,"Voice activity detection in non-stationary Gaussian noise," *Proc. Fourth Int. Conference on Signal Processing*, 1998, vol. 2, pp. 1620–1623.

6. D. Mallah, "System and method for noise threshold adaptation for voice activity detection in nonstationary noise environments," *US Patent No. 5991718*, November 23, 1998.

- **Noise classification**

1. C. Shao, C. and M. Bouchard, "Efficient classification of noisy speech using neural networks", *Proc. of Seventh Int. Symp. on Signal Processing and its Applications (ISSPA)* (Paris), July 2003, pp. 357–360.
2. F. Beritelli, S. Casale and G. Ruggeri, "Hybrid multimode/multirate CS-ACELP speech coding for adaptive voice over IP," *Speech Communication*, 38 (2002), pp. 365–381.
3. V. Peltonen *et al.* "Recognition of everyday auditory scenes: potentials, latencies and cues, " *110th Audio Engineering Society Convention*, Amsterdam, Netherlands, 2001.
4. V. Peltonen, "Computational auditory scene recognition," M.Sc. Thesis, Tampere University of Technology, Dept. of Information Technology, February 2001.
5. F. Beritelli, S. Casale, and G. Ruggeri, "New results in fuzzy pattern classification of background noise," *Proc. Fifth Int. Conference on Signal Processing* (Beijing), August 2000, pp. 1483–1486.
6. J. Sillanpaa *et al.*, "Recognition of acoustic noise mixtures by combined bottom-up and top-down processing," *Proc. European Signal Processing Conf.*, (Tampere, Finland), September 2000.

- **Speech/music discrimination**

1. H. Harb, and L. Chen, "Robust speech/music discrimination using spectrum's first order statistics and neural networks," *Proc. of the Seventh IEEE Int. Symp. on Signal Processing and its Applications*, (Paris), July 2003.
2. H. Jiang, H.-J. Zhang, L. Lu, "Content analysis for audio classification and segmentation," *IEEE Trans. on Speech and Audio Processing*, vol. 10, issue 7, October 2002, pp. 504–516

3. M. Roach, J. Mason, L-Q. Xu, and F. W. M. Stentiford, "Recent trends in video analysis: a taxonomy of video classification problems," *Sixth IASTED Int. Conf. on Internet and Multimedia Systems and Applications* (Hawaii), August 2002.

4. Allamanche *et al.*, "Content-based identification of audio material using MPEG-7 low level description," *2nd Annual Int. Symp. on Music Information Retrieval* (Bloomington, Indiana), October 2001, pp. 15–17.

5. H. Harb, L, Chen, and J.-Y. Auloge, "Speech/music/silence and gender detection algorithm," *Proc. of the 7th Int. conference on Distributed Multimedia Systems*, (Taipei, Taiwan), September 2001, pp. 257–262.

6. L. Lu, H. Jiang, and H.J. Zhang, "A robust audio classification and segmentation method," *Proc. the 9th ACM Int. Multimedia Conference and Exhibition*, August 2001.

7. L. Tancerel, S. Ragot and R. Lefebvre, "Speech/music discrimination for universal audio coding," *20th Biennial Symp. on Communications*, (Kingston, Ontario), May 2000.

References

[1] W. B. Kleijn and K. K. Paliwal, eds., *Speech Coding and Synthesis*. Elsevier, 1995.

[2] A. Gersho, "Advances in speech and audio compression," *Proc. IEEE*, vol. 82, pp. 900–918, June 1994.

[3] A. S. Spanias, "Speech coding: A tutorial review," *Proc. IEEE*, vol. 82, pp. 1541–1582, Oct. 1994.

[4] N. R. Chong, I. S. Burnett, J. F. Chicharo, and M. M. Thomson, "The effect of noise on the waveform interpolation speech coder," in *Proc. of IEEE Region 10 Annual Conf. Speech and Image Technologies for Computing and Telecommunications*, (Brisbane, Australia), pp. 609–612, Dec. 1997.

[5] M. Budagavi and J. D. Gibson, "Speech coding in mobile radio communications," *Proc. IEEE*, vol. 86, pp. 1402–1411, July 1998.

[6] T. Wigren, A. Bergstrom, S. Harrysson, F. Jansson, and H. Nilsson, "Improvements of background sound coding in linear predictive speech coders," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Detroit, MI), pp. 25–29, May 1995.

[7] T. Tanigushi and Y. Yamazaki, "Enhancement of VSELP coded speech under background noise," in *Proc. IEEE Workshop on Speech Coding for Telecommunications*, (Annapolis, MD), pp. 67–68, Sept. 1995.

[8] H. S. P. Yue and R. Rabipour, "Method and apparatus for noise conditioning in digital speech compression systems using linear predictive coding," *US Patent US5642464*, June 1997.

[9] K. Ganesan, H. Lee, and P. Gupta, "Removal of swirl artifacts from CELP-based speech coders," *US Patent US5633982*, May 1997.

[10] R. Hagen and E. Ekudden, "An 8 kbit/s ACELP coder with improved background noise performance," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Phoenix, AZ), pp. 25–28, Mar. 1999.

[11] A. Kataoka, S. Hosaka, J. Ikedo, T. Moriya, and S. Hayashi, "Improved CS-CELP speech coding in a noisy environment using a trained sparse conjugate codebook," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Detroit, MI), pp. 29–32, May 1995.

[12] H. Ehara, K. Yasunaga, Y. Hiwasaki, and K. Mano, "Noise post-processing based on a stationary noise generator," in *Proc. IEEE Workshop on Speech Coding for Telecom.*, (Ibaraki, Japan), pp. 178–180, Oct. 2002.

[13] A. Murashima, M. Serizawa, and K. Ozawa, "A post-processing technique to improve coding quality of CELP under background noise," in *Proc. IEEE Workshop on Speech Coding for Telecom.*, Sept. 2000.

[14] A. Murashima, M. Serizawa, and K. Ozawa, "A multi-rate wideband speech codec robust to background noise," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Istanbul), pp. 1165–1168, June 2000.

[15] H. Tasaki and S. Takahashi, "Post noise smoother to improve low bit rate speech-coding performance," in *Proc. IEEE Workshop on Speech Coding for Telecommunications*, (Porvoo, Finland), pp. 159–161, June 1999.

[16] T. V. Ramabadran, J. P. Ashley, and M. J. McCaughlin, "Background noise suppression for speech enhancement and coding," in *Proc. IEEE Workshop on Speech Coding for Telecommunications*, (Pocono Manor, PN), pp. 43–44, Sept. 1997.

[17] T. Agarwal, "Pre-processing of noisy speech for voice coders," Master's thesis, McGill University, Montreal, Canada, Jan. 2002.

[18] H. W. Gerlich and F. Kettler, "Background noise transmission and comfort noise insertion: The influence of signal processing on speech-quality in complex transmission systems," in *Proc. IEEE/EURASIP Int. Workshop on Acoustic Echo and Noise Control (IWAENC-01)*, 2001.

[19] S. Chennakeshu, R. D. Koilpillai, and E. Dahlman, "Enhancing the spectral efficiency of the American digital cellular system with coded modulation," in *Proc. Asilomar Conf. on Circuits and Systems*, (Pacific Grove, CA), pp. 1001–1005, Oct. 1994.

[20] K. Ivanov, N. Metzner, G. Spring, H. Winkler, and P. Jung, "Frequency hopping spectral capacity enhancement of cellular networks," in *Proc. Asilomar Conf. on Circuits and Systems*, (Pacific Grove, CA), pp. 1267–1272, Oct. 1997.

[21] M. C. Ronchini and E. Gaiani, "Improvement of GSM system performance due to frequency hopping and/or discontinuous transmission," in *Proc. Asilomar Conf. on Circuits and Systems*, (Pacific Grove, CA), pp. 1596–1600, Oct. 1997.

[22] J. Fuhl, A. Kuchar, and E. Bonek, "Capacity increase in cellular PCS by smart antennas," in *Proc. Asilomar Conf. on Circuits and Systems*, (Pacific Grove, CA), pp. 1962–1966, Oct. 1997.

[23] U. Martin and I. Gaspard, "Capacity enhancement of narrowband CDMA by intelligent antennas," in *Proc. Asilomar Conf. on Circuits and Systems*, (Pacific Grove, CA), pp. 90–94, Oct. 1997.

[24] A. Das, E. Paksoy, and A. Gersho, "Multimode and variable-rate speech coding," *in Speech Coding and Synthesis, W. B. Kleijn and K. K. Paliwal Eds., Elsiver*, pp. 257–288, 1995.

[25] E. F. O'Neil, "TASI," *Bell Lab. Rec.*, vol. 37, pp. 83–87, Mar. 1959.

[26] S. J. Campanella, "Digital speech interpolation," *COMSAT Tech. Rev.*, vol. 6, pp. 127–158, 1976.

[27] K. Y. Kou, J. B. O'Neal Jr., and A. A. Nilsson, "Digital speech interpolation for variable rate coders with application to subband coding," *IEEE Trans. Communications*, vol. COM-33, pp. 1100–1108, Oct. 1985.

[28] ETSI TC-SMG, GSM 06.62 Version 6.0.0 Release 1997, *Digital Cellular Telecommunications System (Phase 2+); Discontinuous Transmission (DTX) for Enhanced Full Rate (EFR) Speech Traffic Channels*, 1997.

[29] ETSI TS 126 093 V3.2.0, 3G TS 26.093 version 3.2.0 Release 1999, *Universal Mobile Telecommunications System (UMTS); Mandatory Speech Codec speech processing functions AMR speech codec; Source Controlled Rate operation*, 2000.

[30] M. Mouly and M. Pautet, *GSM System for Mobile Communications: A Comperhensive Overview of the European Digital Cellular Systems*. Telecom Publishing, 1992.

[31] W. C. Y. Lee, "Overview of cellular CDMA," *IEEE Trans. Vehicular Technology*, pp. 291–302, May 1991.

[32] A. Viterbi, *CDMA: Principles of Spread Spetcrum Communication*. Addisson Wesley Publishing, 1995.

[33] A. DeJaco, W. Gardner, P. Jacobs, and C. Lee, "QCELP: The North American CDMA digital cellular variable rate speech coding standard," in *Proc. IEEE Workshop on Speech Coding for Telecommunications*, (Sainte-Adèle, Québec), pp. 5–6, Oct. 1993.

[34] M. C. Recchione, "The enhanced variable rate coder: Toll quality speech for CDMA," *Int. Journal of Speech Technology*, vol. 2, pp. 305–315, May 1999.

[35] C. P. Mammen and B. Ramamurthi, "Capacity enhancement in digital cellular systems using variable bitrate speech coding," in *Proc. IEEE Int. Conf. Communications*, (Montreal), pp. 735–739, June 1997.

[36] A. Benyassine *et al.*, "ITU-T Recommendation G.729 Annex B: A silence compression scheme for use with G.729 optimized for V.70 digital simultaneous voice and data applications," *IEEE Communications Magazine*, vol. 35, pp. 64–73, Sept. 1997.

[37] R. Cox and P. Kroon, "Low bit-rate speech coders for multimedia communication," *IEEE Communications Magazine*, vol. 34, pp. 34–41, Dec. 1996.

[38] S. Jacobs, A. Eleftheriadis, and D. Anastassiou, "Silence detection for multimedia communication systems," *Multimedia Systems*, vol. 7, pp. 157–164, Mar. 1999.

[39] ETSI TS 126 092 V3.0.1, 3G TS 26.092 version 3.0.1 Release 1999, *Universal Mobile Telecommunications System (UMTS); Mandatory Speech Codec speech processing functions AMR speech codec; Comfort noise aspects*, 2000.

[40] K. Swaminathan and B. M. McCarthy, "Comfort noise generation for digital communication systems," *US Patent US5537509*, July 1996.

[41] J. Rotola-Pukkila, K. Jarvinen, P. Kapanen, and V. Ruoppila, "Methods for generating comfort noise during discontinous transmission," *US Patent US5960389*, May 1998.

[42] D. Massaloux, "Process and device for creating comfort noise in a digital speech transmission system," *US Patent US5812965*, Sept. 1998.

[43] A. V. Rao and W. P. LeBlanc, "Method and system for improved discontinuous speech transmission," *US Patent US5794199*, Aug. 1998.

[44] ETSI TC-SMG, GSM 06.81 Version 6.0.0 Release 1997, *Digital Cellular Telecommunications System (Phase 2+); Comfort Noise Aspects for Enhanced Full Rate (EFR) Speech Traffic Channels*, 1997.

[45] TIA/EIA/IS-733, *High Rate Speech Service Option for Wideband Spread Spectrum Communications Systems*, Feb. 1996.

[46] TIA/EIA/IS-127, *Enhanced Variable Rate Codec, Speech Service Option 3 for Wideband Spread Spectrum Digital Systems*, Jan. 1996.

[47] P. Kroon and M. Recchione, "A low-complexity toll-quality variable rate coder for CDMA digital cellular," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Detroit, MI), pp. 5–8, May 1995.

[48] F. Beritelli, "A modified CS-ACELP algorithm for variable-rate speech coding robust in noisy environments," *IEEE Signal Processing Letters*, vol. 6, pp. 31–34, Feb. 1999.

[49] E. Paksoy, A. McCree, and V. Viswanathan, "A variable-rate multimode speech coder with gain-matched analysis-by-synthesis," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Munich, Germany), pp. 751–754, Apr. 1997.

[50] E. W. Yu and C. F. Chan, "Variable bit rate MBELP speech coding via v/uv distribution dependent spectral quantization," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Munich, Germany), pp. 1607–1610, Apr. 1997.

[51] S. McClellan and J. D. Gibson, "Variable-rate CELP based on subband flatness," *IEEE Trans. Speech, and Audio Processing*, vol. 5, pp. 120–130, Mar. 1997.

[52] K. El-Maleh and P. Kabal, "An improved background noise coding mode for variable rate speech coders," in *Proc. IEEE Workshop on Speech Coding for Telecommunications*, (Porvoo, Finalnd), pp. 135–137, June 1999.

[53] K. El-Maleh and P. Kabal, "Natural-quality background noise coding using residual substitution," in *Proc. European Conf. on Speech Commun. and Technology*, (Budapest, Hungary), pp. 2359–2362, Sept. 1999.

[54] K. El-Maleh and P. Kabal, "Method and apparatus for providing background acoustic noise during a discontinued/reduced rate transmission mode of a voice transmission system," *Canadian Patent Application No. CA 2275832 (patent pending)*, June 1999.

[55] K. El-Maleh and P. Kabal, "Method and apparatus for providing background acoustic noise during a discontinued/reduced rate transmission mode of a voice transmission system," *USA Patent Application No. 60/139,751 (patent pending)*, June 1999.

[56] K. El-Maleh, A. Samouelian, and P. Kabal, "Frame-level noise classification in mobile environments," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Phoenix, AZ), pp. 237–240, Mar. 1999.

[57] K. El-Maleh and P. Kabal, "Frame-level noise classification in mobile environments," tech. rep., Document TD 15-E (WP3/12), ITU-T Study Group 12, Question 17 (Noise Aspects in Evolving Networks), Geneva, Nov. 1998.

[58] K. El-Maleh, M. Klein, G. Petrucci, and P. Kabal, "Speech/music discrimination for multimedia applications," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Istanbul), pp. 2445–2448, June 2000.

[59] P. T. Brady, "A model for generating on-off speech patterns in two-way conversation," *Bell Syst. Tech. J.*, pp. 2445–2472, Sept. 1969.

[60] H. P. Stern, S. A. Mahmoud, and K. K. Wong, "A model for generating on-off patterns in conversational speech, including short silence gaps and the effects of interaction between parties," *IEEE Trans. Vehicular Technology*, vol. 43, pp. 1094–1100, Nov. 1994.

[61] J. Gruber, "A comparison of measured and calculated speech temporal parameters relevant to speech activity detection," *IEEE Trans. Communications*, vol. COM-30, pp. 728–738, Apr. 1982.

[62] H. H. Lee and C. K. Un, "A study of on-off characteristics of conversational speech," *IEEE Trans. Communications*, vol. Com-34, pp. 630–637, June 1986.

[63] ITU-T, Geneva, *Recommendation P.59, Artificial Conversational Speech*, Mar. 1993.

[64] Y. Yatsuzuka, "Highly sensitive speech detector and high-speed voiceband data discriminator in DSI-ADPCM systems," *IEEE Trans. Communications*, vol. COM-30, pp. 739–750, Apr. 1982.

[65] ITU-T, Geneva, *Recommendation P.50, Artificial Voices*, Mar. 1993.

[66] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*. Prentice-Hall, 1978.

[67] J. D. Markel and A. H. Gray, *Linear Prediction of Speech*. Berlin, Germany: Springer-Verlag, 1976.

[68] ITU-T, Geneva, *Recommendation P.56, Objective Measurement of Active Speech Level*, Mar. 1993.

[69] P. Kabal, "Measuring speech activity," tech. rep., McGill University, Aug. 1999.

[70] S. F. de Campos Neto, "The ITU-T software tool library," *Int. Journal of Speech Technology*, vol. 2, pp. 259–272, May 1999.

[71] 3GPP2 C.S0030-0, version 1.0, *Selectable Mode Vocoder Service Option for Wideband Spread Spectrum Communication Systems*, Dec. 2001.

[72] Y. Gao, E. Shlomot, A. Benyassine, J. Thyssen, H. Su, and C. Murgia, "The SMV algorithm selected by TIA and 3GPP2 for CDMA aplications," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Salt Lake City, UT), pp. 7–11, vol. 2, May 2001.

[73] B. S. Atal and L. R. Rabiner, "A pattern recognition approach to voiced-unvoiced-silence classification with applications to speech recognition," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. 24, pp. 201–212, June 1976.

[74] J. A. Haigh and J. S. Mason, "Robust voice activity detection using cepstral features," in *Proc. of IEEE Region 10 Annual Conf. Speech and Image Technologies for Computing and Telecommunications*, (Beijing), pp. 321–324, Oct. 1993.

[75] S. A. McClellan and J. D. Gibson, "Spectral entropy: An alternative indicator for rate allocation?," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Adelaide, Australia), pp. 201–204, Apr. 1994.

[76] R. Tucker, "Voice activity detection using a periodicity measure," *IEE Proc.-I*, vol. 139, pp. 377–380, Aug. 1992.

[77] J. Stegmann and G. Schroder, "Robust voice-activity detection based on the wavelet transform," in *Proc. IEEE Workshop on Speech Coding for Telecommunications*, (Pocono Manor, PN), pp. 99–100, Sept. 1997.

[78] K. El-Maleh and P. Kabal, "Comparison of voice activity detection algorithms for wireless personal communications systems," in *Proc. IEEE Canadian Conf. on Electrical and Computer Engineering*, (St. John's, Nfld), pp. 470–473, May 1997.

[79] Y. Ephraim, "Statistical-model-based speech enhancement systems," *Proc. IEEE*, vol. 80, pp. 1526–1555, Oct. 1992.

[80] M. Marzinzik and B. Kollmeier, "Speech pause detection for noise spectrum estimation by tracking power envelope dynamics," *IEEE Trans. Speech, and Audio Processing*, vol. 10, pp. 109–118, Feb. 2002.

[81] A. Fischer and V. Stahl, "On improvement measures for spectral subtraction applied to robust automatic speech recognition in car environments," in *Proc. of the Workshop on Robust Methods for Speech Recognition in Adverse Conditions*, (Tampere, Finalnd), pp. 75–78, May 1999.

[82] J.-C. Junqua, B. Mak, and B. Reaves, "A robust algorithm for word boundary detection in the presence of noise," *IEEE Trans. Speech, and Audio Processing*, vol. 2, pp. 406–412, July 1994.

[83] S. Kuroiwa, M. Naito, S. Yamamoto, and N. Higuchi, "Robust speech detection method for telephone speech recognition system," *Speech Communication*, vol. 27, pp. 135–148, 1999.

[84] J. Sohn and W. Sung, "A voice activity detector employing soft decision based noise spectrum adaptation," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Seattle, WA), pp. 365–368, May 1998.

[85] S. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Processing Letters*, vol. 6, pp. 1–3, Jan. 1999.

[86] Y. K. Cho and A. Kondoz, "Analysis and improvement of a statistical model-based voice activity detector," *IEEE Signal Processing Letters*, vol. 8, pp. 276–278, Oct. 2001.

[87] S. Gazor and W. Zhang, "A soft voice activity detector based on a Laplacian-Gaussian model," *IEEE Trans. Speech, and Audio Processing*, pp. 498–505, Sept. 2003.

[88] M. Rangoussi, A. Delopoulos, and M. Tsatsanis, "On the use of higher-order statistics for robust endpoint detection of speech," in *IEEE Proc. Workshop HOS*, (South Lake Tahoe, CA), pp. 55–60, June 1993.

[89] M. Rangoussi and G. Carayannis, "Higher order statistics based gaussianity test applied to on-line speech processing," in *Proc. Asilomar Conf. on Circuits and Systems*, (Pacific Grove, CA), pp. 303–307, Oct. 1994.

[90] J. Navarro-Mesa, A. Moreno-Bilbao, and E. Lleida-Solano, "An improved speech endpoint detection system in noisy environments by means of third-order spectra," *IEEE Signal Processing Letters*, vol. 6, pp. 224–226, Sept. 1999.

[91] E. Nemer, R. Goubran, and S. Mahmoud, "Robust voice activity detection using higher-order statistics in the LPC residual domain," *IEEE Trans. Speech, and Audio Processing*, vol. 9, pp. 217 – 231, Mar. 2001.

[92] A. Cavallaro, F. Beritelli, and S. Casale, "A fuzzy logic-based speech detection algorithm for communications in noisy environments," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Seattle, WA), pp. 565–568, May 1998.

[93] F. Beritelli, S. Casale, and A. Cavallaro, "A robust voice activity detector for wireless communications using soft computing," *IEEE J. Selected Areas in Comm.*, vol. 16, pp. 1818–1829, Dec. 1998.

[94] D. K. Freeman, G. Cosier, C. B. Southcott, and I. Boyd, "The voice activity detector for the pan-European digital cellular mobile telephone service," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Glasgow, Scotland), pp. 369–372, May 1989.

[95] E. Ekudden, R. Hagen, I. Johansson, and J. Svedberg, "The adaptive multi-rate speech coder," in *Proc. IEEE Workshop on Speech Coding for Telecommunications*, (Porvoo, Finalnd), pp. 117–119, June 1999.

[96] ETSI TS 126 094 V3.0.0 (2000-01), 3G TS 26.094 version 3.0.0 Release 1999, *Universal Mobile Telecommunications System (UMTS); Mandatory Speech Codec speech processing functions AMR speech codec; Voice Activity Detector (VAD)*, 2000.

[97] F. Beritelli, S. Casale, G. Ruggeri, and S. Serrano, "Performance evaluation and comparison of G.729/AMR/fuzzy voice activity detectors," *IEEE Signal processing Letters*, vol. 9, pp. 85–88, Mar. 2002.

[98] N. Doukas, P. Naylor, and T. Stathaki, "Voice activity detection using source separation techniques," in *Proc. European Conf. on Speech Commun. and Technology*, (Rhodes, Greece), pp. 1099–1102, Sept. 1997.

[99] J. Ikedo, "Voice activity detection using neural network," *IEICE Trans. Commun.*, vol. E81-B, pp. 2509–2513, Dec. 1998.

[100] J. D. Hoyt and H. Wechsler, "Detection of human speech in structured noise," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Adelaide, Australia), pp. 237–240, Apr. 1994.

[101] S. G. Tanyer and H. Ozer, "Voice activity detection in nonstationary noises," *IEEE Trans. Speech, and Audio Processing*, vol. 8, pp. 478–482, July 2000.

[102] P. Pollak, "Efficient and reliable measurement and simulation of noisy speech background," in *XI European Signal Processing Conf. (EUSIPCO 2002)*, (Toulouse, France,), Sept. 2002.

[103] F. Beritelli, S. Casale, and A. Cavallaro, "New performance evaluation criteria and a robust algorithm for speech activity detection in wireless communications," in *Int. Conf. on Telecommunications*, (Porto Carras, Greece), pp. 223–227, June 1998.

[104] F. Beritelli, S. Casale, and G. Ruggeri, "A psychoacoustic auditory model to evaluate the performance of a voice activity detector," in *5th Int. Conf. on Signal Processing*, (Beijing, China), pp. 807–810, Aug. 2000.

[105] K. Srinivasan and A. Gersho, "Voice activity detection for cellular networks," in *Proc. IEEE Workshop on Speech Coding for Telecommunications*, (Sainte-Adèle, Québec), pp. 85–86, Oct. 1993.

[106] 3GPP2-C11-20000425-xxx, version 11, *Test Plan and Requirements of the Selectable Mode Vocoder*, 2000.

[107] J. G. Proakis, C. M. Rader, F. Ling, and C. L. Nikias, *Advanced Digital Signal Processing*. Macmillan Publishing Company, 1992.

[108] C. W. Therrien, *Discrete Random Signals and Statistical Signal Processing*. Prentice-Hall, 1992.

[109] D. O'Shaughnessy, *Speech Communication: Human and Machine*. Addison-Wesley Publishing Company, 1987.

[110] K. K. Paliwal and B. S. Atal, "Efficient vector quantization of LPC parameters at 24 bits/frame," *IEEE Trans. Speech, and Audio Processing*, vol. 1, pp. 3–14, Jan. 1993.

[111] A. V. McCree and T. P. Barnwell III, "Mixed excitation LPC vocoder model for low bit rate speech coding," *IEEE Trans. Speech, and Audio Processing*, vol. 3, pp. 242–250, July 1995.

[112] D. W. Griffin and J. S. Lim, "Multiband excitation vocoder," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. ASSP-36, pp. 1223–1235, Aug. 1988.

[113] B. S. Atal and J. R. Remde, "A new model of LPC excitation for producing natural-sounding speech at low bit rates," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Paris), pp. 614–617, May 1982.

[114] B. S. Atal and M. R. Schroeder, "Stochastic coding of speech signals at very low bit rates," in *Proc. IEEE Int. Conf. Communications*, (Amsterdam), pp. 1610–1613, May 1984.

[115] E. Moulines and K. Choukri, "Time-domain procedures for testing that a stationary time-series is gaussian," *IEEE Trans. Signal Processing*, vol. 44, pp. 2010–2025, Aug. 1996.

[116] A. C. Rencher, *Methods of Multivariate Analysis*. John Wiely & Sons, 1995.

[117] A. Papoulis, *Probability, Random Variables, and Stochastic Processes, 2nd edition*. McGraw-Hill, 1984.

[118] G. Kubin, B. Atal, and W. B. Kleijn, "Performance of noise excitation for unvoiced speech," in *Proc. IEEE Workshop on Speech Coding for Telecommunications*, (Sainte-Adèle, Québec), pp. 35–36, Oct. 1993.

[119] B. S. Atal, "Predictive coding of speech at low bit rates," *IEEE Trans. Communications*, vol. COM-30, pp. 600–614, Apr. 1982.

[120] G. Kubin, "On the nonlinearity of linear prediction," in *Proc. European Signal Processing Conf.*, (Rhodes, Greece), 1998.

[121] W. B. Kleijn and K. K. Paliwal, eds., *Speech Coding and Synthesis, Chapter 16*. Elsevier, 1995.

[122] J. A. H. Gray and J. D. Markel, "A spectral-flatness measure for studying the autocorrelation method of linear prediction of speech analysis," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. 22, pp. 207–217, June 1974.

[123] N. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice-Hall, 1984.

[124] B. Kedem, *Time Series Analysis by Higher Order Crossings*. IEEE Press, 1994.

[125] M. Goodwin, "Residual modeling in music analysis-synthesis," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Atlanta, GA), pp. 1005–1008, May 1996.

[126] A. V. Oppenheim and J. S. Lim, "The importance of phase in signals," *Proc. IEEE*, vol. 69, pp. 529–541, May 1981.

[127] H. Pobloth and W. B. Kleijn, "On phase perception in speech," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Phoenix, AZ), pp. 29–32, Mar. 1999.

[128] S. Kim, "Perceptual phase redundancy in speech," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Istanbul), pp. 1383–1386, June 2000.

[129] B. S. Atal and N. David, "On synthesizing natural-sounding speech by linear prediction," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, pp. 44–47, 1979.

[130] B. Elsendoorn and H. Bouma, *Working Models of Human Perception (Chapter 6)*. Academic Press, 1989.

[131] C. Ma and D. O'Shaughnessy, "A perceptual study of source coding of Fourier phase and amplitude of the linear predictive coding residual of vowel sounds," *J. Acoust. Soc. Am.*, vol. 95, pp. 2231–2239, Apr. 1994.

[132] O. Gauthreot, J. S. Mason, and P. Corney, "LPC residual phase investigation," in *Proc. European Conf. on Speech Commun. and Technology*, (Paris), pp. 35–38, Sept. 1989.

[133] I. M. Trancoso, R. Garcia-Gomez, and J. M. Tribolet, "A study on short-time phase and multipulse LPC," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (San Diego, CA), pp. 10.3.1–10.3.4, Mar. 1984.

[134] P. Hedelin, "Phase compensation in all-pole speech analysis," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (New York, NY), pp. 339–342, Apr. 1988.

[135] B. Cheetham, H. Choi, X. Sun, C. Goodyear, F. Plante, and W. Wong, "All-pass excitation phase modlleing for low bit-rate speech coding," in *1997 IEEE Int. Symp. on Circuits and Systems*, (Hong Kong), pp. 2633–2636, June 1997.

[136] T. V. Ramabadran and C. D. Lueck, "Complexity reduction of CELP speech coders through the use of phase information," *IEEE Trans. Communications*, vol. 42, pp. 248–251, feb/mar/apr 1994.

[137] H. Banno, J. Lu, S. Nakamura, K. Shikano, and H. Kawahara, "Efficient representation of short-time phase based on group delay," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Seattle, WA), pp. 861–864, May 1998.

[138] N. Saint-Arnaud and K. Popat, "Analysis and synthesis of sound textures," in *Proc. of Workshop on Computational Auditory Scene Analysis*, (Montreal, Quebec), pp. 125–131, Aug. 1995.

[139] M. R. Schroeder and B. S. Atal, "Code-excited linear predictive (CELP): High quality speech at very low bit rates," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Tampa, FL), pp. 937–940, Mar. 1985.

[140] ETSI TC-SMG, GSM 06.82 Version 6.0.0 Release 1997, *Digital Cellular Telecommunications System (Phase 2+); Voice Activity Detector (VAD) for Enhanced Full Rate (EFR) Speech Traffic Channels*, 1997.

[141] S. McAdams, *Thinking in Sound: The Cognitive Psychology of Human Audition*. Oxford University Press, 1993.

[142] A. S. Bregman, *Auditory Scene Analysis*. MIT Press, 1990.

[143] M. Akbacak and J. Hansen, "Environmental sniffing: Noise knowledge estimation for robust speech systems," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Hong Kong), pp. 113–116, Apr. 2003.

[144] W. C. Treurniet and Y. Gong, "Noise independent speech recognition for a variety of noise types," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Adelaide, Australia), pp. 437–440, Apr. 1994.

[145] N. Nicol, S. Euler, M. Falkhausen, H. Reininger, and D. Wolf, "Noise classification using vector quantization," in *Proc. European Signal Processing Conf.*, (Edinburg, Scotland), pp. 1705–1708, Sept. 1994.

[146] A. Sugiyama, T. P. Hua, M. Kato, and M. Serizawa, "Noise suppression with synthesis windowing and pseudo noise injection," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Orlando-FL), pp. 545–548, May 2002.

[147] S. Kumar, "Smart acoustic volume controller for mobile phones," in *112th AES Convention*, (Munich), May 2002.

[148] ITU-T, Geneva, *COM 12-1-E- List and wording of questions allocated to Study Group 12 for study during the 1997–2000 study period*, Feb. 1997.

[149] J. M. Kates, "Classification of background noises for hearing-aid applications," *J. Acoust. Soc. Am.*, vol. 97, pp. 461–470, Jan. 1995.

[150] C. Couvreur, *Environmental Sound Recognition: A Statistical Approach*. PhD thesis, Faculte Polytechnique de Mons, 1997.

[151] B. Clarkson, N. Sawhney, and A. Pentland, "Auditory context awareness via wearable computing," in *Proc. of the Perceptual User Interface Workshop*, (San Francisco, CA), pp. 37–42, 1998.

[152] F. Beritelli and S. Casale, "Background noise classification in advanced VBR speech coding for wireless communications," in *IEEE Int. Workshop on Intelligent Signal Proc. and Comm. Sys. (ISPACS'98)*, (Melbourne, Australia), pp. 451–455, Nov. 1998.

[153] F. Beritelli, S. Casale, and G. Ruggeri, "New results in fuzzy pattern classification of background noise," in *5th Int. Conf. on Signal Processing*, (Beijing), pp. 1483–1486, Aug. 2000.

[154] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*. Academic Press, 1999.

[155] F. Itakura, "Line spectrum representation of linear prediction coefficients," *J. Acoust. Soc. Am.*, vol. 57, p. S35(A), 1975.

[156] F. K. Soong and B. Juang, "Line spectrum pair (LSP) and speech data compression," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (San Diego, CA), pp. 1.10.1–1.10.4, Mar. 1984.

[157] J.-Y. Tourneret, "Statistical properties of line spectrum pairs," *Signal Processing*, vol. 65, pp. 239–255, Mar. 1998.

[158] P. Kabal and R. P. Ramachandran, "The computation of line spectral frequencies using Chebyshev polynomials," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. ASSP-34, pp. 1419–1426, Dec. 1986.

[159] S. Gracci, *Optimized Implementation of Speech Processing Algorithms*. PhD thesis, University of Neuchatel, IMT, Switzerland, Feb. 1998.

[160] J. S. Erkelens and P. M. T. Broersen, "On the statistical properties of line spectrum pairs," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Detroit, MI), pp. 768–771, May 1995.

[161] K. K. Paliwal, "A study of line spectrum pair frequencies for speech recognition," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (New York, NY), pp. 485–488, Apr. 1988.

[162] K. K. Paliwal, "A study of LSF representation for speaker-dependent and speaker-independent HMM-based speech recognition systems," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Albuquerque, NM), pp. 801–804, Apr. 1990.

[163] C.-S. Liu and M.-T. Lin, "Study of line spectrum pair frequencies for speaker recognition," in *IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Albuquerque, NM), pp. 277–280, Apr. 1990.

[164] F. S. Gurgen, S. Sagayama, and S. Furui, "A study of line spectrum frequency representation for speech recognition," *IEICE Trans. Fundamentals*, vol. 75, pp. 98–102, Jan. 1992.

[165] J. P. Campbell, "Speaker recognition: A tutorial," *Proc. IEEE*, vol. 85, pp. 1437–1462, Sept. 1997.

[166] I. V. McLoughlin and S. Thambipillai, "LSP parameter interpretation for speech classification," in *The 6th IEEE Int. Conf. on Electronics, Circuits and Systems*, (Pafos, Cyprus), pp. 419–422, Sept. 1999.

[167] Y. Lee, M. Ham, and M. Bae, "A study on a reduction of the transmission bit rate by u/v decision using LSP in the CELP vocoder," in *42nd Midwest Symp. on Circuits and Systems*, (Las Cruces, NM), pp. 997–1000, Aug. 1999.

[168] J. J. Parry, I. S. Burnett, and J. F. Chicharo, "The use of LSF-based phonetic classification in low-rate coder design," in *Proc. IEEE Workshop on Speech Coding for Telecommunications*, (Porvoo, Finland), pp. 49–51, June 1999.

[169] M. Elshafei, S. Akhtar, and M. S. Ahmed, "Parametric models for helicopter identification using ANN," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 36, pp. 1242–1252, Oct. 2000.

[170] H. K. Kim, K. C. Kim, and H. S. Lee, "Enhanced distance measure for LSP-based speech recognition," *Electron. Letters*, vol. 29, pp. 1463–1465, Aug. 1993.

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

[172] T. M. Cover and P. E. Hart, "Nearest-neighbor pattern classification," *IEEE Trans. Inform. Theory*, vol. 13, pp. 21–27, Jan. 1967.

[173] B. V. Dasarathi, *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*. IEEE Computer Society Press, Los Alamitos, California, 1991.

[174] C. Decaestecker, "Finding prototypes for nearest neighbour classification by means of gradient descent and deterministic annealing," *Pattern Recognition*, vol. 30, no. 2, pp. 281–288, 1997.

[175] T. Kohonen, *Self-Organizing Maps, 2nd edition*. Springer Series in Information Sciences, 1997.

[176] J. R. Quinlan, *C4.5. Programs for Machine Learning, Morgan Kaufmann Series in Machine Learning*. Morgan Kaufmann Publisher, 1993.

[177] A. Samouelian, "Frame-level phoneme classification using inductive inference," *Computer Speech and Language*, no. 11, pp. 161–186, 1997.

[178] M. Cheung and M. Chiang, "Background noise classification using neural networks," tech. rep., McGill University, Dept. of Electrical and Computer Engineering, Montreal, Canada, Dec. 1998.

[179] S. S. Haykin, *Neural Network: A Comprehensive Foundation*. McMillan College Publishing Company, 1994.

[180] D. Kobayashi, S. Kajita, K. Takeda, and F. Itakura, "Extracting speech features from human speech like noise," in *Proc. Int. Conf. on Spoken Language Processing*, pp. 418–421, Oct. 1996.

[181] C. Couvreur and Y. Bresler, "Classification of mixtures of acoustic noise signals," in *Proc. IEEE 8th Workshop on Signal Processing (DSP'98)*, (Bryce Canyon, UT), Aug. 1998.

[182] J. Sillanpaa, A. Klapuri, J. Seppnen, and T. Virtanen, "Recognition of acoustic noise mixtures by combined bottom-up and top-down processing," in *Proc. X European Signal Processing Conf.*, (Tampere, Finland), Sept. 2000.

[183] A. Sasou and K. Tanaka, "A waveform generation model based approach for segregation of monaural mixture sound," in *XI European Signal Processing Conf. (EUSIPCO 2002)*, (Toulouse, France,), pp. 409–412, Sept. 2002.

[184] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum, New York, 1981.

[185] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Comput. Geosci.*, vol. 10, pp. 191–203, 1984.

[186] J. Saunders, "Real-time discrimination of broadcast speech/music," in *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Atlanta, GA), pp. 993–996, May 1996.

[187] E. Scheirer and M. Slaney, "Construction and evaluation of a robust multifeature speech/music discriminator," in *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Munich, Germany), pp. 1331–1334, Apr. 1997.

[188] G. Williams and D. Ellis, "Speech/music discrimination based on posterior probability features," in *Proc. European Conf. on Speech Commun. and Technology*, (Budapest, Hungary), pp. 687–690, Sept. 1999.

[189] S. A. Ramprashad, "A multimode transform predictive coder (MTPC) for speech and audio," in *Proc. IEEE Workshop on Speech Coding for Telecom.*, (Porvoo, Finalnd), pp. 10–12, June 1999.

[190] L. Tancerel, S. Ragot, and R. Lefebvre, "Speech/music discrimination for universal audio coding," in *Proc. 20th Biennial Symp.on Communications*, (Kingston, Ontario), May 2000.

[191] ISO-IEC, *MPEG-4 Overview (ISO/IEC JTC1/SC29/WG11 N2995 Document)*, Oct. 1998.

[192] R.-Y. Qiao, "Mixed wideband speech and music coding using a speech/music discriminator," in *Proc. of IEEE Region 10 Annual Conf. Speech and Image Technologies for Computing and Telecommunications*, (Brisbane, Qld. , Australia), pp. 605–608, Dec. 1997.

[193] T. Zhang and C.-C. J. Kuo, "Hierarchical classification of audio data for archiving and retrieving," in *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Phoenix, AZ), pp. 3001–3004, Mar. 1999.

[194] K. Minami, A. Akutsu, H. Hamada, and Y. Tonomura, "Video handling with music and speech detection," *IEEE Multimedia*, vol. 5, pp. 17–25, July 1998.

[195] M. J. Carey, E. S. Parris, and H. Lloyd-Thomas, "A comparison of features for speech, music discrimination," in *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Phoenix, AZ), Mar. 1999.

[196] S. Dubnov *et al.*, "Synthesizing sound textures through wavelet tree learning," *IEEE Computer Graphics and Applications*, pp. 38–48, July 2002.

[197] N. Miner, *Creating Wavelet-based Models for Real-time Synthesis of Perceptually Convincing Environmental Sounds*. PhD thesis, University of New Mexico, 1998.

[198] J.-F. C. Cardoso, "Blind signal separation: Statistical principles," *Proc. IEEE*, vol. 86, pp. 2009–2025, Oct. 1998.

[199] B. Bessette *et al.*, "The adaptive multirate wideband speech codec (AMR-WB)," *IEEE Trans. Speech, and Audio Processing*, pp. 620–636, Nov. 2002.