TAMPERE UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF INFORMATION TECHNOLOGY

Vesa Peltonen

COMPUTATIONAL AUDITORY SCENE RECOGNITION

Master of Science Thesis

The subject was approved by the Department of Information Technology
on the 14th of February 2001.

Examiners: Prof. Jaakko Astola (Tampere University of Technology)
M.Sc. Anssi Klapuri (Tampere University of Technology)
Dr. Tech. Jyri Huopaniemi (Nokia Research Center)

Preface

This work was carried out in the Signal Processing Laboratory, Department of Information Technology, Tampere University of Technology, Finland.

I wish to express my gratitude to my thesis advisor and examiner Mr. Anssi Klapuri, for his constant guidance, advice, and patience throughout this work. Also, I thank the other examiners, Professor Jaakko Astola and Dr. Jyri Huopaniemi, for their insightful advice, remarks, and comments.

I would like to thank my colleague Mr. Antti Eronen, for giving useful tips, criticism, proofreading, and for the supporting friendship.

I owe a debt to my colleagues Mr. Mikko Parviainen, Mr. Toni Heittola, and Mr. Juha Tuomi, for carrying out the exhausting audio measurements that made this thesis possible. I want to say particular thanks to Mr. Parviainen for loyally maintaining our Linux operating system.

I am also grateful to the whole staff of Audio Research Group of Tampere University of Technology, for their valuable knowledge and support, and for making the work enjoyable every single day. I want to thank the staff of Nokia Research Center, especially Mr. Timo Sorsa and Mr. Kalle Koivuniemi for their special help.

I take this opportunity to say big thanks to all my friends, who have always been there for me through thick and thin.

I would especially like to thank my family for their prayers, encouragement, support, and endless love.

Last but not least, I would to thank God for giving me everything I have and for walking with me through the trials and tribulations of life.

Tampere, August 2001

Vesa Peltonen

Table of Contents

Tiivistelmä

TAMPEREEN TEKNILLINEN KORKEAKOULU

Tietotekniikan osasto

Signaalinkäsittelyn laitos

Peltonen, Vesa: Kuulomaiseman Automaattinen Tunnistus

Diplomityö, 64 sivua.

Tarkastajat: Prof. Jaakko Astola, DI Anssi Klapuri, TkT Jyri Huopaniemi

Rahoittaja: Nokia Tutkimuskeskus

Elokuu 2001

Avainsanat: kuulomaiseman tunnistus, tietoisuus kontekstista, audion luokittelu, laskennallinen kuulema-analyysi, audion sisältöanalyysi

Kuuntelijan akustinen ympäristö, kuulomaisema, voi välittää informaatiota joka mahdollistaa ympäristön tunnistamisen. Tässä työssä käsitellään kuulomaiseman automaattista tunnistusta, joka on laskennallisen kuulema-analyysin osaongelma. Laskennallisella kuulema-analyysillä tarkoitetaan akustisen ympäristön automaattista analyysiä ja yksittäisten äänitapahtumien tunnistusta. Tämän työn painopiste ei ole yksittäisten äänitapahtumien erittely ja tunnistus (vaikka niitä voidaan hyödyntää luokittelumenetelmissä), vaan kokonaisten akustisten ympäristöjen luokittelu.

Tämä diplomityö kattaa tutkimuksen kaikki eri vaiheet. Näitä ovat kirjallisuustutkimus koskien kuulomaiseman tunnistusta sekä useita muita siihen liittyviä aloja, mittaukset useasta eri ääniympäristöstä, äänitietokannan ohjelmistorajapinnan suunnittelu sekä toteutus, kuuntelukoe koskien ihmisten kykyä tunnistaa kuulomaisemia, äänisignaalien luokittelumenetelmät, algoritmikehitys ja kehitetyn järjestelmän simulointi.

Työn tärkein osa käsittelee automaattista äänisignaalien luokittelua ja signaalinkäsittelyalgoritmien kehitystä. Äänimaiseman tunnistusongelma sisältää useita osaongelmia: tarkoituksenmukainen ympäristöjen ryhmittely, akustisten piirteiden valinta ja irrottaminen, sekä sopivan luokittelualgoritmin löytäminen. Ongelman ratkaisun kriittisin vaihe on löytää riittävän kuvaavia piirteitä, joilla eri luokkien akustinen data voidaan erotella.

Suoritettu kuuntelukoe osoittaa, että ihmiset pystyvät tunnistamaan 25 erilaista ääniympäristöä keskimäärin 70 % tarkkuudella. Tunnistettavat ympäristöt sisälsivät jokapäiväisiä ulko- ja sisäympäristöjä, mm. katuja, toreja, ravintoloita sekä koteja. Automaattisen luokittelumenetelmän tehokkuus testattiin Matlab-simuloinneilla. Paras saatu tunnistustarkkuus oli 56 %, kun luokiteltiin 13 eri ympäristöä. Luokiteltavat ympäristöt valittiin siten, että jokaisesta oli vähintään kolme äänitystä eri paikoista. Laajempia luokkia, nk. metaluokkia luokitellessa saatiin suhteellisen hyviä tuloksia. Esimerkiksi auto voidaan erottaa muista ympäristöistä 95 % tarkkuudella.

Abstract

TAMPERE UNIVERSITY OF TECHNOLOGY

Department of Information Technology

Signal Processing Laboratory

Peltonen, Vesa: Computational Auditory Scene Recognition

Master of Science Thesis, 64 pages.

Examiners: Prof. Jaakko Astola, M.Sc. Anssi Klapuri, Dr.Tech. Jyri Huopaniemi

Financier: Nokia Research Center

August 2001

Keywords: auditory scene recognition, context awareness, audio classification, computational auditory scene analysis, audio content analysis

An acoustic environment surrounding a listener, an auditory scene, can provide contextual cues that enable the recognition of the scene. This thesis concerns the problem of *computational auditory scene recognition*, which is a subproblem of c*omputational auditory scene analysis*. Computational auditory scene analysis refers to the computational analysis of an acoustic environment, and the recognition of distinct sound events in it. In this study, the focus is not in analyzing and recognizing discrete sound events (although they may be used in the recognition process), but in the classification of acoustic environments as whole.

This thesis covers all the different phases of a study that was made at the Signal Processing Laboratory of Tampere University of Technology: a literature review on auditory scene recognition and related fields of research, acoustic measurements that were made in a number of everyday auditory environments, design and implementation of the audio database access software, a listening test examining human abilities in auditory scene recognition, audio signal classification theory, algorithm development and simulations.

The core of this thesis is in the computational audio classification and signal processing algorithm development part. Auditory scene recognition involves correct grouping of similar environments, feature selection and extraction, and the use of a suitable classification algorithm. A crucial step in solving the problem is to determine appropriate features that can discriminate between the acoustic data associated with pre-defined scene classes.

The conducted listening tests show that, on average, humans are able to recognize 25 different scenes with 70 % accuracy. The scenes included everyday outside and inside environments, such as streets, market places, restaurants, and family homes. The performance of the computational classification methods was investigated by conducting Matlab simulations. The best obtained recognition rate for 13 different scenes was 56%, where the classified scenes were selected so that from each one there were at least three recordings from different locations. We also did an experiment of recognizing more general classes (meta-classes), and for certain categorizations of the scenes we obtained relatively good classification results. For example, the meta-class car vs. other was classified correctly in 95% of the cases.

List of acronyms and symbols

ASR	Automatic Speech Recognition
BW	Bandwidth
CASA	Computational Auditory Scene Analysis
CASR	Computational Auditory Scene Recognition
DAT	Digital Audio Tape
DFT	Discrete Fourier Transform
GMM	Gaussian Mixture Model
GSM	Global System for Mobile communication
HMM	Hidden Markov Model
kNN	k-Nearest-Neighbor classifier
LPC	Linear Prediction Coefficients
MAP	Maximum a Posteriori
MFCC	Mel-Frequency Cepstral Coefficients
MPEG	Moving Picture Experts Group
pdf	probability density function
SC	Spectral Centroid
SF	Spectral Flux (Delta Spectrum Magnitude)
SR	Spectral Roll-off point
std	standard deviation
ZCR	Zero-Crossing Rate

$\boldsymbol{x}$	Vector $\boldsymbol{x}$
$\boldsymbol{x}^T$	Transpose of $\boldsymbol{x}$
X	Discrete Fourier transform of $\boldsymbol{x}$
$\bar{\mu}$	Mean vector of $\boldsymbol{x}$
Σ	Covariance matrix
Σ^{-1}	Inverse of covariance matrix
F	Fourier transform
F^{-1}	Inverse Fourier transform
f_s	Sampling frequency
$\boldsymbol{J}$	Cost function
Δx	Derivative of x

1 Introduction

Imagine that you are listening to an audio recording, in which you can hear cars passing by, people walking, somebody talking in the background, and you may even feel a gust of wind blowing into microphones. If you were asked about the location of the recording, you would probably answer that the recording is from a street. In the described situation you would be performing the task of auditory scene recognition. The goal of this thesis is to develop methods that enable a computer to do the same. We refer to this problem as c*omputational auditory scene recognition* (CASR). A scene is also referred to here as environment or a context.

Computational auditory scene *analysis* (CASA) has been an active area of research in the last few years [Ellis96, Brown94, Mellinger91]. It refers to the computational analysis of an acoustic environment, and the recognition of distinct sound sources and events in it. Interpretation of individual events is the point which differentiates the scene recognition problem from scene analysis. Although humans often base their environment recognition on familiar sound events, the CASR task does not necessarily involve analysis down to the level of individual events, even though they may be utilized in the recognition process.

The CASR problem can be approached from the point of view of pattern recognition theory. The basic concept in pattern recognition is to identify objects (scenes in this case) by using some attributes (features) of the objects. The most critical step in designing audio recognition systems is the selection of a representative set of features that are capable of distinguishing between the classes. Figure 1 presents a block diagram of a basic sound classification system, consisting of preprocessing, feature extraction which is often called data reduction, pattern learning, and classification stages.

1.1 Applications

Practical applications of CASR include intelligent wearable devices and hearing aids that sense the environment of their users at a given time. The information about the environment enables the device to provide better service to users' needs. A device can adjust the mode of operation according to the context; for example, a GSM phone may know not to ring when the user is at a lecture or in a meeting, and in contrast, it may ring louder when the user is in a noisy place, e.g. a street. In addition to enhancing the interface, a device may also modify processing parameters according to the context. For example, a speech codec performance can be enhanced by modifying processing parameters according to the background noise, e.g. a silent place or a noisy car.

Some modern hearing aids have a collection of equalization filters for various situations. The switch between the different filters is done manually by the user. A CASR system could

recognize the environment and apply appropriate settings to the hearing aid automatically.

CASR techniques can be also utilized in content-based audio indexing and retrieval at the level of different environments. One of the possible applications is video scene classification using audio tracks [Liu98].

Information about the context can also be utilized to control more specific analysis systems, such as computational auditory scene analysis. The analysis of a scene is easier in such cases where we have high-level information about the scene itself. In other words, if we know the rough category of the scene, we can predict what kind of sound events are likely to occur in that scene.

1.2 State of the art

Audio-based scene recognition has been studied very little compared to speech recognition, for example. However, there are many research fields that are related to CASR and have been studied to a varying extent. The related fields comprise different audio classification problems, such as speech/music discrimination, noise classification, and content-based audio retrieval. Chapter 2 is dedicated to a literature review on CASR and on the related fields of interest.

We managed to find only a few classification schemes that have been proposed to recognize auditory scenes. In a framework suggested by Clarkson, Sawhney, and Pentland [Clarkson98a], an auditory scene is recognized by classifying a temporal sequence of detected and identified sound events. The proposed system was experimented only preliminarily, and is not a complete CASR system as such. Another system based on sound event classification was

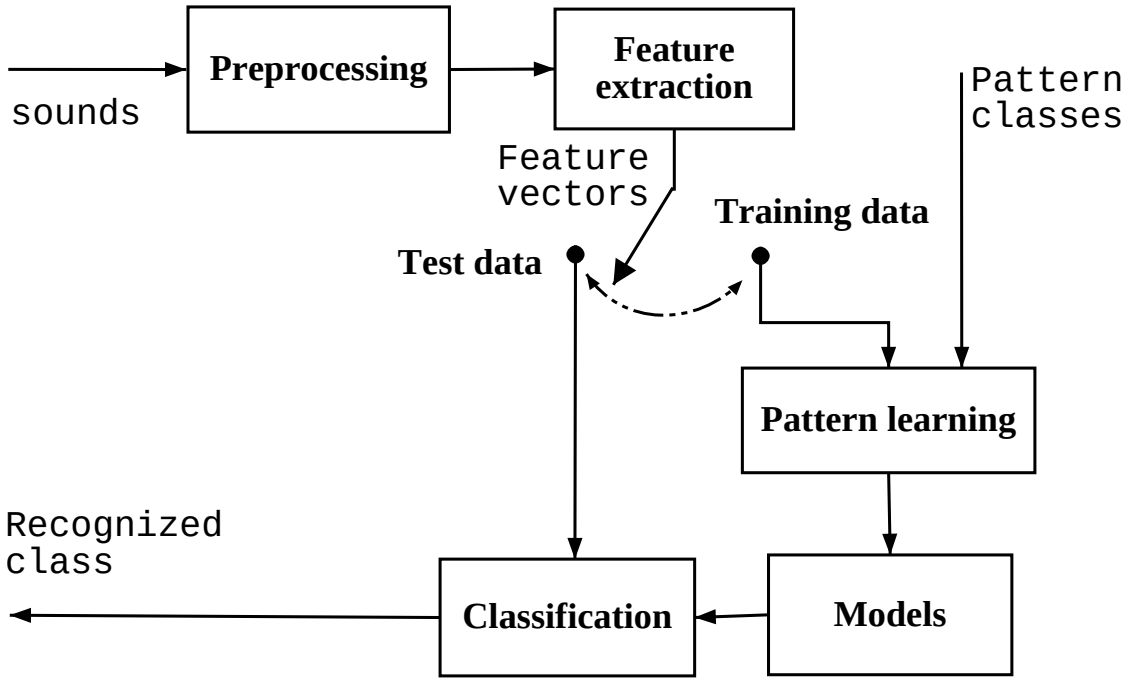

Figure 1. Basic structure of a sound classification system

proposed by Saint-Arnaud and his colleagues [Saint95]. In their system, a sound texture is recognized by modeling short-term and long-term auditory events. With respect to the potential information content, a sound texture refers to an intermediate form between noise and an auditory scene.

As far as we know, the problem of CASR is not even close to be solved yet. We can even state that the research in this field is in its initial phase, and there are a lot of subproblems to research on and to experiement with. A question that arises at this point is: is it meaningful to base scene recognition on acoustic information? One of the motivations in recognizing contexts using audio is to utilize existing sensors in mobile devices, i.e. microphones. There are ongoing studies concerning context awareness using different low-level sensors [Laerhoven99]. However, these solutions require additional hardware. Secondly, humans can fairly well recognize environments by listening only (a listening test is described in Chapter 5). Thus, it is natural to *try* to understand how humans solve these sorts of classification problems, and to attempt to implement computational models which imitate the human auditory system.

1.3 Scope and outline of this thesis

The main objective of this study is to develop a computational model capable of recognizing real-world auditory scenes. The particular objectives that we set for this thesis, and which were achieved to a greater or lesser extent, were to:

1. Give an overview of the CASR field.
2. Gather a representative audio database from everyday auditory environments.
3. Evaluate the human capabilities in recognizing auditory scenes.
4. Apply and evaluate existing pattern recognition techniques in solving the CASR problem.

This thesis is organized into 8 chapters as follows. After the introduction in this chapter, Chapter 2 provides a literature review on CASR and on related research areas. Chapter 3 is dedicated to description of the audio measurements concerning everyday auditory environments. Also the annotation methods of the audio data are presented. In Chapter 4, the implementation of the audio database access software is shortly described. Chapter 5 presents a psychoacoustic listening test studying human abilities in recognizing everyday acoustic environments and draws conclusions from the test. The core of this thesis is in the next two chapters, which are dedicated to the description of the implemented CASR system and to experimental results. Chapter 6 reviews the studied acoustic features and the classification techniques for CASR and Chapter 7 presents the simulations and the results for several approaches. Finally, Chapter 8 summarizes this thesis and discusses future directions for the research.

2 Literature review

The purpose of this chapter is to present an overview of the work done in the field of computational auditory scene recognition (CASR) and in related areas of research. The related fields discussed here are context awareness, computational auditory scene analysis (CASA), and different audio signal classification problems, such as speech/music discrimination, noise classification, sound source recognition, and speech recognition. The mathematical definitions and descriptions of the acoustic features and classifiers mentioned in this chapter are given in Chapter 6.

2.1 CASR

In a project named "*Situational Awareness from Environmental Sounds*" [Sawhney97], Nitin Sawhney investigated techniques that enable machines to recognize a context by extracting and classifying features from environmental sounds. More than three hours of audio material was recorded from different locations. Sawhney analyzed three different classification schemes in recognizing five signal classes. The classes were labeled as people, subway, traffic, voice and other. The classification schemes and the overall recognition rates are listed in Table 1. For the best classification scheme, recognition rates ranged from 40 % (people) to 100 % (voice) for individual classes. In Table 1, the feature vector of the third classification scheme consisted of 100 times subsampled outputs of a Gammatone filter bank.

Table 1: Classification schemes evaluated in [Sawhney97]

Feature	Classifier	Recognition rate (five classes)
Relative spectral (RASTA) coefficients	Recurrent Neural Network	24 %
Power spectral density	k nearest neighbor (kNN)	53 %
Subsampled filterbank output	k nearest neighbors (kNN)	68 %

In 1998, Brian Clarkson and his colleagues proposed a system for extracting context from environmental audio [Clarkson98b]. The classification scheme consisted of the normalized power spectrum as a feature, and hidden Markov models as a classifier. The system was integrated with a wearable messaging system (Nomadic Radio) and used to determine the user's availability by detecting conversation level around the user [Sawhney98]. In the evaluations, the system was proved to reliably detect speech around the user and operated in real-time on an average personal computer.

Audio analysis and recognition techniques have been used also in video scene classification. In 1997, Zhu Liu and his colleagues proposed a method for video scene classification [Liu97]. The system utilized several features extracted from the audio track, and the feature vectors were classified using a neural network. The features explored included volume distribution features, pitch contour, and a couple of spectral features. The volume distribution features consisted of the mean and the standard deviation (std) of the volume within a audio clip, silence ratio and the dynamic range of volume. The dynamic range of volume was defined as $vdr = (max(v) - min(v))/max(v)$, where *max(x)* and *min(v)* are the maximum and the minimum of the volume within a audio clip. The explored spectral features were the spectral centroid, the bandwidth and subband energy ratio. The classification framework was tested in separating audio clips of five different TV programs (advertisement, basketball games, football games, news reports, and weather reports). The average recognition rate for these classes was 88 %.

2.2 Computational auditory scene analysis

Auditory scene analysis refers to the process in human auditory psychology by which we organize the sounds of a complex environment. An excellent source of information in this field is Albert Bregman's comprehensive book "Auditory Scene Analysis - the Perceptual Organization of Sound" [Bregman90]. In the book, Bregman studies the processes in our brain that determine how we hear sounds, how we differentiate between sounds, and how we perceive our auditory environment by grouping sounds into objects. Bregman suggests that there are many phenomena going on in the auditory perception that are similar to those in visual perception, such as *exclusive allocation* (properties belonging to one event only) and *apparent motion* (a motion is perceived, although the stimulus is not moving).

Computational auditory scene analysis (CASA) is a computational approach to auditory scene analysis. CASA deals with automatic analysis of an acoustic environment, the interpretation of discrete sounds events in it, and modelling the sound components. CASA systems are based on, or at least inspired by, psychoacoustic theories of perception. Dan Ellis has contributed a lot to the research on CASA. In his Ph.D. thesis "*Prediction-driven computational auditory scene analysis*" [Ellis96], he presents an approach, in which the analysis is done by matching the predictions of an internal world model and observed acoustic features. The implemented computational model is not a perfect CASA system, but however, there was a good agreement between the events detected by the model and by human listeners. A representative review of recent work on CASA can be found in a book edited by Rosenthal and Okuno [Rosenthal98]. Earlier work on CASA by David Mellinger was more focused on psychoacoustics and psychophysical theories than on implementing a complete computer model for auditory scene analysis [Mellinger91].

2.3 Speech/Music discriminators and classification of general audio data

Speech and music are the two most common and important types of audio signals. Therefore, one of the basic problems in audio classification is to discriminate between speech and music. A lot of work has been done in this field. The investigated classification techniques differ from each other with regard to the set of extracted audio features and to the classifiers used. A set of

Table 2: Features and classifiers used is some speech/music and general audio classifiers

Article	Features	Classifiers	Accuracy
"*Real-Time Discrimination of Broadcast Speech/Music*" [Saunders96]	• Energy contour dip • Four statistical features based on ZCR	Gaussian	98 %
"*Automatic transcription of general audio data: preliminary analyses*" [Spina96]	• MFCC	MAP	94 %
"*Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator*" [Scheirer97]	4Hz Low-Enrg. Roll-Off SC Flux ZCR CRRM Pulse Δ Roll-Off Δ SC Δ Flux Δ ZCR Δ CRRM	MAP GMM k-NN k-d trees	94.0 % 94.4 % 94.7 % 94.3 %
"*A Comparison of Features for Speech/Music Discrimination*" [Carey99]	• MFCC • Amplitude • Pitch • ZCR • Δ MFCC • Δ Amplitude • Δ Pitch • Δ ZCR	GMM	98.8 % with MFCC
"*Speech/Music Discriminator Based on Posterior Probability Features*" [Williams99]	• Entropy • Avg. probability dynamism • Background energy ratio • Phone distribution match	HMM	98.6 %
"*Towards robust features for classifying audio in the Cue-Video system*" [Sriniva99]	• Energy • ZCR • Spectral energy in 4 bands • Harmonic frequencies	Threshold	80 %
"*Hierarchical Classification of Audio Data for Archiving and Retrieving*" [Zhang99]	• Energy • ZCR • Fundamental frequency	HMM	90 % (3 classes)
"*A Fast Audio Classification from MPEG Coded Data*" [Nakajima99]	• Temporal energy density • Bandwidth • Subband centroid	Gaussian	90 % (3 classes)
"*Speech/music discrimination for multimedia applications*" [El-Maleh00]	• LSF • LP-ZCR • HOC • Δ LSF	QGC kNN	95.9 %
"*An investigation of automatic audio classification and segmentation*" [Guojun00]	• Silence Ratio • ZCR	Threshold	87.7 % 75.5 %
"*Using the Fisher kernel method for Web audio classification*" [Moreno00]	• MFCC • Δ MFCC	SVM	81.8 %

Table 3: The acronyms used in Table 2

4Hz	Four Hertz modulation energy	LSF	Line Spectral Frequency
CRRM	Cepstrum resynthesis residual magnitude	MAP	MAP Gaussian estimator
Flux	Spectral Flux	MFCC	Mel Frequency Cepstral coefficients
Gaussian	Multivariate-Gaussian classifier	Pulse	Pulse Metric
GMM	Gaussian Mixture Models	QGC	Quadratic Gaussian Classifier
HOC	Higher Order Crossing (ZCR of filtered signal)	Roll-Off	Roll-Off frequency
k-NN	k-Nearest Neighbor classifier	SC	Spectral Centroid
Low-Enrg.	Percentage of low-energy frames	SVM	Support Vector Machine
LP-ZCR	Linear Prediction Zero Crossing Rate	ZCR	Zero Crossing Rate

recently proposed speech/music discrimination and general audio classification schemes are summarized in Table 2. The recognition rates listed in the table are not comparable as such, due to the lack of a common test bench, differences in the duration of the classified audio excerpts (frame-based vs. clip-based), and additional sound classes in some schemes (e.g. applause). The meanings of the acronyms used in the table are given in Table 3.

Scheirer and Slaney [Scheirer97] compared several classification strategies and found that there is not much difference between the performances of different classifiers. For four different classifiers the accuracy ranged between 94.0 % and 94.4 %. Thus, they concluded that the selection of the features is actually much more critical to the classification performance than the used classifier.

Speech and music have quite different temporal characteristics and probability density functions (pdf). Therefore, it is not very difficult to reach a relatively high discrimination accuracy, as can be noticed from Table 2, only a few approaches reported discrimination accuracy below 90 %. Figure 2 illustrates the difference between speech and music: the waveforms and sample histograms of speech and music signals are plotted side-by-side. The speech signal is a fragment of a Finnish sentence spoken by a female, and the music signal is two second excerpt of classical music (J. S. Bach). The histograms are fairly good approximations of the pdf of the signal samples.

The problem of speech/music discrimination is practically solved and the recent research interest is turning towards classification of general audio data. General audio data may include any auditory class and it is not restricted only to speech and music. Li and his colleagues recently proposed a classification scheme for general audio data [Li01]. Their system is able to classify audio into seven categories consisting of silence, single speaker speech, music, environmental noise, multiple speakers' speech, simultaneous speech and music, and speech and noise. A total of 143 features were studied, and it was found that cepstral based features outperformed the temporal and spectral features. The best classification accuracy (85.3 %) was

obtained using 20 Mel-frequency cepstral coefficients and a Bayesian classifier. Also a segmentation-pooling scheme was proposed in the paper, which tries to detect transitions from one type of audio to another type, and with the help of this to reduce the loss in accuracy at the class borders. Li and his colleagues reported that the pooling-scheme increased the accuracy about 5 % ending up with the accuracy of 90.1 %.

Zhang and Kuo proposed a content-based audio classification system to be used in audiovisual data parsing in their research monograph [Zhang00]. The audio data is classified hierarchically at two levels. In the first, coarse-level, the data is segmented into basic audio types consisting of silence and audio with or without music components. The segmentation is based on statistical analysis of the energy function, average zero-crossing rate, fundamental frequency and spectral peak tracks. The monograph included detailed description of the use of various features. The classification itself is done using heuristic threshold rules. In the second level, the

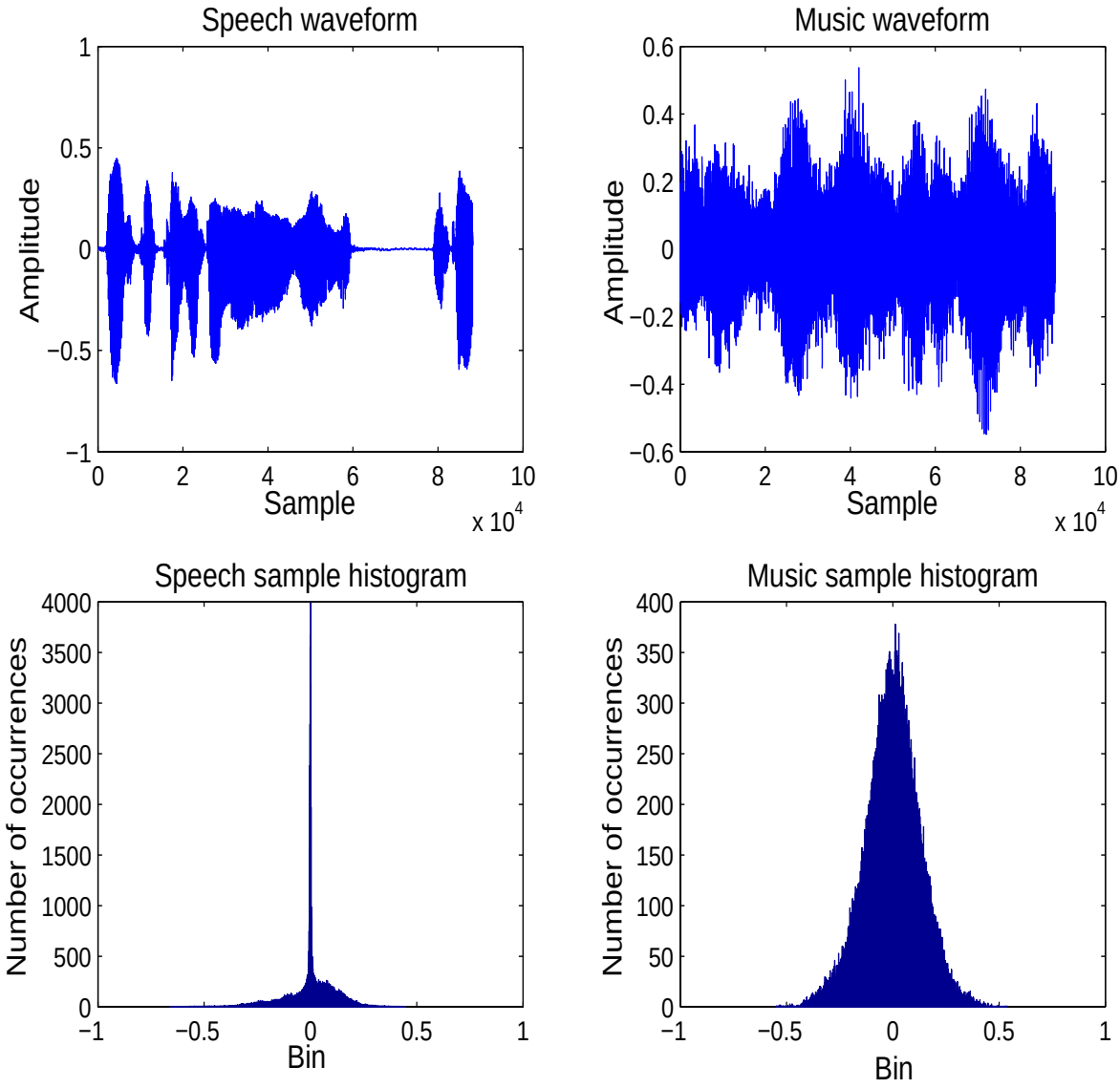

Figure 2. The waveforms and sample histograms of speech and music samples. The speech sample is a fragment of a Finnish sentence spoken by a female, and the music sample is a two second excerpt of classical music (J. S. Bach).

coarse-level categories are further classified into finer classes using hidden Markov models (excluding the silence class). These classes are harmonic environmental sound, non-harmonic environmental sound, environmental sound with music, pure music, song, speech with music, and pure speech. The overall recognition rate obtained for these eight classes was 90.7%. In recognizing sound effects within a query-by-example scheme, Zhang and Kuo obtained a performance of 86% for 18 classes. The sound effect classes were such us applause, footstep, explosion and raining. Query-by-example refers to the retrieval of sound effects that are similar to given example sound. In the research, Zhang and Kuo also worked on the analysis of the visual part of audiovisual data.

2.4 Sound source and sound effect classification

Many systems have been proposed for the recognition of sound effects and sound sources, such as human speakers, environmental sounds, and musical instruments. However, within the scope of this thesis, only few systems are reviewed.

In the 1999, Martin developed a computer system that is capable of recognizing musical instruments among 25 possibilities [Martin99]. Martin studied various different features, such as pitch, spectral centroid, onset duration, vibrato frequency, slope of onset, and others. The investigated classifier was an enhanced Bayesian belief network. The recognition accuracy was comparable to other proposed computer systems, but was not as good as the recognition performance of musically skilled human subjects, especially in instrument family classification.

For more approaches on musical instrument recognition see [Brown99, Eronen01, Fraser99, Fujinaga00, Kostek99, and Martin98]. Herrera and his colleagues have reviewed in the different techniques proposed for musical instrument classification [Herrera00].

The Sound Understanding Testbed is a system for recognizing specific household and environmental sounds suggested by Klassner in 1996 [Klassner96]. The system is based on a blackboard architecture, which implements two interacting search processes. The first searches for an explanation for a signal, the second looks for an appropriate front-end configuration for analyzing it. The system had a library of 40 sounds from which the test material was constructed by placing four independent sounds on a five-second recording, after which the systems had to identify which sounds occurred and when. The recognition accuracy was reported to be around 60 %.

Utilizing acoustic information for vehicle recognition has been also studied in papers “Mobile object recognition based on acoustic information” [Jarnicki98] and “Vehicle sound signature recognition by frequency vector principal component analysis” [Wu98].

2.5 Noise classification

In a paper entitled “*Automatic Classification of Environmental Noise Events by Hidden Markov Models*” [Gaunard98], Paul Gaunard and his colleagues used hidden Markov models to recognize five types of environmental noise events (car, truck, moped, aircraft and train). The best recognition rate (95.3 %) was obtained using a five-state hidden Markov model as a classifier and linear prediction coefficient (LPC) based cepstral coefficients as features. The

optimal LPC analysis order was 10, the same as often used in speech recognition. The paper also describes a listening test which showed that on the average, humans can recognize these noise classes in 91.8 % of the cases, given the possible categories.

Another similar noise classification scheme was introduced by El-Maleh and his colleagues in 1999 [El-Maleh99]. The sound classes considered where car, street, babble, factory and bus. Among the different LPC-based features experimented in the work, the best discrimination between the classes was obtained using the line spectral frequencies (LSF's). A quadratic Gaussian classifier outperformed the other classifiers, achieving 89 % recognition rate.

In [Saint95], Nicolas Saint-Arnaud and Kris Popat presented a cluster-based model to characterize and resynthesize *sound texture*s. A sound texture is defined as a two level phenomenon: simple sound elements at a low-level and their distribution and arrangement at a higher level. The difference between noise, speech, music, and a sound texture is illustrated in Figure 3. Sound of rain or babble are examples of sound textures. The resynthesized sound textures were in some cases perceptually similar to the original sound. The model can be used also as a sound texture classification system, as proposed by Saint-Arnaud and Popat.

2.6 Context awareness using multi-modal sensors

In the past few years a lot of work has been done in the field of context awareness in general. The target of this research area is to enable mobile computing and communication devices to be aware of their context and to change their behavior according to this information providing better services to the user. The term "awareness" refers to the knowledge of the minute-to-minute context in which the device is used. Some of the context awareness research has been restricted to geographical location awareness only. These systems are based either on global positioning systems (GPS) or on radio beacons. Beacons can be, for example, base stations of

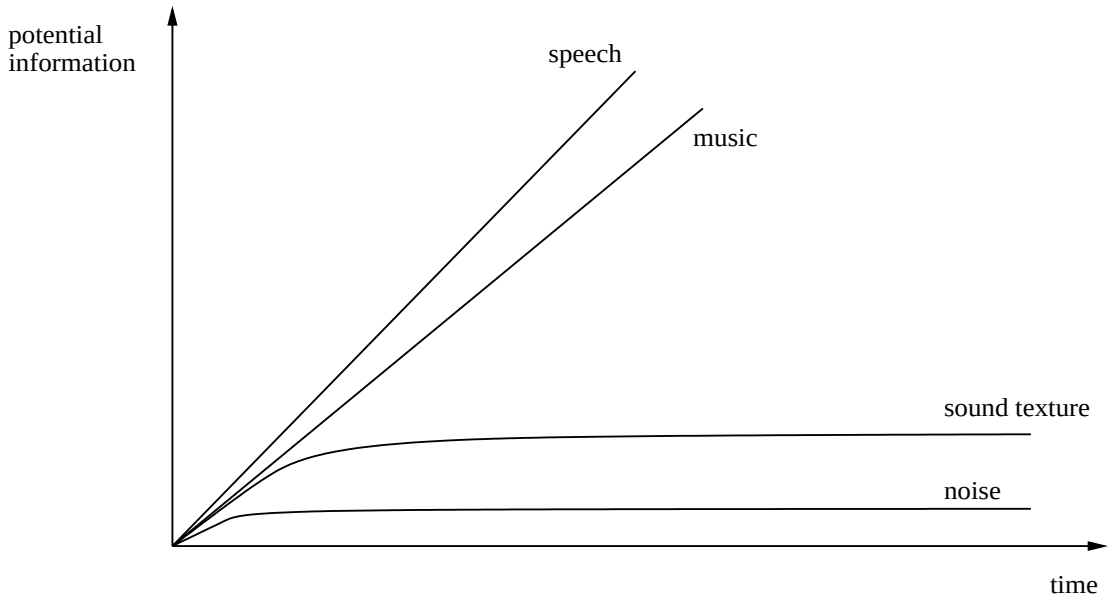

Figure 3. Constant long-term characteristics of sound textures and noise as described by Saint & Popat. Reprinted by permission.

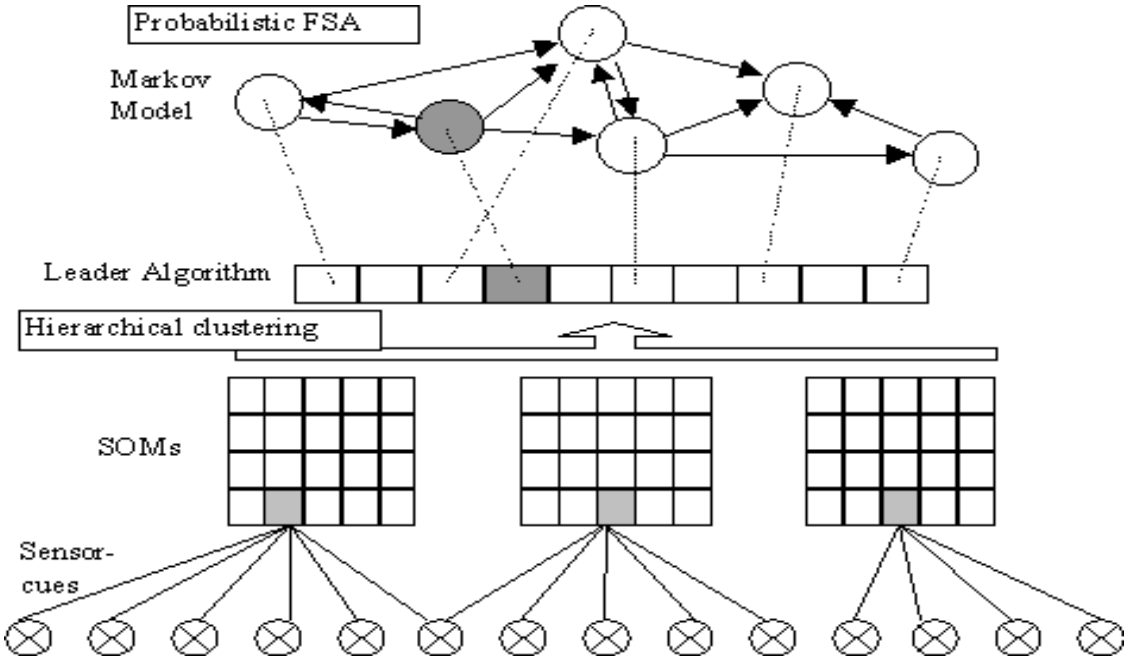

Figure 4. The architecture of the context awareness system of the TEA project. Reprinted by permission from [Laerhoven99].

cellular network or any network of transmitters.

The objective of TEA project (Technology for Enabling Awareness), described in the thesis of Van Laerhoven [Laerhoven99], was to recognize context at a higher level of abstraction, such as “walking outside”. The system described is based on low-level sensors, such as accelerometers, photodiodes, temperature sensors, touch sensors, microphones, and some other sensors as well. The raw-data from these sensors is further processed by feature extraction and clustering, and classification algorithms are used to accomplish the context recognition. The data is first clustered with Kohonen Self-Organizing Maps and then classified with hidden Markov models. The architecture of the system is described in Figure 4 as presented in [Laerhoven99].

To summarize this Chapter, the actual CASR problem has been studied very little. However, there exist many related audio signal classification problems, which have been studied to a varying extent. Although the feature extraction and classification algorithms used depend on the classification task, the reviewed classification schemes provide some directions for the construction of a CASR system.

3 Acoustic measurements

An audio database was collected, which contains recordings from a variety of different auditory contexts. The recordings were made in everyday domestic and business environments, including e.g. family homes, vehicles, business buildings, and different outside environments such as streets, market places and roads (for a list of the scenes, see Table 4). The target of these measurements was to gather real data which is suitable for a subjective listening test (Chapter 5) and for the development of a CASR system (Chapter 7).

The measurements were made using two different recording setups. The first setup consisted of an 8-channel recording configuration including binaural, stereo and B-format setups. These are explained in detail later. The first setup was difficult to transfer: it was not movable by one person, and an electric line supply was needed to operate it. Therefore, we constructed a more compact and mobile setup by which a large volume of contexts could be recorded swiftly. The second setup consisted of a two-channel stereo recording equipment only and could be operated without an electric line supply.

3.1 Equipment

Setup I: 8-channel recording

The first recording configuration consisted of a binaural setup (2 channels), stereo setup (2 channels), and the B-format setup (4 channels). A short description of each of these is given below. The setup was originally developed by Zacharov and Koivuniemi [Zacharov01].

- **The binaural setup** included a standard Brüel & Kjær 4128 head and torso simulator with an associated Brüel & Kjær Nexus amplifier. The head and torso simulator was attached to a stand and employed at an approximate height of a standing person. The microphones mounted in the ears of the dummy head enable a realistic reproduction of an auditory scene.
- **The stereo setup** consisted of two omni-directional microphones (AKG C460B), separated by a distance of 1 m. As a voltage source for the microphones, we used a 48 V phantom power supply. The construction was attached to the dummy head, as shown in the sketch presented in Figure 5.
- **The B-format** recordings were done using the MkV SoundField microphone and an associated processor. The B-format is a three dimensional representation of a sound in a certain point of the sound field. The four channels of the B-format consist of the three dimensional components (X,Y, and Z) and a reference channel (W). By preserving these four components of a particular point in the space, we have all the necessary information for reproducing the sound precisely in that point [Streicher98]. The reference component is needed to determine the absolute sound pressure, which can be thought of as a kind of an origin to the directional pressure components.

All the acoustic material was recorded into digital multitrack recorder in 16-bit and 48kHz sampling rate format. The recorders used were Sony PCM-800 and TASCAM DA-78HR. TASCAM can offer 24-bit high resolution recordings, but a 16-bit mode was used to ensure compatibility with the other recordings and the software.

Setup II: stereo recording

The second setup consisted of a stereo recording configuration. The stereo setup was created using the same microphones as in the first setup (AKG C460B) with a 48V phantom power supply. However, in this configuration the distance between the microphones was only 10 cm. The material was recorded with a Sony (TCD-D10) digital audio tape in 16-bit and 48kHz sampling rate format. A picture of the setup is presented in Figure 6.

In the both recording setups, the interconnections between the microphones, the amplifiers, and the recorder were made using balanced wires with XLR connectors. The connection between the recorder and a Silicon Graphics Octane audio workstation was made with an optic fibre.

3.2 Measurements

The first set of recordings was made during the summer 2000 in two middle-sized cities in Finland (Tampere and Helsinki). A total of 7 hours and 46 minutes of audio was captured in 56 recordings from tens of different scenes. The recordings were made in authentic environments, except for a few sessions which were simulated. The simulated sessions were coffee break, office ambience and meeting. In some cases, the measurement itself affected the recorded environment a little due to the visibility of the recording equipment. For example, in a restaurant, the people sitting close to the equipment spoke more quietly than normally.

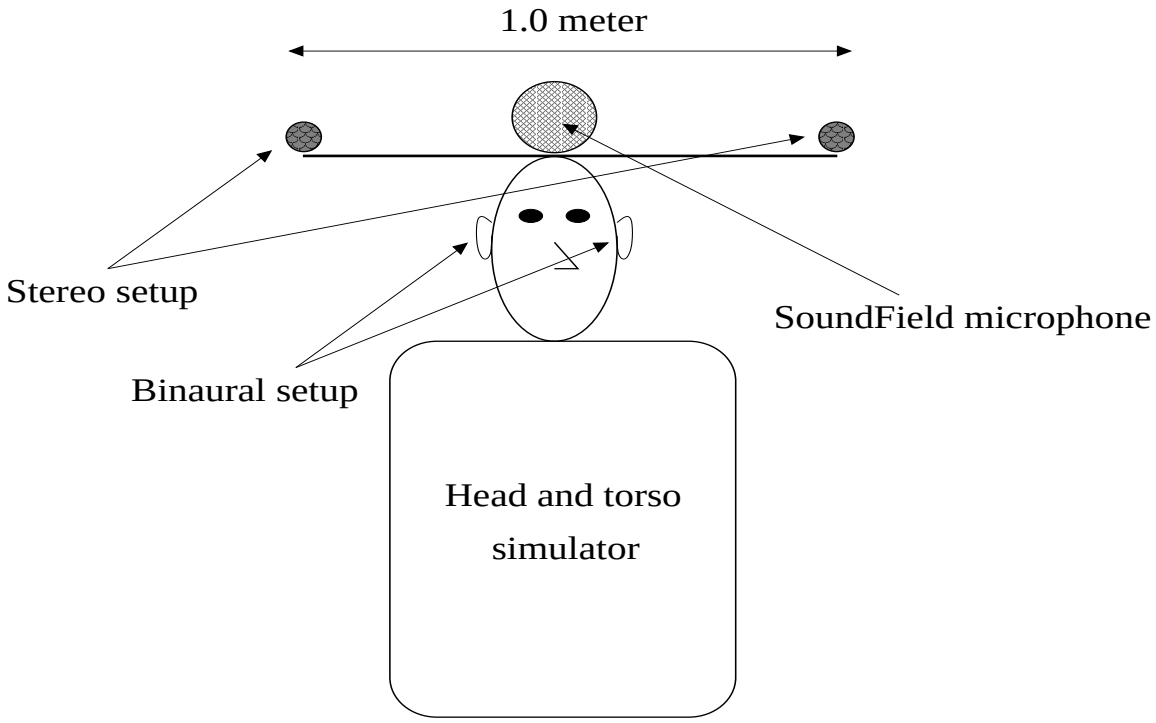

Figure 5. Sketch of the 8-channel recording setup

The second phase of the measurements started in the spring 2001 and at the time of writing this thesis it was still going on. The recorded scenes were the same as in the first stage. Calibration signals were also recorded for each recording session. These were not recorded regularly in the first recording phase. As a sound source for calibration signal we used Brüel & Kjær 4231 acoustical calibrator.

In Table 4 the different scenes and the number of recordings from each are listed. The contexts are categorized at two hierarchical levels: a higher abstraction level and a lower one. The higher level is the rough context, representing the environment itself or a common characteristic of the individual scenes. The classes at the higher level were: outdoors, public, home, music, vehicles, offices, and reverberant spaces. The lower level represents the individual scenes. The lower level classes are such as restaurant, street, and library. The categorization of the individual scenes to main contexts was somewhat ambiguous. A lower level scene can be associated with more than one higher level class. For instance, the scene amusement park has characteristics of the main context public as well as of the context outdoors. However, the scenes in Table 4 are put under one main context only.

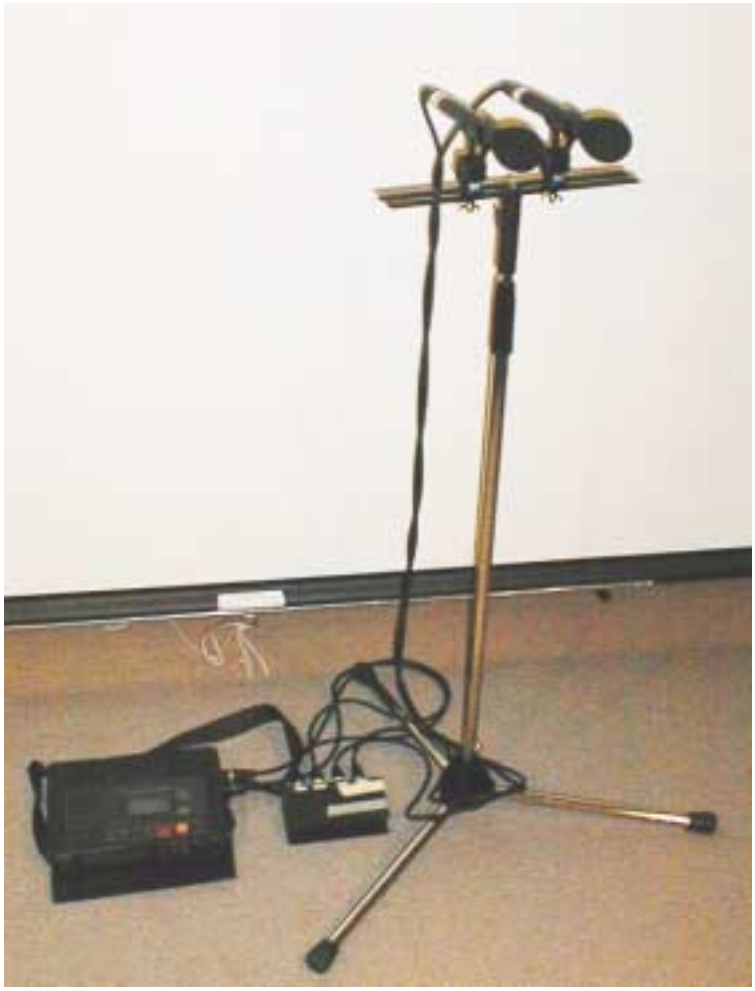

Figure 6. The stereo recording setup

Table 4: List of the recorded auditory scenes divided into main context and the number of recordings form each.

Main context	Scene	1st recording setup	2nd recording setup	Total
Outdoors (37)	Street	3	3	6
	Street/trams	3		3
	Street/under a bridge		1	1
	Railway station/platform		1	1
	Market place	1		1
	Road	2	4	6
	Construction site	1	7	8
	Nature	1	9	10
	Wind	1		1
Public/Social places (24)	Restaurant	2	7	9
	Pub	1		1
	Cafeteria	1	2	3
	Crowds of people/indoors	1	1	2
	Crowds of people/amusement park	1		1
	Lecture pause		1	1
	Coffee break	1		1
	Supermarket, department store	3	3	6
Home (12)	Living room	1	1	2
	Kitchen	1	3	4
	Bathroom	1	5	6
Music (2)	Classical	1		1
	Rock	1		1
Vehicles (20)	Car	17	1	18
	Train	2		2
Offices/meeting rooms/ quiet places (22)	Office ambience	2	6	8
	Meeting		1	1
	Library	1	2	3
	Lecture	1	9	10
Reverberant spaces (7)	Church	4		4
	Railway station/waiting hall	1	1	2
	Metro station	1		1
Total		56	68	124

Table 5: Examples of annotation fields

Annotation field	Comments
Context = Public	#Main context
Scene = Restaurant	#The scene
Location = Hervanta/Tapsantori	
Content = chairs moving @ 21 - 24	#@ indicates time interval
Length = 600	#Length of the recording in seconds
Fs = 48000	#Sampling frequency in Hz
Date = 14.06.2000	
Time = 12:39	
Microphone = AKG	#The microphone used
channel1_gain = 30dB	#The gain in the 1st channel

3.3 Annotations

A structured descriptor, an annotation file, was created for each acoustic measurement. This annotation file describes the measurement process, equipment used and the content itself. The annotated information can be divided into two sections: compulsory and complementary information. Compulsory information comprises such data as the sampling rate, time, and location. Complementary information gives description of the content; incongruous events.

The annotation file consists of annotation fields. Each field represents a single event and occupies one line in the annotation file. The format of the field is the following:

<*label*> = <*value*> [@<*interval*>],

where *label* represents the type of the annotated data, *value* is numeric or verbal description of the data, and the optional parameter is a time interval for which the value is valid. The interval is given at a precision of one second. An absence of interval denotes that this annotation is valid for the entire audio file. Table 5 gives several examples of annotation fields. Free text can also be included in the annotation file as a comment. A comment is denoted with an hash mark (#) at the beginning of each comment line.

4 Database access software

A front-end software was implemented for managing the audio database described in Chapter 3. The interface makes it possible to search the database for any annotated information, such as specific sound events. The query results can be saved and processed later on. The interface is composed of two main programs: *search* and *save*, and hence this chapter is divided into two parts: description of the algorithms in these two main modules. However, we are not going to describe the code-level implementations, but the discussion is kept at the level of data flow and objects. Figure 7 shows the flow of the search and the save processes. The programs were implemented using the C++ programming language and imported into Linux, Unix and Windows operating systems. The user interface is a command-line interface.

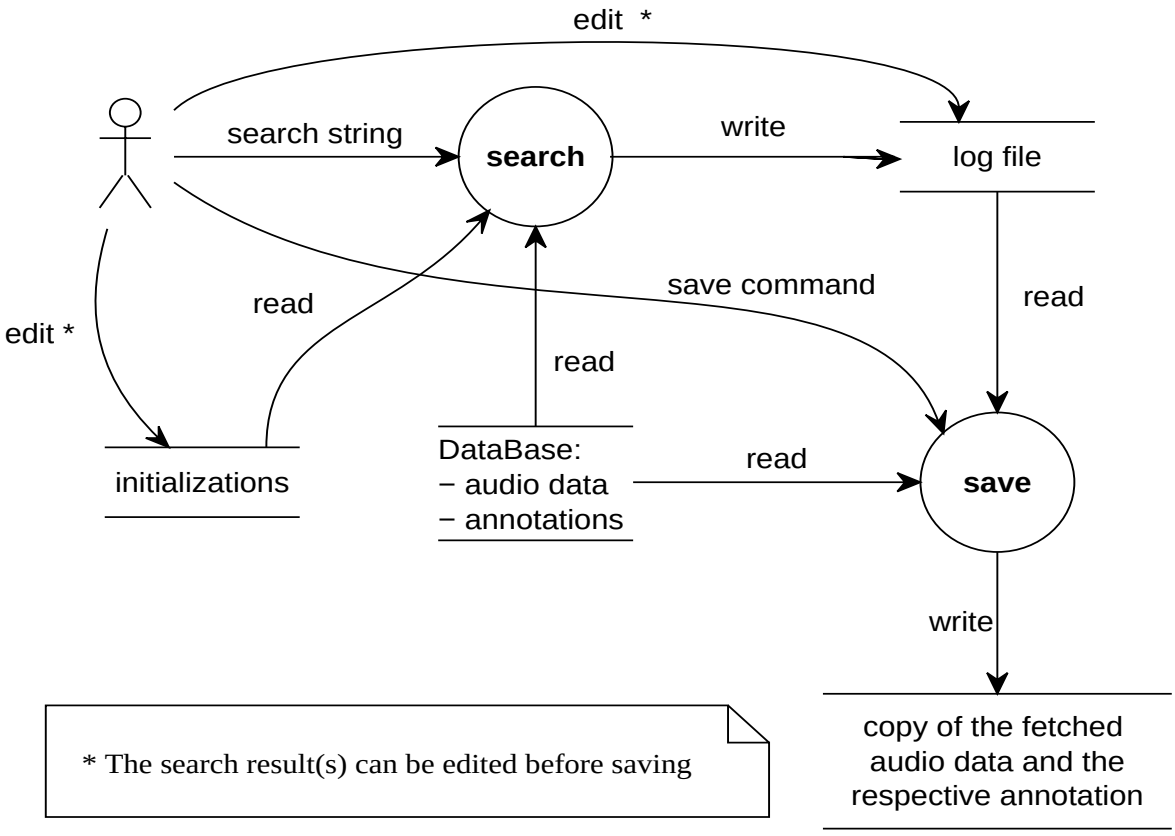

Figure 7. The database management tool: searching and saving

4.1 The search module

The *search* command is used to retrieve audio excerpts that match given search criteria. An example of search criteria, expressed verbally, can be such as: "*find all the scenes containing speech and music concurrently, but only in audio files that are recorded at sampling rate of 48 kHz*". The results of the query are the names of the audio files and the time intervals in which the recordings match the search criteria. The command-line format of the search command is the following:

search <search string>

The target of the search is encoded into the search string. The search string is composed of conditions, which we term *basic atoms*. The format of a basic atom is the following:

< label > < comparator > < value>,

where *label* is a pre-defined field in the annotations, *comparator* is the equality or the inequality sign, and the *value* is the desired value of the given label. In the *label* and *value* fields, an asterisk (*) can replace any substring. A few examples of basic atoms are listed below:

- *content = door**
- *scene = train*
- *fs <= 48000*

These basic atoms can be connected with boolean operands to form extended search strings. The supported operands are: NOT, AND, OR and parentheses. The syntaxes of these operands in the search string are: "!", "&", "|" and "()" correspondingly. The operands are executed in their common preference order, which is: parentheses, NOT, AND, and OR. The number of nested parentheses is not limited, enabling extremely complicated queries. If we convert the verbal search criteria, given as an example above, to a schematic search string, it would look like the following:

(content = speech & music) & (fs = 48000)

Description of the search algorithm

The main steps of the search algorithm are listed below and explained one-by-one in the next paragraph.

Step 1. Preprocessing and validity checks

Step 2. Splitting the search string into basic atoms

Step 3. Calculation of stationary time intervals

Step 4. Boolean value evaluation of each basic atom in stationary time intervals

Step 5. Boolean value evaluation of the whole query string in stationary intervals

Step 6. Combination of true values in adjacent intervals

(1) First, we do some string manipulation to the search string, including, for example, removal of whitespaces and changing upper-case letters to lower-case letters. After that, the legality of the search string is checked. Validity checks include checking the following parameters: number of parentheses, legality of operands (e.g. “&&”, “!&“, “||” are not legal), and checking if the search string is empty. **(2)** If these checks are passed without fatal errors, then the search string is split into basic atoms. **(3)** At this point we know the basic conditions of which the

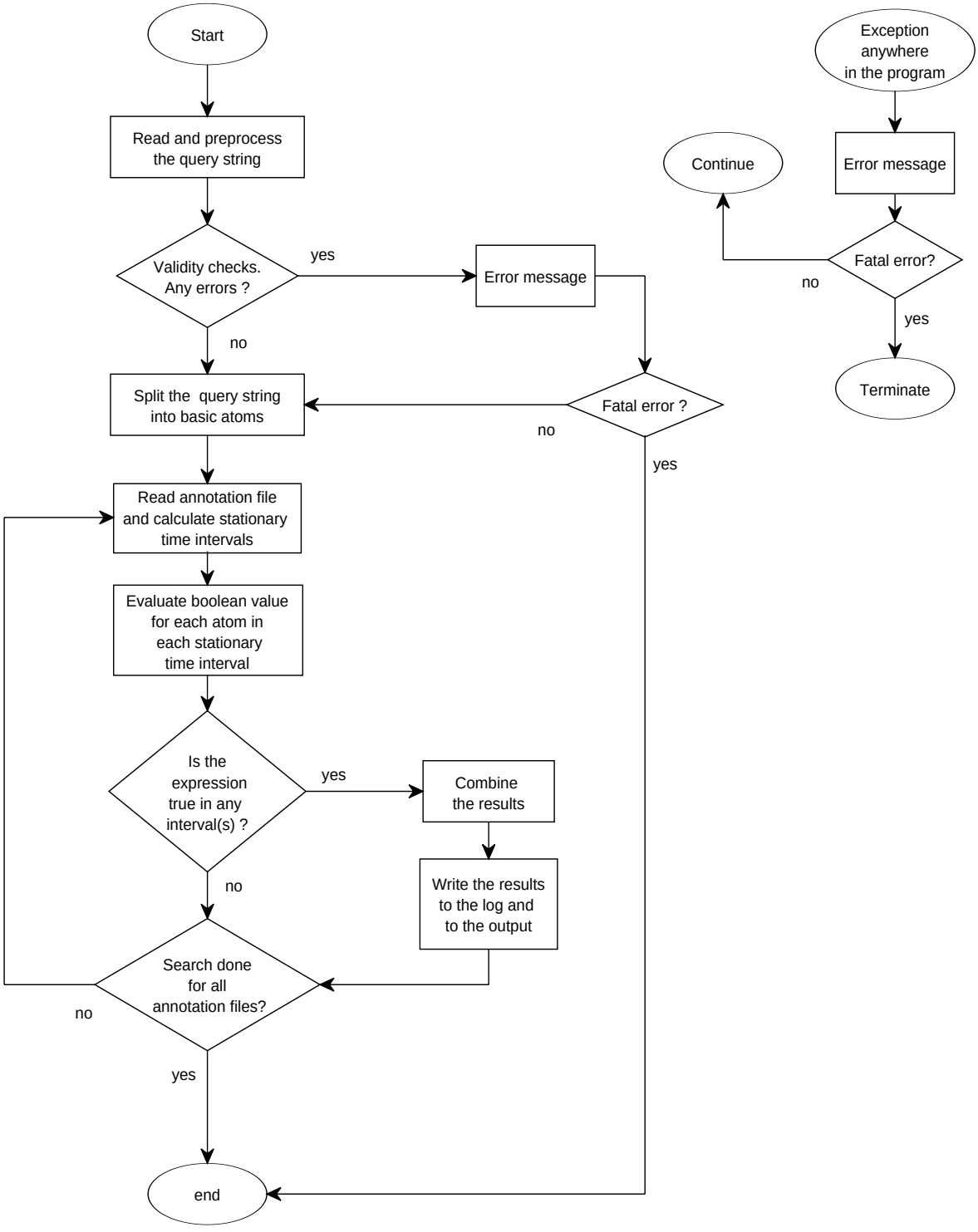

Figure 8. A data flow diagram of the search algorithm

search string consists. Each annotation file is divided into stationary time intervals based on these conditions. Here, a stationary time interval refers to an interval in which all searched basic atoms are constant. **(4)** After that, for each stationary interval, the boolean value of each basic atom is evaluated and the basic atoms in the original search string are replaced with these values. Evaluation of the boolean value at this stage is quite straightforward: the given label is compared to the searched value with the given comparator. **(5)** Next, the boolean value of the whole expression can be evaluated for each interval. The evaluation is done by using the common preference order of the operands. At the end of this stage, we have a true or false value for each interval. **(6)** Finally we combine the results, that is, neighboring true values are combined into one result. Thus, for one annotation file there can be more than one result; each isolated interval is a separate result. A data flow diagram of the algorithm is presented in Figure 8.

Implementation of the search algorithm

The algorithm was implemented using the C++ programming language and utilizing the object-oriented properties of the language. We do not describe the exact implementation of the algorithm in detail, but give only a short description of the main objects and their member functions that construct the kernel of the program. The static diagram of the main objects is illustrated in Figure 9.

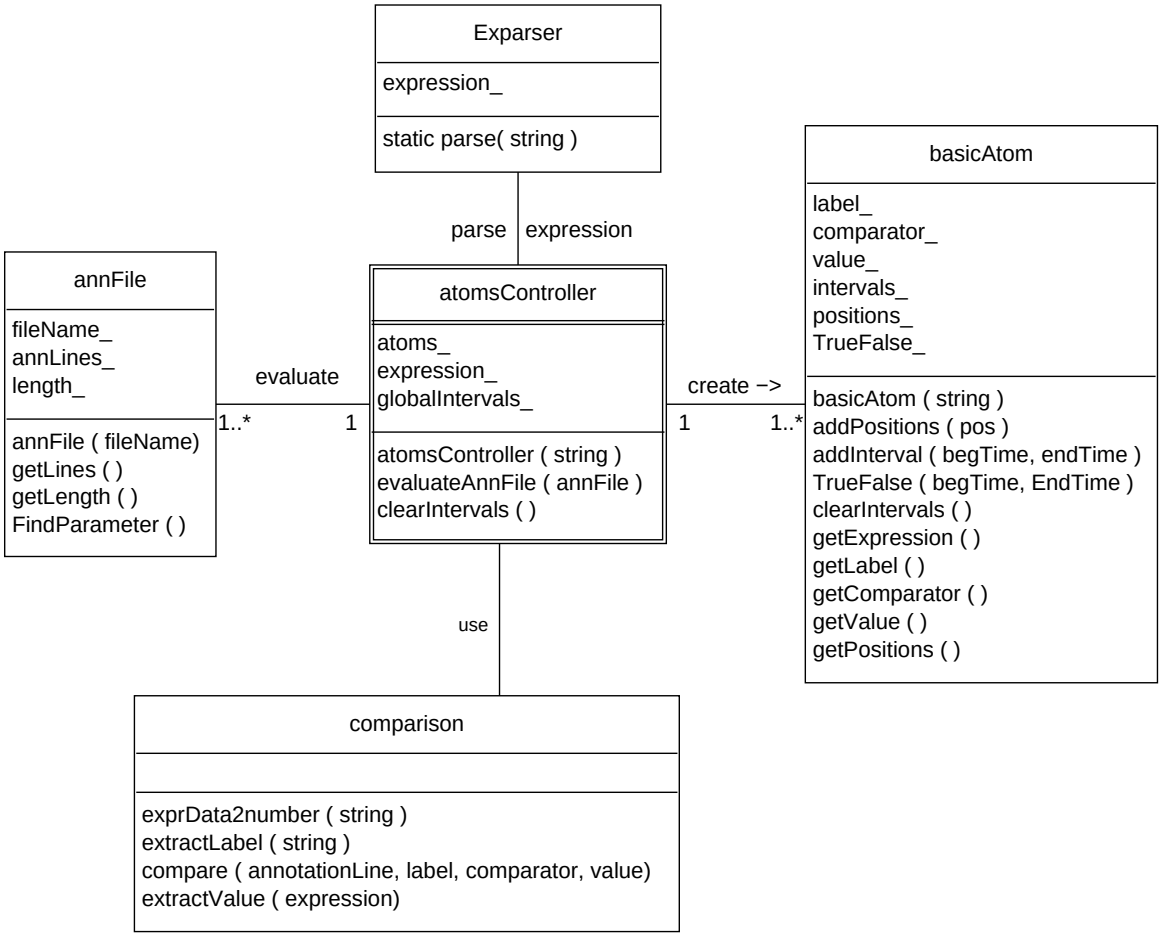

Figure 9. A static structure diagram of the main objects in the search module

The concept "*basic atom*" described in the previous section, is represented with an object named *"basicAtom"*. This object is responsible for all the data and the processing functions associated with an individual basic atom. The *basicAtom* object contains data fields such as the label, comparator, the value of the atom, and other information as well. Member functions are simple data storage and retrieval functions. Each annotation file is represented by an object called "*annFile*". This object is responsible for handling single annotation files, configured with the required data variables and member functions. Two more abstract objects are "*Exparser*" and "*comparison*". The task of "*Exparser*" is to evaluate the boolean value of an expression consisting of true/false values and operands. The piece of work allocated to "*comparison*" is to compare given label and value with a specified comparator. The core of the program is encoded into an object named "*atomsController*", which is responsible for control-

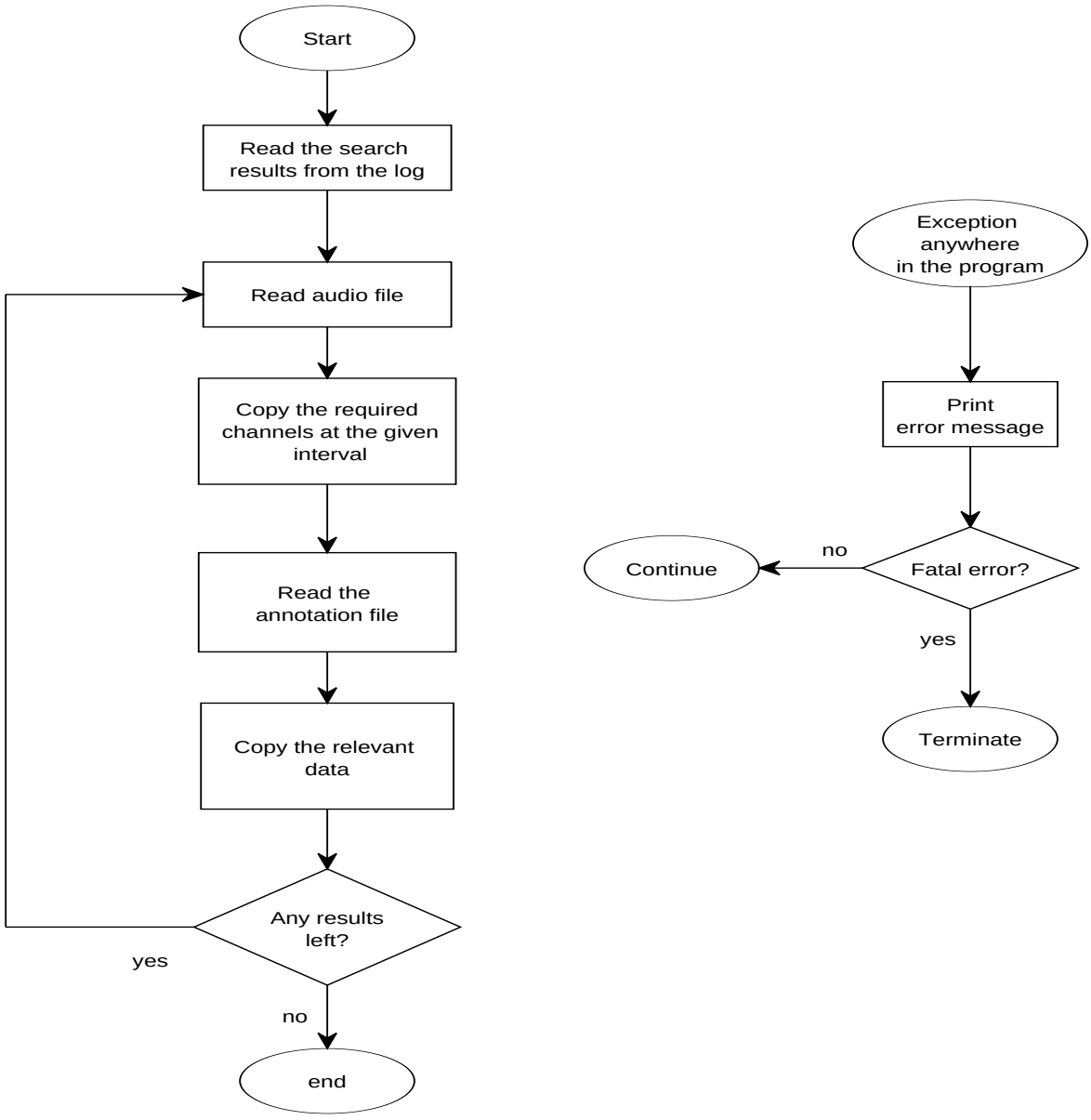

Figure 10. A flow diagram of the save module

ling the program flow and scheduling the tasks to the other objects. Objects not described here are such as string manipulation functions, error handling, initialization etc.

4.2 The save module

The save program is used for making temporary copies of the audio material fetched with the search module. Each search result is copied into an individual file, even though the results would be different time periods of a single audio file. For each saved audio file, a new annotation file is also created. The new annotation file contains only the lines that describe the retrieved time period. The command-line format of the save command is the following:

save 'log file' [*output prefix*] [*path to save*] [*channels to save*]

The log is output of the search module, containing the resulted audio files and the time intervals matching the search criteria.

Description of the save algorithm

The data flow diagram of the save module is presented in Figure 10. The program flow is a quite straightforward technical implementation of the specifications of the audio format. In this implementation, only the WAVE audio format is supported. The results are copied one-by-one, and two main steps are done at each iteration:

Step 1. Copying the queried audio excerpt.

Step 2. Creation of a new annotation file from the existing one.

First, the audio file is opened, and the resulted section is saved into a new audio file. Only the requested channels from the original audio file are copied. However, a channel can be saved more than once, for example, a two channel recording (dummy stereo) can be created from a mono recording by specifying the first channel twice. Then a new annotation file is created by copying and manipulating the data fields in the original file. All the fields of the data that are valid for the whole recording, are copied as such. These field are, for example, the sampling frequency, the context, the microphones used, etc. The rest of the fields are copied only if their occurrence time hits inside the closed time interval of the fetched section. If needed, the beginning and end times are also corrected to suit the length of the new audio file. The beginning and end times are corrected as following: for the beginning time of an event we choose the maximum of beginning of the event and of the fetched audio section. For the finishing time we choose the minimum of the end of the event and the end of the fetched section.

Implementation of the save algorithm

The save module was also implemented using the C++ language. In Figure 11, the static structure diagram of the two main objects is shown. The objects are "*riffFile*" and "*WaveFile*". The object "*riffFile*" represents a file in the format of Microsoft's RIFF (Resource Interchange File Format) specification. The object "*WaveFile"* is inherited from the "*riffFile*", and it represents an audio file of WAVE format. The tasks of the member functions are such as opening and closing a file, reading and writing to a file, seeking specified spot in a file, etc.

Manipulation of the annotation files was implemented by making use of the object "*annFile*" described in the section 4.2, in the paragraph "Implementation of the search algorithm".

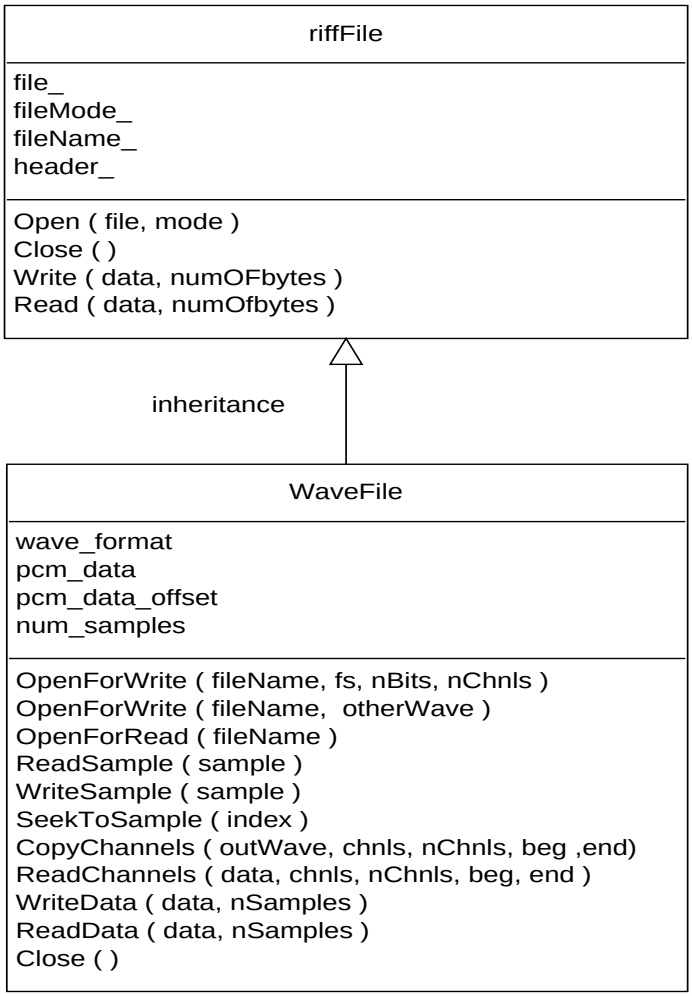

Figure 11. Static structure diagram of the main objects in the save module

5 Psychoacoustic listening test

A listening test was conducted to study the human abilities in recognizing everyday auditory scenes by listening to binaural recordings. The goal of this listening test was to find answers to the following questions:

1. How well are humans able to recognize environments based on binaural recordings only?
2. How fast is this recognition done? (response time)
3. What is the basis for the recognition?

Thus, this chapter is divided into three main parts, each dealing with one of the above questions. The results of the test have been already reported in [Peltonen01].

The motivation for this test was to find a baseline for the performance of a CASR system. Human abilities give a good idea of how accurately and fast an artificial system could potentially operate, and what applications would be realizable. Earlier work in psychoacoustics has mainly concerned human abilities in recognizing single, isolated sound events instead of complex sound mixtures from different environments [Ballas93]. To summarize the results of our listening test, the average correct recognition rate for 19 subjects was 70 % for 25 different scenes, and the average recognition time was 20 seconds. The list of correct scenes was not provided to the subjects. In most cases, the test subjects reported that the recognition was based on prominent identified sound events.

5.1 Experiment method

Stimuli

The stimuli used consisted of binaural recordings selected from the audio database described in Chapter 3. A total of 25 different environments were included in the two-part listening test, and some scenes had multiple instances. In the first experiment, 34 samples of one minute in duration were used and in the second experiment, a subset of 20 samples of these environments was applied. The average duration of the samples in this experiment was three minutes. These samples were picked from the same recordings as those in the first experiment; however, different temporal sections were selected and the samples were presented in a different order. The environments of the stimuli used in the both experiments are listed in Table 6 in the same order in which they appeared in the first experiment. The subset of samples used in the second experiment is highlighted. A total of 92 minutes of binaural data was selected and downsampled to a sampling rate of 44.1 kHz. The levels of the recordings were appropriately normalized before they were written on two audio-CDs.

Table 6: Environments used in the listening test in the order they were presented in the 1st experiment. The environments presented in the 2nd experiment are highlighted.

Scene		Scene	
1.	**Traffic, 80 km/h route**	18.	Street traffic
2.	**Railway station**	19.	**Wind (water tower)**
3.	**Church, concert**	20.	**Traffic, trams passing**
4.	**Car & Radio, 80km/h**	21.	Supermarket
5.	**Train**	22.	Traffic, 80 km/h route
6.	**Street café**	23.	**Market place**
7.	**Subway station**	24.	Restaurant
8.	**Amusement park**	25.	Children playing, home
9.	Car & speech, 40km/h	26.	**Nature**
10.	**Library**	27.	Department store
11.	**Street traffic**	28.	**People indoors**
12.	**Construction site**	29.	**Pub**
13.	**Supermarket**	30.	Traffic, trams passing
14.	**Restaurant**	31.	Office
15.	Presentation	32.	Traffic, trams passing
16.	Car, 40km/h	33.	Replayed music
17.	**Kitchen**	34.	Bathroom

Subjects

A total of 19 normally hearing subjects participated the test, five of which were blind. Blind people were recruited in order to test the hypothesis that they would be more experienced in analyzing environments by listening only. Both male and female subjects between 22 and 58 years of age were involved. All except two subjects were Finns. Some of the subjects were involved in audio engineering, but none had significant experience in listening tests.

Test setup

Most of the tests were conducted in a listening room at Tampere University of Technology, Signal Processing Laboratory. The blind subjects were tested in the premises of Tampereen näkövammaiset ry, (Tampere association for the blind and the visually impaired). A few tests were also performed at the subjects' houses. In the listening room, the amount of background noise and other interfering elements was negligible. The other premises had some occasional background noise sources evident due to e.g. air conditioning, but care was taken to make the locations as silent as possible and to avoid any interfering movements and noise.

A CD player with adjustable volume control and high-quality headphones were used as test equipment. The listening level was adjusted so that the subjects felt comfortable. If desired, the volume was adjusted again during the test. A supervising person controlled the test and wrote down the answers.

Test Procedure

The objective of the first experiment was to find out how accurately and how fast an environment could be recognized. In this part, 34 samples of one minute in duration were used. The subjects were instructed to try to recognize the scene as fast as possible. The subjects were not provided with a list of correct answers, but they were told that the scenes include everyday places, such as vehicles, public and private buildings and outside environments. If the subjects could not give any exact answer, they were asked to tell whether the place was public or private and inside or outside. The answers and the time to give them were written down by a supervising person, letting the subjects to concentrate on the audition. The response times were measured using the display of the CD player at a precision of one second.

Besides studying the recognition accuracy, the second test aimed at discovering the cues used by human listeners in auditory environment recognition. In this test, 20 samples of three minutes in duration were used. The subjects were allowed to make guesses and refine their answers while listening to the sample. In addition to that, they were asked to tell what information they based their guesses on. Again, the answers and the response times were written down by a supervising person. The subjects were allowed to listen to the whole sample if they desired. The subjects were not asked whether they were acquainted with the scenes, assuming that they are familiar with these everyday auditory environments.

5.2 Results

Analysis of the Results

The answers from both tests were processed and the following information was collected for each subject and scene pair:

- The claimed scene
- The response time

Additionally, the cues mentioned by the subjects were collected in the second test. None of the subjects or the answers was discarded in the analysis process.

Accuracy

In Table 7, the confusion matrix for the first test is shown. The columns of the matrix list the presented scenes and the rows describe the subjects' responses. The values in the matrix are recognition percentages. The overall recognition rate for the first test was 66 % and the recognition accuracy for individual environments ranged from 0 % to 100 %.

Table 7: Confusion matrix for the first experiment

Presented / Responded	1. road	2. railway stat.	3. church	4. in car	5. in train	6. street cafe	7. metro stat.	8. amusement p.	9. library	10. street	11. constr. site	12. supermarket	13. restaurant	14. presentation	15. kitchen	16. wind	17. trams	18. market	19. home	20. nature	21. lobby	22. office	23. coffee break	24. music	25. bathroom
1. road	**87**									8															
2. railway stat.		**84**					11			3							11								
3. church			**84**																						
4. in car	3			40						5															
5. in train				7	37												5								
6. street cafe						0																			
7. metro stat.							**53**										3								
8. amusement p.							5	**100**									5								
9. library									0																
10. street	10				11		5			**82**							11								
11. constr. site											**53**						3								
12. supermarket		5				**58**			21			**75**									5				
13. restaurant						21			5				**93**								11		11		
14. presentation														**95**											
15. kitchen															**74**										
16. wind				4												**95**	3								
17. trams					5		5										**37**								
18. market																		47							
19. home						5					5	2		5	11				**58**			5			
20. nature																5		5		**84**					
21. lobby				2					5												**53**				
22. office					5	5			5		11	2										**84**			
23. coffee break																							**84**		
24. music																								32	
25. bathroom															5										**100**
26. concert			16																					**63**	
27. others	0	11	0	**47**	**42**	11	21	0	**64**	2	31	21	7	0	10	0	22	**48**	42	16	31	11	5	5	0

Confusions between scenes were relatively rare. The highest off-diagonal value is reached when a “street café” is recognized as a “supermarket” (58 %). This may be due to the fact that sounds of a cash register and jingling sounds of change are heard clearly in that recording. The erroneous answers were mostly environments, which do not appear in the test set. An extreme case is the scene “replayed music” which was recognized as a concert in 63 % of the cases. This context was a recording of classical music replayed from a CD player in a silent listening room.

Table 8: Confusion matrix for the second experiment

Presented / Responded	1. in train	2. restaurant	3. trams	4. supermarket	5. in car	6. kitchen	7. road	8. lobby	9. street	10. market	11. subway station	12. amusement park	13. nature	14. construction site	15. church	16. library	17. railway station	18. wind	19. street cafe
1. in train	68																		
2. restaurant		95																	5
3. trams			11																
4. supermarket				95												21			
5. in car	5				74				5										
6. kitchen						100													
7. road	5		5				95												
8. lobby		3						74								11			
9. street			84				5		74	16									5
10. market										42									
11. subway station											79	5							
12. amusement park											5	84							
13. nature										5			89						
14. construction site														95					
15. church															89				
16. library																16			
17. railway station											16						100		
18. wind										11								100	
19. street cafe																			79
20. concert															11				
21. in a vehicle	11				16														
22. harbor										26									
23. parking lot									16										
24. office		2														11			
25. home				5									5			5			
26. airplane					5														
27. roomy space								16								5			
28. others	11	0	0	0	5	0	0	10	5	0	0	11	6	5	0	31	0	0	11

Table 8 presents the confusion matrix for the second experiment. The overall recognition rate for the second test was 78 %. In this case, the recognition accuracy for individual environment ranged from 16 % to 100 %. In the first test, the three best recognized scenes were bathroom (100 %), amusement park (100 %), and presentation (95 %). On the other hand, the three most difficult scenes were “library” (0 %), street café (0 %) and replayed music (32 %). The best recognized scenes for the second test were kitchen (100 %), railway station (100 %) and wind (water tower, 100 %). The worst cases were library (16 %), market place (42 %) and train (68 %).

Table 9: Recognition accuracies of different subject classes

	Overall	Seeing subjects	Blind subjects	Best	Worst
First test	66	68	62	85	47
Second test	78	78	78	90	60

Some of the reasons for confusions seem obvious. One of them was misleading sound events. For example, the scene “market place”, which was recorded in Helsinki at the market of Hakaniemi near the sea, was frequently recognized as a harbor. In that recording, screams of seagulls can be heard very loudly, which is likely to have misled 26 % of the subjects. The scene “trams” was a recording of trams passing by at a distance of one meter. However, there were other sound sources as well, like cars and buses. For some subjects the sound event was unfamiliar and they suggested a train as an answer. In the first test, this scene was often confused (see Table 7). In the second test, there was less confusion; however, the scene was recognized as a street in 84 % of the cases, which can be treated as a correct answer. Another possible source of errors is a lack of prominent sound events. The recording from the library was very quiet with very few loud sound events. The sound events were such as footsteps, beeps of a bar code reader and noises caused by handling of books. Some of the subjects described the scene as “there is nothing going on”. Differences in the recognition abilities between subjects are presented in Table 9. In contrast to our initial assumption that blind subjects would be likely to perform better than seeing subjects, no significant differences can be noticed between the recognition rate of these two groups. However, due to the limited number of subjects, a final conclusion cannot be made. In both tests, the best recognition performance was achieved by a young female subject, whereas the worst recognition rate belongs to an older male subject.

Latencies

In Figure 12, average latencies of the recognition process are shown for a couple of scenes in the first test. The bold line in the figure is the latency averaged over all the scenes. On the average, successful recognition took 20 seconds in this test.

The latency curve was calculated as a cumulative sum of the correct answers at a given time. A point on the curve may represent more than one correct answer. This is because the response times were measured at a precision of one second.

A sharp increase in some curves was often caused by a recognized prominent sound event at that particular instant. A good example is the bathroom environment, where a sound of toilet flushing is heard about five seconds from the beginning. Actually, none of the subjects recognized the scene before this particular sound event.

The average identification time for the second test was 46 seconds. The purpose of the second test was not to examine the response times. Therefore, the subjects were not hastened to give their answers which, in turn, resulted in longer response times.

Cues

In the second test, the subjects were asked to describe what were the cues they used in the recognition process. It turned out that the cues were most often described in terms of familiar sound sources or events. These answers were analyzed and categorized as shown in Table 10.

In Table 11, the categorized cues reported by the subjects are shown. Values in the table are percentages, which indicate how many times each cue was mentioned in a successful recognition. The percentages are averaged over all the scenes and weighted by the number of correct identifications. The only difference, which can be noticed between seeing and blind subjects, is that the blind subjects mentioned spatial information more often as a cue than seeing subjects (27 % and 12 % correspondingly). Some of expressions they used were “this is a big place”, “there is a stone flooring” and “it is close to a wall”.

On the average, transient sounds were reported as a cue in 47 % of the cases (a subclass of “prominent event”, see Table 10). “Many speaking” was reported as a cue in 24 % and “human activity” in 21 % of the cases. In contrast, “one speaking” was reported only in 2 % and “understood content” in 5 % of the cases.

In Table 12, the cues used in different scenes are listed. Percentages in the table indicate how many times each category of cues was reported having to been used. Only scenes that were correctly identified were counted. For example, 100 % means that the cue was mentioned by all subjects that recognized the scene correctly.

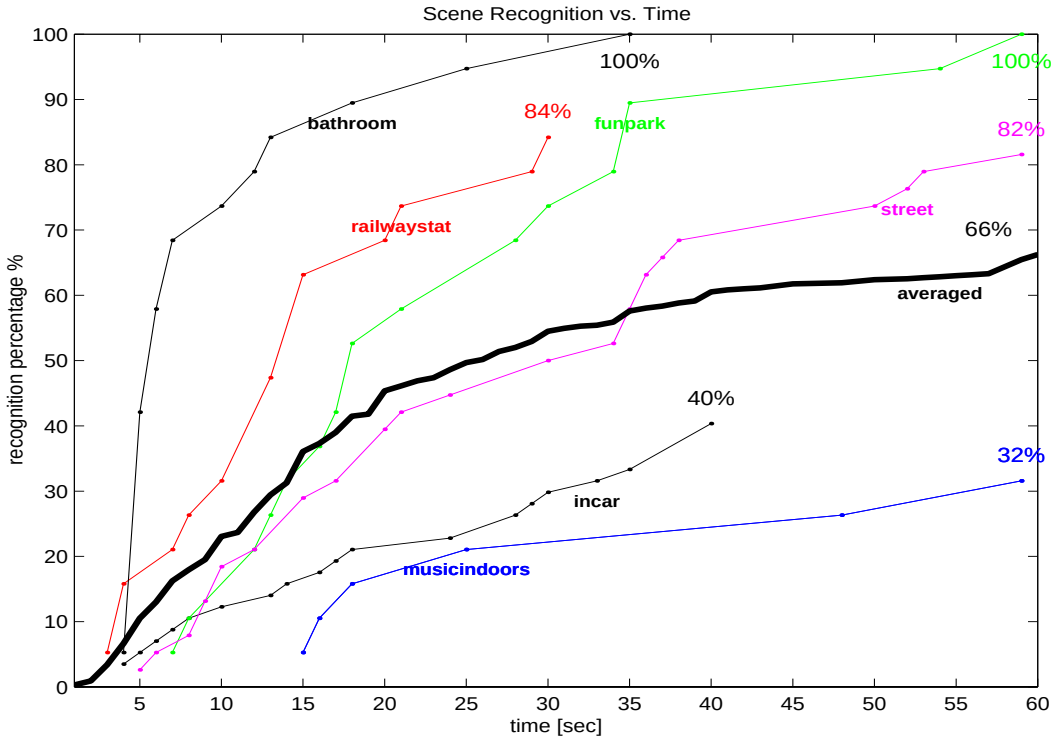

Figure 12. Averaged response times in the first experiment

Table 10: Categorization of the cues

Class	Subclasses
Human	one speaking, many speaking, other human noise (screaming, laughing), understood content, human activity (steps, pottering around)
Vehicles	(cars, bikes, planes, etc.)
Continuous noise	(car engine, rail noise)
Nature	organic (animals), inorganic (wind, water)
Spatial information	(reverberation)
Prominent event	transients (dishes, coins, crash, boom & bang), noise (paper bag, machine tools, microwave), not live performance (announcements, radio), live performance (live music, lecturing)

Table 11: Cues used by the subjects

	All subjects	Seeing	Blind
Human	41	42	40
Vehicles	29	28	31
Continuos noise	9	9	9
Natural	20	21	17
Spatial information	16	12	27
Prominent event	69	68	72

5.3 Conclusions from the listening test

The listening test showed that humans are able to recognize everyday auditory scenes in 70 % of cases on average. The confusions were between scenes that had same types of prominent sound events, and were usually not fatal from the point of view of computational auditory scene recognition applications. This can be seen from the confusion matrices in Table 7 and Table 8.

Recognition latency was of the order of 20 seconds. This suggests that an accurate automatic recognizer for similar material should utilize relatively long excerpts of input signals in making the inferences, that is, tens of seconds of audio. In addition, it would seem advantageous to focus the recognition process to distinct sound events. Analysis of the different cues used by humans provides directions in constructing the feature extractors. On the other hand, humans are so much oriented towards analyzing auditory scenes into distinct sound sources that it is not surprising that they described what they heard in term of sources. It is quite probable that amateur listeners are not able to report the use of low-level qualities of the sound scene, although these may have affected the recognition process.

Table 12: Cues used in the recognition of different environments

	Human	Vehicles	Continuous noise	Natural	Spatial information	Prominent event
In train	42		95	5	11	58
Restaurant	55				13	95
Trams	100	100		21	5	26
Supermarket	58				11	95
In car			89	5	5	95
Kitchen	42			58	5	100
Road		100		16		
People indoors	84				58	21
Street	11	89		11	11	37
Market place	79	74		95	5	32
Subway station	16	89	5	5	5	11
Amusement park	89	11			5	74
Nature	68	63	11	79		21
Construction site	11				5	95
Church	11				47	89
Library	53				47	84
Railway station	26	11			47	89
Wind	47	26		95		68
Street café	68	89			16	84

6 Acoustic feature extraction and classifiers

Certain fundamental problems have to be considered when designing a pattern recognition system. Tou and Gonzalez state these as [Tou74]:

1. The Sensing problem: how do we measure the object?
2. Feature extraction problem: extraction of characteristic features.
3. Determination of the decision procedure: how the classes can be differentiated with the given features?

In audio classification systems, the measurements are normally done using microphones, thus the sensing problem is solved. This chapter is devoted to the definition of several physical audio features and their mathematical definitions, and to the description of tested classification and clustering algorithms. In addition, a short discussion on the correlation between the features is provided.

6.1 Description of the acoustic features

The acoustic attributes can be divided into two groups: physical features and perceptual features. The perceptual features describe the sensation of a sound in subjective terms, i.e. perception of sounds by human beings. Examples of these features are loudness, pitch, brightness and timbre. Physical features refer to features calculated mathematically from the sound wave, such as intensity, fundamental frequency, spectral centroid and spectrum. In this section, we will provide mathematical definitions of different physical features examined in this work. The physical features are further grouped into two categories according to the domain in which they are calculated. These categories are spectral features (frequency-domain) and temporal features (time-domain).

As we have mentioned before, the choice of a feature set is the crucial step in building a pattern classification system. Therefore, the selection of the features is sometimes left to the brute force of an algorithm to find the features that are the most capable to discriminate between the classes. To read more on the automatic selection of feature sets, see [Schürmann96].

Spectral features

1. Spectral Centroid represents the “balancing point”, or the midpoint of the spectral power distribution. It is related to the brightness of a sound. The higher the centroid, the brighter the sound. The spectral centroid can be calculated for a discrete signal as

$$SC = \frac{\sum_{n=0}^{M} n \cdot |X(n)|^2}{\sum_{n=0}^{M} |X(n)|^2}, \quad (1)$$

where X is the discrete Fourier transform (DFT) of the time domain signal and M is the index of the highest frequency sample (which is half of the DFT order N). The resolution of the centroid is same as the resolution of the DFT, which is f_s */(N-1)* Hz. Here f_s refers to the sampling frequency. This yields the maximum error of f_s */2(N-1)* Hz. Figure 13 shows the power spectrum and illustrates the calculated spectral centroids for an unvoiced and voiced phoneme.

2. Signal bandwidth is defined as the width of the range of frequencies that the signal occupies. The bandwidth of a signal frame is given as

$$BW = \sqrt{\frac{\sum_{n=0}^{M} (n - SC)^2 \cdot |X(n)|^2}{\sum_{n=0}^{M} |X(n)|^2}}, \quad (2)$$

where SC is the spectral centroid given by Equation (1), X is the DFT of the signal and M is the index of the highest frequency sample.

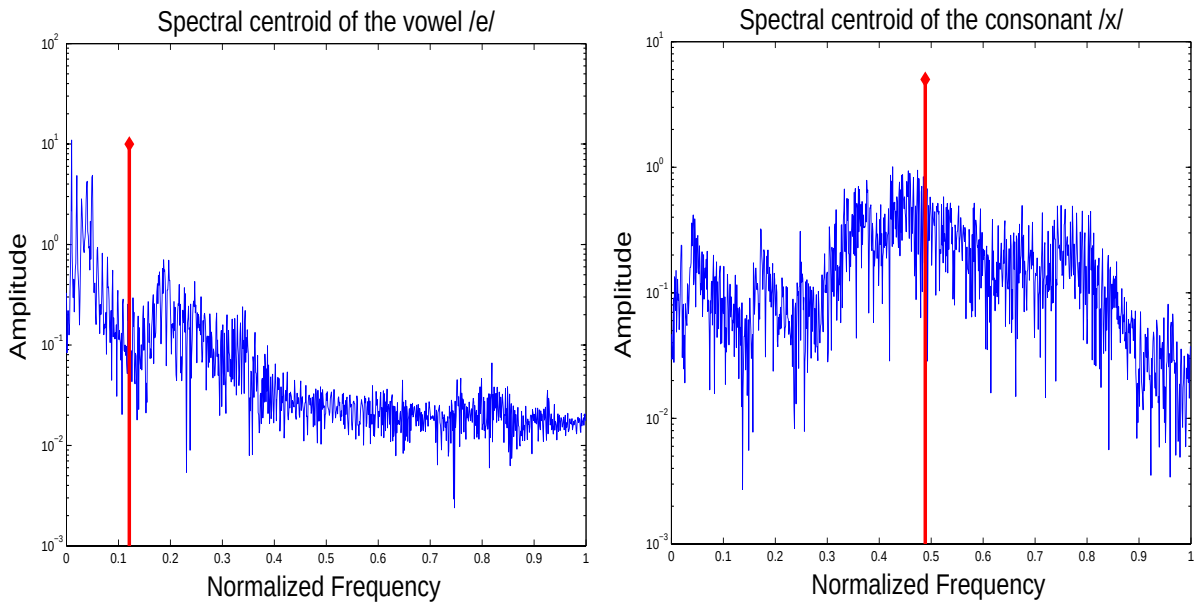

Figure 13. Spectral centroid for a voiced and unvoiced phoneme

3. Spectral Rolloff Point. This feature measures the frequency below which a specific amount of the power spectrum resides. It measures the "skewness" of the spectral shape. The rolloff point is calculated as

$$SR = max\left[K \mid \sum_{n=0}^{K} |X(n)|^2 \leq TH \cdot \sum_{n=0}^{M} |X(n)|^2 \right], \quad (3)$$

where TH is a threshold between 0 and 1. A commonly used value for the threshold is 0.95. In our experiments, we examined values between 0.90 and 0.98.

4. Spectral Flux or **Delta Spectrum Magnitude** measures the change in the shape of the power spectrum. The spectral flux is calculated as a difference between power spectra of successive frames, more exactly

$$SF_k = \sum_{n=1}^{M-1} \left\| |X_k(n)| - |X_{k-1}(n)| \right\| \quad , \quad (4)$$

where *k* is the index of the frame.

Spectral flux is an efficient feature for speech/music discrimination, since in speech the frame-to-frame spectra fluctuate more than in music, particularly in unvoiced speech [Scheirer97].

5. Band Energy Ratio (BER) is the ratio of the energy in a certain frequency band to the total energy. The band energy ratio of the i^{th} subband is calculated as

$$BER_i = \frac{\sum_{n \in S_i} |X(n)|^2}{\sum_{n=0}^{M} |X(n)|^2}, \quad (5)$$

where S_i is the set of Fourier transform coefficients belonging to the i^{th} subband. In our experiments, we used four logarithmic subbands. Denoting the Nyquist frequency with f_{max} (i.e. $f_{max} = f_s/2$), the four subbands were the following: [0, $f_{max}/8$], ($f_{max}/8$, $f_{max}/4$], ($f_{max}/4$, $f_{max}/2$] and ($f_{max}/2$, f_{max}]. We chose this partition because in natural acoustic signals, the lower frequency bands posses more energy than the higher ones and it is also perceptually more correct to use the logarithmic frequency scale.

6. Linear Prediction Coefficients (LPC)

Linear predictive analysis is one of the most used and successful speech analysis tools [Markel76]. It has also been applied to filter design, spectral analysis, and system identification. The basic idea behind linear prediction is that the next signal sample is predicted from a weighted sum of *p* previous samples, given as follows:

$$\hat{s}(n) = \sum_{i=1}^{p} a_i s(n-i), \tag{6}$$

where the set $\{a_i\}$ are the prediction coefficients and *s(n-i)* is a sample at time instant *n-i*. In other words, each sample of a signal is modeled as a linear combination of previous samples, which is equivalent to all-pole IIR filtering. The prediction coefficients are determined by minimizing the mean squared error between the actual sample and the prediction. The prediction error signal is given by:

$$e(n) = s(n) - \hat{s}(n) = s(n) - \sum_{i=1}^{p} a_i s(n-i), \tag{7}$$

and the squared error for a segment of the sample waveform is:

$$E_n = \sum_m [e_n(m)]^2 = \sum_m \left[s(n) - \sum_{i=1}^{p} a_i s(n-i) \right]^2, \tag{8}$$

where *m* is the length the signal segment. To find the minimum of the error we set the partial derivation of E_n with respect to the a_i to zero, that is $\partial E_n / \partial a_i = 0$, for *i=1,2,..., P*, and obtain the following set of *p* normal equations

$$\sum_{i=1}^{p} a_i \sum_m s_n(m-k) s_n(m-i) = \sum_m s_n(m-k) s_n(m), \qquad k = 1,2,...,p. \tag{9}$$

The expression $\sum s_n(m-k) s_n(m-i)$ is actually the autocorrelation of the signal *s(n)* at lag $|k-i|$. Denoting the autocorrelation with *r(k)* and substituting it into the normal equations, we obtain

$$\sum_{i=1}^{p} a_i r(|k-i|) = r(k) \quad , \qquad k = 1,2,...,p. \tag{10}$$

This system of equations can be with the solved so-called Levinson-Durbin recursion [Ljung87].

The LPC analysis order *p* (i.e. the number of the coefficients) determines how finely the LPC filter 1/*A(z)* models the original magnitude spectrum. Figure 14 illustrates the magnitude spectrum of the FFT and the frequency responses of five different LPC filters of the orders 4, 6, 10, 12 and 20 of the vowel /e/. As can be noticed from the graphs, the higher the analysis order, the more the spectrum fine structure is modeled.

7. Cepstral Coefficients. The cepstrum is defined as the inverse Fourier transform of the logarithm of magnitude spectrum, given by

$$\boldsymbol{c} = F^{-1}\{\log|F\{\boldsymbol{x}\}|\}, \tag{11}$$

where $\boldsymbol{x}$ is the time domain signal and F denotes the Fourier transform. The cepstral coefficients can be calculated directly using the above equation or they can be estimated efficiently by using the LP analysis, and converting the LPC coefficients into LP-based cepstral coefficients. The conversion from p LP coefficients to N cepstral coefficients can be done using the following recursion:

Step 1. The first coefficient is the autocorrelation at lag 0, i.e. $c_0 = r(0)$.

Step 2. The next coefficients are calculated as $c_m = -a_m - \sum_{k=1}^{m-1} \frac{k}{m} c_k a_{m-k}$, for $m \geq 1$, where $a_0 = 1$ and $a_k = 0$ for $k > p$ (p is the order of the LP analysis).

The zeroth cepstral coefficient represents the energy of the signal, the lower coefficients reflect the macro structure and the higher ones reflect the microstructure of the spectrum.

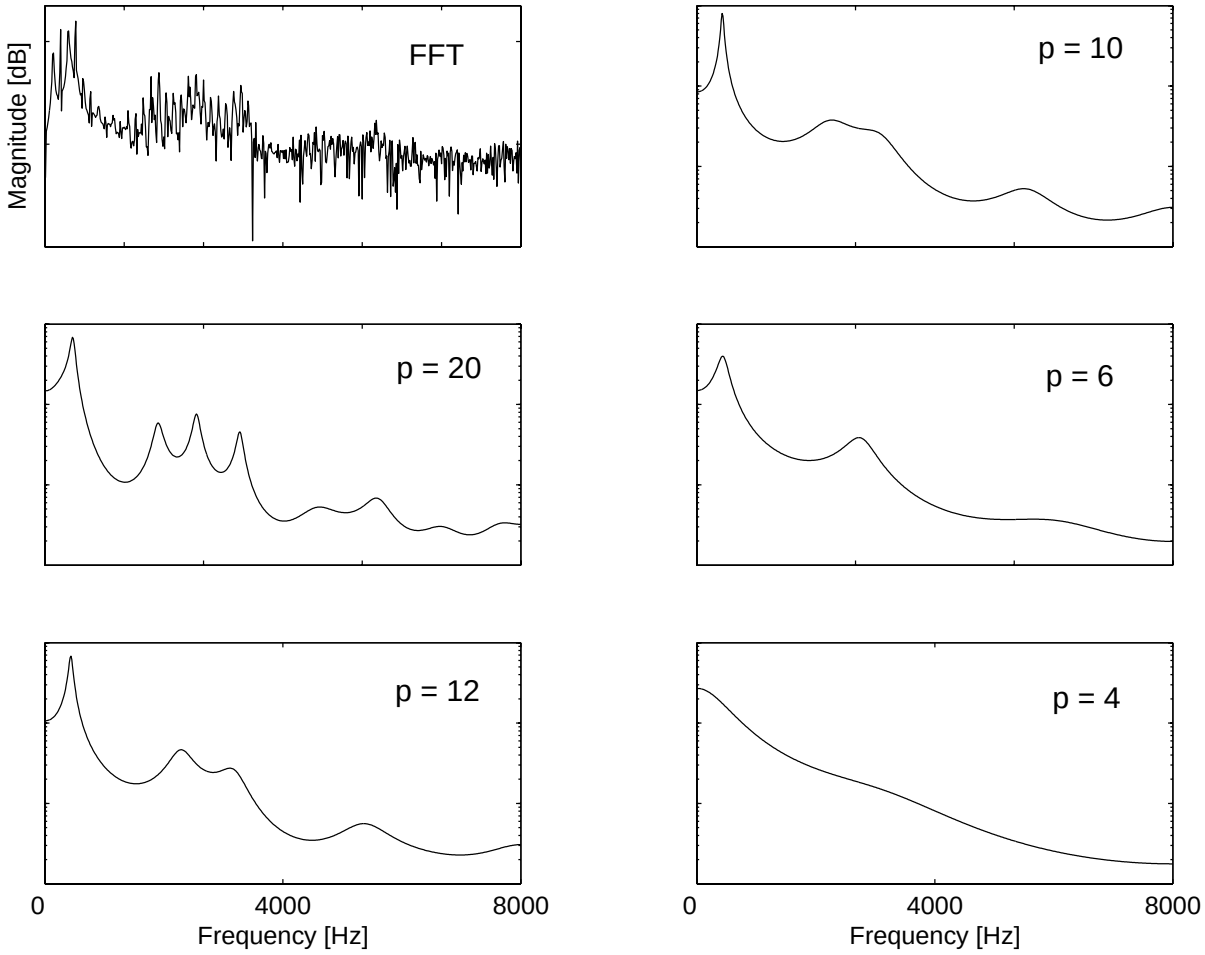

Figure 14. The effect of LPC analysis order. The first graph is the logarithmic magnitude spectrum of the FFT of the vowel /e/ and the next five graphs are the frequency responses of five different LPC filters of the orders 4, 6, 10, 12, and 20.

Cepstral coefficients are known to be more robust and reliable features in speech analysis than LPCs. This is because the cepstral coefficients are generally uncorrelated (see Figure 17). This advantage can be explained with a contrary example: lets suppose that two dimensions (coefficients) of a feature vector are highly correlated (i.e. one is linearly dependent on the other), then they are not providing any additional information for the classification process. Thus, we would like to have uncorrelated dimensions, in hope that they are independent and each of the dimensions provides dissimilar information. Furthermore, statistical classifiers assume uncorrelated features, e.g. the Gaussian mixture model with diagonal covariance matrices. A short discussion on correlation between the features is provided in Section 6.2.

8. Mel-frequency cepstral coefficients (MFCC).

In Figure 15, a simplified block diagram of a MFCC feature extractor is presented. The first step, pre-processing, consists of pre-emphasizing, frame blocking and windowing of the signal. After a discrete Fourier transform, the power spectrum is transformed to mel-frequency scale. This is done by using a filter bank consisting of triangular filters, spaced uniformly on the mel-scale. An approximate conversion between a frequency value in Hertz (f) and in mel is given by:

$$Mel(f) = 2595\log_{10}\left(1 + \frac{f}{700}\right) \qquad (12)$$

Finally, the cepstral coefficients are calculated from the mel-spectrum by taking the discrete cosine transform (DCT) of the logarithm of the mel-spectrum. This calculation is given in by

$$c_i = \sum_{k=1}^{K} (\log S_k) \cdot \cos\left(\frac{i\pi}{K}\left(k - \frac{1}{2}\right)\right) ,$$

$$i = 1, 2, \ldots K , \qquad (13)$$

where c_i is the i^{th} MFCC, S_k is the output of k^{th} filterbank channel (i.e. the weighted sum of the power spectrum bins on that channel) and K is the number of coefficients. A more detailed description of the mel-frequencycepstralcoefficientscanbefoundin[Rabiner93].

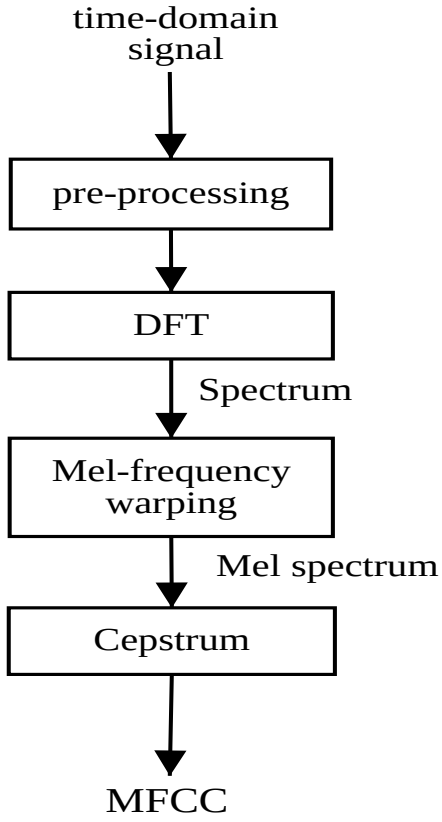

Figure 15. Block diagram of the MFCC feature extractor

Time-domain features

1. Zero-Crossing Rate (ZCR) is defined as the number of time-domain zero crossings within a processing frame. This feature correlates with the spectral centroid, since both features measure the frequency content of the signal. This dependency is illustrated in Figure 16, where

both the ZCR and centroid are calculated for the speech sample "*Bond, James Bond*". In this thesis, the ZCR is calculated as follows

$$ZCR = \left(\sum_{n=1}^{M-1} |\mathrm{sgn}(x(n)) - \mathrm{sgn}(x(n-1))| \right) / 2M \quad , \tag{14}$$

where x is the time-domain signal, sgn is the signum function, and M is the size of the processing frame. The signum function in our implementation is defined as

$$\mathrm{sgn}(x) = \begin{cases} 1 & , x > 0 \\ 0 & , x = 0 \\ -1 & , x < 0 \end{cases} . \tag{15}$$

This definition counts also the cases where the signal only touches the zero voltage level.

One of the most attractive properties of the ZCR is that it is very fast to calculate. As being a time-domain feature, there is no need to calculate the spectra. Furthermore, a system which uses only the ZCR-based features would not even need digital-to-analog conversion, but only the information whenever the sign of the signal changes.

2. Short-time average energy. In this study, the average energy for a frame was calculated as follows:

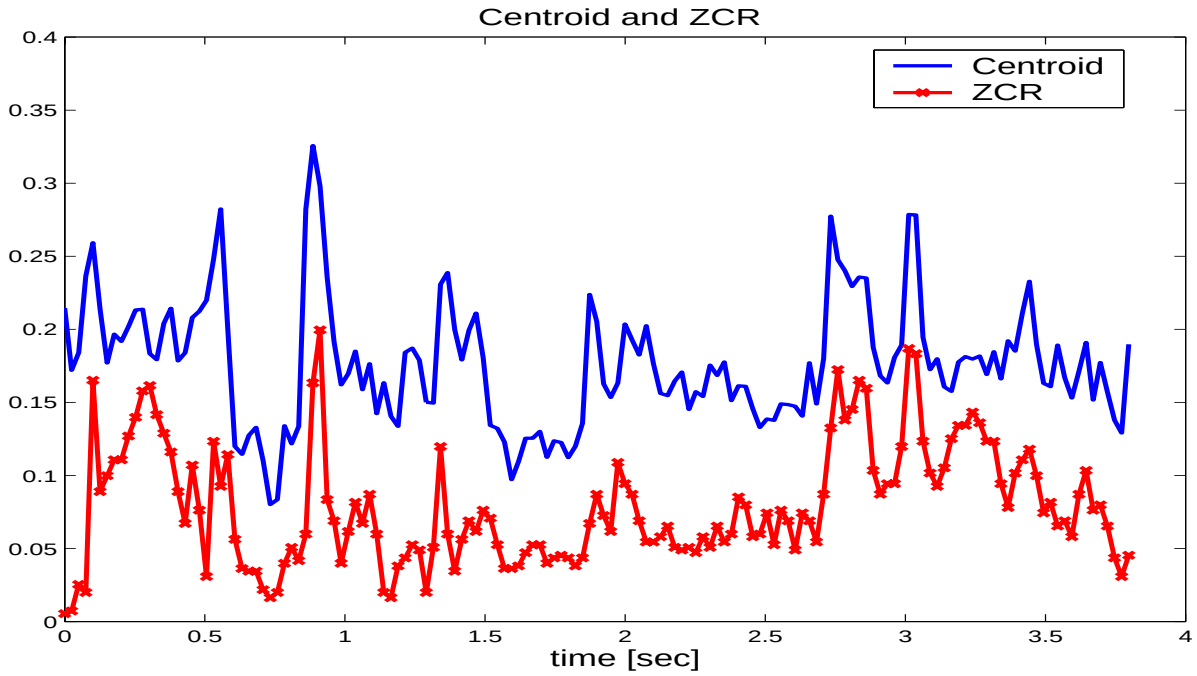

Figure 16. Frame-wise spectral centroid and zero-crossing rate values over the speech sentence "*Bond, James Bond*". The correlation between the features can be clearly seen.

$$E = \frac{1}{N}\sum_{k=0}^{N-1} x(k)^2 , \qquad (16)$$

where x is the time domain signal and N is the size of the processing window. Thus, the feature depends on the recording gain.

3. Percentage of Low-energy frames measures the proportion of low energy frames within a one-second window. In this thesis, a low-energy frame is defined as a frame with average energy less than 50 % of the mean energy in the one-second window.

6.2 Correlation between the features

As we mentioned before, different features may be correlated. One way to measure the correlation between the features is to compute their covariance matrix. For uncorrelated features the covariance matrix is diagonal. A covariance matrix for a N dimensional feature vector $\boldsymbol{x}$ can be estimated as follows

$$\Sigma = \frac{1}{N}\sum_{i=1}^{N} x_i x_i^T - \bar{\mu}\bar{\mu}^T \qquad , \qquad (17)$$

where $\bar{\mu}$ is the mean vector estimated by

$$\bar{\mu} = \frac{1}{N}\sum_{i=1}^{N} x_i \quad . \qquad (18)$$

The derivation of Equations (17) and (18) are given in appendix A.1. In Figure 17, a plots of two covariance matrices are illustrated. The first is a covariance matrix of cepstral coefficients and the second is a covariance matrix of linear prediction coefficients estimated from the same signal sample. The correlation trend can be seen clearly in the covariance matrix estimated from the linear prediction coefficients as parallel lines to the diagonal.

6.3 Clustering algorithms

Clustering is defined as a process by which objects are partitioned into subgroups, such that the objects within the same subgroup have similar characteristics, and the objects within different clusters have distinct characteristics. Within a class, clustering can be seen as a sort of quantization, i.e. reduction of dimensions. The instances of the class are presented with fewer parameters, which are the cluster centers. Different kinds of clustering algorithms have been developed for different purposes and varying performance requirements [Tou74]. The clustering algorithm we tested in this work, *k-means*, is reviewed shortly after introducing the performance index.

Performance index

The performance of a clustering algorithm is often evaluated using a cost function that is the

sum of the squared errors, calculated as

$$\boldsymbol{J} = \sum_{i=1}^{N} \sum_{x \in S_i} \left\| \boldsymbol{x} - \boldsymbol{m}_i \right\|, \tag{19}$$

where N is the number of cluster centers, S_i is the set of samples belonging to the i^{th} cluster center, and $\boldsymbol{m}_i$ is the mean of the samples belonging to the set S_i.

K-means clustering algorithm

This clustering algorithm tries to minimizes the performance index introduced above. The algorithm is simple, intuitive, and fast. However, the main drawback of the algorithm is that the number of cluster centers is pre-defined. The performance of this algorithm depends on the number of the clusters, on the initialization of the cluster centers and on the geometrical properties of the data. The algorithm consists of the following steps:

Step 1. Choose arbitrary samples to represent K initial cluster centers

Step 2. Distribute all the samples to their nearest cluster center using a selected distance measure (for example Euclidean distance).

Step 3. Compute new centers for each cluster, by calculating the mean of the samples belonging to each cluster.

Step 4. Terminate if there are no changes in cluster centers or if a maximum number of iterations is reached. Otherwise return to step 2.

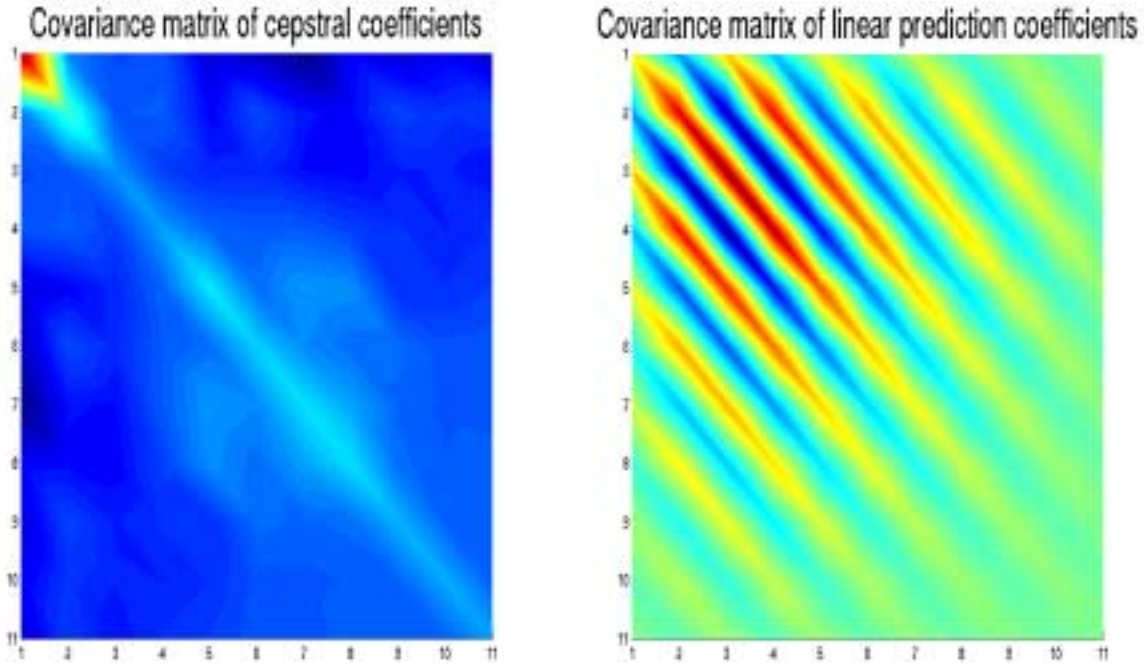

Figure 17. Covariance matrices of cepstral coefficients and linear prediction coefficients estimated from the same data.

6.4 Classification algorithms

K-Nearest Neighbor classifier (k-NN)

The k-nearest-neighbor classifier simply places the points of the training set in the feature space, and the classification decision is made according to "voting" of the nearest *k* neighbors to a test point [Cover67]. The voting is done by picking the *k* points nearest to the test point, and the chosen class is the class that is most often picked. The distance can be measured with different metrics, the most often used are the well known Euclidean distance and the Mahalanobis distance. In our implementation, the distance is measured using the Mahalanobis metric. The square Mahalanobis distance between the vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ is given by:

$$r_M^{\,2} = (\boldsymbol{x}-\boldsymbol{y})^T C^{-1}(\boldsymbol{x}-\boldsymbol{y}), \tag{20}$$

where C is the covariance matrix of the training data. The squared Euclidean distance is calculated as

$$r_E^{\,2} = \|\boldsymbol{x}-\boldsymbol{y}\| = (\boldsymbol{x}-\boldsymbol{y})^T(\boldsymbol{x}-\boldsymbol{y}). \tag{21}$$

There are several advantages in using the Mahalanobis metric instead of the Euclidean metric. First, the Mahalanobis metric automatically scales the coordinate axes of the feature space. Second, it decorrelates the different features. However, this decorrelation is done to the whole set of training samples as one entity, and not for every class separately. This relies on the assumption that the covariance matrix is the same for all classes, which is not true for a majority of the practical cases. Third, the Mahalanobis metric is able to accommodate curved as well as linear decision boundaries [Therrien89].

Figure 18 illustrates the decision boundaries of a 1-NN classifier for four real scenes: inside a moving car, restaurant, church, and nature. The extracted features were the mean and the standard deviation of the bandwidth over a one second window. The bandwidth was calculated in 30ms frames, with a 15ms overlap. In the first plot (upper left corner), the distance is measured using the Euclidean metric, and the coordinate axes of the feature space are not normalized. This causes an undesired weighting of the features, as seen in the figure as elongated horizontal contours. However, in the second plot (upper right corner), the axes of the feature space are normalized, and therefore the distance metric is not influenced by the range of the fluctuation of the individual features. The third and the forth plot (lower left and right panels) represent exactly the same decision boundary, which is estimated using the Mahalanobis distance metric. The decision boundary on the left is plotted in the original feature space, whereas the one on the right is plotted in the transformed feature space. The transformation here refers to the multiplication with C^{-1} in Equation (20) that performs the scaling and the rotation. As can be seen from the plot in bottom-left, the Mahalanobis distance automatically accounts for the scaling of the axes (coordinate axes are in the same range as in the first plot). The Mahalanobis distance rotates the feature space to decorrelate the different features. The decision boundaries in the rotated feature space are shown in the last plot. As mentioned above, the covariance matrix was estimated from the whole training set, and therefore the features of an individual class may be still correlated.

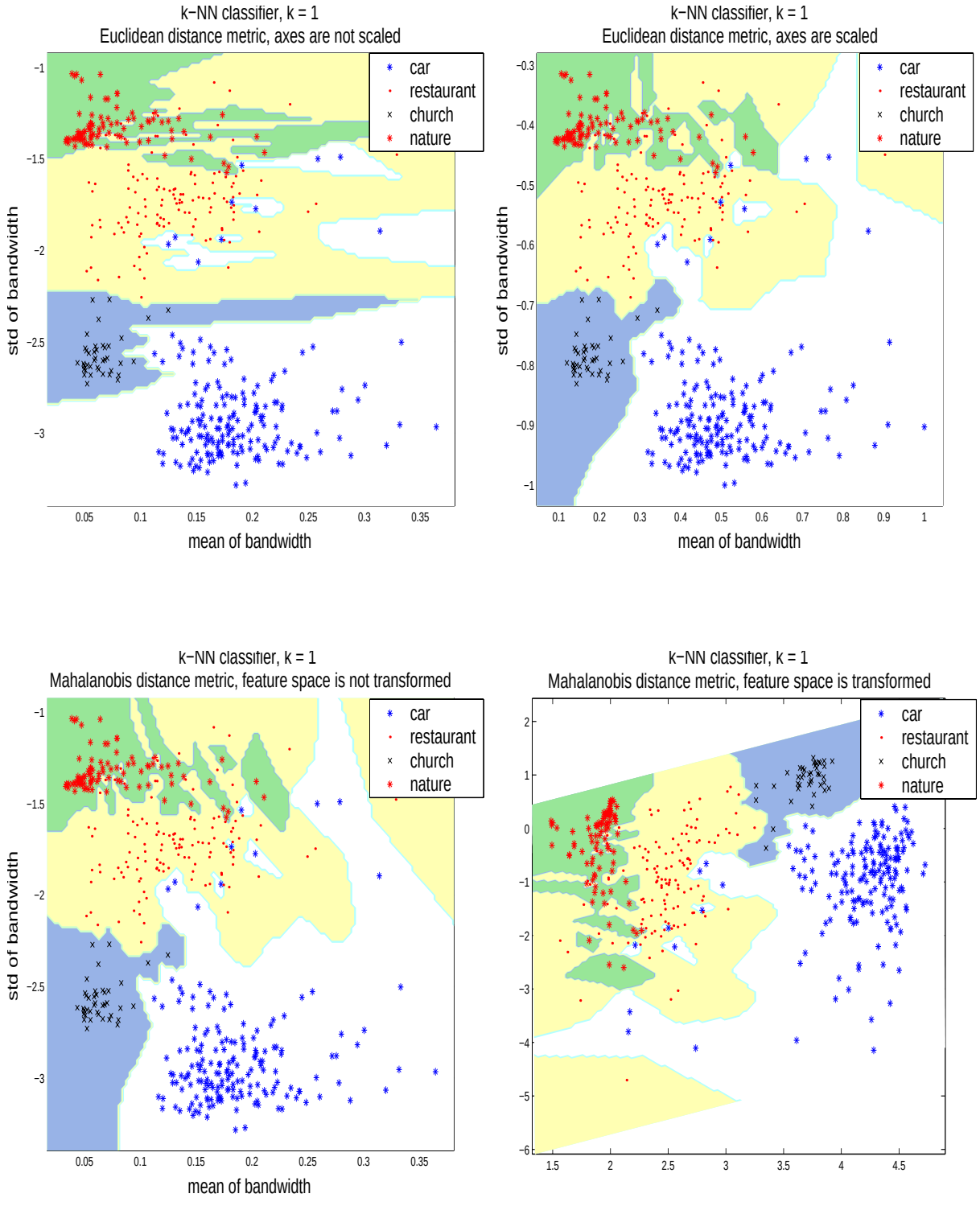

Figure 18. Decision boundaries for a 1-NN classifier for four cases (a) Euclidean distance metric without normalization of the coordinate axes (up-left), (b) Euclidean metric with normalization of the axes (up-right), (c) Mahalanobis distance metric in the original feature space (bottom-left), and (d) Mahalanobis metric in the transformed feature space (bottom-right).

Multivariate Gaussian classifier

An assumption behind the Gaussian classifier is that the samples of each class ω_i follow a Normal distribution. The probability density function (pdf) of a one dimensional Gaussian distribution with the mean *m* and variance σ^2 is given by

$$P(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-m)^2}{2\sigma^2}}, \tag{22}$$

The pdf for a d-dimensional vector $\boldsymbol{x}$ is given by

$$P(\boldsymbol{x}) = \frac{1}{(2\pi)^{d/2}\left|\Sigma_i\right|^{1/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\bar{\mu}_i)^T \Sigma_i^{-1}(\boldsymbol{x}-\bar{\mu}_i)\right\}, \tag{23}$$

where $\bar{\mu}_i$ is d-dimensional mean vector and Σ_i is *d* x *d* dimensional covariance matrix of the i^{th} Normal distribution. The estimation equations for the mean vector and the covariance matrix were given in Equation (17) and Equation (18) respectively.

The classification decision is done by minimizing the probability of error by choosing the class with maximum a posteriori probability [Kay93], given as

$$\hat{\omega} = \arg\max_{\omega_i} \; p(\omega_i|\boldsymbol{x}) \tag{24}$$

where $\hat{\omega}$ is the classified class and $p(\omega_i|\boldsymbol{x})$ is the posterior probability. That is the probability of class ω_i after the data vector $\boldsymbol{x}$ have been observed. The $p(\omega_i|\boldsymbol{x})$ is generally not know, therefore using the Bayes formula, given by

$$p(\omega_i|\boldsymbol{x}) = \frac{p(\omega_i)p(\boldsymbol{x}|\omega_i)}{p(\boldsymbol{x})} \tag{25}$$

we obtain

$$\hat{\omega} = \arg\max_{\omega_i} \; p(\boldsymbol{x}|\omega_i)\cdot p(\omega_i), \tag{26}$$

where $p(\boldsymbol{x}|\omega_i)$ is the so called the likelihood function of class ω_i (when the pdf is viewed as a function of the unknown parameter, it is termed as the likelihood function). Note that the denominator $p(\boldsymbol{x})$ in Equation (19) is the same for all ω_i, so it can be omitted in the substitution.

Gaussian Mixture Models (GMMs)

The problem with single Gaussians is that they can model only Normal distributions. They can not model distributions with multiple modes or distributions with nonlinear correlation. A

Gaussian mixture density is able to approximate an arbitrary pdf with a weighted sum of N multivariate Gaussian pdf's [Reynolds95]. The Gaussian mixture density with a model order M is given by

$$p(\boldsymbol{x}|\lambda) = \sum_{i=1}^{M} p_i b_i(\boldsymbol{x}), \tag{27}$$

where $\boldsymbol{x}$ is a d-dimensional random vector, $b_i(\boldsymbol{x})$ are the M Gaussian pdf's, and p_i are the M mixture weights. The sum of the mixture weights is one and the pdf of the i^{th} d-variate normal distribution, $b_i(\boldsymbol{x})$, is given in Equation (18). A GMM is completely represented with three parameters: the mean vectors, the covariance matrices, and the mixture weights. These parameters are collectively represented by the notation

$$\lambda = \{p_i, \bar{\mu}_i, \Sigma_i\}, \qquad i = 1,...,M. \tag{28}$$

The parameters are estimated using the Expectation Maximization (EM) algorithm such that the likelihood is maximized [Dempster77]. The EM algorithm consists of two main steps:

Step 1. Estimate the statistics of the complete data given the observed data and current parameters

Step 2. Maximize the likelihood given the new statistics

The algorithm guarantees a monotonically non decreasing likelihood and it converges at least to a local maximum of the underlying likelihood function. The main disadvantage of the EM algorithm is the slow convergence. For a sequence of T training vectors, $X = \{\boldsymbol{x}_1,...,\boldsymbol{x}_T\}$, the GMM likelihood is as follows

$$p(X|\lambda) = \prod_{t=1}^{T} p(\boldsymbol{x}_t|\lambda). \tag{29}$$

The GMM is used as a classifier as follows. The GMM parameters for each class are estimated by analyzing the training data, which is the learning stage. The probability of each class, given the observation, is evaluated, and the class that gives the highest probability is chosen as the classification result.

6.5 Hidden Markov models

We did not experimented hidden Markov models (HMM) in this thesis, and thus we are not presenting the theory of this classification method. For the underlying theory of HMMs see [Baum70 and Baum72].

There were several reasons for not investigating the HMMs, one of them being the lack of time and resources that would have been needed to properly assess the suitability of HMMs to the given task. However, the efficiency of this approach should be studied in the future. HMMs can be used at two different levels: to model individual sound events, or to model the order of

distinct events. Solving the first problem would require the annotation of distinct events in audio files, which is a huge task. Estimating the transition probabilities between individual sound events, in turn, would require relatively large amount of training data, and, on the other hand, we were sceptical whether there usually exists a regular time pattern for the sound events in different scenes.

7 Simulation results

In this chapter, we describe several examined classification schemes and present the obtained results. The evaluated schemes differ from each other with respect to the selected set of acoustic features, the classification decision methods, and the partition of recorded environments into more general classes called *metaclasses*. An intuitive and a simple example of a metaclass partition is *outdoors* vs. *indoors* environments.

This chapter is divided into three entities. In the first part, we compare the performance obtained with individual features used one at a time with the same data set and classification method. After that, the recognition performance of different classification approaches is evaluated and the recognition accuracy as a function of test sequence length is also studied. In the third part, we report results for the recognition of various metaclasses using a couple different feature sets.

All the tested algorithms were implemented in Matlab, which is an efficient tool for testing and simulating signal processing algorithms. However, a fundamental drawback using Matlab is its relatively low speed of computation, which caused some limitations in simulations.

7.1 Method of evaluation

The scenes

The utilized audio database was described in detail in Chapter 3. The training set included 25 different environments (124 samples), among which 13 environments were classified. The environments to be classified were chosen according to the criterion that each of them had to have at least three samples from different recording sessions. The classified environments and the number of recorded instances are listed in Table 13. We did not allow multiple class labels for one recording. Therefore, each recording is listed under one environment only, even though it would have had characteristics of more than one scene.

Evaluation procedure

When not otherwise mentioned, all the recorded material, except the one to be tested, was included in the training phase of the classification procedure, even though not all the scenes were included in the test set. That is, we used the “leave-one-out” testing method, where a classifier is trained with all instances except the one that is left out for classification. This procedure is repeated until all the instances are classified. In this way, the training data is maximally utilized but the system has never heard that particular recording before.

Table 13: The scenes and number of samples used in the simulations

Scene	Number of samples
1. Inside a moving car	18
2. Restaurant, cafeteria, dinning room	13
3. Nature	11
4. Lecture, presentation	10
5. Street	10
6. Office	8
7. Construction site	8
8. Supermarket, department store	6
9. Road	6
10. Bathroom	6
11. Kitchen	4
12. Church	4
13. Library	3
14. **Other**	**17**
Total	124

The recognition rates were computed as follows:

Step 1. For each class, the recognition rate was calculated as a percentage of correctly classified instances among all the classified instances.

Step 2. The overall recognition rate is calculated as the arithmetic mean of recognition rates of the individual classes.

It follows that a varying number of instances in different classes does not affect the recognition rate, but the same weight is given to the all classes. For comparison, the recognition rates obtained by random guessing are presented. The random recognition rate was calculated as the average of the probability of individual classes to be classified randomly. The random probability of each class was estimated as the probability of the class: n_i / n_t, where n_i is the number of all the samples from class i, and n_t is the total number of trained samples. The class probability was chosen instead of a uniform distribution, because with the k-NN classifier the number of training instances actually affects the classification rate.

7.2 Comparison of different features

Preprocessing

The mean value of the signal was removed in some feature extractors, such as zero-crossing rate, band-energy ratio, and spectral centroid. Also a pre-emphasizing filtering was applied in some feature extractors (cepstral coefficients and MFCC).

Based on preliminary experiments, we noticed that changes in the short-time signal processing parameter settings had only minor effect on the performance. The tested windowing functions were hanning and hamming, and the analysis window length varied between 20 ms and 50 ms. In most cases, the window length was fixed to 30 ms and the windowing function was hanning. The overlap between successive frames was fixed to 50 % of the window length.

Classification method and feature vector formation

We evaluated the discriminative power of each studied feature separately. The scenes were classified using two methods. The first was a frame-based approach. The training set of the k-NN classifier included the acoustic features from all the frames as such, without averaging, clustering or modeling the features before classifying them. The frames of the unknown signal were classified one-by-one. The recognition result for the whole clip was the class most often given as an answer for individual frames. In the case of the 1-NN (only one neighbor), we did an experiment in which we weighted the frame-level classification results. The weighing was as following: for each result we gave a weight inversely proportional to the distance between the classified point and the nearest neighbor. We did not notice any considerable improvement in the recognition rates using this weighting, in some cases the rates even decreased.

In the second and more successful classification method we estimated the mean and standard deviation (std) of the features over one second windows. The values were used as new features and each one second frame was classified separately using the 1-NN classifier. For clips longer than one second, the final result was again chosen by the majority rule. Also in this case, weighting the individual results did not yield any improvement. We tried several window lengths for estimating the mean and the std. The tested window lengths were 0.05, 0.125, 0.25, 0.5, 1, 2, 4 and 8 seconds. The recognition rates for the 13 scenes as a function of the window length are listed in Table 14. The length of the test signals was 24 seconds and the band-energy ratio was used as a feature. The best results were obtained using a window length of one second, although there was no great difference in the performance obtained with the lengths of 0.125 and 0.25 seconds. Anyhow, the window length of one second was chosen, because the classification stage requires less computation time when a longer window length is used.

Table 14: Recognition rates as a function of the averaging window length

Window length [s]	0.05	0.125	0.25	0.5	1	2	4	8
Recognition rate %	47.34	48.04	52.32	51.36	**54.38**	46.04	44.13	37.72

K-means clustering

We tested the k-means clustering algorithm, described in the section 6.3, along with the 1-NN classifier. The number of clusters were chosen such that the amount of the data of the raw features was reduced 20 times. The performance evaluation was done using the leave-one-out testing method, explained in Section 7.1. The feature vectors of the sample to be classified were not clustered, but compared as such to the cluster centers of the training data. The preliminary experiment showed that this approach does not perform very well. For example, we obtained a recognition rate of 34 % for the 13 scenes (listed in Table 13) using the band-energy ratio as a feature for analysis duration of 15 seconds. The corresponding accuracy for the k-NN

Table 15: Recognition rates for individual features using 1-NN and GMM classifiers for 13 scenes.

Feature	raw features, 1-NN (30ms)	mean and std, 1-NN (15 seconds)	Δ mean and Δ std 1-NN (15seconds)	raw features, GMM (15 seconds)
Band energy ratio	34 %	**56 %**	33 %	30 %
MFCC's (12 coefficients)	12 %	33 %	18 %	**48 %**
MFCC's (4 coefficients)	16 %	36 %	17 %	**44 %**
Cepstral coefficients	18 %	30 %	22 %	**44 %**
Bandwidth	17 %	**38 %**	21 %	30 %
LPC (12 coefficients)	13 %	26 %	11 %	**38 %**
Spectral Centroid	14 %	**36 %**	24 %	29 %
Spectral rolloff point	15 %	**33 %**	19 %	13 %
Zero-crossing rate	20 %	**28 %**	18 %	11 %
Spectral flux	9 %	**27 %**	18 %	6 %
Short-time average energy	12 %	18 %	**25 %**	6 %
low-energy frames	8 %	**12 %**	9 %	8 %
Random Guess probab.	**6.64 %**			

(mean+std) method was 56 %. For spectral centroid the accuracy was 23 % (36 %), for cepstral coefficients 8 % (30 %), and for MFCC's it was also only 8% (36 %). The percentages in the parentheses are the equivalent recognition rates obtained with the k-NN (mean+std) classification approach (see the results in Table 15).

Classification results

The classification results using individual features with several classification approaches are shown in Table 15. The recognition rates for a single frame were computed using the frame-based classification approach. The results for a 15 second test signal were computed using the k-NN (mean+std) method. In addition, we also computed the recognition rates with the feature derivatives (Δx) using the k-NN (mean+std) method. The derivatives were obtained by $\Delta x = x(n) - x(n-1)$, where *x(n)* is the value of the feature in the frame *n*.

For comparison, we listed in Table 15 also the recognition rates obtained with the second promising classification method, Gaussian mixture model, which is presented later in this chapter.

The best recognition result was achieved using the band-energy ratio and the 1-NN (mean+std)

Table 16: Best recognition rates for Individual scenes, with the 1-NN (mean+std) classification method.

Scene	Best recognition	With feature(s)
1. Inside a moving car	100 %	MFCC's (33 %), cepstral coeff. (30 %), flux (27 %)
2. Restaurant, cafeteria	77 %	MFCC's (33 %), cepstral coefficients (30 %)
3. Nature	91 %	Delta of the cepstral coefficients (22 %)
4. Lecture, presentation	80 %	BER (56 %)
5. Street	70 %	BER (56 %), bandwidth (38 %), flux (27 %)
6. Office	62.5 %	BER (56 %)
7. Construction site	62.5 %	Bandwidth (38 %), MFCC's (33 %)
8. Supermarket	67 %	BER (56 %)
9. Road	67 %	Roll-off (33 %), ZCR (28 %)
10. Bathroom	67 %	BER (56 %), centroid (36 %), Roll-off (33 %)
11. Kitchen	0 %	with all features
12. Church	100 %	BER (56 %), Bandwidth (38 %)
13. Library	0 %	with all features

classification method (56 %), which is approximately eight times better than the random guess rate (6.64 %). With the following three features we obtained almost the same performance: the bandwidth (38 %), spectral centroid (36 %), and MFCC's with four coefficients (36 %).

The relative efficiency of different features depends to some extent on the classifier itself. This is especially true if the feature vector contains several values whose weighting is affected by the classifier. In an experiment described in the next section, we obtained a 57 % recognition rate using the MFCC's as a feature vector and the Gaussian mixture models (GMMs) as a classifier for an analysis duration of 30 seconds. This result differs from the recognition rate (48 %) reported in Table 15, because the lengths of the test sequences were different, 30 seconds and 15 seconds correspondingly. A short discussion on recognition accuracy as a function of test sequence length is provided separately later in this chapter, in Section 7.4.

With the short-time average energy we did not obtain good results. As mentioned before, this feature depends on the channel gain of the recording process, and due to the lack of calibration information, we did not do any normalization to the feature values. Many contexts have a characteristic loudness, for example a noisy street or a quiet library. Therefore, if the energy information were employed correctly, it might enhance the recognition performance considerably.

Best performance for the detection of individual scenes

Table 16 shows the best recognition rates for individual scenes and the features used to obtain the results. The percentages in parentheses are the overall recognition rates obtained with the corresponding feature. The recognition rates are obtained with the 1-NN (mean+std) classification method.

Only two contexts were classified perfectly: *car* and *church*, and even those results were not obtained with the same features. The other extreme is the scenes that were not recognized at all. There were two scenes, *library* and *kitchen*, that were never classified correctly with any feature with the given classification method. There may be many underlying reasons for the poor performance. One evident reason is the small number of samples from these scenes (see Table 13). As discussed earlier, we do not make use of the loudness information, which might bring the needed extra information.

The individual recognition rates for the rest of the scenes ranged from 62.5 % to 95 %. Most often the best recognition rate for a single scene was obtained with the band energy ratio (6 times). That is consistent with the overall recognition results. Next comes bandwidth, MFCC's, and cepstral coefficients reported three times as giving the best recognition performance. They are followed by the spectral roll-off point (2 times), spectral flux (2 times), and spectral centroid (once).

7.3 Comparison of the examined classifiers

We tested three elementary classifiers, and a few variations of each. The tested classifiers were the k-NN, the multivariate Gaussian classifier and the Gaussian mixture model (GMM). A description of these classifiers was given in Chapter 6, in the section 6.4.

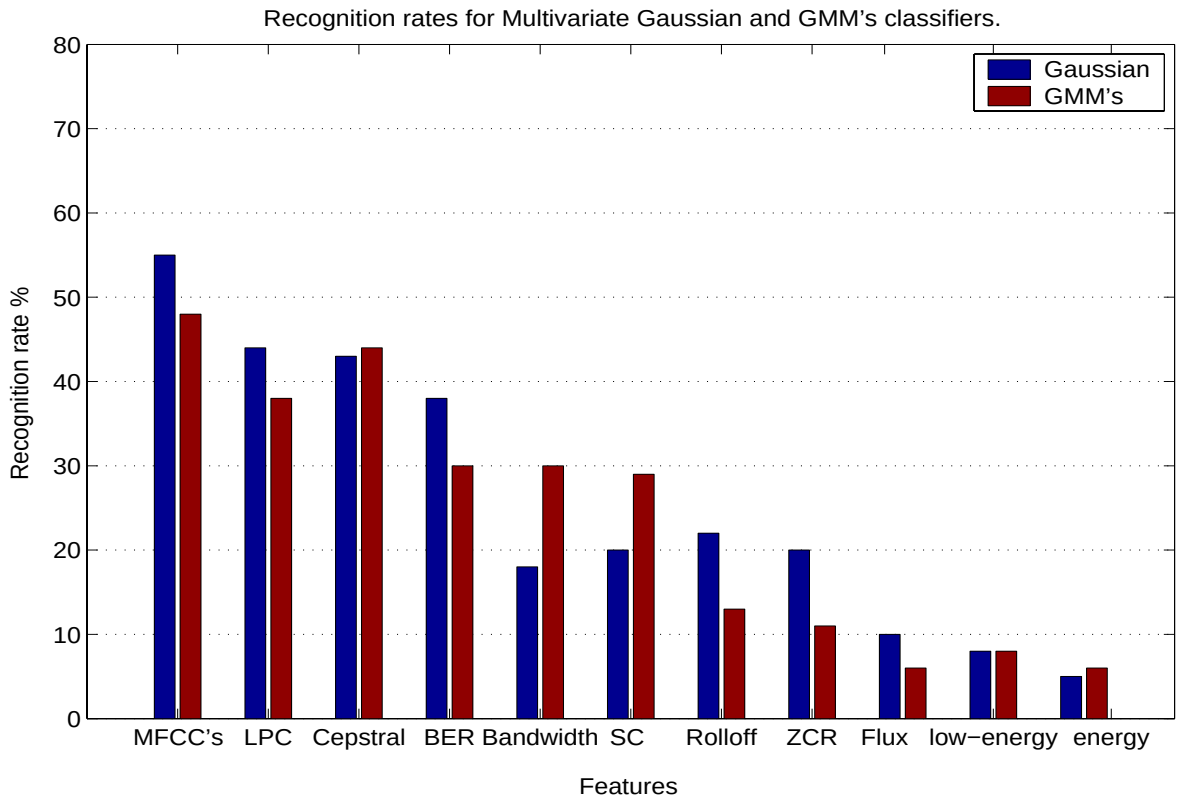

Figure 19. Recognition rates for Multivariate Gaussian classifier and GMM's classifier. The length of the tested audio sequence was 15 seconds.

Evaluation of the multivariate Gaussian classifier

The multivariate Gaussian classifier is basically a GMM classifier with one Gaussian model. Our preliminary results showed that there is no great difference between the two classifiers for short excerpts of test signals. In Figure 19, the recognition rates obtained with the multivariate Gaussian and the GMM classifiers for 11 different features are shown. The length of the test sequence was 15 seconds for the both cases, which was not adequate for accurately training the models in the GMM classifier.

Evaluation of the GMM classifier

The GMM classifier was tested with a varying number of Gaussians distributions in the mixture model. The 13 scenes described in section 7.1 were classified using model orders of 2, 4, 8, 16, and 32. The classified scenes were 30 seconds in duration and MFCC's of order 12 were used as a feature. The number of iterations in the expectation maximization (EM) algorithm was 50. In Table 17, the recognition rates as a function of the model order are presented. The classification performance seemed to saturate when the number of models was four, and then decreased. The decrease in the performance is due to an insufficient amount of training data compared to the number of parameters to be estimated.

Table 17: GMM model order vs. recognition rate

Model order	2	4	8	16	32
Recognition rate	52.03 %	56.79 %	55.19 %	53.73 %	53.09 %

The best performance for the GMM classifier was obtained using a vector of MFCC's as a feature. A 57 % correct classification rate was achieved when 30-second excerpts were used for training and testing. We tested the performance of some other features as well, using 30 seconds of audio. We obtained a recognition accuracy of 30 % using the band-energy ratio, 36 % using the band-energy ratio along with the bandwidth, 24 % using spectral centroid, and 23 % using the bandwidth alone.

Comparison of the confusions and recognition rates of the k-NN and GMM classifiers

In the next page, two confusion matrices are shown in Table 18 and Table 19. The first matrix represents the confusions of the GMM classifier for the 13 scenes. MFCC's were used as features and the analysis excerpt duration was 30 seconds. The number of instances from each class is denoted after the scene names. The rectangular box encloses the outside environments from the others.

The next matrix shows the confusions for the same 13 scenes classified with the k-NN and the mean+std approach using the band energy ratio as a feature. The analysis duration of the classified signal was 15 seconds.

As can be noticed from the matrices, most of the confusions are not of the kind a human

Table 18: Confusion matrix for MFCC's classified with the GMM classifier using 4 Gaussians. The average recognition rate was 57 %, and the length of the used audio sequence was 30 seconds. The number of instances is denoted after each context.

Presented / Classified	1. Car	2. Street	3. Road	4. Nature	5. Construction	6. Kitchen	7. Bathroom	8. Restaurant	9. Supermarket	10. Lecture	11. Office	12. Library	13. Church
1. Car (18)	**94**												
2. Street (10)		**70**	17		12.5			8					
3. Road (6)			**83**		12.5								
4. Nature (11)				**36**									
5. Construction (8)		10			**50**				17				
6. Kitchen (4)						**0**	17		16		25	33	
7. Bathroom(6)				9			**83**						
8. Restaurant (13)	6	10						**69**			25		25
9. Supermarket (6)									**50**			67	
10. Lecture (10)						50		8		**80**			
11. Office (8)		10			12.5	25					**50**		
12. Library (3)						25						**0**	
13. Church (4)													**75**
14. Others (17)				55	12.5			15	17	20			

Table 19: Confusion matrix for band-energy ratio, classified with the 1-NN classifier. The average recognition accuracy was 56 %, and the length of the used audio test sequence was 15 seconds.

Presented / Classified	1. Car	2. Street	3. Road	4. Nature	5. Construction	6. Kitchen	7. Bathroom	8. Restaurant	9. Supermarket	10. Lecture	11. Office	12. Library	13. Church
1. Car (18)	**89**												
2. Street (10)	11	**70**			13			8					
3. Road (6)			**33**	9									
4. Nature (11)			17	**64**		25	16					33	
5. Construction (8)		10	33		**37**								
6. Kitchen (4)						**0**		8					
7. Bathroom(6)						50	**67**						
8. Restaurant (13)		10		9	37	25	17	**62**	17	20			
9. Supermarket (6)				9				8	**67**		12		
10. Lecture (10)								14		**80**	25		
11. Office (8)		10							16		**63**	67	
12. Library (3)				9								**0**	
13. Church (4)													**100**
14. Others (17)			17		13								

listener would make, but more like random. Some irrational confusions were such as a construction site being recognized as a restaurant, a street as an office, and so on. However, there are a few confusions which make sense. One example is the library scene that was classified two times as an office with the 1-NN classifier (Table 19). These sound scenes have some similarities; they both have silent background with some transient sound events, such as the sound of typing or footsteps. Another example is the kitchen scene which was confused with a bathroom, both having sounds of a water faucet. There is not much correlation between the confusions of the two matrices, except the case of the street scene which has exactly the same confusions for the both classification methods.

There is a high correlation between the recognition rates of the individual scenes (diagonal values of the matrix). The average absolute difference between the recognition rates of the two classification methods was around 13 %. Figure 20 illustrates the statistical distribution of the recognition rates of the two classification methods, for individual scenes. The box in the figure outlines the lower quartile and upper quartile recognition rates, the line inside the box marks out the median value, and the recognition rates outside the box are marked with asterisks. The correlation can be seen clearly from Figure 21, where the recognition rates of individual scenes obtained with the GMM and 1-NN (mean+std) classifiers are shown side-by-side.

7.4 Recognition rate as a function of the test sequence length

We also studied the recognition rates as a function of test sequence length. We calculated the recognition rates for five different features using the k-NN (mean+std) classification method and for one feature using the GMM classifier. The features were the spectral centroid, the band-energy ratio, the bandwidth, and the MFCC with 4 and 12 coefficients for the 1-NN (mean+std) method, and MFCC with 12 coefficients for the GMM classifier. The reason for calculating only one feature for the GMM classifier was that it required very much computa-

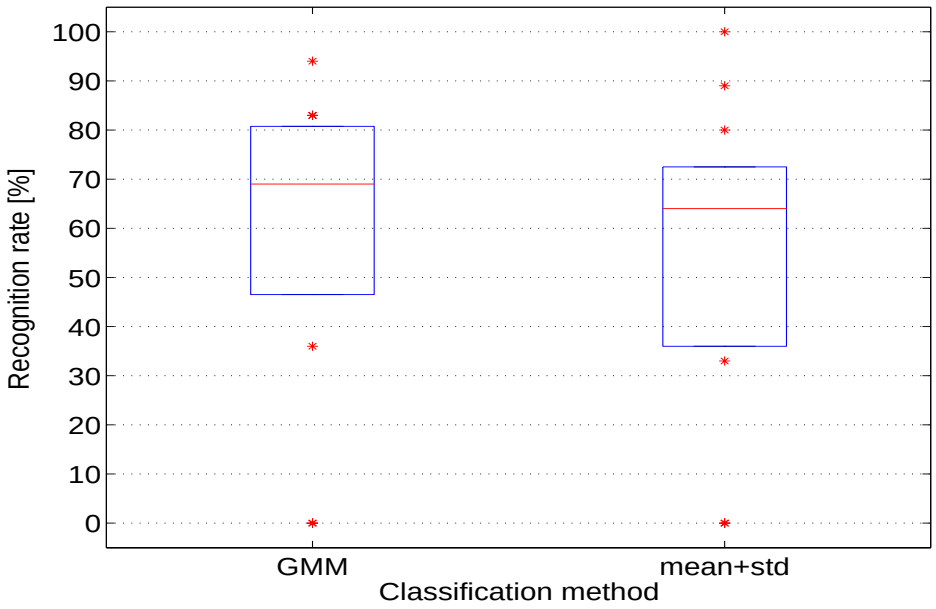

Figure 20. The distribution of the recognition rates of the individual scenes. The box outlines the lower and upper quartile values, the line inside the box marks out the median value, and the outliers are marked with asterisks.

tion time, which in turn resulted from the use of the leave-out-one testing method. The maximum length of test sequences was 60 seconds and the step size for the 1-NN classifier was one second and for the GMM classifier it was the ten seconds.

In Figure 22, the recognition rates of are plotted as a function of the test sequence length. As expected, increasing the sequence duration improves the overall classification performance. As can been seen from the figure, the trend of all curves on average is ascending. For band energy ratio, the recognition curve seems to converge at 15 seconds to the level of 55 %. However, a small increase can be noticed from 55 to 60 seconds, ending up at a recognition rate of 58 %. From the figure we can also notice a correlation between the results obtained using the spectral centroid and the bandwidth. The learning curves of these features are very close to each other, and their behavior as a function of the sequence length is very similar. Also the behavior of MFCCs with four coefficients is very close to the behavior of these two.

A curious thing which can be noticed from the figure is that the behavior of the recognition rate of the band-energy ratio is very similar to that of the MFCCs & GMM, although these two features were classified with two completely different classifiers (1-NN and GMM respectively). Also, both curves resemble the human response time as described in Figure 12 (Chapter 5). This gives some evidence that the classification systems indeed model the auditory scenes and the sound events in to some precision.

Another interesting point is the fall down in the MFCC&GMM curve at 15 seconds. Although there may be many underlying reasons, one intuitive reason is the following. At the given time there occurred new sound events in the analyzed audio excerpt and the GMM classifier did not have enough data to model these events. After accumulating more data, the sound events are modelled more accurately and the recognition rate bounces back to the initial trend of the curve. A solution to inadequate modelling is to adjust the number of Gaussians in the GMM

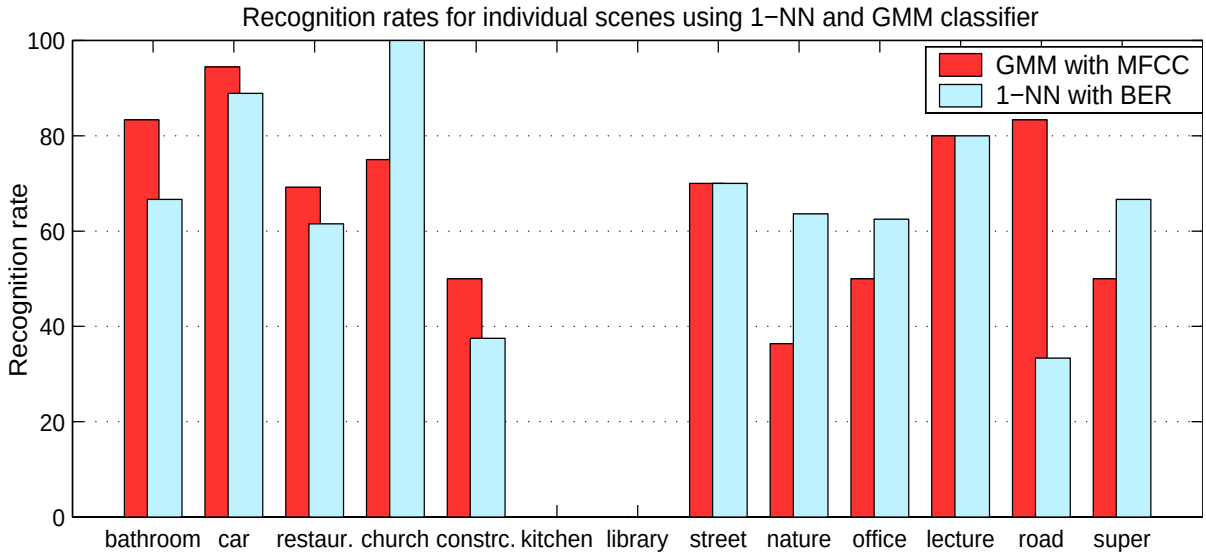

Figure 21. Recognition rates for the 13 scenes using GMM and 1-NN classifiers. The used features were MFCC's and band energy ratio respectively. The obtained recognition rate for 1-NN was 56 % (15 s) and for GMM 57 % (30 s).

according to the amount of the available data. However, this is not a simple task without knowing the statistics and the behavior of the data as a function of the time.

7.5 Recognition of metaclasses

We did experiments of recognizing more general contexts, meta-classes, using the k-NN (mean+std) classification approach. Analysis segment duration was 15 seconds. The classifier used was 1-NN and the feature vector consisted of spectral centroid (one number) and band-energy ratio (four numbers). The selection of the features was based only on a few experiments and the classification approach was selected based on previous results emphasizing the calculation efficiency. Partitioning the scenes into the metaclasses was somewhat problematic, because categorization was not explicit for all the scenes. For example, when we grouped the scenes into outdoors and indoors classes, we had several borderline cases, such as a street cafe recorded at the entrance door. These experiments are only preliminary and a lot of research has to be done in order to get conclusive results. Particularly, emphasis has to be put on finding a sensible and exploitable partition of the scenes to metaclasses.

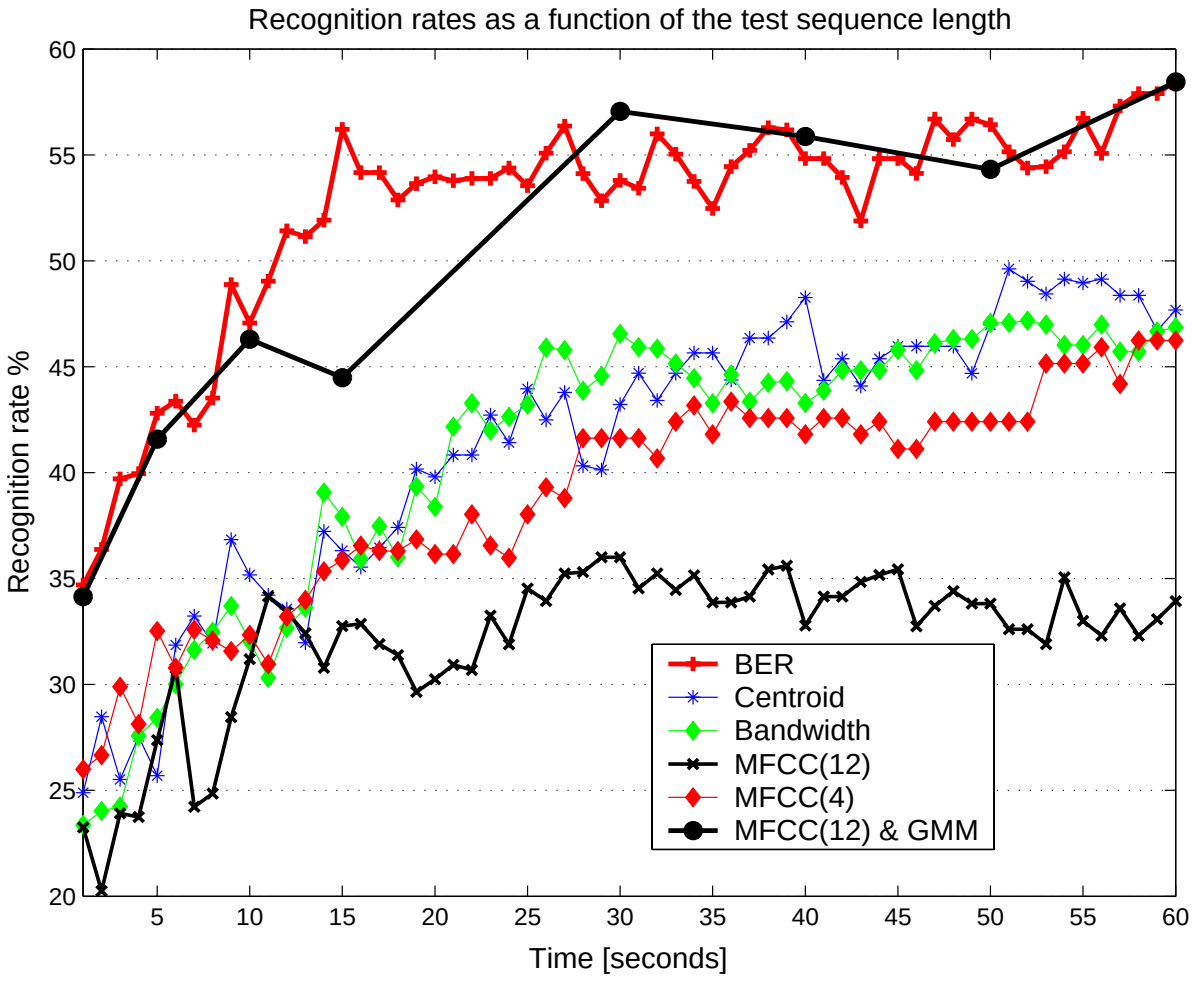

Figure 22. Recognition rates as a function of the sequence length for the following features: the band energy ratio, the spectral centroid, the bandwidth, and the MFCCs classified with the 1-NN classifier and MFCCs classified with the GMM classifier.

Table 20: Recognition rates for six meta-classes

Class	Number of samples	Recognition rate	Example
Private	33	66.7 %	home, office
Public	25	72 %	restaurant, shop
Car	18	88.9 %	inside a car
Reverberant place	10	40 %	church, railway stat.
Outdoors	20	30 %	nature
Traffic	18	27.8 %	street
Total	124	54.2 %	

The first metaclass set consisted of three classes: indoors (67 instances), outdoors (39 instances) and car (18 instances). The performance evaluation was done using the leave-one-out method and the length of the used audio sequence was 15 seconds. The overall recognition rate for this partition was 81.5 %. The recognition rates for each context were the following: indoors 94 %, car 89 %, and outdoors 61.5 % (the probability to guess the correct class randomly was 33.33 %). We did not find any consistence in the confusions between the three scenes.

The second meta-class partition included only two classes: outdoors (39) and indoors (85). The recordings of car were treated as indoors environments. The average recognition rate this time was only 76.5 %; indoors was recognized correctly in 94 % of the cases and outdoors in 59 % of the cases.

We obtained a recognition rate of 94.5 % when we tried to discriminate between the cars (18) and the other scenes (106). In this experiment only two instances of car were not recognized correctly; thus, the recognition rates for the two classes were 88.9 % and 100 % correspondingly.

The last examined partition included six classes. In Table 20 the classes, the number of samples from each, the recognition rates and examples of scenes in each class are listed. The evaluation process was the same as described in the previous experiments. The average recognition rate was 54.2 %.

8 Summary and conclusions

The problem of computational auditory scene recognition (CASR) has been studied, with the aim of developing techniques that enable a machine to be aware of the context by acoustic information only. The focus of this work was to study and investigate the usability of different acoustic features and pattern classification methods in CASR systems.

An indispensable part of the work was to collect a comprehensive audio database suitable for listening tests and for the development of CASR systems. The acoustic measurements consumed a lot of time and resources, but were worth the effort, since the amount of the training data is the first bottleneck in evaluating a new classification system.

The listening test showed that humans are able to recognize an auditory scene in 70 % on the average and that the response time was of the order of 20 seconds. The subjects reported that the recognition was based on prominent identified sound events. These results suggests us that an accurate CASR system should analyze relatively long fragments of audio in making the inferences, i.e. tens of seconds of audio, and the focus on the recognition process should be put on modeling distinct sound events.

In this thesis, we have presented two promising classification schemes for CASR systems. The first scheme was based on averaging the band-energy ratio over one second windows and classifying the features with the k-nearest neighbor classifier. The features used in the second classification scheme were mel-frequency cepstral coefficients and they were classified using Gaussian mixture models. The experimental results show that the firstly suggested classification system provides a recognition accuracy of 56 % for 13 scenes and for analysis duration of 15 seconds. With the second method, we achieved about the same accuracy (57 %), for analysis duration of 30 seconds. Both of the two presented methods can be understood to model individual prominent sound events to a certain level of accuracy. The similar results of the two different classification approaches indicate that it is difficult to obtain considerably better results with this audio material and these scenes. Preliminary results were given for the recognition of more general classes, called metaclasses. The results show that for certain categorizations, a very good recognition accuracy can be obtained. For example, a car environment can be distinguished from all others with 95 % accuracy.

The research presented in this thesis, is in its earliest stage and gave only a launch pad for the study of the problem. A lot of work is still to be done before we have a complete and reliable CASR system. Although further research is needed to investigate the presented approaches, a promising direction is to recognize scenes by identifying distinct sound events and associating them to specific auditory environments. A future vision is to construct an adaptive system capable of learning new environments and of applying higher-level knowledge in making the decisions.

References

[Ballas93] Ballas, J.A. “Common Factors in the Identification of an Assortment of Brief Everyday Sounds”. Journal of Experimental Psychology: Human Perception and Performance, 19(2), pp. 250-267, 1993.

[Baum70] Baum, L. E., Petrie, T., Soules, G.m and Weiss, N. “A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains”.The Annals of Mathematical Statistics, 41(1):164-171, 1970.

[Baum72] Baum, L. E. “An inequality and associated maximization technique in statistical estimation for probabilistic functions of a markov process”. Inequalities, 3:1-8, 1972.

[Bregman90] Bregman, A. S. “Auditory Scene Analysis: The Perceptual Organization of Sound”. Cambridge, Massachusetts: MIT Press, 1990.

[Brown94] Brown, G.J. and Cooke, M.P. “Computational auditory scene analysis”. Computer Speech and Language, 8, 297-336, 1994.

[Brown99] Brown, J. C. “Computer identification of musical instruments using pattern recognition with cepstral coefficients as features”. J. Acoust. Soc. Am. 105(3), March 1999.

[Carey99] Carey M.J., Parris, E.S., and Lloyd-Thomas, H. “A comparison of features for speech, music discrimination”. In Proceedings of the 1999 IEEE international Conference on Acoustics, Speech, and Signal Processing, volume 1, pp. 149 - 15, 1999.

[Clarkson98a] Clarkson, B. P., Sawhney, N., and Pentland, A. “Auditory Context Awareness via Wearable Computing”. Proceedings of the Perceptual User Interfaces Workshop, San Francisco, CA, 1998.

[Clarkson98b] Clarkson, B. P. and Pentland, A. “Extracting Context From Environmental Audio”. Proceedings of the 2nd International Symposium on Wearable Computers Pittsburgh, Pennsylvania, October 1998.

[Cover67] Cover, T. M., and Hart P. E. “Nearest Neighbor Pattern Classification”. IEEE Transactions on Information Theory, volume 13, no. 1, pp. 21-27, January 1967.

[Dempster77] Dempster, A. P., Laird, N. M., and Rubin, D. B. “Maximum likelihood for incomplete data via the EM algorithm”. Journal of the Royal Statistical Society, Series B, volume 39, pp. 1-38, 1977.

[Ellis96] Ellis, D.P.W. “Prediction-driven computational auditory scene analysis”. Ph.D. Thesis, Massachusetts Institute of Technology, 1996.

[El-Maleh99] El-Maleh, K., Samouelian, A., and Kabal, P. “Frame level noise classification in mobile environments”. In Proceedings of the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing, volume 1, pp. 237 - 240.

[El-Maleh00] El-Maleh, K., Klein, M., Petrucci, G., and Kabal, P. “Speech/music discrimina-

tion for multimedia applications". In Proceedings of the 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing, volume 4, pp. 2445 -2448, Istanbul, 2000.

[Eronen01] Eronen, A. "Comparison of features for musical instrument recognition". In Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2001.

[Fujinaga00] Fujinaga, I. "Realtime recognition of orchestral instruments". Proceedings of the International Computer Music Conference, 2000.

[Fraser99] Fraser, A. and Fujinaga, I. "Towards real-time recognition of acoustic musical instruments". Proceedings of the International Computer Music Conference, 1999.

[Gaunard98] Gaunard, P., Mubikangiey, C.G., Couvreur, C., and Fontaine, V. "Automatic classification of environmental noise events by hidden Markov models". In the Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, volume 6, pp. 3609 - 3612, May 1998.

[Goldhor93] Goldhor, R. S. "Recognition of Environmental Sounds". Proceedings of the IEEE, volume 1, pp. 149-152, 1993.

[Guojun00] Guojun L. and Hankinson, T. "An investigation of automatic audio classification and segmentation". Proceedings of the 5th International Conference on Signal Processing (WCCC-ICSP 2000), volume 2, pp. 1026 -1032, 2000.

[Herrera00] Herrera, P., Amatriain, X., Batlle, E., and Serra, X. "Towards instrument segmentation for music content description: a critical review of instrument classification techniques". In Proceedings of the International Symp. on Music Information Retrieval (ISMIR2000). Plymouth, MA. October 2000.

[Jarnicki98] Jarnicki, J., Mazurkiewicz, J., and Maciejewski, H. "Mobile Object Recognition Based on Acoustic Information". In Proceedings of the 24th Ann. Conf. of the IEEE Industrial Electronics Society, IECON'98, volume 3, pp. 1564-1569, 1998.

[Kaminskyj95] Kaminskyj, I., and Materka, A. "Automatic Source Identification of Monophonic Musical Instrument Sounds". Proceedings of the IEEE Int. Conf. on Neural Networks, 1995.

[Kay93] Kay, S. M. "Fundamentals of Statistical Signal Processing: Estimation Theory". Prentice Hall, 1993.

[Klassner96] Klassner, F. "Data Reprocessing in Signal Understanding Systems". Ph.D. thesis, University of Massachusetts at Amherst, Amherst, Massachusetts, September 1996.

[Kostek99] Kostek, B. "Soft Computing in Acoustics: Applications of Neural Networks, Fuzzy Logic and Rough Sets to Musical Acoustics". Physica-Verlag, 1999.

[Laerhoven99] Van Laerhoven, K. "Online Adaptive Context Awareness". Licentiate thesis at the University of Brussels (VUB), Brussels, 1999.

[Li01] Li, D., Sethi, I.K., Dimitrova, N., and McGee, T. "Classification of general audio data for content-based retrieval". Pattern Recognition Letters (22), No. 5, pp. 533-544, April 2001.

[Liu97] Liu, Z., Huang, J., Wang, Y., and Chen, T. "Audio Feature Extraction & Analysis for Scene Classification". Workshop on Multimedia Signal Processing, IEEE Signal Processing Society, Princeton, New Jersey, USA, June 1997.

[Liu98] Liu, Z., Wang, Y., and Chen, T. “Audio Feature Extraction and Analysis for Scene Segmentation and Classification”. Journal of VLSI Signal Processing System, June 1998.

[Ljung87] Ljung, L. “System Identification: Theory for the User”. Englewood Cliffs, NJ: Prentice Hall, pp. 278-280, 1987.

[Markel76] Markel, J. and Gray, A. “Linear Prediction of Speech”. Springer-Verlag, New York, 1976.

[Martin98] Martin, K. and Kim, Y. “Musical instrument identification: A pattern-recognition approach”. Proc. of the 136th meeting of the Acoustical Society of America, 1998.

[Martin99] Martin, K. “Sound Source Recognition: A Theory and Computational Model”, Ph.D. thesis, MIT, 1999.

[Mellinger91] Mellinger, D. K. “Event Formation and Separation in Musical Sounds”. Ph.D. Thesis, Report No. STAN-M-77, Department of Music, Stanford University, CA, 1991.

[Moreno00] Moreno, P.J. and Rifkin, R. “Using the Fisher kernel method for Web audio classification”. Proceedings of the 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing, volume 4, pp. 2417 - 2420, June 2000.

[Nakajima99] Nakajima, Y., Yang Lu, Sugano, M., Yoneyama, A., Yamagihara, H., and Kurematsu, A. “A fast audio classification from MPEG coded data”. Proceedings of the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing, volume 6, pp. 3005 - 3008, March 1999.

[Peltonen01] Peltonen,V., Eronen, A., Parviainen, M., and Klapuri, A. “Recognition of Everyday Auditory Scenes: Potential, Lantencies and Cues”. 110th Audio Engineering Society Convention, Amsterdam, Netherlands, May 2001.

[Rabiner93] Rabiner, L., and Juang, B.-H. “Fundamentals of Speech Recognition”. Prentice-Hall, Englewood Cliffs, NJ, 1993.

[Reynolds95] Reynolds, D. A. “Automatic Speaker Recognition Using Gaussian Mixture Speaker Models”. Lincoln Laboratory Journal, volume. 8, No. 2, pp. 173-192, Fall 1995.

[Rosenthal98] Rosenthal, D. F. and Okuno, H. G. (editors) “Computational Auditory Scene Analysis”. Lawrence Erlbaum Associates, Mahwah, New Jersey, 1998.

[Saint95] Saint-Arnaud, N. “Classification of Sound Textures”. M.S. Thesis in Media Arts and Sciences, MIT, September 1995.

[Saunders96] Saunders, J. “Real-time discrimination of broadcast speech/music”. In Proceedings of the 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing, volume 2, pp. 993 - 996, Atlanta, GA, May 1996.

[Sawhney97] Sawhney, N. “Situational Awareness from Environmental Sounds”. Project Report, Speech Interface Group, MIT Media Lab, June 1997.

[Sawhney98] Sawhney, N. “Contextual Awareness, Messaging and Communication in Nomadic Audio Environments”. M.S. Thesis in Media Arts and Sciences, MIT Media Lab, May 1998.

[Scheirer97] Scheirer, E. D. and Slaney, M. “Construction and Evaluation of A Robust Multifeature Speech/Music Discriminator”. In Proceedings of the 1997 IEEE Conference on Acoustics, Speech and Signal Processing, volume 2, pp. 1331 - 1334,

Munich, Germany, April 1997.

[Spina96] Spina, M. and Zue, V. “*A*utomatic Transcription of General Audio Data: Preliminary Analysis”. In Proceedings of the Fourth International Conference on Spoken Language (ICSLP 1996), volume 2, pp. 594-597, October 1996.

[Sriniva99] Srinivasan, S., Ponceleon, D., and Petkovic, D. “Towards Robust Features for Classifying Audio in the CueVideo System”. Proceedings of the seventh ACM international conference on Multimedia, pp. 393 - 400, Orlando, FL USA, 1999.

[Streicher98] Streicher, R. and Everest, F. A. “The New Stereo Soundbook”. Audio Engineering Associates, California, 1998. ISBN 0-8306-3903-9.

[Schürmann96] Schürmann, J. “Pattern Classification: A Unified View of Statistical and Neural Approaches”. John Wiley and Sons Inc., Toronto, 1996.

[Therrien89] Therrien, C. W. “Decision estimation and classification”. John Wiley and Sons, New York, 1989.

[Tou74] Tou, J. T. and Gonzalez, R. C. “Pattern Recognition Principles”. Addison-Wesley Publishing Company, Massachusetts, 1974.

[Williams99] Williams, G. and Ellis, D. “Speech/music discrimination based on posterior probability features”. Presented at Eurospeech-99, volume 2, pp. 687-690. Budapest, Hungary, September 1999.

[Zhang99] Zhang, T. and Kuo, C.-C. J. “Hierarchical classification of audio data for acrhiving and retrieving”. In 1999 IEEE International Conference on Acoustics, Speech and Signal Processing, volume 6, pp 3001-3004, March 1999.

[Zhang00] Zhang, Tand Kuo, C.-C. J. “Content-based audio classification and retrieval for audiovisual data parsing”. The Kluwer International Series in Engineering and Computer Science, 160 pp., December 2000.

[Dufournet98] Dufournet, D; Jouenne, P.; Rozwadowski, A. (1998), “Automatic Noise Source Recognition”. In Proceedings of the 16th International Congress on Acoustics and 135th Meeting Acoustical Society of America (ICA/ASA 1998), Seattle, Washington, 1998.

[Wang00] Wang, Y.; Liu, Z.; Huang, J-C. “Multimedia Content Analysis”. In IEEE Signal Processing Magazine, November 2000.

[Wu98] Wu, H., Siegel, M., and Khosla, P. “Vehicle Sound Signature Recognition by Frequency Vector Principal Component Analysis”. Proceedings of the IEEE Instrumentation and Measurement Technology Conference, 1998.

[Zacharov01] Zacharov, N. and Koivuniemi, K. “Unraveling the Perception of Spatial Sound Reproduction: Techniques and Experimental Design”. Audio Engineering Society 19th International Conference on Surround Sound, Techniques, Technology and Perception, Schloss Emlau, Germany, June 2001.

Appendix A: Derivations

Estimation of the mean vector $\bar{\mu}$ and the covariance matrix Σ

Here we provide the derivation of the estimators of the mean vector $\bar{\mu}$ and the covariance matrix Σ. If we approximate the expected value as the sample average, then the mean value can be written as

$$\bar{\mu} = E\{\boldsymbol{x}\} = \int_{\boldsymbol{x}} \boldsymbol{x} p(\boldsymbol{x}) d\boldsymbol{x} \approx \frac{1}{N} \sum_{i=1}^{N} x_i = \hat{\mu}, \tag{A1}$$

where N is the number of samples of vector $\boldsymbol{x}$ and $\hat{\mu}$ is the estimated value of the mean vector. The covariance matrix for vector $\boldsymbol{x}$ is given by

$$\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & & & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix} \tag{A2}$$

with the element of Σ being defined as

$$\sigma_{ij} = E\{(x_i - \mu_i)(x_j - \mu_j)\}, \tag{A3}$$

where x_i, x_j and μ_i, μ_j are the ith and jth elements of $\boldsymbol{x}$ and $\bar{\mu}$. The covariance matrix may be expressed in the following vector form:

$$\begin{aligned} \Sigma &= E\{\langle \boldsymbol{x} - \bar{\mu} \rangle \langle \boldsymbol{x} - \bar{\mu} \rangle\}^T \\ &= E\{\boldsymbol{x}\boldsymbol{x}^T - 2\boldsymbol{x}\bar{\mu}^T + \bar{\mu}\bar{\mu}^T\} \\ &= E\{\boldsymbol{x}\boldsymbol{x}^T\} - E\{2\boldsymbol{x}\bar{\mu}^T\} + E\{\bar{\mu}\bar{\mu}^T\} \end{aligned} \tag{A4}$$

Approximating again, the expected value with the sample averages yields to

$$\Sigma \approx \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{x}_i - \hat{\mu})(\boldsymbol{x}_i - \hat{\mu})^T \tag{A5}$$

And finally, we have the estimation value for the covariance matrix

$$\hat{\Sigma} = \frac{1}{N} \sum_{i=1}^{N} x_i x_i^T - \hat{\mu}\hat{\mu}^T. \tag{A6}$$