# CONTENT-BASED AUDIO SEARCH:
# FROM FINGERPRINTING TO SEMANTIC AUDIO RETRIEVAL

A DISSERTATION SUBMITTED TO THE
TECHNOLOGY DEPARTMENT OF POMPEU FABRA UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR FROM THE POMPEU FABRA UNIVERSITY
—
DOCTORATE PROGRAM OF
COMPUTER SCIENCE AND DIGITAL COMMUNICATION

Pedro Cano

2007

Dipòsit legal: B.42899-2007
ISBN: 978-84-691-1205-2

THESIS DIRECTION

**Thesis Advisor**

Dr. Xavier Serra
Departament de Tecnologia
Universitat Pompeu Fabra, Barcelona

# Abstract

This dissertation is about audio content-based search. Specifically, it is on developing technologies for bridging the semantic gap that currently prevents wide-deployment of audio content-based search engines. Audio search engines rely on metadata, mostly human generated, to manage collections of audio assets. Even though time-consuming and error-prone, human labeling is a common practice. Audio content-based methods, algorithms that automatically extract description from audio files, are generally not mature enough to provide a user friendly representation for interacting with audio content. Mostly, content-based methods are based on low-level descriptions, while high-level or semantic descriptions are beyond current capabilities. This dissertation has two parts. In the first one we explore the strengths and limitation of a pure low-level audio description technique: audio fingerprinting. We prove, by implementation of different systems, that automatically extracted low-level description of audio are able to successfully solve a series of tasks such as linking unlabeled audio to corresponding metadata, duplicate detection or integrity verification. We show that the different audio fingerprinting systems can be explained with respect to a general fingerprinting framework. We then suggest that the fingerprinting framework, which shares many functional blocks with content-based audio search engines, can eventually be extended to allow for content-based similarity type of search, such as find similar or "query-by-example". However, low-level audio description cannot provide a semantic interaction with audio contents. It is not possible to generate a verbose and detailed descriptions in unconstraint domains, for instance, for asserting that a sound corresponds to "fast male footsteps on wood" but rather some signal-level descriptions.

In the second part of the thesis we hypothesize that one of the problems that hinders the closing the semantic gap is the lack of methods that make use of common sense knowledge and that the inclusion of such a knowledge base is a primary step toward bridging the semantic gap. For the specific case of sound effects, we propose a general sound classifier capable of generating verbose descriptions in a representation that computers and users alike can understand. We conclude the second part with the description of a complete sound effects retrieval system which leverages both low-level and semantic technologies and that allows for intelligent interaction with audio collections.

# Resum

Aquesta tesis tracta de cercadors d'audio basats en contingut. Específicament, tracta de desenvolupar tecnologies que permetin fer més estret l'interval semàntic o "semantic gap" que, a avui dia, limita l'ús massiu de motors de cerca basats en contingut. Els motors de cerca d'àudio fan servir metadades, en la gran majoria generada per editors, per a gestionar col.leccions d'àudio. Tot i ser una tasca àrdua i procliu a errors, l'anotació manual és la pràctica més habitual. Els mètodes basats en contingut àudio, és a dir, aquells algorismes que extreuen automàticament etiquetes descriptives de fitxers d'àudio, no són generalment suficientment madurs per a permetre una interacció semàntica. En la gran majoria, els mètodes basats en contingut treballen amb descriptors de baix nivell, mentre que els descriptors d'alt nivell estan més enllà de les possibilitats actuals.

Aquesta dissertació té dos parts. En la primera explorem els avantatges i limitacions d'una tècnica que treballa amb descriptors de baix nivell: audio fingerprinting. Provem, mitjançant la implementació de diversos sistemes, que l'extracció automàtica de descriptors d'àudio de baix nivell és suficient per resoldre una sèrie de tasques com ara identificació d'àudio, detecció de duplicats o verificació d'integritat. Mostrem que els diversos sistemes de fingerprinting es poden explicar amb un marc general. Suggerim llavors que el marc o diagrama de blocs proposat, el qual comparteix molts blocs funcionals amb els cercadors basats en contingut, es pot extendre per acomodar cerques de semblança. No obstant això, els descriptors de baix nivell no poden oferir una interacció semàntica amb col.leccions d'àudio. No és possible generar una descripció suficientment detallada com ara: aquest so correspon a "passes ràpides d'home sobre fusta", sinó una descripció més a nivell de senyal.

En la segona part de la tesi hipotitzem que un dels problemes que complica l'estretament de l'interval semàntic és la manca de mètodes que incorporin coneixement de sentit comú i que la inclusió d'aquest coneixement és un pas previ per abordar l'interval semàntic. Pel cas específic d'efectes de so, proposem un sistema de descripció d'àudio per a qualsevol tipus de so de manera expressiva en una format de representació que puguin entendre tant els humans com els ordinadors. Concluïm la segona part amb la descripció d'un cercador d'efectes de so complert que aprofita tant tecnologies basades en descriptors de baix nivell així com tecnologies semàntiques i que permete una avançada amb col.leccions d'àudio.

# Resumen

Esta disertación trata de búsqueda de áudio basada en el contenido. Específicamente, trata de desarrollar tecnologías que permitan estrechar el intérvalo semántic o “semantic gap” que a día de hoy limita el uso masivo de motores de búsqueda basados en contenido. Los motores de búsqueda de áudio utilizan metadatos, en su mayoría generado por editores, para gestionar las colecciones de áudio. Pese a ser una tarea árdua y proclive a errores, el etiquetado manual es la práctica común. Los métodos basados en contenido áudio, es decir, aquellos algoritmos que extraen automáticamente etiquetas descriptivas de ficheros de áudio, son generalmente inmaduros para proveer una interacción semántica. En su mayoría, los métodos basados en contenido, trabajan con descriptores de bajo nivel, mientras que los descriptores de alto nivel están más allá de las prestaciones actuales.

Esta disertación consta de dos partes. En la primera, exploramos las ventajas y limitaciones de una técnica que trabaja con descriptores de bajo nivel: audio fingerprinting. Probamos, mediante la implementación de diversos sistemas, que la extracción automática de descriptores de áudio de bajo nivel es suficiente para resolver un serie de tareas tales como identificación de áudio, detección de duplicados o verificación de integridad. Mostramos que los diferentes sistemas de fingerprinting se pueden explicar con respecto a un marco general. Sugerimos entonces que el marco o diagrama de bloques propuesto, el cual comparte muchos bloques funcionales con los motores de búsqueda basados en contenido, puede extenderse para acomodar búsquedas de semejanza. Sin embargo, los descriptores de bajo nivel no pueden proporcionar una interacción semántica con los contenidos de áudio. No es posible generar una descripción lo suficientemente expresiva y detallada tal como: este sonido corresponde a ”pasos rápidos de hombre sobre madera” sino una descripción más cercana a la señal.

En la segunda parte de la tesis hipotizamos que uno de los problemas que complica el estrechamiento del intérvalo semántico es la falta de métodos que incorporen conocimiento de sentido común y que la inclusión de dicho conocimiento es un paso previo para abordar el intérvalo semántico. Para el caso específico de los efectos de sonido, proponemos un sistema de extración de descripciones de cualquier tipo de sonido de manera expresiva en una representación capaz de ser comprendida tanto por humanos como por computadores. Concluímos la segunda parte con la descripción de un buscador de efectos de sonidos completo que aprovecha tanto tecnologías basadas en descriptores de bajo nivel así como tecnologías semánticas y que permite una interacción avanzada con colecciones de áudio.

# Acknowledgements

I would first like to thank my supervisor Dr. Xavier Serra for giving me the opportunity of being part of the Music Technology Group back 1997 and the guidance and confidence to pursue this work. Research on audio fingerprinting was partly supervised by Eloi Batlle, so I thank him along the rest of the AIDA crew: Jose Pedro Mahedero, Vadim Tarasov, Jaume Masip and Enric Guaus. I thank as well the partners of the RAA project, specially Helmut Neuschmied and Harald Mayer. In reviewing fingerprinting algorithms I owe very helpful collaboration and discussions to Ton Kalker and Jaap Haitsma. To understand the relationship between fingerprint and watermarking I am in debt to Emilia Gómez, Leandro de C.T. Gomes and Madeleine Bonnet. With them I also collaborated in the mixed fingerprinting and watermarking integrity method. This thesis would certainly have not been possible without the work of the AudioClas team, specially Markus Koppenberger and Nicolas Wack. The audio database used in the sound experiments is courtesy of the SoundEffectsLibrary of the Tape Gallery. I need to thank Lloyd Billing, Peter Stanfield and the rest of the Tape Gallery' stuff for their professional counseling, feedback and support. José Lozano, the studio man at the MTG, generously shared his audio post-production expertise. I would specially like to thank Fabien Gouyon for ever reading and discussing manuscripts and being a source of inspiration as well as Perfecto Herrera, a renaissance man, always ready to give advice. I thank Oscar Celma for his collaboration and advice in ontologies and standards. I thank Gunnar Holmberg and Wei Li for throughly reviewing this work and providing critical comments. Thanks to my first colleagues at the MTG Alex Loscos for his fine tuned irony and for making me laugh of myself and Jordi Bonada, an amazing source of resources. Thanks to Lars Fabig and Alvaro Barbosa for their collaboration in the ambiance generation. Thanks to Salvador Gurrera for his confidence and

help. Thanks to Cristina and Joana for their support. Thanks the technical squat Ramon and Carlos. Thanks to the long list of researchers around the globe I have met in different conferences. Specially thanks to the researchers and staff of the Music Technology Group. Thanks to my family and thanks to Chiara.

This research has been pursued in the Music Technology Group of the Universitat Pompeu Fabra partially funded by the following EU projects:

- RAA (IST-1999-12585, http://raa.joanneum.ac.at) Recognition and Analysis of Audio,
- AUDIOCLAS (Eureka Project E!2668, http://audioclas.iua.upf.edu), automatic audio classification,
- SALERO (IST FP6-027122, http://www.salero.eu) Semantic Audiovisual Entertainment Reusable Objects.
- CANTATA (ITEA 05010) Content Analysis & Networked Technologies toward Advanced and Tailored Assistance.

# Contents

# List of Figures

# Chapter 1

# Introduction

## 1.1 Motivation

The standardization of personal computers, the ubiquity of high-storage devices, the proliferation of Peer2Peer networks and world-wide low-latency networks have dramatically increased digital audio growth and access. Major music labels now provide their music catalogs in nearly CD audio quality formats through on-line distributors such as Apple iTunes, eMusic or Yahoo! Music. The media industry is demanding tools to help organize large and growing amounts of digital audio content more efficiently and automatically. Users are starting to own thousands of audio files in their PCs and portable devices.

Access and management of audio content has mainly relied on manual annotations. Manual annotation of content raises a number of difficulties. It is a time-consuming and error-prone task when performed by humans. Moreover, different individuals tend to tag with different conventions. Meanwhile, the amount of digital content is skyrocketing. There are 30,000 CDs being released every year. There are millions of audio tracks in P2P networks.[1]

At the same time while the amount of audio content explodes, a great deal of cultural heritage content of great interest remains undisclosed because content providers' lack of confidence in wide content distribution through the web due to fears for unauthorized content usage and missed returns on investment in content creation, acquisition, transformation and

[1] see http://www2.sims.berkeley.edu/research/projects/how-much-info-2003

distribution. These and other problems call for automatic solutions to analyze, describe and index assets.

In the field of audio content-based retrieval there has been a plethora of interesting research and developments aiming at solving the above issues. Most systems use low-level features such as spectral centroid or mel-frequency coefficients while users prefer to interact at a higher semantic level: "I need a sound of a big dog barking angrily". Low-level features relate to the acoustic properties of audio and hence can be extracted automatically using signal processing and machine learning techniques. High-level features are more user-friendly descriptions that can currently be derived automatically only on very constraint domains (a few classes such as speech/music, or percussive instruments) where domain-specific knowledge is leveraged in a top-down approach. One of the major failings of current automatic media annotation systems relates to the *semantic gap* which refers to the discontinuity between the simplicity of features or content descriptions that can be currently computed automatically and the richness of semantics in user queries posed for media search and retrieval (Dorai and Venkatesh, 2001).

In a first part, this dissertation explores the capabilities of a pure low-level audio description technique for managing media assets: audio fingerprinting. We prove its usefulness implementating several fingerprinting systems.

In a second part we explore ways of bridging the semantic gap by developing automatic methods for high-level description and interaction. We validate the different proposals on a specific audio type: isolated sound effects. Existing systems for high-level description mainly rely on automatic classification. These systems require a classification scheme and examples of each class (e.g.: a taxonomy of musical genres with some examples of tracks per genre). These approaches achieve good results but in constrained domains. They are generally not scalable— they operate on a few tens of classes at most when a real-world sound effects dataset has tens of thousands of classes— nor adaptable should new classes be added to the system. To overcome the scalability and adaptability we propose a memory-based classifier. Moreover, we hypothesize that another of the issues that hinders bridging the semantic gap is the lack of general knowledge base that encodes common sense information, such as the fact that "some doors are made of wood", "cars have engines and horns" or that "cats miaow, purr and hiss". Pursuing this goal we design a general sound classifier with such a knowledge base whose performance is validated on a large database of sound effects. A complete sound effects retrieval system which leverages both low-level and semantic technologies is also proposed, implemented and validated.

## 1.2 Application scenarios

In order to evaluate the technologies proposed in this thesis we have worked in different specific scenarios for content-based retrieval: audio fingerprinting and sound effects management. In this section, we justify that there is a real need for the technologies researched.

### 1.2.1 Audio fingerprinting

In this subsection we justify some of the market needs for fingerprinting technologies. Music copyright enforcement is not a new issue. The recording industry has been fighting piracy since its very early times. However, the digital revolution in audio has brought this fight to a new level, as music in digital format can be copied and distributed easily and with no degradation. Electronic distribution, particularly the Internet, associated with efficient compression algorithms (such as MP3 and AAC) and peer-to-peer file-sharing systems create an environment that is prone to music piracy.

*Watermarking* has been proposed as a potential solution to this problem. It consists in embedding a mark, the watermark, into the original audio signal. This mark should not degrade audio quality, but it should be detectable and indelible. Compliant devices should check for the presence of a watermark before proceeding to operations that could result in copyright infringement. Research in this field has been very active over the last years. In particular, the Secure Digital Music Initiative consortium (SDMI), which brings together the major actors in the recording and consumer-electronics industries, has recently released technology specifications intended to protect, by means of a watermark, the playing, storing and distribution of music in digital format (SDMI, 2001). This technology was submitted to public evaluation through a “challenge” inviting individuals to defeat SDMI’s protection system, a goal that was shortly achieved, showing that the technology was not ready for commercial purposes (Craver et al., 2001; Boeuf and Stern, 2001).

Another approach to the copyright-protection problem, quite different from watermarking or any of the other alternatives (see Figure 1.1) in its conception, consists in analyzing an audio signal and constructing a “fingerprint” that is uniquely associated with this signal. *Automatic music recognition* or *fingerprinting* systems (see Chapter 2) can identify a song by searching for its fingerprint in a previously constructed database. Such systems are being used, for example, to monitor music transfers in Napster-like file-sharing facilities, blocking transfers of copyrighted material or collecting the corresponding royalties, and to track audio content played by broadcasters.

At the same time that we experience an explosion of available content, there is a great deal of content of major interest that is not distributed. Some factors are currently preventing a widespread distribution from the contents providers to make accessible cultural heritage content:

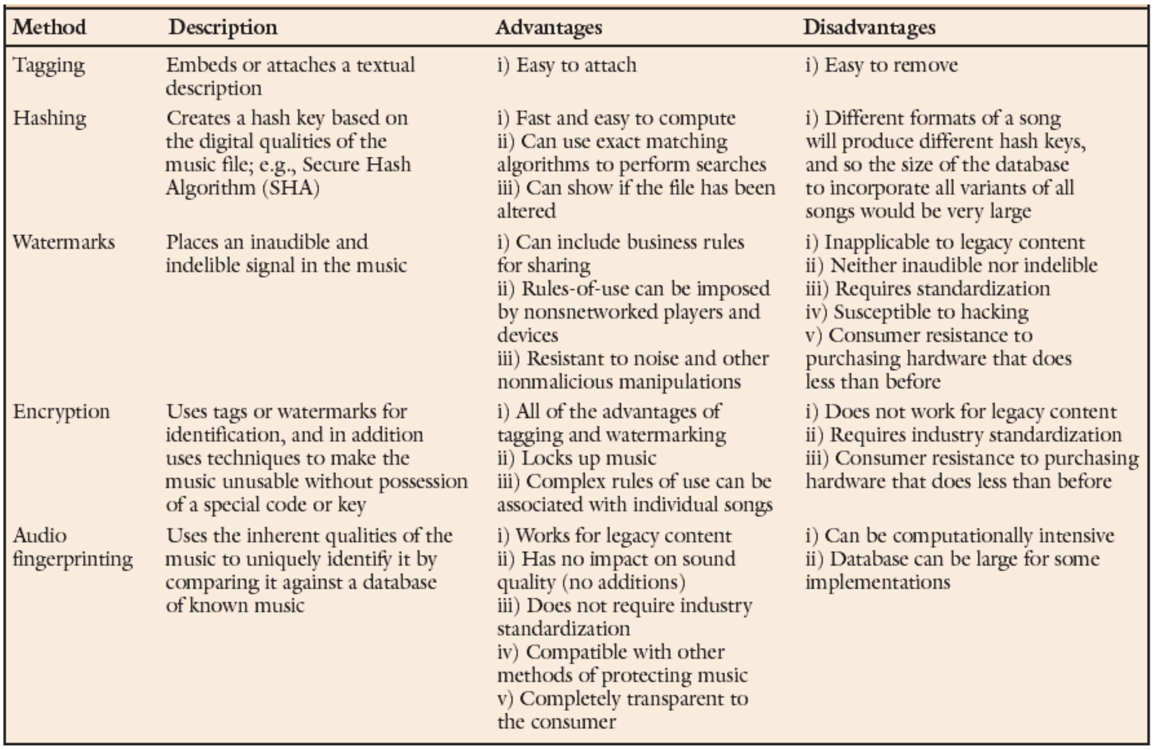
- Content providers’ lack of confidence in wide content distribution through the web due to fears for unauthorized content usage and missed returns on large investment in content creation, acquisition / transformation and distribution.

- Interoperability issues at level of content formats and metadata, through networks, organizations, public/private and among market sectors.

There is great interest to develop systems that are capable of tracking and monitor audio content regardless of the distortion and media format for digital right management. However both in the industry as well as consumer level there is a demand for added-value services that automatically connect audio content to the corresponding metadata, detect duplicates in large databases or even check the integrity of certain audio content.

### 1.2.2 Sound effects management

The audio component is a fundamental aspect in an audiovisual production. According to the staff of the Tape Gallery, a post-production facility in London, around 75% of the sound effects (SFX) of a movie are added during post-production. Originally captured sounds are frequently useless due to the noise in the recording session and some are simply not picked up by the production microphones. Sometimes sounds are replaced in order to



| Method | Description | Advantages | Disadvantages |
|---|---|---|---|
| Tagging | Embeds or attaches a textual description | i) Easy to attach | i) Easy to remove |
| Hashing | Creates a hash key based on the digital qualities of the music file; e.g., Secure Hash Algorithm (SHA) | i) Fast and easy to compute<br>ii) Can use exact matching algorithms to perform searches<br>iii) Can show if the file has been altered | i) Different formats of a song will produce different hash keys, and so the size of the database to incorporate all variants of all songs would be very large |
| Watermarks | Places an inaudible and indelible signal in the music | i) Can include business rules for sharing<br>ii) Rules-of-use can be imposed by nonsnetworked players and devices<br>iii) Resistant to noise and other nonmalicious manipulations | i) Inapplicable to legacy content<br>ii) Neither inaudible nor indelible<br>iii) Requires standardization<br>iv) Susceptible to hacking<br>v) Consumer resistance to purchasing hardware that does less than before |
| Encryption | Uses tags or watermarks for identification, and in addition uses techniques to make the music unusable without possession of a special code or key | i) All of the advantages of tagging and watermarking<br>ii) Locks up music<br>iii) Complex rules of use can be associated with individual songs | i) Does not work for legacy content<br>ii) Requires industry standardization<br>iii) Consumer resistance to purchasing hardware that does less than before |
| Audio fingerprinting | Uses the inherent qualities of the music to uniquely identify it by comparing it against a database of known music | i) Works for legacy content<br>ii) Has no impact on sound quality (no additions)<br>iii) Does not require industry standardization<br>iv) Compatible with other methods of protecting music<br>v) Completely transparent to the consumer | i) Can be computationally intensive<br>ii) Database can be large for some implementations |

Figure 1.1: Ecosystem of digital rights management systems (Venkatachalam et al., 2004)

improve the dramatic impact, e.g.: arrow sounds of the "Lord of the Rings" are replaced by "whooshes". There are also artistic reasons, for example, in the movie "All the President's Men", in order to strengthen the message that the pen is mightier that the sword, the typewriter keys sounds were mixed with the sound of gunfire (Weis, 1995). Many occasions, not only movies but also computer games, audio-visual presentations, web-sites require sounds. These sounds can be recorded as well as recreated using Foley techniques—for the sound of the knife entering the body in Psycho' shower scene, Hitchcock used a melon (Weis, 1995). Another possibility is the use of already compiled SFX libraries. Accessing library sounds can be an interesting alternative to sending a team to record sounds—think of recording Emperor penguin in their natural habitat). Recreate sounds in a studio using Foley techniques requires a Foley pit and the rest of the recording equipment (L.Mott, 1990). A number of SFX providers, such as www.sounddogs.com, www.sonomic.com or www.sound-effects-library.com, offer SFX on-line. The technology behind these services is standard text-search. Librarians tag sounds with descriptive keywords that the users may search for.

Some companies also keep directories or categories—such as “automobiles”, “horror” or “crashes”—to ease the interaction with the collections. The text-based approach presents several limitations. The work of the librarian is error-prone and a very time-consuming task. Another source of problems is due to the imprecision and ambiguity of natural languages. Natural languages present polysemy—“bike” can mean both “bicycle” and “motorcycle”—and synonymy—both “elevator” and “lift” refer to the same concept. This, together with the difficulty associated to describing sounds with words, affects the quality of the search. The user has to guess how the librarian has labeled the sounds and either too many or too few results are returned. Solutions have been proposed to manage media assets from an audio content-based perspective, e.g.: with “query-by-example” or “find similar” type of techniques, both from the academia and the industry (e.g. www.findsounds.com). However none seems to have impacted in professional sound effects management systems. Finally, a major issue when running sound effects systems is the annotation time by librarians. The staff of the Sound-Effects-Library [2] estimated that it would take 60 years for a librarian to manually label a collection of 2 million sounds.

## 1.3 Goals and methodology

The general goals of this dissertation are presented below. The methodology to test the hypothesis follows an implementation plus evaluation approach.

1. Survey on a low-level description audio content technique: audio fingerprinting and its applications.

2. Justify the usefulness of low-level description of audio content as well as its limitations with example applications: e.g. they provide identification of distorted recordings but are not able to bridge the semantic gap.

3. Underline open issues in state-of-the-art automatic sound annotation as a method to bridge the semantic gap, mainly its limitations to working conditions in limited domains: a few musical instruments or sound ambiances and restricted to a few number of classes.

[2] http://www.sound-effects-library.com

4. We propose a general scalable memory-based method together with a real-world taxonomy: WordNet for overcoming the semantic gap.

5. Implement and evaluate content and concept-based search in a production size sound effects search engine as well as provide an example of an intelligent application that uses the framework: Automatic generation of background ambiances.

## 1.4 Organization of the thesis

This dissertation is organized as follows (see Figure 1.2). In Chapter 2 we explore the applicability of a purely low-level description of audio and its limitations: Audio fingerprinting. We do it by thoroughly reviewing existing methods (see Section 2.2) as well as by implementing audio fingerprinting systems and applications. We illustrate its usefulness with several applications, namely: broadcast audio monitoring (Section 2.3 and integrity verification (Section 2.4). We explore how fingerprinting constraints can be relaxed to allow for similarity type of search and navigation in music collections in Section 2.5.

In Chapter 3 we aim at closing *the semantic gap* when interacting with audio content. In the previous chapter we demonstrate several uses of low-level audio description. Humans use high-level descriptions when searching audio content while automatic methods normally work with low-level descriptions (e.g. overall sound quality, timbral characteristics, and so on). In section 3.1, we explore the use of a general-purpose knowledge database: WordNet, as an ontology-backbone for audio description. The ontology not only provides concept versus keyword search as well other new ways of intelligently interacting with media assets but also provides the classification scheme for a general sound classification methodology. In section 3.2, we present a classifier that is scalable with a huge number of classes. In section 3.3 we describe a SFX search engine that leverages the above mentioned techniques. The search engine is not only able of semantic navigation of sounds but it constitutes a foundation to build other applications. Section 3.4 introduces an example application, automatic sound ambiance generation, which builds on top of the sound search engine.

We conclude this dissertation summarizing the contributions and highlighting promising lines of research (see Chapter 4). In the Appendix A, a list of publications by the author relevant to the dissertation is outlined.

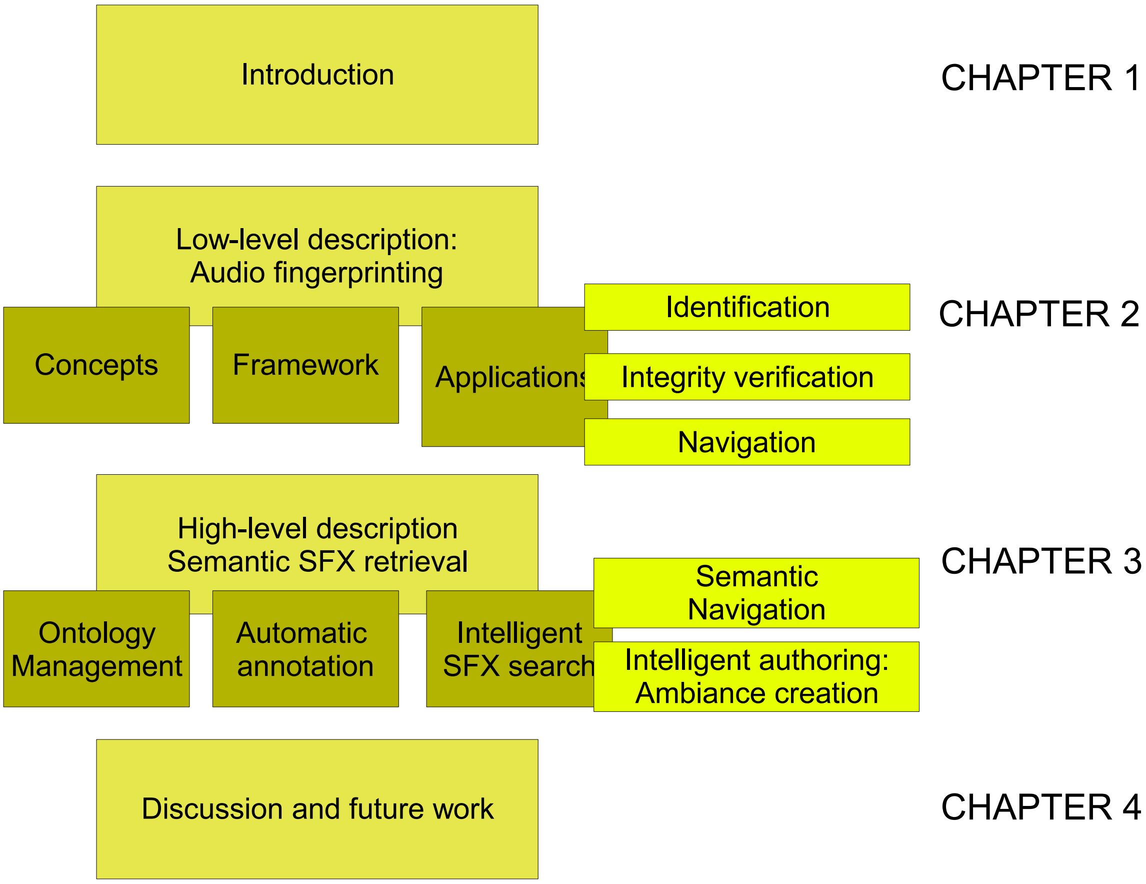


Figure 1.2: Schema of the thesis

## 1.5 Contributions

We summarize the contributions of this dissertation to the state-of-the-art in content-based audio management.

### 1.5.1 Low-level audio description: Fingerprinting

An audio fingerprint is a unique and compact digest derived from perceptually relevant aspects of a recording. Fingerprinting technologies allow the monitoring of audio content without the need of metadata or watermark embedding. However, additional uses exist for audio fingerprinting. In this dissertation we give an overview on Audio Fingerprinting. The rationale is presented along with the differences with respect to watermarking. The main requirements for fingerprinting systems are described. The basic modes of employing audio fingerprints, namely identification, authentication, content-based secret key generation for watermarking and content-based audio retrieval and processing are depicted. The overview includes concrete scenarios and business models where the technology is or can be deployed.

The different approaches to fingerprinting are usually described with different rationales and terminology depending on the background: Pattern matching, Multimedia (Music) Information Retrieval or Cryptography (Robust Hashing). In this thesis, we review different techniques mapping functional parts to blocks of a unified framework.

In order to assess the usefulness of signal-level type of audio description we have implemented or collaborated in different implementations of audio fingerprinting. In the identification mode of fingerprinting, the author contributed to understand the distortions that the audio undergoes when broadcast by radio stations and proposed methods for efficient approximate matching, enhancements over its scalability as well as methods for the detection of false positives.

We proposed and implemented a method to detect whether an audio asset had been modified or tampered with. The method exploited both fingerprinting and watermarking.

Finally, some experiments that hint that the general audio fingerprinting methodology can be extended to allow for similarity search in music and audio databases.

### 1.5.2 Semantic audio description: Sound Effects management

As we show in the dissertation with the example of audio fingerprinting, low-level content audio description have certainly clear uses and there are some success stories. However, rather than dealing with low-level descriptions when interacting with media collections, users prefer high-level descriptions. In this context, we review methods of describing sound, propose a classification engine capable of automatic annotation and implemented both content and knowledge-based techniques in a professional sound effects search engine.

The first part, when trying to develope computational methods that mimic humans in labeling, includes a review of how users describe sound and how to code this information in a way that can be processed by humans and computers alike. Indeed, sounds are multifaceted, multirepresentional and usually difficult to describe in words. We review some taxonomic proposals for audio description found in the literature together with an analysis of the types of descriptions actually found in SFX commercial systems. In order for computers to process the descriptions, it is necessary to code the information in certain way. We review how the issue is dealt within a multimedia standardization process such as MPEG-7 (Manjunath et al., 2002). We have then proposed a management scheme that uses and extends a general purpose ontology: WordNet.

The next contribution after having understood how people describe sounds is building computational models that can automatize the work. In the industry, annotation of audio content is done manually, which is an arduous task. Automatic annotation methods, normally fine-tuned to reduced domains such as musical instruments or reduced sound effects taxonomies, are not mature enough for labeling with great detail any possible sound. A general sound recognition tool requires: first, a taxonomy that represents common sense knowledge of the world and, second, thousands of classifiers, each specialized in distinguishing little details. We propose and report experimental results on a general sound annotator. To tackle the taxonomy definition problem we use WordNet, a semantic network described in 3.1 that organizes real world knowledge. In order to overcome the need of a huge number of classifiers to distinguish many different sound classes, we use a nearest-neighbor classifier with a database of isolated sounds unambiguously linked to WordNet concepts. A 30% concept prediction is achieved on a database of over 50,000 sounds and over 1,600 concepts.

Content-based audio tools such as those described with fingerprinting offer perceptual ways of navigating the audio collections, like “find similar sound”, even if unlabeled, or query-by-example, possibly restricting the search to a semantic subspace, such as “vehicles‘”. The proposed content-based technologies also allow semi-automatic sound annotation and hence close the semantic gap. We demonstrate the integration of semantically-enhanced management of metadata using WordNet together with content-based methods in a commercial sound effect management system.

We finally show how a semantic enabled search engine is capable of semi-automatic ambiance generation. Ambiances are background recordings used in audiovisual productions to make listeners feel they are in places like a pub or a farm. Accessing to commercially available atmosphere libraries is a convenient alternative to sending teams to record ambiances yet they limit the creation in different ways. First, they are already mixed, which reduces the flexibility to add, remove individual sounds or change its panning. Secondly, the number of ambient libraries is limited. We propose a semi-automatic system for ambiance generation. The system creates ambiances on demand given text queries by fetching relevant sounds from a large sound effect database and importing them into a sequencer multi track project. Ambiances of diverse nature can be created easily. Several controls are provided to the users to refine the type of samples and the sound arrangement.

# Chapter 2

# Low-level audio retrieval: Fingerprinting

In this Chapter we explore capabilities and limitations of a well known low-level content-based technique: Audio Fingerprinting. We introduce in Section 2.1 its definition and its applications. In Section 2.2 we propose a general framework that outlines the main functional blocks with which we critically review existing state-of-the-art. Section 2.3 and Section 2.4 we present systems that illustrate the usefulness of fingerprinting: broadcast monitoring and integrity verification. The final Section, 2.5, illustrates how a fingerprinting scheme can be used for content-based retrieval and navigation, not just identification of distorted recordings.

This chapter relates to the dissertation goals 1 and 2 as presented in Section 1.3.

## 2.1 Audio Fingerprinting

Audio fingerprinting is best known for its ability to link unlabeled audio to corresponding meta-data (e.g. artist and song name), regardless of the audio format. Audio fingerprinting or content-based audio identification (CBID) systems extract a perceptual digest of a piece of audio content, i.e. a fingerprint and store it in a database. When presented with unlabeled audio, its fingerprint is calculated and matched against those stored in the database. Using fingerprints and matching algorithms, distorted versions of a recording can be identified as the same audio content.

A source of difficulty when automatically identifying audio content derives from its high dimensionality, the significant variance of the audio data for perceptually similar content and the necessity to efficiently compare the fingerprint with a huge collection of registered fingerprints. The simplest approach that one may think of – the direct comparison of the digitalized waveform – is neither efficient nor effective. A more efficient implementation of this approach could use a hash method, such as MD5 (Message Digest 5) or CRC (Cyclic Redundancy Checking), to obtain a compact representation of the binary file. In this setup, one compares the hash values instead of the whole files. However, hash values are fragile, a single bit flip is sufficient for the hash to completely change. Of course this setup is not robust to compression or minimal distortions of any kind and, in fact, it cannot be considered as content-based identification since it does not consider the content, understood as information, just the bits.

An ideal fingerprinting system should fulfill several requirements. It should be able to accurately identify an item, regardless of the level of compression and distortion or interference in the transmission channel. Depending on the application, it should be able to identify the titles from excerpts of only a few seconds. The fingerprinting system should also be computationally efficient. Efficiency is critical in a real application both in the calculation of the fingerprint of the unknown audio and, even more so, in the search for a best match in huge repository of fingerprints. This computational cost is related to the size of the fingerprints, the complexity of the search algorithm and the complexity of the fingerprint extraction.

The design principles and needs behind audio fingerprinting are recurrent in several research areas. Compact signatures that represent complex multimedia objects are employed in Information Retrieval for fast indexing and retrieval. In order to index complex multimedia objects it is necessary to reduce their dimensionality (to avoid the "curse of dimensionality") and perform the indexing and searching in the reduced space (Baeza-Yates and Ribeiro-Neto, 1999; Subramanya et al., 1999; Kimura et al., 2001). In analogy to the cryptographic hash value, content-based digital signatures can be seen as evolved versions of hash values that are robust to content-preserving transformations (Haitsma and Kalker, 2002b; Mihçak and Venkatesan, 2001). Also from a pattern matching point of view, the idea of extracting the essence of a class of objects retaining its main characteristics is at the heart of any classification system (Cano et al., 2002a; Allamanche et al., 2001; Sukittanon and Atlas, 2002; Theodoris and Koutroumbas, 1999; Picone, 1993).

This section aims to give a vision on Audio Fingerprinting. The rationale along with the differences with respect to watermarking are presented in 2.1.1. The main requirements of fingerprinting systems are described in 2.1.1. The basic modes of employing audio fingerprints, namely identification, authentication, content-based secret key generation for watermarking and content-based audio retrieval are commented in Section 2.1.2. We then present in Section 2.1.3 some concrete scenarios and business models where the technology is used. In the lasts subsections (from Subsection 2.2 to Subsection 2.2.4), we introduce the main contribution of the dissertation: a general framework of audio fingerprinting systems. Although the framework focuses on identification, some of its functional blocks are common to content-based audio retrieval or integrity verification.

### 2.1.1 Definition of audio fingerprinting

An audio fingerprint is a compact content-based signature that summarizes an audio recording. Audio fingerprinting has attracted a lot of attention for its audio identification capabilities. Audio fingerprinting technologies extract acoustic relevant characteristics of a piece of audio content and store them in a database. When presented with an unidentified piece of audio content, characteristics of that piece are calculated and matched against those stored in the database. Using fingerprints and matching algorithms, distorted versions of a single recording can be identified as the same music title (RIAA, 2001).

The approach differs from an alternative existing solution to identify audio content: *Audio Watermarking.* In audio watermarking (Boney et al., 1996a), research on psychoacoustics is conducted so that an arbitrary message, the watermark, can be embedded in a recording without altering the perception of the sound. The identification of a song title is possible by extracting the message embedded in the audio. In audio fingerprinting, the message is automatically derived from the perceptually most relevant components of sound. Compared to watermarking, it is ideally less vulnerable to attacks and distortions since trying to modify this message, the fingerprint, means alteration of the quality of the sound. It is also suitable to deal with legacy content, that is, with audio material released without watermark. In addition, it requires no modification of the audio content. As a drawback, the computational complexity of fingerprinting is generally higher than watermarking and there is the need of a connection to a fingerprint repository. In addition, contrary to watermarking, the message is not independent from the content. It is therefore for example not possible to distinguish between perceptually identical copies of a recording. Just like with watermarking technology, there are more uses to fingerprinting than identification. Specifically, it can also be used for verification of content-integrity; similarly to fragile watermarks.

At this point, we should clarify that the term "fingerprinting" has been employed for many years as a special case of watermarking devised to keep track of an audio clip's usage history. Watermark fingerprinting consists in uniquely watermarking each legal copy of a recording. This allows to trace back to the individual who acquired it (Craver et al., 2001). However, the same term has been used to name techniques that associate an audio signal to a much shorter numeric sequence (the "fingerprint") and use this sequence to e.g. identify the audio signal. The latter is the meaning of the term "fingerprinting" in this article. Other terms for audio fingerprinting are robust matching, robust or perceptual hashing, passive watermarking, automatic music recognition, content-based digital signatures and content-based audio identification. The areas relevant to audio fingerprinting include information retrieval, pattern matching, signal processing, databases, cryptography and music cognition to name a few (Dannenberg et al., 2001).

The requirements depend heavily on the application but are useful in order to evaluate and compare different audio fingerprinting technologies. In their *Request for Information on Audio Fingerprinting Technologies* (RIAA, 2001), the IFPI (International Federation of the Phonographic Industry) and the RIAA (Recording Industry Association of America) tried to evaluate several identification systems. Such systems have to be computationally efficient and robust. A more detailed enumeration of requirements can help to distinguish among the different approaches (CBID, 2002; Kalker, 2001):

**Accuracy:** The number of correct identifications, missed identifications, and wrong identifications (false positives).

**Reliability:** Methods for assessing that a query is present or not in the repository of items to identify is of major importance in play list generation for copyright enforcement organizations. In such cases, if a song has not been broadcast, it should not be identified as a match, even at the cost of missing actual matches. In other applications, like automatic labeling of MP3 files (see Section 2.2), avoiding false positives is not such a mandatory requirement.

**Robustness:** Ability to accurately identify an item, regardless of the level of compression and distortion or interference in the transmission channel. Other sources of degradation are pitching, equalization, background noise, D/A-A/D conversion, audio coders (such as GSM and MP3), etc.

**Granularity:** Ability to identify whole titles from excerpts a few seconds long. It requires to deal with shifting, that is lack of synchronization between the extracted fingerprint and those stored in the database and it adds complexity to the search (it needs to compare audio in all possible alignments).

**Security:** Vulnerability of the solution to cracking or tampering. In contrast with the robustness requirement, the manipulations to deal with are designed to fool the fingerprint identification algorithm.

**Versatility:** Ability to identify audio regardless of the audio format. Ability to use the same database for different applications.

**Scalability:** Performance with very large databases of titles or a large number of concurrent identifications. This affects the accuracy and the complexity of the system.

**Complexity:** It refers to the computational costs of the fingerprint extraction, the size of the fingerprint, the complexity of the search, the complexity of the fingerprint comparison, the cost of adding new items to the database, etc.

**Fragility:** Some applications, such as content-integrity verification systems, may require the detection of changes in the content. This is contrary to the robustness requirement, as the fingerprint should be robust to content-preserving transformations but not to other distortions (see subsection 2.1.2.2).

Improving a certain requirement often implies losing performance in some other. Generally, the fingerprint should be:

- A perceptual digest of the recording. The fingerprint must retain the maximum of perceptually relevant information. This digest should allow the discrimination over a large number of fingerprints. This may be conflicting with other requirements, such as complexity and robustness.

- Invariant to distortions. This derives from the robustness requirement. Content-integrity applications, however, relax this constraint for content-preserving distortions in order to detect deliberate manipulations.

- Compact. A small-sized representation is interesting for complexity, since a large number (maybe millions) of fingerprints need to be stored and compared. An excessively short representation, however, might not be sufficient to discriminate among recordings, affecting thus accuracy, reliability and robustness.

- Easily computable. For complexity reasons, the extraction of the fingerprint should not be excessively time-consuming.

### 2.1.2 Usage modes

#### 2.1.2.1 Identification

Independently of the specific approach to extract the content-based compact signature, a common architecture can be devised to describe the functionality of fingerprinting when used for identification (RIAA, 2001).

The overall functionality mimics the way humans perform the task. As seen in Figure 2.1, a memory of the recordings to be recognized is created off-line (top); in the identification mode (bottom), unlabeled audio is presented to the system to look for a match.

**Database creation:** The collection of recordings to be recognized is presented to the system for the extraction of their fingerprint. The fingerprints are stored in a database and can be linked to editorial information such as artist, album, track name or other meta-data relevant to each recording.

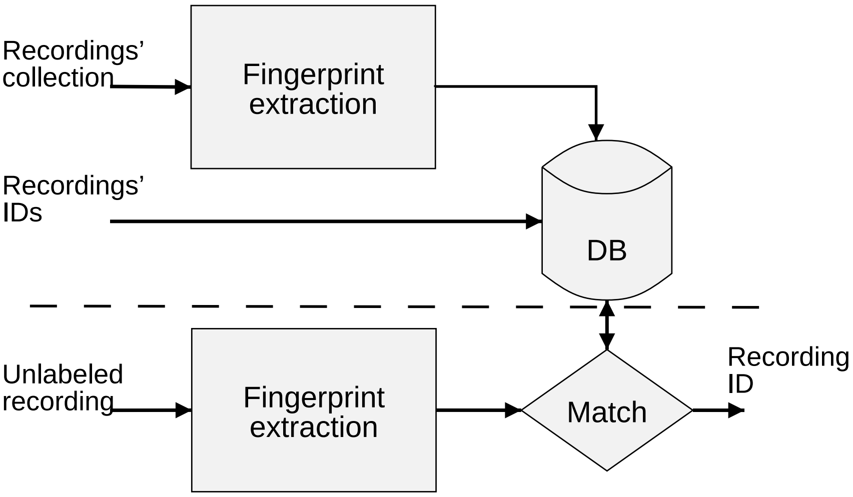


Figure 2.1: Audio fingerprinting framework

**Identification:** The unlabeled recording is processed in order to extract a fingerprint. The fingerprint is subsequently compared with the fingerprints in the database. If a match is found, the meta-data associated with the recording is obtained from the database. Optionally, a reliability measure of the match can be provided.

#### 2.1.2.2 Integrity verification

Integrity verification aims at detecting the alteration of data. As described in section 2.4 there are cases where it is necessary to assess the integrity of audio recordings, e.g. check that they have not been tampered for malicious reasons. The overall functionality (see Figure 2.2) is similar to identification. First, a fingerprint is extracted from the original audio. In the verification phase, the fingerprint extracted from the test signal is compared with the fingerprint of the original. As a result, a report indicating whether the signal has been manipulated is output. Optionally, the system can indicate the type of manipulation and where in the audio it occurred. The verification data, which should be significantly smaller than the audio data, can be sent along with the original audio data (e.g. as a header) or stored in a database. A technique known as *self-embedding* avoids the need of a database or a special dedicated header, by embedding the content-based signature into the audio data using watermarking (see Figure 2.3). An example of such a system is described in (Gómez et al., 2002).

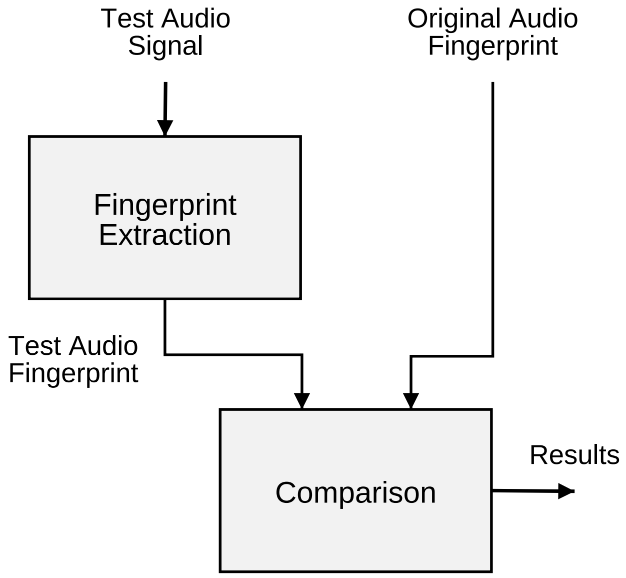


Figure 2.2: Integrity verification framework

#### 2.1.2.3 Watermarking support

Audio fingerprinting can assist watermarking. Audio fingerprints can be used to derive secret keys from the actual content. As described by Mihçak and Venkatesan (2001), using the same secret key for a number of different audio items may compromise security, since each item may leak partial information about the key. Audio fingerprinting / perceptual hashing can help generate input-dependent keys for each piece of audio. Haitsma and Kalker (2002b) suggest audio fingerprinting to enhance the security of watermarks in the context of copy attacks. Copy attacks estimate a watermark from watermarked content and transplant it to unmarked content. Binding the watermark to the content can help to defeat this type of attacks. In addition, fingerprinting can be useful against insertion/deletion attacks that cause desynchronization of the watermark detection: by using the fingerprint, the detector is able to find anchor points in the audio stream and thus to resynchronize at these locations (Mihçak and Venkatesan, 2001).

#### 2.1.2.4 Content-based audio retrieval and processing

Deriving compact signatures from complex multimedia objects is an essential step in Multimedia Information Retrieval. Fingerprinting can extract information from the audio signal at different abstraction levels, from low level descriptors to higher level descriptors. Especially, higher level abstractions for modeling audio hold the possibility to extend the fingerprinting usage modes to content-based navigation, search by similarity, content-based processing and other applications of Music Information Retrieval. In a query-by-example scheme, the fingerprint of a song can be used to retrieve not only the original version but also "similar" ones (Cano et al., 2002b).

### 2.1.3 Application Scenarios

Most of the applications presented in this section are particular cases of the identification usage mode described above. They are therefore based on the ability of audio fingerprinting to link unlabeled audio to corresponding meta-data, regardless of audio format.

#### 2.1.3.1 Audio Content Monitoring and Tracking

In this section we outline different monitoring setups.

**Monitoring at the distributor end**

Content distributors may need to know whether they have the rights to broadcast certain content to consumers. Fingerprinting helps identify unlabeled audio in TV and Radio channels repositories. It can also identify unidentified audio content recovered from CD plants and distributors in anti-piracy investigations (e.g. screening of master recordings at CD manufacturing plants) (RIAA, 2001).

**Monitoring at the transmission channel**

In many countries, radio stations must pay royalties for the music they air. Rights holders are eager to monitor radio transmissions in order to verify whether royalties are being properly paid. Even in countries where radio stations can freely air music, rights holders are interested in monitoring radio transmissions for statistical purposes. Advertisers are also willing to monitor radio and TV transmissions to verify whether commercials are being broadcast as agreed. The same is true for web broadcasts. Other uses include chart compilations for statistical analysis of program material or enforcement of “cultural laws” (e.g. in France a certain percentage of the aired recordings needs to be in French). Fingerprinting-based monitoring systems can be and are actually being used for this purpose. The system “listens” to the radio and continuously updates a play list of songs or commercials broadcast by each station. Of course, a database containing fingerprints of all songs and commercials to be identified must be available to the system, and this database must be updated as new songs come out. Examples of commercial providers of such services are: Broadcast Data System (www.bdsonline.com), Music Reporter (www.musicreporter.net), Audible Magic (www.audiblemagic.com), Yacast (www.yacast.fr) or BMAT (www.bmat.com).

Napster and Web-based communities alike, where users share music files, have proved to be excellent channels for music piracy. After a court battle with the recording industry, Napster was enjoined from facilitating the transfer of copyrighted music. The first measure taken to conform with the judicial ruling was the introduction of a filtering system based on file-name analysis, according to lists of copyrighted music recordings supplied by the recording companies. This simple system did not solve the problem, as users proved to be extremely creative in choosing file names that deceived the filtering system while still allowing other users to easily recognize specific recordings. The large number of songs with identical titles was an additional factor in reducing the efficiency of such filters. Fingerprinting-based monitoring systems constitute a well-suited solution to this problem. Napster actually adopted a fingerprinting technology (see www.relatable.com) and a new file-filtering system relying on it. Additionally, audio content can be found in ordinary web pages. Audio fingerprinting combined with a web crawler can identify this content and report it to the corresponding right owners (e.g. www.baytsp.com).

**Monitoring at the consumer end**

In usage-policy monitoring applications, the goal is to avoid misuse of audio signals by the consumer. We can conceive a system where a piece of music is identified by means of a fingerprint and a database is contacted to retrieve information about the rights. This information dictates the behavior of compliant devices (e.g. CD and DVD players and recorders, MP3 players or even computers) in accordance with the usage policy. Compliant devices are required to be connected to a network in order to access the database.

#### 2.1.3.2 Added-value services

Content information is defined as information about an audio excerpt that is relevant to the user or necessary for the intended application. Depending on the application and the user profile, several levels of content information can be defined. Here are some of the situations we can imagine:

- Content information describing an audio excerpt, such as rhythmic, timbrical, melodic or harmonic descriptions.

- Meta-data describing a musical work, how it was composed and how it was recorded. For example: composer, year of composition, performer, date of performance, studio recording/live performance.

- Other information concerning a musical work, such as album cover image, album price, artist biography, information on the next concerts, etc.

Some systems store content information in a database that is accessible through the Internet. Fingerprinting can then be used to identify a recording and retrieve the corresponding content information, regardless of support type, file format or any other particularity of the audio data. For example, MusicBrainz, Id3man or Moodlogic (www.musicbrainz.org, www.id3man.com, www.moodlogic.com) automatically label collections of audio files; the user can download a compatible player that extracts fingerprints and submits them to a central server from which meta data associated to the recordings is downloaded. Gracenote (www.gracenote.com), who has been providing linking to music meta-data based on the TOC (Table of Contents) of a CD, recently offered audio fingerprinting technology to extend the linking from CD's table of contents to the track level. Their audio identification method is used in combination with text-based classifiers to enhance the accuracy.

Another example is the identification of an audio excerpt by mobile devices, e.g. a cell phone; this is one of the most demanding situations in terms of robustness, as the audio signal goes through radio distortion, D/A-A/D conversion, background noise and GSM coding, and only a few seconds of audio might be available available (e.g. www.shazam.com).

#### 2.1.3.3 Integrity verification systems

In some applications, the integrity of audio recordings must be established before the signal can actually be used, i.e. one must assure that the recording has not been modified or that it is not too distorted. If the signal undergoes lossy compression, D/A-A/D conversion or other content-preserving transformations in the transmission channel, integrity cannot be checked by means of standard hash functions, since a single bit flip is sufficient for the output of the hash function to change. Methods based on fragile watermarking can also provide false alarms in such a context. Systems based on audio fingerprinting, sometimes combined with watermarking, are being researched to tackle this issue. Among some possible applications (Gómez et al., 2002), we can name: Check that commercials are broadcast with the required length and quality, verify that a suspected infringing recording is in fact the same as the recording whose ownership is known, etc.

### 2.1.4 Watermarking

Watermarking was proposed as a solution to copyright enforcement before fingerprinting methods were widely developed. Watermarking consists in embedding into the audio signal an inaudible mark containing copyright information. In this subsection we introduce the concept of watermarking as well as describes potential applications of both methodologies: watermarking and fingerprinting, showing which one is more suitable for each application.

**Definition of Watermarking**

As in cryptography, a *key* is generally used during the construction of the watermark, and another key (which may or may not be identical to the first one) is required for watermark detection. Despite this similarity, watermarking differs from cryptography in its essence. While an encrypted audio file is useless without the corresponding decryption key, no such information is necessary in order to listen to a watermarked audio file. The important point is that the watermark is always present in the signal — even in illegal copies of it — and the protection that is offered by a watermarking system is therefore of a permanent kind. The same is not true for a cryptographic system, as audio files must be decrypted (and thus unprotected) in order to become usable.

Let us clarify the utilization of a watermarking system through an example. Audio

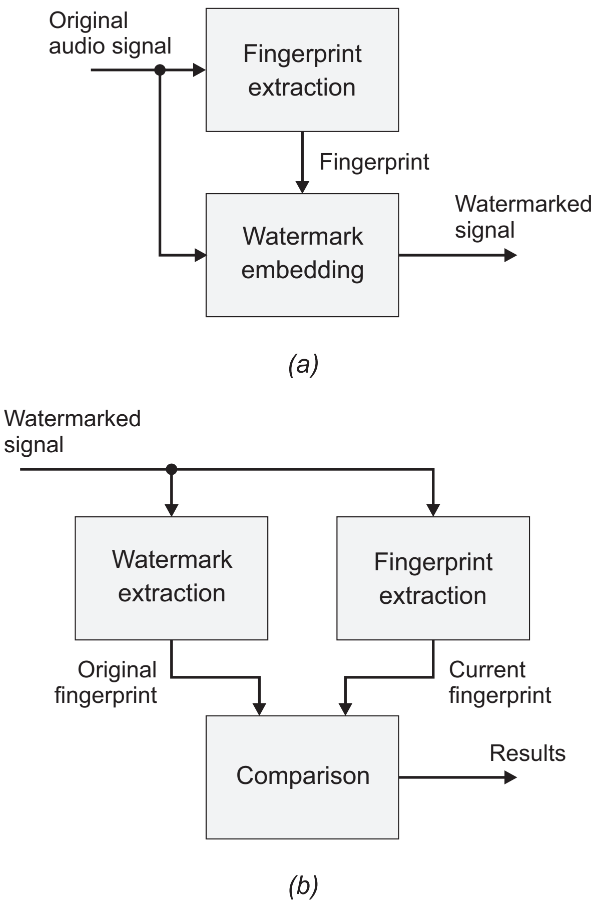


Figure 2.3: Self-embedding integrity verification framework: (a)fingerprint embedding and (b) fingerprint comparison.

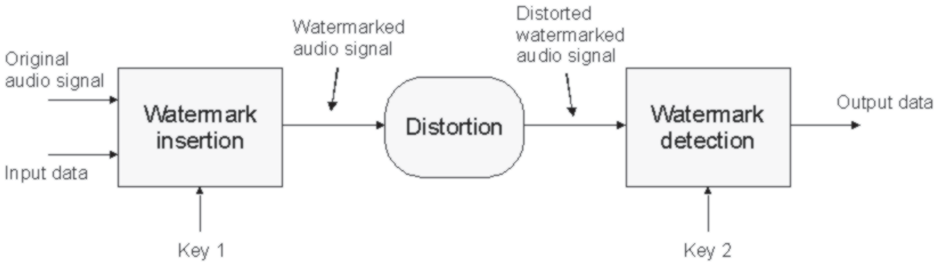


Figure 2.4: General watermarking scheme.

content can be watermarked with a “copy-never” watermark. A compliant CD-writer device will analyze the input audio signal and check for the presence of the watermark before recording. If no watermark is found, the content is assumed to be copyright-free and the CD is recorded; otherwise, the equipment refuses to perform the requested operation. A more sophisticated system could admit multiple degrees of protection, ranging from “copy-never” to “copy-freely”. For instance, audio marked as “copy-twice” could be duplicated, but the resulting copy would have its watermark set to the “copy-once” state. If a second copy were made from this first copy, it would be marked as “copy-never” and would not be reproducible. This would limit the number of generations in the duplication process — if you have an original CD, you can burn a copy for a friend, but he might not be able to do the same from the copy you gave him.

A watermarking system is *symmetric* if the same key is used for both watermark insertion and detection. When these keys are different from each other, the system is *asymmetric.* Symmetric watermarking systems are suitable for *private* watermarking, where the key is kept secret; in contrast, asymmetric watermarking is appropriate for *public* watermarking, where a private (secret) key is used for watermark insertion and a public key for watermark detection. As in public encryption systems, in particular the RSA system (Boneh, 1999), the idea of a non-invertible function is present: the public key is derived from the private key, but the private key cannot be deduced from the public key.

The requirements that an audio watermarking system must satisfy are application-dependent and often conflicting. As general requirements, we can mention:

- **Inaudibility:** watermarking should not degrade sound quality.
- **Robustness:** the watermark should resist any transformations applied to the audio signal, as long as sound quality is not unacceptably degraded.
- **Capacity:** the watermark bit rate should be high enough for the intended application, which can be conflicting with inaudibility and robustness; a trade-off must be found.
- **Reliability:** data contained in the watermark should be extracted with acceptable error rates.
- **Low complexity:** for real-time applications, watermarking algorithms should not be excessively time-consuming.

All these requirements are to be respected to a certain extent, according to the application. Some applications (such as low bit-rate audio over the Internet) might admit the watermark to introduce a small level of sound quality degradation, while others (such as high bit-rate audio) would be extremely rigorous on that matter. Resistance to signal-processing operations such as filtering, resampling or coding is usually necessary. For copyright protection, resistance to malicious attacks aimed at preventing watermark detection is also required; for example, if a piece of the signal is deleted, the watermark should still be detectable. However, for integrity-verification applications (e.g. of testimonies recorded before a court), the watermark must no longer be recognized when the audio content is modified in any way. In that case, robustness is no longer required; on the contrary, the watermark must be fragile.

**How It Works**

Watermarking can be viewed as a communication system: the watermark is the information-bearing signal and the audio signal plays the role of channel noise. In conventional communication systems, the useful signal is usually stronger than the noise, and the latter is often assumed to be Gaussian and white. This is not the case in watermarking. To avoid audible distortion, the watermark signal must be much weaker (some tens of decibels) than the audio signal. Furthermore, the audio signal is generally non-stationary and strongly colored.

Several approaches for audio watermarking have been proposed in the literature (Miller et al., 1999). For example, we can mention:

- **Spread-spectrum watermarking**: As in spread-spectrum communication systems (Dixon, 1976; Haykin, 1988), the idea consists in spreading the watermark in frequency to maximize its power while keeping it inaudible and increasing its resistance to attacks (Boney et al., 1996b).

- **Echo-hiding watermarking**: Temporal masking properties are exploited in order to render the watermark inaudible. The watermark is an "echo" of the original signal (Bender et al., 1996).

- **Bit stream watermarking**: The watermark is inserted directly in the bit stream generated by an audio coder. For example, in  Lacy et al. (1998), the watermark consists in the modification of scale factors in the MPEG AAC bit stream.

Many variations of these basic schemes have been proposed. For example, rather than adding the watermark to the audio signal in the time domain, some systems perform this operation in the frequency domain by directly replacing spectral components (García, 1999).

**Psychoacoustic Models**

*Psychoacoustics* is the study of the perception of sound. Through experimentation, psychoacousticians have established that the human ear presents several limitations. In particular, when two tones, close to each other in frequency, are played simultaneously, *frequency masking* may occur: if one of the tones is sufficiently loud, it *masks* the other one (Zwicker and Fastl 1990).

Psychoacoustic models generalize the frequency-masking effect to non-tonal signals. From an audio signal $u(t)$, these models calculate a curve $M_u(f)$ called *masking threshold* that is homogeneous to a power spectral density (PSD) (Perreau Guimarães 1998). If the PSD $V(f)$ of a signal $v(t)$ is below $M_u(f)$ for all frequencies, then $v(t)$ is masked by $u(t)$. This means that the listener is unable to perceive any difference between $u(t)$ and $u(t) + v(t)$ (Fig. 2).

These models are widely used in lossy compression methods, such as MP3 or MPEG-AAC (ISO 1997, Bosi et al. 1997), to render quantization noise inaudible, thus providing high quality audio at low bit rates.

In audio watermarking, psychoacoustic models are often used to ensure inaudibility of the watermark. The watermark is constructed by shaping in frequency a nearly-white signal according to the masking threshold. After this operation, the PSD of the watermark is always below the masking threshold and the watermark should not be heard in the presence of the original audio signal. Thanks to psychoacoustic models, inaudibility can be reached at signal-to-watermark power ratios of approximately 20 dB. In contrast, a white watermark would require much higher ratios to ensure inaudibility, thus rendering detection more difficult.

Masking can also occur in the time domain with pre or post-masking. If two sounds are close to each other in time and one of them is sufficiently loud, it will mask the other one. This effect is exploited in lossy compression methods to further increase the compression

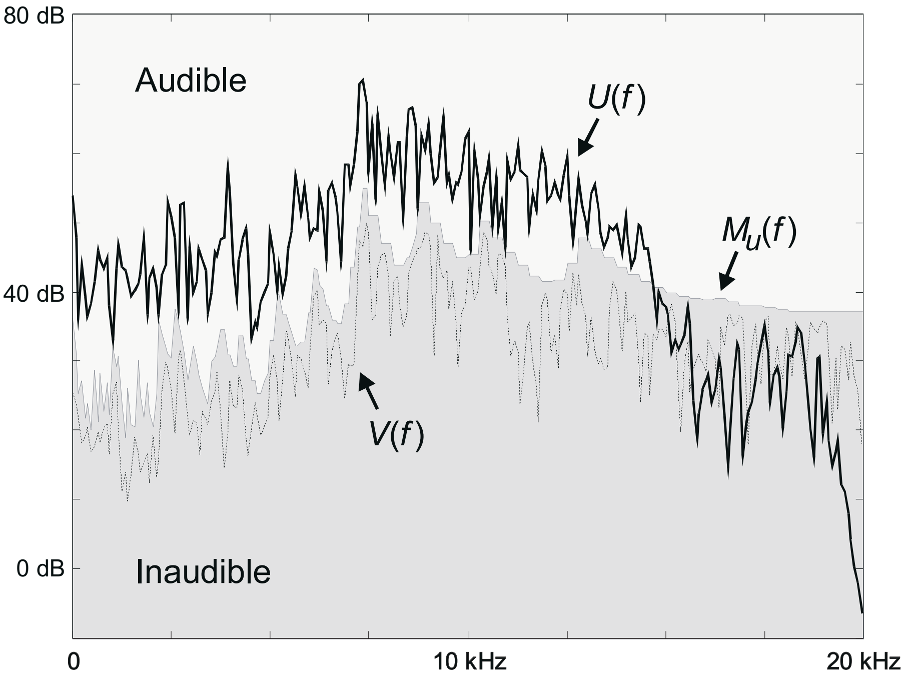


Figure 2.5: PSDs of the masking and masked signals ($U(f)$, continuous line, and $V(f)$, dotted line, respectively), as well as the masking threshold $M_u(f)$. In the upper part of the spectrum, the masking threshold (and the masked signal) often surpasses the masking signal due to the low sensibility of the human ear to high frequencies.

rate (ISO 1997). Post-masking is also used in "echo-hiding" watermarking systems: the watermark is a delayed and attenuated version of the audio signal, and the delay between the audio signal and this "echo" is used as a means of coding information.

Psychoacoustic models are useful in many other applications. To name a few: echo cancellation, automatic audio quality evaluation and hearing aids for the deaf.

#### 2.1.4.1 Audio watermarking versus audio fingerprinting

In this section, we summarize the major differences and similarities between audio watermarking and audio fingerprinting.

**Modification of the Audio Signal**

Audio watermarking modifies the original audio signal by embedding a mark into it, whereas fingerprinting does not change the signal at all but rather analyzes it and constructs a hash (the fingerprint) uniquely associated with this signal. In watermarking, there is a trade-off between watermark power (and audibility), data rate and detection performance. In fingerprinting, there is no such trade-off: the system "listens" to the music, constructs a description of it and searches for a matching description in its database.

**Requirement of a Fingerprint Repository**

A human listener can only identify a piece of music if he has heard it before, unless he has access to more information than just the audio signal. Similarly, fingerprinting systems require previous knowledge of the audio signals in order to identify them, since no information other than the audio signal itself is available to the system in the identification phase. Therefore, a musical knowledge database must be built. This database contains the fingerprints of all the songs the system is supposed to identify. During detection, the fingerprint of the input signal is calculated and a matching algorithm compares it to all fingerprints in the database. The knowledge database must be updated as new songs come out. As the number of songs in the database grows, memory requirements and computational costs also grow; thus, the complexity of the detection process increases with the size of the database.

In contrast, no database is required for detection in a watermarking system, as all the information associated with a signal is contained in the watermark itself. The detector checks for the presence of a watermark and, if one is found, it extracts the data contained therein. Hence, watermarking requires no update as new songs come out, and the complexity of the detection process is not changed when new audio signals are watermarked.

**Requirement of Previous Processing**

For several applications, the need of previously processing audio signals, i.e. watermark embedding, is a severe disadvantage of watermarking systems. For example, watermarking-based distribution-monitoring systems would only be able to detect copyright infringements if the copyrighted signals had been previously watermarked, which means that old non-watermarked material would not be protected at all. Additionally, new material would have to be watermarked in all its distribution formats, as even the availability of a small number of non-watermarked copies might compromise system security. This is not an issue for audio fingerprinting systems, since no previous processing is required.

**Robustness**

In watermark detection, the signal that contains useful information corresponds to a small fraction of the input power, as the watermark is much weaker than the original audio signal due to the inaudibility constraint. In addition, noise that might be added to the watermarked signal (by MP3 compression or analog transmission, for example) can be as strong, or even stronger, as the watermark. In case of severe channel perturbation or piracy attack, the watermark may no longer be detectable.

In contrast, detection in fingerprinting systems is based on the audio signal itself, which is strong enough to resist most channel perturbations and is less susceptible to piracy attacks. Such systems are thus inherently more robust. As long as the original audio in the knowledge database sounds approximately the same as the piece of music that the system is “listening” to, their fingerprints will also be approximately the same. The definition of “approximately” depends on the fingerprint extraction procedure; therefore, the robustness of the system will also depend on it. Most fingerprinting systems use a psychoacoustic front-end approach to derive the fingerprint. By doing so, the audio to analyze (and identify) can be strongly distorted with no decrease in system performance.

**Independence Between Signal and Information**

The information contained in the watermark may have no direct relationship with the carrier audio signal. For example, a radio station could embed the latest news into the songs it airs through a watermark; at reception, the news would appear on a small screen while the songs are played. In contrast, a fingerprint is correlated with the audio signal from which it was extracted; any change in the audio signal that is perceivable to a human listener should cause a change in the fingerprint. This fact is behind most differences in applications between the two approaches: while watermarks can carry any kind of information, fingerprints always represent the audio signal.

This independence between signal and information is derived from the fact that watermarking systems only deal with information that has been previously added, given that no connection to a database is provided. This information can be either related or not with the audio signal in which it has been embedded. Fingerprinting can extract information from the audio signal at different abstraction levels, depending on the application and the usage scenario. The higher level abstractions for modeling the audio and thus the fingerprinting hold the possibility to extend the applications to content-based navigation, search by similarity and other applications of Music Information Retrieval.

#### 2.1.4.2 Summary

Audio watermarking allows one to embed information into an audio signal. Although initially intended for copyright protection, watermarking is useful for a multitude of purposes, particularly for the transport of general-purpose information. Audio fingerprinting does not add any information to the signal, since it uses significant acoustic features to extract a unique fingerprint from it. In conjunction with a database, this fingerprint can be used to identify the audio signal, which is useful in many applications (copyright-related or not).

While information retrieved from a database by means of a fingerprint is always related to a specific piece of music, information embedded into the signal by means of a watermark may be of any kind. Watermarking can even be used as a replacement (or a complement) for cryptography in secure communications. Watermarking has therefore a broader range of applications than fingerprinting.

On the other hand, fingerprinting is inherently more robust than watermarking: while the fingerprint extraction procedure makes use of the full audio signal power, watermark detection is based on a fraction of the watermarked signal power (the watermark, which is several times weaker than the original audio signal due to the inaudibility constraint). This means that fingerprinting will resist distortion at higher levels than watermarking, which is a particularly attractive characteristic in copyright-related applications. When both techniques apply, robustness may be a strong argument in favor of fingerprinting. In addition, fingerprinting does not add any information to the audio signal; one may track and identify a piece of audio already released (watermarked or not), in any recognizable format, by presenting one example of the audio excerpt to the system. Assuming the cost of higher computational requirements and the need of a repository of the fingerprints, this approach represents a flexible solution for copyright- and content-related applications.

An important lesson has been (re)learned from recent research on audio watermarking: absolute protection against piracy is nothing more than an illusion. Sooner or later (probably sooner), pirates will find their way into breaking new protection schemes. The actual goal is to render piracy a less attractive (i.e. more expensive) activity, and to “keep honest people honest”. Neither fingerprinting-based protection systems may claim absolute invulnerability.

Copyright-related applications are still central to the research on both watermarking and fingerprinting. However, recently-proposed added-value applications tend to become more and more prominent in the years to come.

In the next Section we will review existing approaches for audio fingerprinting and propose a general framework for the critical comparison of systems.

## 2.2 General Fingerprinting Framework

In this section we will review audio fingerprinting algorithms. In the literature, the different approaches to fingerprinting are usually described with different rationales and terminology depending on the background: Pattern matching, Multimedia (Music) Information Retrieval or Cryptography (Robust Hashing). In this section, we review different techniques mapping functional parts to blocks of a unified framework. In spite of the different rationales behind the identification task, methods share certain aspects. As depicted in figure 2.2, there are two fundamental processes: the fingerprint extraction and the matching algorithm. The fingerprint extraction derives a set of relevant perceptual characteristics of a recording in a concise and robust form. The fingerprint requirements include:

- Discrimination power over huge numbers of other fingerprints,
- Invariance to distortions,
- Compactness,
- Computational simplicity.

The solutions proposed to fulfill the above requirements imply a trade-off between dimensionality reduction and information loss. The fingerprint extraction consists of a front-end and a fingerprint modeling block (see Figure 2.2.1.2). The front-end computes a set of measurements from the signal (see Section 2.2.1). The fingerprint model block defines the final fingerprint representation, e.g: a vector, a trace of vectors, a codebook, a sequence of indexes to HMM sound classes, a sequence of error correcting words or musically meaningful high-level attributes (see Section 2.2.2).

Given a fingerprint derived from a recording, the matching algorithm searches a database of fingerprints to find the best match. A way of comparing fingerprints, that is a similarity measure, is therefore needed (see Section 2.2.3.1). Since the number of fingerprint comparisons is high in a large database and the similarity can be expensive to compute, we require methods that speed up the search. Some fingerprinting systems use a simpler similarity measure to quickly discard candidates and the more precise but expensive similarity measure for the reduced set of candidates. There are also methods that pre-compute some distances off-line and build a data structure that allows reducing the number of computations to do on-line (see Section 2.2.3.2). According to (Baeza-Yates and Ribeiro-Neto, 1999), good searching methods should be:

- Fast: Sequential scanning and similarity calculation can be too slow for huge databases.
- Correct: Should return the qualifying objects, without missing any — i.e: low False Rejection Rate (FRR).
- Memory efficient: The memory overhead of the search method should be relatively small.
- Easily up-datable: Insertion, deletion and updating of objects should be easy.

The last block of the system – the hypothesis testing (see Figure 2.2) – computes a reliability measure indicating how confident the system is about an identification (see Section 2.2.4).

### 2.2.1 Front-End

The front-end converts an audio signal into a sequence of relevant features to feed the fingerprint model block (see Figure 2.2.1.2). Several driving forces co-exist in the design of the front-end:

- Dimensionality reduction
- Perceptually meaningful parameters (similar to those used by the human auditory system)

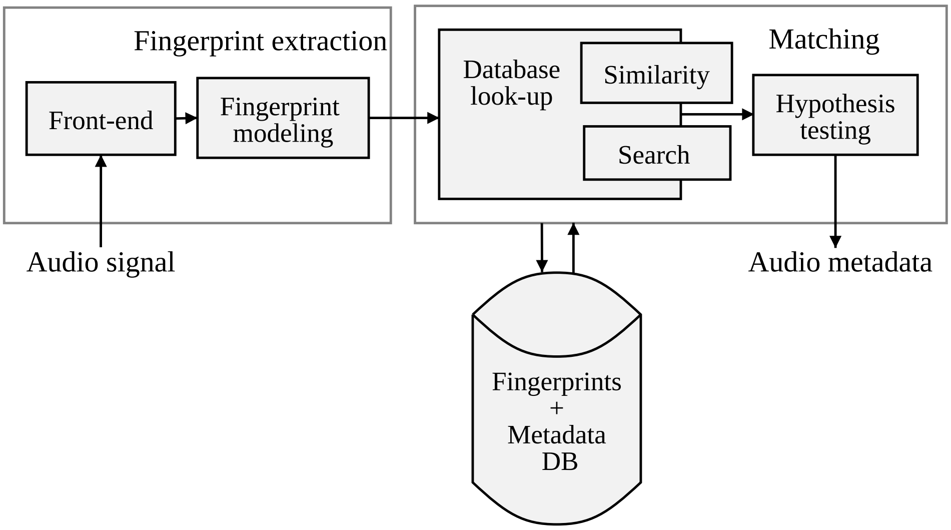


Figure 2.6: Content-based Audio Identification Framework.

- Invariance / robustness (to channel distortions, background noise, etc.)

- Temporal correlation (systems that capture spectral dynamics).

In some applications, where the audio to identify is coded, for instance in mp3, it is possible to by-pass some of the following blocks and extract the features from the audio coded representation.

#### 2.2.1.1 Preprocessing

In a first step, the audio is digitalized (if necessary) and converted to a general format, e.g: mono PCM (16 bits) with a fixed sampling rate (ranging from 5 to 44.1 KHz). Sometimes the audio is preprocessed to simulate the channel, e.g: band-pass filtered in a telephone identification task. Other types of processing are a GSM coder/decoder in a mobile phone identification system, pre-emphasis, amplitude normalization (bounding the dynamic range to (-1,1)).

#### 2.2.1.2 Framing and Overlap

A key assumption in the measurement of characteristics of an audio signal is that the signal can be regarded as stationary over an interval of a few milliseconds. Therefore, the signal is divided into frames of a size comparable to the variation velocity of the underlying acoustic events. The number of frames computed per second is called frame rate. A tapered window function is applied to each block to minimize the discontinuities at the beginning and end. Overlap must be applied to assure robustness to shifting (i.e. when the input data is not perfectly aligned to the recording that was used for generating the fingerprint). There is a trade-off between the robustness to shifting and the computational complexity of the system: the higher the frame rate, the more robust to shifting the system is but at a cost of a higher computational load.

#### 2.2.1.3 Linear Transforms: Spectral Estimates

The idea behind linear transforms is the projection of the set of measurements to a new set of features. If the transform is suitably chosen, the redundancy is significantly reduced. There are optimal transforms in the sense of information packing and decorrelation properties, like Karhunen-Loève (KL) or Singular Value Decomposition (SVD) (Theodoris and Koutroumbas, 1999). These transforms, however, are problem dependent and computationally complex. For that reason, lower complexity transforms using fixed basis vectors are common. Most CBID methods therefore use standard transforms from time to frequency domain to facilitate efficient compression, noise removal and subsequent processing. Lourens (1990), (for computational simplicity), and Kurth et al. (2002), (to model highly distorted sequences, where the time-frequency analysis exhibits distortions), use power measures. The power can still be seen as a simplified time-frequency distribution, with only one frequency bin.

The most common transformation is the Discrete Fourier Transform (DFT). Some other transforms have been proposed: the Discrete Cosine Transform (DCT), the Haar Transform or the Walsh-Hadamard Transform (Subramanya et al., 1999). Richly *et al.* did a comparison of the DFT and the Walsh-Hadamard Transform that revealed that the DFT is generally less sensitive to shifting (Richly et al., 2000). The Modulated Complex Transform

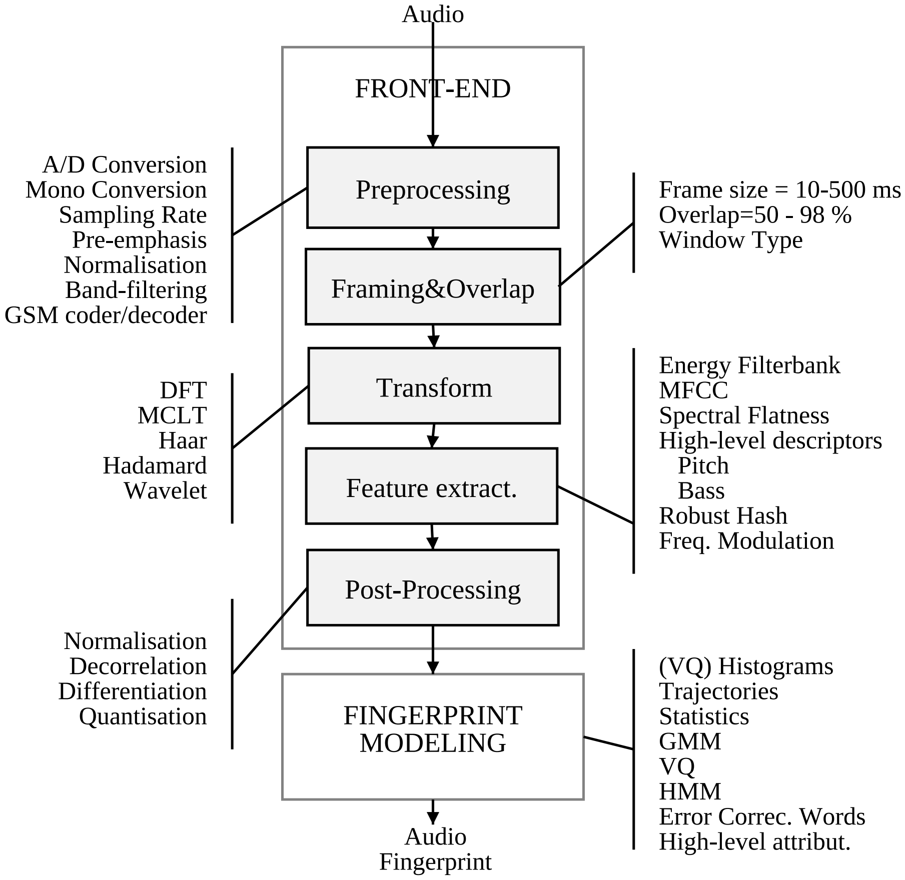


Figure 2.7: Fingerprint Extraction Framework: Front-end (top) and Fingerprint modeling (bottom).

(MCLT) used by Mihçak and Venkatesan (2001) and also by Burges et al. (2003) exhibits approximate shift invariance properties (Mihçak and Venkatesan, 2001).

#### 2.2.1.4 Feature Extraction

Once on a time-frequency representation, additional transformations are applied in order to generate the final acoustic vectors. In this step, we find a great diversity of algorithms. The objective is again to reduce the dimensionality and, at the same time, to increase the invariance to distortions. It is very common to include knowledge of the transduction stages of the human auditory system to extract more perceptually meaningful parameters.

Therefore, many systems extract several features performing a critical-band analysis of the spectrum (see Fig.3). In (Cano et al., 2002a; Blum et al., 1999), Mel-Frequency Cepstrum Coefficients (MFCC) are used. In (Allamanche et al., 2001), the choice is the Spectral Flatness Measure (SFM), which is an estimation of the tone-like or noise-like quality for a band in the spectrum.Papaodysseus et al. (2001) presented the "band representative vectors", which are an ordered list of indexes of bands with prominent tones (i.e. with peaks with significant amplitude). Energy of each band is used by Kimura et al. (2001). Normalized spectral subband centroids are proposed by Seo et al. (2005). Haitsma *et al.* use the energies of 33 bark-scaled bands to obtain their "hash string", which is the sign of the energy band differences (both in the time and the frequency axis) (Haitsma et al., 2001; Haitsma and Kalker, 2002b).

Sukittanon and Atlas claim that spectral estimates and related features only are inadequate when audio channel distortion occurs (Sukittanon and Atlas, 2002; Sukittanon et al., 2004). They propose modulation frequency analysis to characterize the time-varying behavior of audio signals. In this case, features correspond to the geometric mean of the modulation frequency estimation of the energy of 19 bark-spaced band-filters.

Approaches from music information retrieval include features that have proved valid for comparing sounds: harmonicity, bandwidth, loudness (Blum et al., 1999).

Burges *et al.* point out that the features commonly used are heuristic, and as such, may not be optimal (Burges et al., 2002). For that reason, they use a modified Karhunen-Loève transform, the Oriented Principal Component Analysis (OPCA), to find the optimal features in an "unsupervised" way. If PCA (KL) finds a set of orthogonal directions which maximize the signal variance, OPCA obtains a set of possible non-orthogonal directions which take some predefined distortions into account.

#### 2.2.1.5 Post-processing

Most of the features described so far are absolute measurements. In order to better characterize temporal variations in the signal, higher order time derivatives are added to the signal model. In (Cano et al., 2002a) and (Batlle et al., 2002), the feature vector is the concatenation of MFCCs, their derivative (delta) and the acceleration (delta-delta), as well as the delta and delta-delta of the energy. Some systems only use the derivative of the

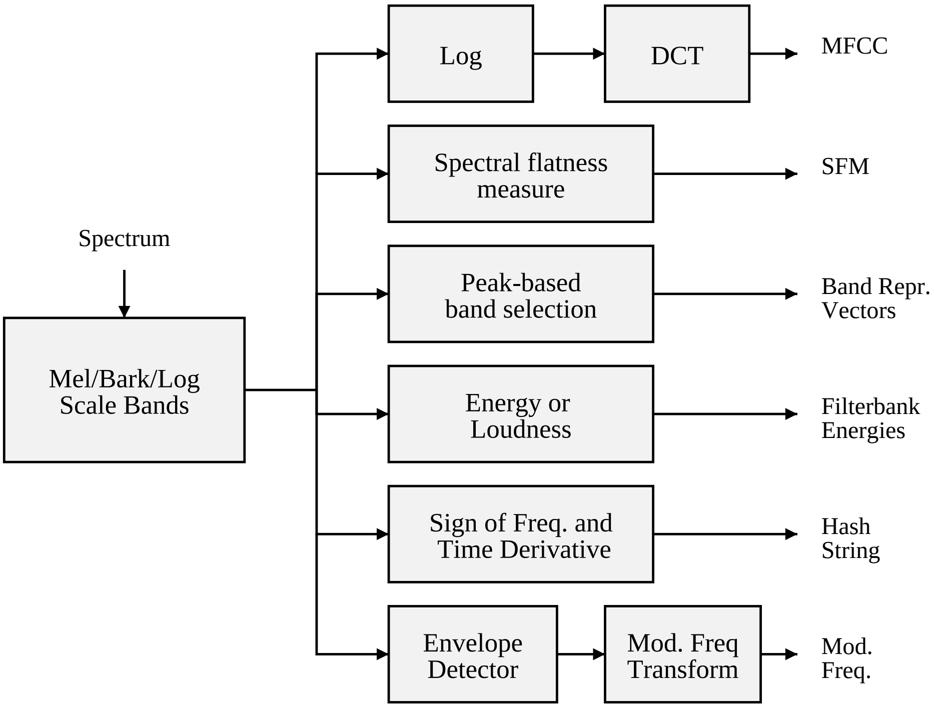


Figure 2.8: Feature Extraction Examples

features, not the absolute features (Allamanche et al., 2001; Kurth et al., 2002). Using the derivative of the signal measurements tends to amplify noise (Picone, 1993) but, at the same time, filters the distortions produced in linear time invariant, or slowly varying channels (like an equalization). Cepstrum Mean Normalization (CMN) is used to reduce linear slowly varying channel distortions in (Batlle et al., 2002). If Euclidean distance is used (see Section 2.2.3.1), mean subtraction and component wise variance normalization are advisable.Park et al. (2006) propose a frequency-temporal filtering for a robust audio fingerprinting schemes in real-noise environments. Some systems compact the feature vector representation using transforms (e.g: PCA (Cano et al., 2002a; Batlle et al., 2002)).

It is quite common to apply a very low resolution quantization to the features: ternary (Richly et al., 2000) or binary (Haitsma and Kalker, 2002b; Kurth et al., 2002). The purpose of quantization is to gain robustness against distortions (Haitsma and Kalker, 2002b; Kurth et al., 2002), normalize (Richly et al., 2000), ease hardware implementations, reduce the memory requirements and for convenience in subsequent parts of the system. Binary sequences are required to extract error correcting words utilized in (Mihçak and Venkatesan, 2001; Kurth et al., 2002). In (Mihçak and Venkatesan, 2001), the discretization is designed to increase randomness in order to minimize fingerprint collision probability.

### 2.2.2 Fingerprint Models

The fingerprint modeling block usually receives a sequence of feature vectors calculated on a frame by frame basis. Exploiting redundancies in the frame time vicinity, inside a recording and across the whole database, is useful to further reduce the fingerprint size. The type of model chosen conditions the similarity measure and also the design of indexing algorithms for fast retrieval (see Section 2.2.3).

A very concise form of fingerprint is achieved by summarizing the multidimensional vector sequences of a whole song (or a fragment of it) in a single vector. Etantrum (Etantrum, 2002) calculates the vector out of the means and variances of the 16 bank-filtered energies corresponding to 30 sec of audio ending up with a signature of 512 bits. The signature along with information on the original audio format is sent to a server for identification. Relatable's TRM signature (TRM, 2002) includes in a vector: the average zero crossing rate, the estimated beats per minute (BPM), an average spectrum and some more features to represent a piece of audio (corresponding to 26 sec). The two examples above are computationally efficient and produce a very compact fingerprint. They have been designed for applications like linking mp3 files to editorial meta-data (title, artist and so on) and are more tuned for low complexity (both on the client and the server side) than for robustness (cropping or broadcast streaming audio).

Fingerprints can also be sequences (traces, trajectories) of features. This fingerprint representation is found in (Blum et al., 1999), and also in (Haitsma and Kalker, 2002b) as binary vector sequences. The fingerprint in (Papaodysseus et al., 2001), which consists on a sequence of "band representative vectors", is binary encoded for memory efficiency.

Some systems, include high-level musically meaningful attributes, like rhythm ( (Kirovski and Attias, 2002)) or prominent pitch (see (TRM, 2002) and (Blum et al., 1999)).

Following the reasoning on the possible sub-optimality of heuristic features, Burges et al. (2002) employ several layers of OPCA to decrease the local statistical redundancy of feature vectors with respect to time. Besides reducing dimensionality, extra robustness requisites to shifting[1] and pitching[2] are accounted in the transformation.

"Global redundancies" within a song are exploited in (Allamanche et al., 2001). If we assume that the features of a given audio item are similar among them (e.g: a chorus that repeats in a song probably hold similar features), a compact representation can be generated by clustering the feature vectors. The sequence of vectors is thus approximated by a much lower number of representative code vectors, a codebook. The temporal evolution of audio is lost with this approximation. Also in (Allamanche et al., 2001), short-time statistics are collected over regions of time. This results in both higher recognition, since some temporal dependencies are taken into account, and a faster matching, since the length of each sequence is also reduced.

Cano et al. (2002a) and Batlle et al. (2002, 2004) use a fingerprint model that further exploits global redundancy. The rationale is very much inspired on speech research. In speech, an alphabet of sound classes, i.e. phonemes can be used to segment a collection of raw speech data into text achieving a great redundancy reduction without "much" information loss. Similarly, we can view a corpus of music, as sentences constructed concatenating sound classes of a finite alphabet. "Perceptually equivalent" drum sounds, say for instance a hi-hat, occurs in a great number of pop songs. This approximation yields a fingerprint which consists in sequences of indexes to a set of sound classes representative of a collection of recordings. The sound classes are estimated via unsupervised clustering and modeled with Hidden Markov Models (HMMs) (Batlle and Cano, 2000). Statistical modeling of the signal's time course allows local redundancy reduction. The fingerprint representation as sequences of indexes to the sound classes retains the information on the evolution of audio through time.

In (Mihçak and Venkatesan, 2001), discrete sequences are mapped to a dictionary of error correcting words. In (Kurth et al., 2002), the error correcting codes are at the basis of their indexing method.

[1] Shifting refers to a displacement of the audio signal in time with respect to the original signal

[2] Pitching is playing an audio file faster. It produces frequency distortions

### 2.2.3 Similarity measures and Searching Methods

#### 2.2.3.1 Similarity measures

Similarity measures are very much related to the type of model chosen. When comparing vector sequences, a correlation metric is common. The Euclidean distance, or slightly modified versions that deal with sequences of different lengths, are used for instance in (Blum et al., 1999). In (Sukittanon and Atlas, 2002), the classification is Nearest Neighbor using a cross entropy estimation. In the systems where the vector feature sequences are quantized, a Manhattan distance (or Hamming when the quantization is binary) is common (Haitsma and Kalker, 2002b; Richly et al., 2000). Mihçak and Venkatesan (2001) suggest that another error metric, which they call “Exponential Pseudo Norm” (EPN), could be more appropriate to better distinguish between close and distant values with an emphasis stronger than linear.

So far we have presented an identification framework that follows a template matching paradigm (Theodoris and Koutroumbas, 1999): both the reference patterns – the fingerprints stored in the database – and the test pattern – the fingerprint extracted from the unknown audio – are in the same format and are compared according to some similarity measure, e.g: hamming distance, a correlation and so on. In some systems, only the reference items are actually “fingerprints” – compactly modeled as a codebook or a sequence of indexes to HMMs (Allamanche et al., 2001),(Batlle et al., 2002, 2003). In these cases, the similarities are computed directly between the feature sequence extracted from the unknown audio and the reference audio fingerprints stored in the repository. In (Allamanche et al., 2001), the feature vector sequence is matched to the different codebooks using a distance metric. For each codebook, the errors are accumulated. The unknown item is assigned to the class which yields the lowest accumulated error. In (Batlle et al., 2002), the feature sequence is run against the fingerprints (a concatenation of indexes pointing at HMM sound classes) using the Viterbi algorithm. The most likely passage in the database is selected.

#### 2.2.3.2 Searching methods

A fundamental issue for the usability of a fingerprinting system is how to efficiently do the comparison of the unknown audio against the possibly millions of fingerprints. A brute-force approach that computes the similarities between the unknown recording's fingerprint and those stored in the database can be prohibitory. The time for finding a best match in this linear or sequential approach is proportional to $Nc\left(d\left(\right)\right)+E$, where $N$ is the number of fingerprints in the repository and $c\left(d\left(\right)\right)$ the time needed for a single similarity calculation and $E$ accounts for some extra CPU time.

**Pre-computing distances off-line** One cannot pre-calculate off-line similarities with query fingerprint because the fingerprint has not been previously presented to the system. However one can pre-compute distances among the fingerprints registered in the repository and build a data structure to reduce the number of similarity evaluations once the query is presented. It is possible to build sets of equivalence classes off-line, calculate some similarities on-line to discard some classes and search exhaustively the rest(see for example (Kimura et al., 2001)). If the similarity measure is a metric, i.e: the similarity measure is a function that satisfies the following properties: positiveness, symmetry, reflexivity and the triangular inequality, there are methods that reduce the number of similarity evaluations and guarantee no false dismissals (see (Chávez et al., 2001)). Vector spaces allow the use of efficient existing spatial access methods (Faloutsos et al., 1994). An example of such an indexing scheme is (Miller et al., 2005).

**Filtering unlikely candidates with a cheap similarity measure** Another possibility is to use a simpler similarity measure to quickly eliminate many candidates and the more precise but complex on the rest, e.g: in (Kenyon, 1993; Kastner et al., 2002). As demonstrated in (Faloutsos et al., 1994), in order to guarantee no false dismissals, the simple ( coarse) similarity used for discarding unpromising hypothesis must lower bound the more expensive (fine) similarity.

**Inverted file indexing** A very efficient searching method use of inverted files indexing. Haitsma *et al.* proposed an index of possible pieces of a fingerprint that points to the positions in the songs. Provided that a piece of a query's fingerprint is free of errors (exact match), a list of candidate songs and positions can be efficiently retrieved to exhaustively search through (Haitsma and Kalker, 2002b,a). In (Cano et al., 2002a), indexing and heuristics similar to those used in computational biology for the comparison of DNA are used to speed up a search in a system where the fingerprints are sequences of symbols. Kurth et al. (2002) present an index that use code words extracted from binary sequences representing the audio. Sometimes these approaches, although very fast, make assumptions on the errors permitted in the words used to build the index which could result in false dismissals.

**Candidate pruning** A simple optimization to speed up the search is to keep the best score encountered thus far. We can abandon a similarity measure calculation if at one point we know we are not going to improve the best-so-far score (see for instance (Kimura et al., 2001; Kashino et al., 2003)).

**Other approaches** Some similarity measures can profit from structures like suffix trees to avoid duplicate calculations (Baeza-Yates and Ribeiro-Neto, 1999). In one of the setups of (Wang and SmithII, 2002), the repository of fingerprints is split into two databases. The first and smaller repository holds fingerprints with higher probability of appearance, e.g: the most popular songs of the moment, and the other repository with the rest. The queries are confronted first with the small and more likely repository and only when no match is found does the system examine the second database. Production systems actually use several of the above depicted speed-up methods. (Wang and SmithII, 2002) for instance, besides searching first in the most popular songs repository, uses an inverted file indexing for fast accessing the fingerprints along with a heuristic to filter out unpromising candidates before it exhaustively searches with the more precise similarity measure.

### 2.2.4 Hypothesis Testing

This last step aims to answer what is the confidence for a positive match as well as whether the query is present or not in the repository of items to identify. During the comparison of the extracted fingerprint to the database of fingerprints, scores (resulting from similarity measures) are obtained. In order to decide that there is a correct identification, the score needs to be beyond a certain threshold. It is not easy to choose a threshold since it depends on: the used fingerprint model, the discriminative information of the query, the similarity of the fingerprints in the database, and the database size. The larger the database, the higher the probability of wrongly indicating a match by chance, that is a false positive[3] Approaches to deal with false positives have been explicitly treated for instance in (Haitsma and Kalker,

[3] The false positive rate is also named false acceptance rate (FAR) or false alarm rate. The false negative rate appears also under the name of false rejected rate (FRR). The nomenclature is related to the Information Retrieval performance evaluation measures: Precision and Recall (Baeza-Yates and Ribeiro-Neto, 1999).

2002b; Cano et al., 2001; Lourens, 1990).

### 2.2.5 Summary

We have presented a review of the research carried out in the area of audio fingerprinting. An audio fingerprinting system generally consists of two components: an algorithm to generate fingerprints from recordings and algorithm to search for a matching fingerprint in a fingerprint database. We have shown that although different researchers have taken different approaches, the proposals more or less fit in a general framework.

In the following Section we present in detail a fingerprinting implementation originally conceived for broadcast audio monitoring.

## 2.3 Identification of Broadcast Music

This section describes the development of an audio fingerprint called AudioDNA designed to be robust against several distortions including those related to radio broadcasting. A complete system, covering also a fast and efficient method for comparing observed fingerprints against a huge database with reference fingerprints is described. The promising results achieved with the first prototype system observing music titles as well as commercials are presented. The system was developed in the context of the European project RAA (http://raa.joanneum.at).

### 2.3.1 Introduction

A monitoring system able to automatically generate play lists of registered songs can be a valuable tool for copyright enforcement organizations and for companies reporting statistics on the music broadcast.

The difficulty inherent in the task of identifying broadcast audio material is mainly due to the difference of quality of the original titles, usually stored on Audio CD and the quality of the broadcast ones. The song is transmitted partially, the presenter talks on top of different fragments, the piece is maybe played faster and several manipulation effects are applied to increase the listeners' psychoacoustic impact (compressors, enhancers, equalization, bass booster). Moreover, in broadcast audio streams there are no markers indicating the begin and the end of the songs.

Such a system also has to be fast because it must do comparisons with several thousand songs. This affects the memory and computation requisites since the system should observe several radio stations, give results on-line and should not be very expensive in terms of hardware.

The section describes an approach to the problem. It proposes a modeling of audio aimed at being robust to different distortions in an adverse environment: radio broadcasting. Along with an explanation of the fingerprint matching algorithms, we present some results and conclusions.

The overall functionality of this system—as well as any fingerprinting system—mimics the way humans perform the task. Off-line a memory of the songs to be recognized is created; in the identification mode, unlabeled audio is presented to the system to look for a match. It is possible to distinguish two operating modes:

*Building the database*: The collection of songs to be recognized is presented to the system. The system processes the audio signals extracting unique representations based on their acoustic characteristics. This compact and unique representation is stored in a database and each fingerprint is linked with a tag or other metadata relevant to each recording.

*Actual Audio Identification*: The unlabeled audio is processed in order to extract the fingerprint. The fingerprint is then compared to the fingerprints of the database. If a match is found, the tag associated with the work is obtained from the database. A confidence of the match is also provided.

Again, there is a trade-off between robustness and computational costs. Some methods are designed to be very scalable but less flexible with respect to the distortions on the audio, or the need for the whole song for a correct identification. The proposed system falls into the category of robust fingerprinting technologies. It is designed to be identifying audio titles even if a fragment that has undergone distortions is used as query. The distortions robust systems aim to answer are enumerated for instance in Mihçak and Venkatesan (2001); Haitsma et al. (2001); Allamanche et al. (2001); Papaodysseus et al. (2001). What we present in the next section is a description of the manipulations radio stations perform (Plaschzug et al., 2000).

### 2.3.2 Broadcast audio processing distortions

Radio stations use complex sound processing to get more loudness and a more impressive sound. The major intention of these stations is to attract listeners by sending at higher average energy than other stations in a certain area. How this is achieved in detail is different at each radio station. Most radio stations use bass compressions and enhancements, equalizing, frequency selective compressions and exciting, full-range compressions and exciting.

Typically radio stations can get out 10dB more over-all average level and more with these sound effects. Moreover, they enlarge the stereo base which enhances the impression of stereo listening, even if the two loudspeakers are separated by a small distance only. Some stations use “pitching”, i.e. they play the songs faster. In order to save memory, digital audio data are coded using compression algorithms. In Europe, the most commonly used sound processors are the“Optimod” and the “Omnia” system. Unfortunately, all radio stations consider their way of using these devices (sequence of effects, parameter settings) as their intellectual property, and there is no way of getting any details. A number of effect devices are used in order to prepare a signal for broadcasting.

**CompressorLimiter**

These effects are used in order to ensure the headroom of digital systems while avoiding clippings. If one reduces the dynamic range of a signal and defines a lower maximum level it is possible to push the whole range to a higher average level. This attracts attention to particular radio stations while “zapping” through the offer.

Every recording has its specific dynamic range – even in one stylistic area you have different song characters and different ways of producing the mix. In order to unify the average levels of different pieces dynamic compressors are used, mostly in combination with limiters in order to achieve a fixed maximum level.

**Stereo Base-width**

All stations enlarge the stereo base-width. This is done to achieve the feeling of sitting in a living room even while listening music in a car. The effect devices delay each channel and mix the result crosswise with the opposite channel. This causes changes of the signals-to-phase relations of both stereo channels. In extreme cases the mono compatibility decreases. That means that parts of the full frequency range disappear in a mono mix. Sometimes whole instruments vanish.

**Exciter Enhancer**

Exciters, also named enhancers, add psycho acoustic effects on audio material by changing the hull curve without changing levels. The principle of work is to add generated harmonics to the original. The signal is cut off at a frequency threshold between 2 and 6 kHz. The remaining spectrum above is the base for the harmonic generator. This produces an addition of distortion. How sharp or mellow this is perceived depends on the mix between even and odd numbered harmonics. After that the effect is mixed with the original. Modern exciters do not add the distortions over the whole hull curve. They just select strong signal transients. The effects are a more brilliant sound without adding more high frequency level, a better speech understanding, direct attacks of percussive sounds and a wider angle for tweeters without adding undesired sharpness.

**Pitching**

"Pitching" is an effect with which the songs in a radio broadcast are played faster. Sometimes stations use pitching up to 2,5% to achieve two goals: playing more songs per hour and getting more "kick/attraction" for listeners.

In Figure 2.9 the results of applying pitching are shown. The song corresponds to *Believe* by Cher and it has been broadcast 1.7% faster than the original CD from a German radio station. The figure the energy content per each semitone, the x-axis corresponds to time and the y-axis corresponds to semitones. A leaking of energy into the upper semitones is appreciated in the broadcast version. Although this effect is less used nowadays, the distortion can affect the performance of some fingerprinting systems depending on the features and modeling used.

**Sound Processors**

Professional radio stations use complex sound processors which comprise all of the effects devices discussed above (e.g. Omnia or Optimod). Sometimes, the results are better if the signal is compressed a little bit (2:1) before going through the sound processor, so other compressors are used before sending the signal into the sound processor. Also the stereo

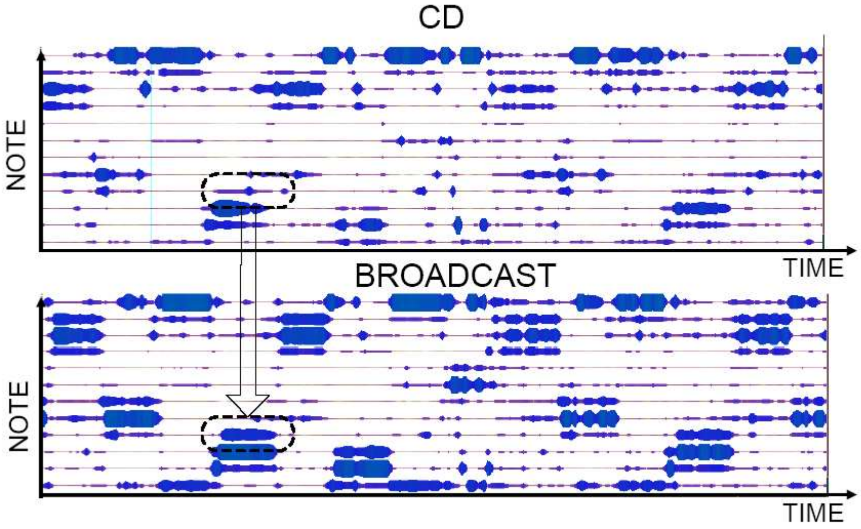


Figure 2.9: Pitching Effect in Energy per Semitone

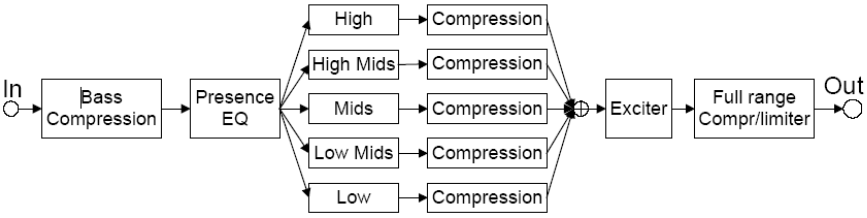


Figure 2.10: Sample setup in a radio station

base-width is often enlarged by other systems. Figure 2.10 visualizes a sample setup of sound processing effects. However, high-end sound processors allow for changing both the parameters of each of the effects devices as well as their sequence.

In figure 2.10, we see a selective bass compression with optimized parameters, followed by a parametric equalizer for the low mid spectrum (often called “presence”). The next block includes a crossover frequency splitting (5 bands) for selective compressions. Sometimes, in the upper bands, one can find additional exciters. After summing up the different edited frequency bands we have a full range exciter. After a final compressor/limiter the signal goes out to the FM-modulation.

### 2.3.3 System Overview

In this scenario, a particular abstraction of audio to be used as robust fingerprint is presented: audio as sequence of acoustic events (Batlle and Cano, 2000). Such a sequence identifies a music title. In analogy to the biological terminology the acoustic events are named AudioGenes (Neuschmied et al., 2001). A piece of music is composed of a sequence of audio genes which is called the AudioDNA. As an analogy we can take the speech case, where speech events are described in terms of phonemes. However, in music modeling it is not so straightforward. For example the use of musical notes would have disadvantages: Most often notes are played simultaneously and music samples contain additional voices or other sounds. The approach is to learn the relevant acoustic events, AudioGenes, through unsupervised training that is, without any previous knowledge of music events. The training is performed through a modified Baum- Welch algorithm on a corpus of representative music (Batlle and Cano, 2000).

Shortly the whole system works as follows, off-line and out of a collection of music representative of the type of songs to be identified, an alphabet of sounds that best describes the music is derived. These audio units are modeled with Hidden Markov Models (HMM) (Rabiner, 1990). The unlabeled audio and the set of songs are decomposed in these audio units ending up with a sequence of symbols for the unlabeled audio and a database of sequences representing the original songs. By approximate string matching the song sequences that best resembles the sequence of the unlabeled audio is obtained.

#### 2.3.3.1 Fingerprint Extraction: AudioDNA

The audio data is Prue-processed by a front-end in a frame-by-frame analysis. In the first block a set of relevant feature(s) vectors are extracted from the sound. Within the front end, a normalization of feature vectors as well as some other processing is done before the decoding block (Haykin, 1996). In the decoding block, the feature vectors are run against the statistical models of the AudioGenes using the Viterbi algorithm (Viterbi, 1970). As a result, the most likely AudioDNA sequence is produced (see Figure 2.11).

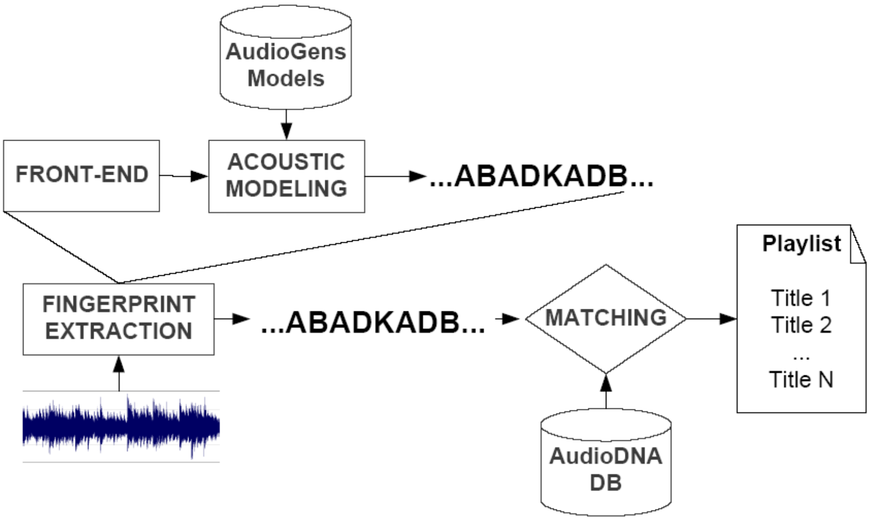


Figure 2.11: Block diagram of the fingerprinting system

#### 2.3.3.2 Front-end

The first stage in a classification system is the obtainment of a set of values that represent the main characteristics of the audio samples. A key assumption made at this step is that the signal can be regarded as stationary over an interval of a few milliseconds. Thus, the prime function of the front-end parameterization stage is to divide the input sound into blocks and from each block derive some features, like a smoothed spectral estimate.

The spacing between blocks is around 10 ms and blocks are overlapped to give a longer analysis window, typically 25 ms. As with all processing of this type, a tapered window function (e.g. Hamming) is applied to each block so as to minimize the signal discontinuities at the beginning and end of each frame (Oppenheim and Schafer, 1989).

It is well known that the human ear performs some kind of signal processing before the audio signal enters the brain. Since this processing has proved to be robust in front of several kinds of noises and distortions in the area of speech recognition, it seems reasonable to use a similar signal front-end processing for music in the system. The required spectral estimates are computed via Fourier analysis and there are a number of additional transformations that can be applied in order to generate the final acoustic vectors. To illustrate one typical arrangement, the figure 2.12 shows the front-end to generate Mel-Frequency

Cepstral Coefficients (MFCCs).

To compute MFCC coefficients, the Fourier spectrum is smoothed by integrating the spectral coefficients within triangular frequency bins arranged on a non-linear scale called the Mel-scale. The Mel-scale is designed to approximate the frequency resolution of the human ear being linear up to 1,000 Hz and logarithmic thereafter (Ruggero, 1992). In order to make the statistics of the estimated song power spectrum approximately Gaussian, logarithmic (compression) conversion is applied to the filter-bank output.

The final processing stage is to apply the Discrete Cosine Transform to the log filter-bank coefficients. This has the effect of compressing the spectral information into the lower order coefficients and it also de-correlates them (Batlle et al., 1998).

The acoustic modeling based on HMM assumes that each acoustic vector is independent with its neighbors. This is a rather poor assumption since physical constraints of the musical instruments ensure that there is continuity between successive spectral estimates. However, appending the first and second order differentials to the basic static coefficients will greatly reduce the problem. Obviously, there are more acoustic features that can be extracted to feed the models.

#### 2.3.3.3 Fingerprint modeling: HMM Acoustic-models

The purpose of the acoustic models is to provide a method of calculating the likelihood of any sequence of AudioDNA given a vector sequence Y. Each individual AudioGenes is represented by a Hidden Markov model (HMM) (Rabiner, 1990). An HMM is most easily understood as a generator of vector sequences. It is a finite state machine which changes state once every time unit and each time t that a state $j$ is entered, an $n$ acoustic vector $y_t$ is generated with probability density $b_j(y_t)$. Furthermore, the transition from state $i$ to state $j$ is also probabilistic and governed by the discrete probability $a_ij$. In the figure 2.13 we show an example of this process where the model moves through the state sequence $X = 1, 1, 2, 2, 2, 2, 2, 3, 3, 3$ in order to generate the 10 observation vectors of k-index model.

The joint probability of a vector sequence $Y$ and state sequence $X$ given some model $M$ is calculated simply as the product of the transition probabilities and the output probabilities. The joint probability of an acoustic vector sequence $Y$ and some state sequence $X =$

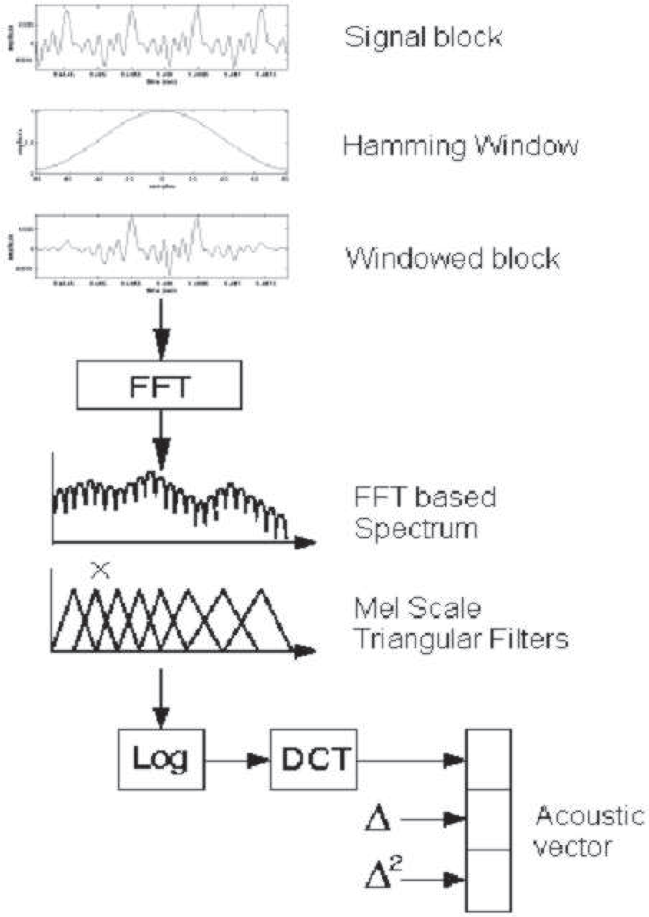


Figure 2.12: Front-end feature extraction.

$x(1),\ x(2),\ x(3),\ \ldots\ ,\ x(T)$ is:

$$P(Y, X|M) = a_{x(0)x(1)} \prod_{t=1}^{T} b_{x(t)}(y_t) a_{x(t)x(t+1)} \tag{2.1}$$

In practice only the observation sequence $Y$ is known and the underlying state sequence $X$ is hidden. This is why it is called *Hidden Markov Model*.

For the decoding of AudioGenes sequences, the trained models are run against the feature vectors using the Viterbi algorithm (Viterbi, 1970). As a result, the most probable path through the models is found, providing a sequence of AudioGenes and the points in

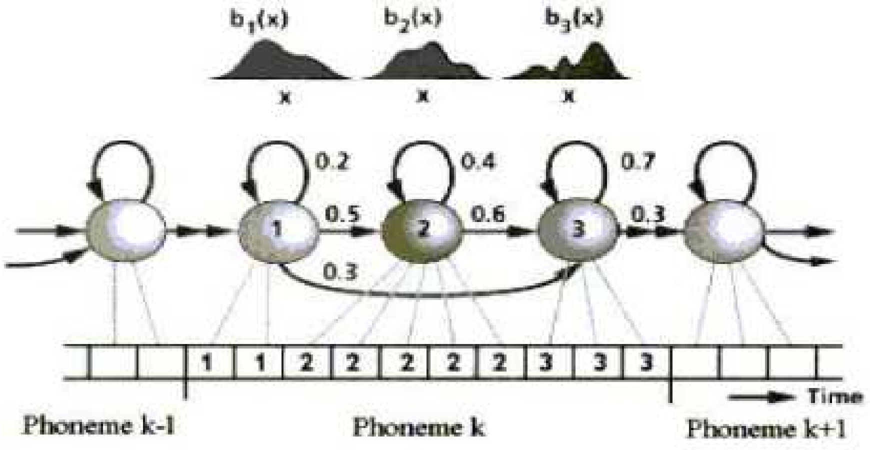


Figure 2.13: Hidden Markov Model process example

time for every transition from one model to the following.

The Viterbi algorithm is an efficient algorithm to find the state sequence that most likely produced the observations. Let $\phi_j(t)$ represent the maximum likelihood of observing acoustic vectors $y_1$ to $y_t$ and being in state $j$ at time $t$. This partial likelihood can be computed using the following recursion

$$\phi_j(t) = \max_i\{\phi_j(t-1) \cdot a_{ij}\} b_j(y_t) \tag{2.2}$$

where

$$\phi_1(1) = 1 \tag{2.3}$$

$$\phi_j(1) = a_{1j} b_j(y_1) \tag{2.4}$$

for $1 < j < N$. The maximum likelihood $P'(Y|M)$ is then given by

$$\phi_N(T) = \max_i\{\phi_j(T) a_{iN}\} \tag{2.5}$$

By keeping track of the state $j$ giving the maximum value in the above recursion formula, it is possible, at the end of the input sequence, to retrieve the states visited by the best path, thus obtaining the most probable AudioDNA sequence for the given input frames.

As it has been shown, the algorithm performs the backtracking at the end of the audio data. When dealing with streaming audio data, like when observing a radio broadcast, it is necessary to provide the AudioDNA sequence in real time (plus a little latency time). Fortunately, it is possible to modify the algorithm to work on-line (Loscos et al., 1999). To do so, the backtracking is adapted to determine the best path at each frame iteration instead of waiting until the end of the utterance. Doing so it is possible to detect that the sequence of AudioGenes up to a point has converged, that is to say, it has become stationary and included in the best path from a certain time on, they can be extracted. The time of convergence is variable and depends on the probabilistic modeling of the sound with the current HMMs.

The AudioDNA representation results then in a sequence of letters, the Gens, and temporal information, start time and duration (see Figure 2.14). The actual number of different Gens as well as the output rate can be adjusted. The setup used in the Experimental Results section corresponds to AudioDNA of 32 different Gens and an average output rate of 800 Gens per minute.

#### 2.3.3.4 Similarity search: Approximate Matching

After the extraction of the fingerprint, the next important part of the proposed system is the matching component, i.e. the module which compares fingerprints from observed audio signals against reference fingerprints in a database. As the main requirement on the proposed fingerprint system is robustness against several kinds of signal distortions, the actual fingerprint from an observed signal will not be fully identical to the reference database (see Figure 2.14). And as the system is designed to monitor audio streams with unknown begin/end time of particular titles, the matching component has to take care about this fact as well. Every database management system (DBMS) has to be able to perform exact matching, i.e. retrieve data records according to a given query, usually specified as SQL statement. In addition to these exact matching possibilities there are usually some enhancements to support typical text applications, e.g. string search techniques which sometimes contain approximate matching capabilities, like phonetic search. However, these methods cannot be applied to our requirements, as those algorithms are optimized to support certain languages and are based on special dictionaries. Besides the problem of approximate

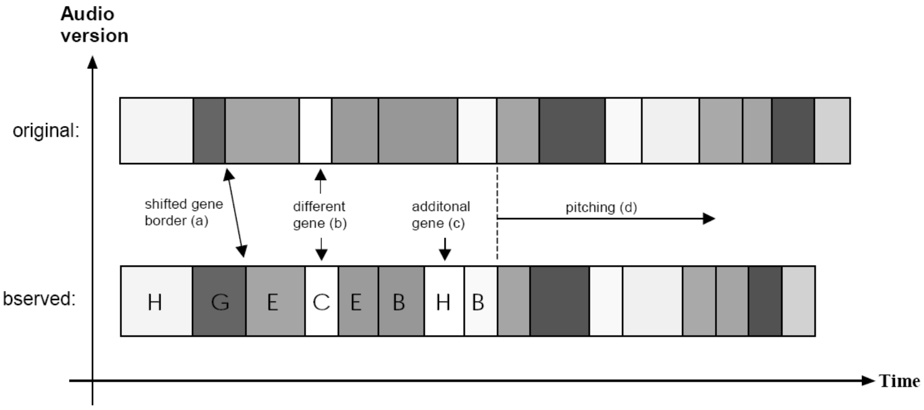


Figure 2.14: Possible changes in the broadcast vs. original fingerprint

matching, there is also another important design goal: the algorithm has to be very fast and efficient, i.e. identifying one title against a database with some 100,000 fingerprints considerably faster than real-time. Such scalability features allow the observation of several audio streams in parallel.

**AudioDNA Properties** AudioGenes have additional time information, which is a significant difference to standard string applications, but this information can be exploited in the matching algorithm. Figure 6 shows the most important cases besides the identical match, which may occur when comparing observed with original AudioDNA fingerprints:

1. identical genes are detected, but their time borders are different

2. parts having different genes

3. additional genes are present in the same time interval

4. the average length of genes is different (typical case with pitching effects)

String algorithms which are necessary to approach the above issues are a traditional area of study in computer science. In recent years their importance has grown dramatically with the huge increase of electronically stored text and of molecular sequence data produced by various genome projects. In our applied matching algorithm there are two main processing steps, which use the same principal method as FASTA (Pearson and Lipman, 1988; Gusfield, 1997), one of the most effective practical database search methods for biological gene sequences:

1. reduce the search space by exact matching of short subsequences

2. apply approximate matching algorithms starting at the positions of the previously found positions

**Matching Process** Short subsequences of AudioDNA from an observed audio stream are continuously extracted and compared with the fingerprints in the database. If an identical subsequence is found, the matching result will be stored in a balanced tree data structure for further processing steps. One node of this tree contains a list of exact matching results of one title. Only results appearing in the right chronological order will be appended to a node. In addition the time length information is used to discard non-promising nodes. If a node contains a certain amount of exact matching results, an approximate matching method is applied to detect similarities of longer sequences starting at the position of the exact matches. The calculation of the similarity of sequences is a standard application of dynamic programming. Again the embedded time information of audio genes is used to apply a very fast and simple approximate matching method.

The exact matching process yields a time base for aligning the two AudioDNA to be compared. This alignment allows determining the time intervals $\Delta t_{equal}$ where equal audio genes occur. The similarity $S$ of the AudioDNA sequences in a specific time period $\Delta t_{obs}$ is given by the equation:

$$S(\Delta t_{obs}) = \frac{\sum_{t=1}^{n} \Delta t_{equal}(i)}{\Delta t_{obs}} \tag{2.6}$$

where $\Delta t_{equal}(1)$ is the first and $\Delta t_{equal}(n)$ is the last $\Delta t_{equal}(i)$ in the time period $\Delta t_{obs}$. $S$ can be computed in $O(N)$ time where $N$ is the length of the compared sequence.

The actual result (matching music title or "unknown") of the approximate matching process is finally derived from an empiric model using these similarity values.

| Compression | Rate | False Negatives | False Positives |
|---|---|---|---|
| MP3 | 128 kbps | 0 | 0 |
| | 96 kbps | 0 | 0 |
| | 72 kbps | 0 | 0 |
| | 48 kbps | 1 | 0 |
| | 32 kbps | 15 | 0 |
| RealAudio | 128 kbps | 0 | 0 |
| | 96 kbps | 0 | 0 |
| | 72 kbps | 0 | 0 |
| | 48 kbps | 2 | 0 |
| | 32 kbps | 20 | 0 |

Table 2.1: Detection results for different audio coding formats and compression rates

### 2.3.4 Experiments

The database of reference fingerprints used to perform experiments contains roughly 50,000 music titles. For the following described robustness tests it will be expected that an audio title can be observed at least for six seconds. Principally the minimal length of a song which can be detected depends on the quality and significance of the observed audio signal. If the system fails to detect a title (result 'unknown') which is included in the database, the test result is called a "false negative". On the other hand, the detection of wrong titles is called "false positive", which is more annoying when deploying the system for copyright enforcement and to share the royalties among the artists whose music was broadcast.

**Radio Broadcast** The audio signal captured from radio broadcast is influenced by different transmission and manipulation effects. In a preliminary experiment 12 hours of continuously broadcast material of different stations were captured to test the recognition performance of the system The evaluation of this test material yielded very promising results. All titles included in the reference database (104 titles) were detected. Especially important, there were no false positive results. At this point, intensive testing with many more hours of radio is compulsory.

**Coding Formats & Compression** The system was also tested against different audio coding formats and compression rates. 200 titles have been chosen and stored in several formats.

No false positives were detected and it can be said that down to 48 kbps the system is

| Task | duration [s] per title (240 sec) | real time factor |
|---|---|---|
| AudioDNA extraction | 77,0 | 3,12 |
| AudioDNA matching | 4,8 | 50,00 |
| AudioDNA import | 1,2 | 200,00 |

Table 2.2: Calculation time of the main processing tasks

able to recognize the music title correctly. Having higher compression rates, the recognition-rate drops significantly.

**Commercials** The detection of commercials is another challenging problem. They are very short and contain both music and speech. The AudioDNA of 134 commercials has been stored in the database, some of them appear twice with only slight changes in a short section of one or two seconds inside the spot. Again public radio stations have been observed for testing. All recorded spots have been detected and no false detections occurred. The similar spots have been detected twice when one of them appeared.

**Performance Tests** Table 2 shows some performance figures of the main processing tasks with the system. All measurements have been performed on a standard Pentium III 933 MHz PC having two GB of main memory. It can be seen that the most time consuming task is currently the AudioDNA extraction, however this module has not been optimized for performance yet. The other modules reach processing times which clearly make them suitable for the operation of a service, where several audio streams are observed and analyzed in parallel. All numbers have been normalized to typical duration of a music title, i.e. four minutes, the reference database contained some 50,000 titles for the matching and import measurements. Importing is the task of putting a new AudioDNA into the reference database, which is important when setting up a new reference database based on music titles within a digital asset management system.

### 2.3.5 Conclusion

The results with the prototype proved that the chosen approach to extract a fingerprint from an audio signal robust against various distortions is valid. The system can be implemented on inexpensive standard PC hardware (Pentium III, 1 GHz), allowing the calculation of at least four fingerprints at the same time. The appliance of system is not limited to radio broadcast, but has also been successfully tested with Internet radio and Internet download facilities. It could be useful as monitoring tool for Internet service providers (ISP), ensuring the proper usage of their customers Internet access. The research involved around the AudioDNA opened several future lines of development. Among these we can find the inclusion of other musical-based parameters (like rhythm and melodic trajectories) into the pattern matching algorithm as well as improvements into the HMM structure in order to better fit the musical needs.

In the next Section we will show another application of the AudioDNA fingerprinting scheme: Integrity verification.

## 2.4 Integrity Verification

In this Section we introduce a method for audio-integrity verification based on a combination of watermarking and audio fingerprinting. As we described in previous Sections an audio fingerprint is a perceptual digest that holds content information of a recording and that ideally allows to uniquely identify it from other recordings. Integrity verification is performed by embedding the fingerprint into the audio signal itself by means of a watermark. The original fingerprint is reconstructed from the watermark and compared with a new fingerprint extracted from the watermarked signal. If they are identical, the signal has not been modified; if not, the system is able to determine the approximate locations where the signal has been corrupted. The watermarked signal could go through content preserving transformations, such as D/A and A/D conversion, resampling, etc...without triggering the corruption alarm.

### 2.4.1 Introduction

In many applications, the integrity of an audio recording must be unquestionably established before the signal can actually be used, i.e. one must be sure that the recording has not been modified without authorization.

Some application contexts of these integrity verification systems dealing with speech are the following ones:

- integrity verification of a previously recorded testimony that is to be used as evidence before a court of law;
- integrity verification of recorded interviews, which could be edited for malicious purposes.

Regarding music applications, some examples are:

- integrity verification of radio or television commercials;
- integrity verification of music aired by radio stations or distributed on the internet;

Integrity verification systems have been proposed as an answer to this need. Two classes of methods are well suited for these applications: *watermarking,*, which allows one to embed data into the signal, and *fingerprinting,* which consists in extracting a "signature" (the fingerprint) from the audio signal.

After a conceptual description of integrity verification schemes based solely on fingerprinting and watermarking, we propose a mixed approach that takes advantage of both technologies.

### 2.4.2 Integrity Verification Systems: A Conceptual Review

#### 2.4.2.1 Watermarking-Based Systems

We define three classes of integrity-verification systems based on watermarking:

**1. Methods based on fragile watermarking,** which consist in embedding a fragile watermark into the audio signal (e.g. a low-power watermark). If the watermarked signal is edited, the watermark must no longer be detectable. By "edited", we understand any modification that could corrupt the content of a recording. "Cut-and-paste" manipulations (deletion or insertion of segments of audio), for example, must render the watermark undetectable. In contrast, content-preserving manipulations, such as lossy compression with reasonable compression rates or addition of small amounts of channel noise, should not prevent watermark detection (as long as the content is actually preserved). In order to do so, the original watermark must be stored elsewhere.

Extremely fragile watermarks can also be used to verify whether a signal has been manipulated in any way, even without audible distortion. For example, a recording company can watermark the content of its CDs with a very fragile watermark. If songs from this CD are compressed (e.g. in MPEG format), then decompressed and recorded on a new CD, the watermark would not be detected in the new recording, even if the latter sounds exactly as the original one to the listener. A CD player can then check for the presence of this watermark; if no watermark is found, the recording has necessarily undergone illicit manipulations and the CD is refused. The main flaw in this approach is its inflexibility: as the watermark is extremely fragile, there is no margin for the rights owner to define any allowed signal manipulations (except for the exact duplication of the audio signal).

**2. Methods based on semi-fragile watermarking,** which are a variation of the previous class of methods. The idea consists in circumventing the excessive fragility of the watermark by increasing its power. This semi-fragile watermark is able to resist slight modifications in the audio signal but becomes undetectable when the signal is more significantly modified. The difficulty in this approach is the determination of an appropriate "robustness threshold" for each application.

**3. Methods based on robust watermarking,** which consist in embedding a robust watermark into the audio signal. The watermark is supposed to remain detectable in spite of any manipulations the signal may suffer. Integrity is verified by checking whether the information contained in the watermark is corrupted or not.

Watermarking-based integrity-verification systems depend entirely on the reliability of the watermarking method. However, an audio signal often contains short segments that are difficult to watermark due to localized unfavorable characteristics (e.g. very low power or ill-conditioned spectral characteristics); these segments will probably lead to detection errors, particularly after lossy transformations such as resampling or MPEG compression. In integrity-verification applications, this is a serious drawback, since it may not be possible to decide reliably whether unexpected data are a consequence of intentional tampering or "normal" detection errors.

#### 2.4.2.2 Fingerprinting-Based Systems

**Audio fingerprinting** or **content-based identification** (CBID) methods extract relevant acoustic characteristics from a piece of audio content. The result is a perceptual digest, the *fingerprint,* that acts as a kind of signature of the audio signal. If the fingerprints of a set of recordings are stored in a database, each of these recordings can be identified by extracting its fingerprint and searching for it in the database.

In fingerprinting-based integrity-verification systems, the integrity of an audio signal is determined by checking the integrity of its fingerprint. These systems operate in three steps: (1) a fingerprint is extracted from the original audio recording, (2) this fingerprint is stored in a trustworthy database, and (3) the integrity of a recording is verified by extracting its fingerprint and comparing it with the original fingerprint stored in the database. Whenever the transmission is digital, the fingerprint can be send within a header (Wu and Kuo, 2001).

Fingerprinting or content-based digital signature methods evolve from the traditional cryptographic hash methods. The direct application of hashing methods results in a type of integrity-verification systems:

**Methods sensitive to data modification,** based on hashing methods such as MD5. This class of methods is appropriate when the audio recording is not supposed to be modified at all, since a single bit flip is sufficient for the fingerprint to change. Some robustness to slight signal modifications can be obtained by not taking into account the least-significant bits when applying the hash function.

In order to be insensitive to common content preserving operations there has been an evolution toward content-based digital signatures or fingerprints:

**Methods sensitive to content modification,** based on fingerprinting methods that are intended to represent the content of an audio recording (such as AudioDNA (Cano et al., 2002a)). This class of methods is appropriate when the integrity check is not supposed to be compromised by operations that preserve audio content (in a perceptual point of view) while modifying binary data, such as lossy compression, D/A and A/D conversion, resampling...

The main disadvantage of fingerprinting-based methods is the need of additional metadata (the original fingerprint) in the integrity-check phase. This requires the access to a database or the insertion of the fingerprint in a dedicated field in a header (not appropriate for analog streams of audio) (Wu et al., 2001).

### 2.4.3 A Combined Watermarking-Fingerprinting System

The branch of integrity-verification that combines watermarking and fingerprinting is known as *self-embedding* (Wu et al., 2001). The idea consists in extracting the fingerprint of an audio signal and storing it in the signal itself through watermarking, thus avoiding the need of additional metadata during integrity check.

Some methods based on this idea have already been described in the literature, specially for image and video (Dittmann et al., 1999; Dittmann, 2001). Shaw proposed a system (Shaw, 2000) that embedded an encrypted hash into digital documents, also including audio. This approach inherits the limitations of hashing methods with respect to fingerprinting: hashing methods are sensitive to content preserving transformations (see section *II.B*).

We propose an integrity verification approach that combines a fingerprinting method representing the content of an audio recording and a robust watermarking algorithm. Figure 2.15 presents a general scheme of this mixed approach.

First, the fingerprint of the original recording is extracted; this fingerprint, viewed as a sequence of bits, is then used as the information to be embedded into the signal through watermarking. As the watermark signal is weak, the watermarked recording should have the same fingerprint as the original recording. Thus, the integrity of this recording can be verified by extracting its fingerprint and comparing it with the original one (reconstructed from the watermark). This procedure will be detailed in the following sections.

We mention below some of the requirements that are expected to be satisfied by the integrity-verification system and its components:

- the fingerprint should not be modified when transformations that preserve audio content are performed;

- the watermarking scheme must be robust to such transformations;

- the bit rate of the watermarking system must be high enough to code the fingerprint information with strong redundancy;

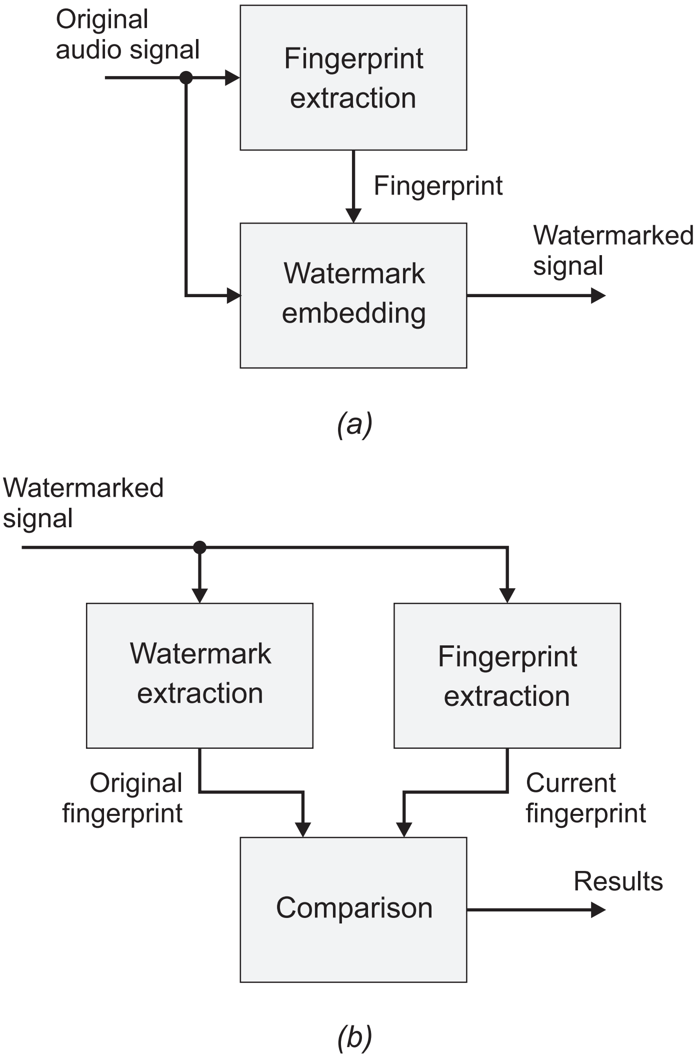


Figure 2.15: Block diagram of the mixed approach for audio integrity verification: (a) embedding; (b) detection.

- the method should be suitable for use with streaming audio, as the total length of the audio file is unknown in applications such as broadcasting Gennaro and Rohatgi (1997).

As will be shown in the following sections, the first three requirements are fulfilled by the system. The last one is also satisfied, as both the watermark and the fingerprint can be processed “on the fly”.

We will also show experiments to detect structural manipulations of audio signals. This is, for example, the kind of tampering that must be avoided in the case of recorded testimonies or interviews.

Nevertheless, some promising results are obtained regarding the detection of distortions that perceptually affect the signal have been handled by the system, as for instance:

- time stretching modification;
- pitch shifting of an audio segment;
- severe distortion through filtering;
- addition of strong noise.

The system is not only able to detect tampering, but it can also determine the approximate location where the audio signal was corrupted.

### 2.4.4 Implementation

**Fingerprint Extraction**

The key idea of employed fingerprinting scheme consists in considering audio as a sequence of *acoustic events.*

As we described in Section 2.3 in detail, the system works as follows. An alphabet of representative sounds is derived from the corpus of audio signals (constructed according to the kind of signals that the system is supposed to identify). These audio units are modeled by means of Hidden Markov Models (HMM).

The audio signal is processed in a frame-by-frame analysis. A set of relevant-feature vectors is first extracted from the sound. These vectors are then normalized and sent to the decoding block, where they are submitted to statistical analysis by means of the Viterbi algorithm. The output of this chain — the fingerprint — is the most likely ADU sequence for this audio signal. This process is illustrated in Figure 2.4.4.

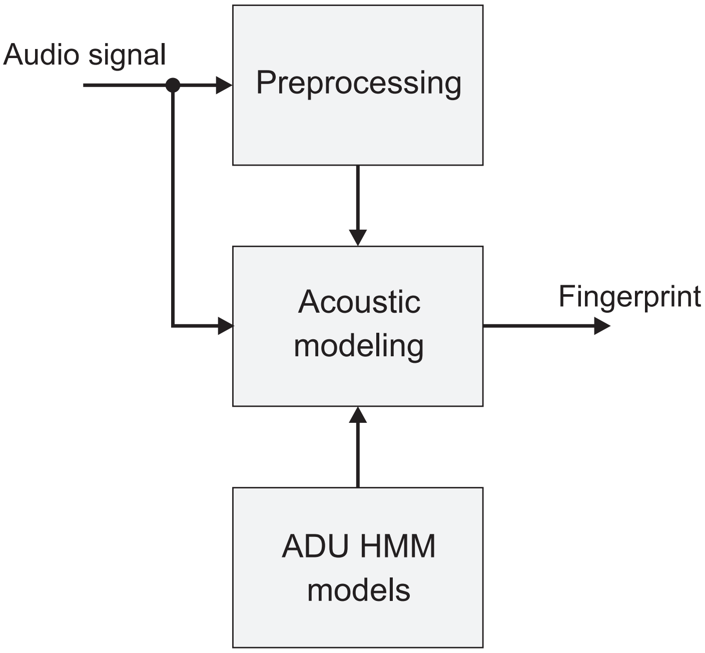


Figure 2.16: Fingerprint extraction.

The resulting fingerprint is therefore a sequence of symbols (the ADUs) and time information (start time and duration). The number of different ADUs available to the system can be adjusted, as well as the output rate. The setup used in our experiments corresponds to 16 different ADUs (G0, G1, . . . , G15) and an average output rate of 100 ADUs per minute.

**Fingerprint Encoding and Watermark Embedding**

Each 8-s segment of the audio signal is treated individually in order to allow for streaming-audio processing. The fingerprint is converted into a binary sequence by associating a unique four-bit pattern to each of the 16 possible ADUs; thus, the average fingerprint bit rate is approximately 7 bits/s. In our experiments, the watermark bit rate is set to 125 bits/s, allowing the fingerprint information to be coded with huge redundancy (which minimizes the probability of error during its extraction). A simple repetition code is employed, with a particular 6-bit pattern (011110) serving as a delimiter between repetitions. To avoid confusion between actual data and delimiters, every group of four or more consecutive bits "1" in the data receives an additional bit "1", which is suppressed in the detection phase.

Fingerprint data is embedded into the audio signal by means of a watermark. The watermarking system used in our experiments is represented in Figure 2.17.

The analogy between watermarking and digital communications is emphasized in the figure: watermark synthesis corresponds to transmission (with the watermark as the information-bearing signal), watermark embedding corresponds to channel propagation (with the audio signal as channel noise), and watermark detection corresponds to reception.

The watermark signal is synthesized from the input data by a modulator. In order to obtain a watermark that is spread in frequency (so as to maximize its power and increase its robustness), a codebook containing white, orthogonal Gaussian vectors is used in the modulator. The number of vectors is a function of the desired bit rate. Each codebook entry is associated with a specific input binary pattern. The modulator output is produced by concatenating codebook vectors according to the input data sequence.

To ensure watermark inaudibility, the modulator output is spectrally shaped through filtering according to a masking threshold (obtained from a psychoacoustic model). This procedure, repeated for each window of the audio signal ($\approx$ 10 ms), produces the watermark. The watermarked signal is obtained by adding together the original audio signal and the watermark.

As transmission and reception must be synchronized, the transmitted data sequence also carries synchronization information. This sequence is structured in such a way that detected data is syntactically correct only when the detection is properly synchronized. If synchronism is lost, it can be retrieved by systematically looking for valid data sequences. This resynchronization scheme, based on the Viterbi algorithm, is detailed in Gómez (2000) and de C. T. Gomes et al. (2001).

**Watermark Detection and Fingerprint Decoding**

For each window of the received signal, the watermark signal is strengthened through Wiener-filtering and correlation measures with each codebook entry are calculated. The

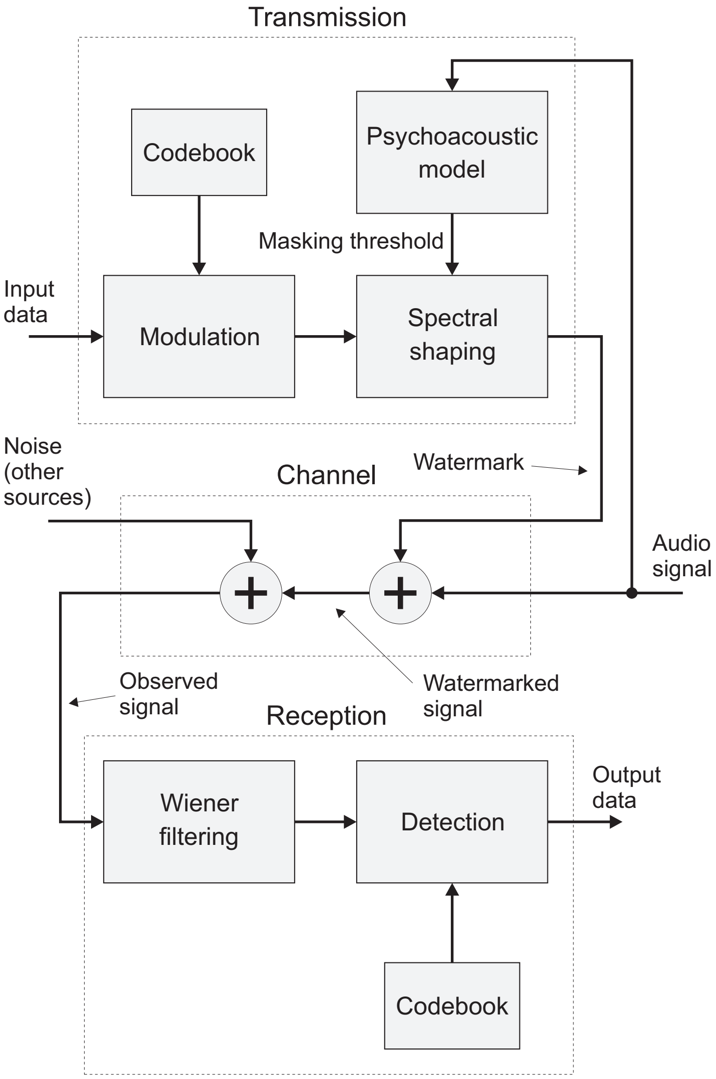


Figure 2.17: Watermarking system.

binary pattern associated with the codebook entry that maximizes the correlation measure is selected as the received data. The syntactic consistency of the data is constantly analyzed to ensure synchronization, as described in the previous section.

The output binary sequence is then converted back into AudioDNAs. For each 8-s audio segment, the corresponding fingerprint data is repeated several times in the watermark (16 times in average). Possible detection errors (including most errors caused by malicious attacks) can then be corrected by a simple majority rule, providing a replica of the original fingerprint of the signal.

**Matching and Report**

Finally, the fingerprint of the watermarked signal is extracted and compared with the original fingerprint obtained from the watermark. If the two sequences of AudioDNAs match perfectly, the system concludes that the signal has not been modified after watermarking; otherwise, the system determines the instants associated to the non-matching AudioDNAs, which correspond the approximate locations where the signal has been corrupted. Identical AudioDNAs slightly shifted in time are considered to match, since such shifts may occur when the signal is submitted to content-preserving transformations.

### 2.4.5 Simulations

#### 2.4.5.1 Experimental conditions

Results of cut-and-paste tests are presented for four 8-s test signals: two songs with voice and instruments (signal “cher”, from Cher’s “Believe”, and signal “estrella_morente”, a piece of flamenco music), one song with voice only (signal “svega”, Suzanne Vega’s “Tom’s diner”, a cappella version), and one speech signal (signal “the_breakup”, Art Garfunkel’s “The breakup”). The signals were sampled at 32 kHz and were inaudibly watermarked with a signal to watermark power ratio of 23 dB in average.

#### 2.4.5.2 Results

Figure 2.18 shows the simulation results for all test signals. For each signal, the two horizontal bars represent the original signal (upper bar) and the watermarked and attacked signal (lower bar). Time is indicated in seconds on top of the graph. The dark-gray zones correspond to attacks: in the upper bar, they represent segments that have been *inserted* into the audio signal, whereas in the lower bar they represent segments that have been *deleted* from the audio signal. Fingerprint information (i.e. the AudioDNAs) is marked over each bar.

For all signals, the original fingerprint was successfully reconstructed from the watermark. Detection errors introduced by the cut-and-paste attacks were eliminated by exploiting the redundancy of the information stored in the watermark.

A visual inspection of the graphs in Figure 2.18 shows that the AudioDNAs in the vicinities of the attacked portions of the signal were always modified. These corrupted AudioDNAs allow the system to determine the instant of each attack within a margin of approximately ±1 second.

For the last signal ("the_breakup"), we also observe that the attacks induced two changes in relatively distant AudioDNAs (approximately 2 s after the first attack and 2 s before the second one). This can be considered a false alarm, since the signal was not modified in that zone.

### 2.4.6 Advantages of the Mixed Approach

In this subsection, we summarize the main advantages of the mixed approach in comparison with other integrity-verification methods:

- No side information is required for the integrity test; all the information needed is contained in the watermark or obtained from the audio signal itself. This is not the case for systems based solely on fingerprinting, since the original fingerprint is necessary during the integrity test. Systems based solely on watermarking may also require side information, as the data embedded into the signal cannot be deduced from the signal itself and must be stored elsewhere;

- Slight content-preserving distortions do not lead the system to "false alarms", since the fingerprint and the watermark are not affected by these transformations. Hashing methods (such as MD5) and fragile watermarks generally do not resist such transformations;

- In general, localized modifications in the audio signal also have a localized effect on the fingerprint, which enables the system to determine the approximate locations where the signal has been corrupted. This is not the case for simple hashing methods, since the effects of a localized modification may be propagated to the entire signal;

- Global signal modifications can also be detected by the system; in this case, the entire fingerprint will be modified and/or the watermark will not be successfully detected;

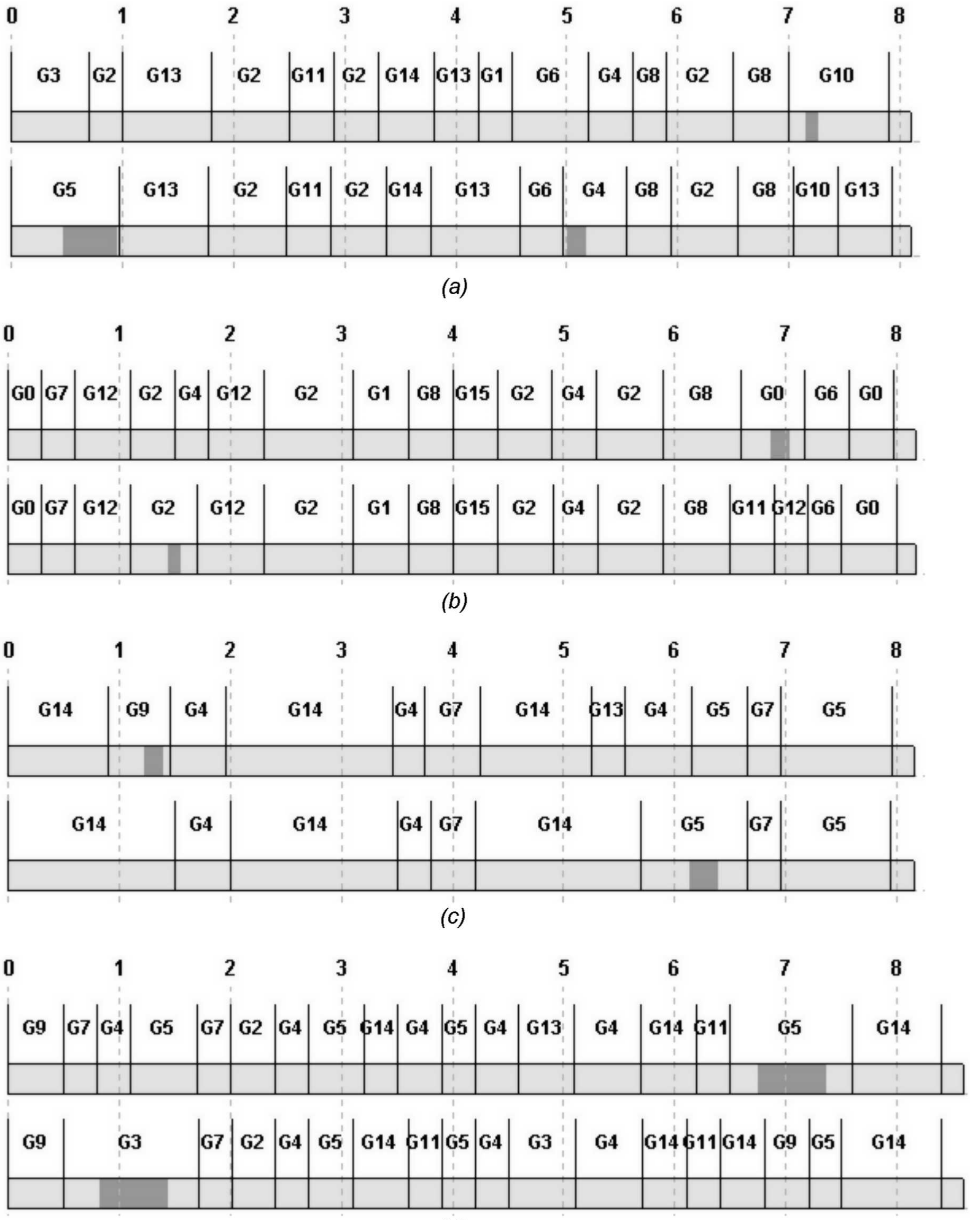
- This method is well suited for streaming audio, since all the processing can be done in real time.

### 2.4.7 Conclusions

In this section, we have presented a system for integrity verification of audio recordings based on a combination of watermarking and fingerprinting. By exploiting both techniques, our system avoids most drawbacks of traditional integrity-verification systems based solely on fingerprinting or watermarking. Unlike most traditional approaches, no side information is required for integrity verification. Additionally, the effect of localized modifications generally do not spread to the rest of the signal, enabling the system to determine the approximate location of such modifications. Experimental results confirm the effectiveness of the system.

As next steps in this research, we will consider possible developments in order to further increase overall system reliability, particularly in what concerns false alarms (i.e. signal modifications detected after content-preserving transformations or in zones where the signal was not modified). More efficient coding schemes will also be considered for fingerprint encoding prior to embedding.

In the next Section we stretch the boundaries of usual fingerprinting usage. The identification framework will be further extended to allow for similarity type of search and navigation. Additionally a dimensionality reduction tool will be evaluated for its applicability in audio asset visualization.


0 1 2 3 4 5 6 7 8
G3 G2 G13 G2 G11 G2 G14 G13 G1 G6 G4 G8 G2 G8 G10
G5 G13 G2 G11 G2 G14 G13 G6 G4 G8 G2 G8 G10 G13
0 1 2 3 4 5 6 7 8
G0 G7 G12 G2 G4 G12 G2 G1 G8 G15 G2 G4 G2 G8 G0 G6 G0
G0 G7 G12 G2 G12 G2 G1 G8 G15 G2 G4 G2 G8 G11 G12 G6 G0
0 1 2 3 4 5 6 7 8
G14 G9 G4 G14 G4 G7 G14 G13 G4 G5 G7 G5
G14 G4 G14 G4 G7 G14 G5 G7 G5
0 1 2 3 4 5 6 7 8
G9 G7 G4 G5 G7 G2 G4 G5 G14 G4 G5 G4 G13 G4 G14 G11 G5 G14
G9 G3 G7 G2 G4 G5 G14 G11 G5 G4 G3 G4 G14 G11 G14 G9 G5 G14

(a)

(b)

(c)

(d)

Figure 2.18: Simulation results: (a) signal "cher"; (b) signal "estrella_morente"; (c) signal "svega"; (d) signal "the_breakup".

## 2.5 Content-based Retrieval

In this section we experiment with another possible application of fingerprinting. Deriving compact signatures from complex multimedia objects and efficient similarity metrics are essential steps in a content-based retrieval system. Fingerprinting could be expanded to extract information from the audio signal at different abstraction levels, from low level descriptors to higher level descriptors. Especially, higher level abstractions for modeling audio hold the possibility to extend the fingerprinting usage modes to content-based navigation, search by similarity, content-based processing and other applications of Music Information Retrieval. In a query-by-example scheme, the fingerprint of a song can be used to retrieve not only the original version but also "similar" ones. In this section, the fingerprinting scheme described in section 2.3 is used together with a heuristic version of Multidimensional Scaling (MDS) named *FastMap* to explore for its potential uses in audio retrieval and browsing. *FastMap*, like MDS, maps objects into an Euclidean space, such that similarities are preserved. In addition of being more efficient than MDS it allows query-by-example type of query, which makes it suitable for a content-based retrieval purposes.

### 2.5.1 Introduction

The origin of this experiment is the research on a system for content-based audio identification. Details on the system were described in Section 2.3 and (Cano et al., 2002a). Basically the system decomposes songs into sequences of an alphabet of sounds, very much like speech can be decomposed into phonemes. Once having converted the audio into sequences of symbols, the identification problem results in finding subsequences in a superstring allowing errors, that is, approximate string matching. If we compare one sequence—corresponding to an original song in the database—to the whole database of sequences we retrieve a list of sequences sorted by similarity to the query. In the context of an identification system, this list reflects which songs the query—a distorted version of an original recording (Cano et al., 2002a)—can be more easily confused with. Of course, studying this for each song is a tedious task and it is difficult to extract information on the matching results for the whole database against itself. Indeed, the resulting distances displayed in a matrix are not very informative at first sight. One possible way to explore these distances between songs by mere visual inspection is Multidimensional Scaling. MDS makes it possible to view a database of complex objects as points in an Euclidean space where the distances between points correspond approximately to the distances between objects. This plot helps to discover some structure in the data in order to study methods to accelerate the song matching search, like quick discarding of candidates or spatial indexing methods. It can also be used as a test environment to compare different audio parameterization as well as their corresponding intrinsic distances independently of the metrics. Finally, it also provides an interesting tool for content-based browsing and retrieval of songs.

### 2.5.2 Related Work

Other projects that offer visual interfaces for browsing are the *Sonic Browser* (Maidin and Fernström, 2000), *Marsyas3D* (Tzanetakis and Cook, 2001) or *Islands of Music* (Pampalk et al., 2002a). The *Sonic Browser* uses sonic spatialization for navigating music or sound databases. In (Maidin and Fernström, 2000) melodies are represented as objects in a space. By adding direct sonification, the user can explore this space visually and aurally with a new kind of cursor function that creates an aura around the cursor. All melodies within the aura are played concurrently using spatialized sound. The authors present distances for melodic similarity but they acknowledge the difficulty to represent the melodic distances in an Euclidean space. *Marsyas3D* is a prototype audio browser and editor for large audio collections. It shares some concepts with the *Sonic Browser* and integrates them in an extended audio editor. To solve the problem of reducing dimensionality and mapping objects into 2D or 3D spaces, Principal Component Analysis (PCA) is proposed. Pampalk et al. (2002a) proposed the use of Self-Organizing Map (SOM), an artificial neural network, which models biological brain functions, to perform a non-linear mapping of the song space into 2D. A drawback of these solutions is that the object must be a vector of features and thus it does not allow the use of the edit distance or any other arbitrary distance metrics. In the next subsection, the use of Multidimensional Scaling and *FastMap* are presented.

### 2.5.3 Mapping complex objects in euclidean spaces

#### 2.5.3.1 Multidimensional Scaling

Multidimensional scaling (MDS)(Shepard, 1962; Kruskal, 1964; Faloutsos and Lin, 1995; Basalaj, 2001) is used to discover the underlying (spatial) structure of a set of data from the similarity, or dissimilarity, information among them. It has been used for some years in e.g. social sciences, psychology, market research, physics. Basically the algorithm projects each object to a point in a k-dimensional space trying to minimize the *stress* function:

$$stress = \sqrt{\frac{\sum\limits_{i,j}(\widehat{d}_{ij} - d_{ij})^2}{\sum\limits_{i,j} d_{ij}^2}}$$

where $d_{ij}$ is the dissimilarity measure between the original object $O_i$ and $O_j$ and $\widehat{d}_{ij}$ is the Euclidean distance between the projections $P_i$ and $P_j$. The *stress* function gives the relative error that the distances in k-dimensional space suffer from, on average. The algorithm starts assigning each item to a point in the space, by random or using some heuristics. Then, it examines each point, computes the distances from the other points and moves the point to minimize the discrepancy between the actual dissimilarities and the estimated distances in the Euclidean space. As described in (Faloutsos and Lin, 1995), the MDS suffers from two drawbacks:

- It requires $O(N^2)$ time, where $N$ is the number of items. It is therefore impractical for large datasets.

- If used in a 'query by example' search, each query item has to be mapped to a point in the k-dimensional space. MDS is not well-suited for this operation: Given that the MDS algorithm is $O(N^2)$, an incremental algorithm to search/add a new item in the database would be $O(N)$ at best.

#### 2.5.3.2 FastMap

To overcome these drawbacks, Faloutsos and Lin propose an alternative implementation of the MDS: *FastMap.* Like MDS, *FastMap* maps objects into points in some k-dimensional space, such that the (dis)similarities are preserved. The algorithm is faster than MDS (being linear, as opposed to quadratic, w.r.t. the database size $N$), while it additionally allows indexing. They pursue fast searching in multimedia databases: mapping objects into points in k-dimensional spaces, they subsequently use highly fine-tuned spatial access methods (SAMs) to answer several types of queries, including the 'Query by Example' type. They aim at two benefits: efficient retrieval, in conjunction with a SAM, as discussed above, visualization and data-mining.

### 2.5.4 Results and Discussion

To evaluate the performance of both least squares MDS and *FastMap*, we used a test bed consisting of 2 data collections. One collection consists in 1840 popular songs and the second collection in 250 isolated instrument sounds (from IRCAM's Studio OnLine). Several dissimilarity matrices were calculated with different distance metrics. In Figure 2.19 the representation of the song collection as points calculated with MDS and *FastMap* is shown. The MDS map takes a considerably longer time to calculate than the *FastMap*'s (894 vs 18.4 seconds) although several runs of *FastMap* are sometimes needed to achieve good visualizations. We did not objectively evaluate *FastMap* and MDS (objective evaluations of data representation techniques are discussed in (Basalaj, 2001)), but on an preliminary check of the results, MDS maps seem to be of higher quality. MDS, on the other hand, presents a high computational cost and do not account for the indexing/retrieval capabilities of the *FastMap* approach.

We have presented the use of the existing *FastMap* method for improving a content-based audio identification system. The tool proves to be interesting, not only for audio fingerprinting research, but also as a component of a search-enabled audio browser. It allows the browsing and retrieval of audio repositories using heterogeneous mixes of attributes and arbitrary distances.

Visually exploring the representational space of audio data may reveal the possible weakness of a specific parameterization. We tested the tool with audio objects less complex than songs, such as harmonic or percussive isolated sounds, for which perceptually-derived distances exist. In this case the results are excellent. But songs have a multidimensional nature, they account for many aspects of interest: melody, rhythm, timbre, and so on. Such audio data calls for powerful and complex representations, new paradigms of representation and other interfaces are necessary to allow a user to browse flexibly. That includes visualization tools that accept any data representation or distance definitions, from physical feature vectors (e.g. spectral flatness), up to subjective distances defined by experts (respecting e.g. the "mood").

## 2.6 Conclusions

We have presented a review of the research carried out in the area of audio fingerprinting. Furthermore a number of applications which can benefit from audio fingerprinting technology were discussed. An audio fingerprinting system generally consists of two components: an algorithm to generate fingerprints from recordings and algorithm to search for a matching fingerprint in a fingerprint database. We have shown that although different researchers have taken different approaches, the proposals more or less fit in a general framework. In this framework, the fingerprint extraction includes a front-end where the audio is divided into frames and a number of discriminative and robust features is extracted from each frame. Subsequently these features are transformed to a fingerprint by a fingerprint modeling unit which further compacts the fingerprint representation. The searching algorithm finds the best matching fingerprint in a large repository according to some similarity measure. In order to speed up the search process and avoid a sequential scanning of the database, strategies are used to quickly eliminate non-matching fingerprints. A number of the discussed audio fingerprinting algorithms are currently commercially deployed, which shows the significant progress that has been made in this research area. There is, of course, room for improvement in the quest for more compact, robust and discriminative fingerprints and efficient searching algorithms. It also needs to be seen how the identification framework can be further extended to browsing and similarity retrieval of audio collections.

We have described in detail an implementation of a fingerprinting system designed for identification: broadcast monitoring, and for integrity verification. We have shown one possible extension of fingerprinting for content-based navigation of music collections.

During the next Chapter we attempt to move from low-level audio description toward higher-level descriptors.

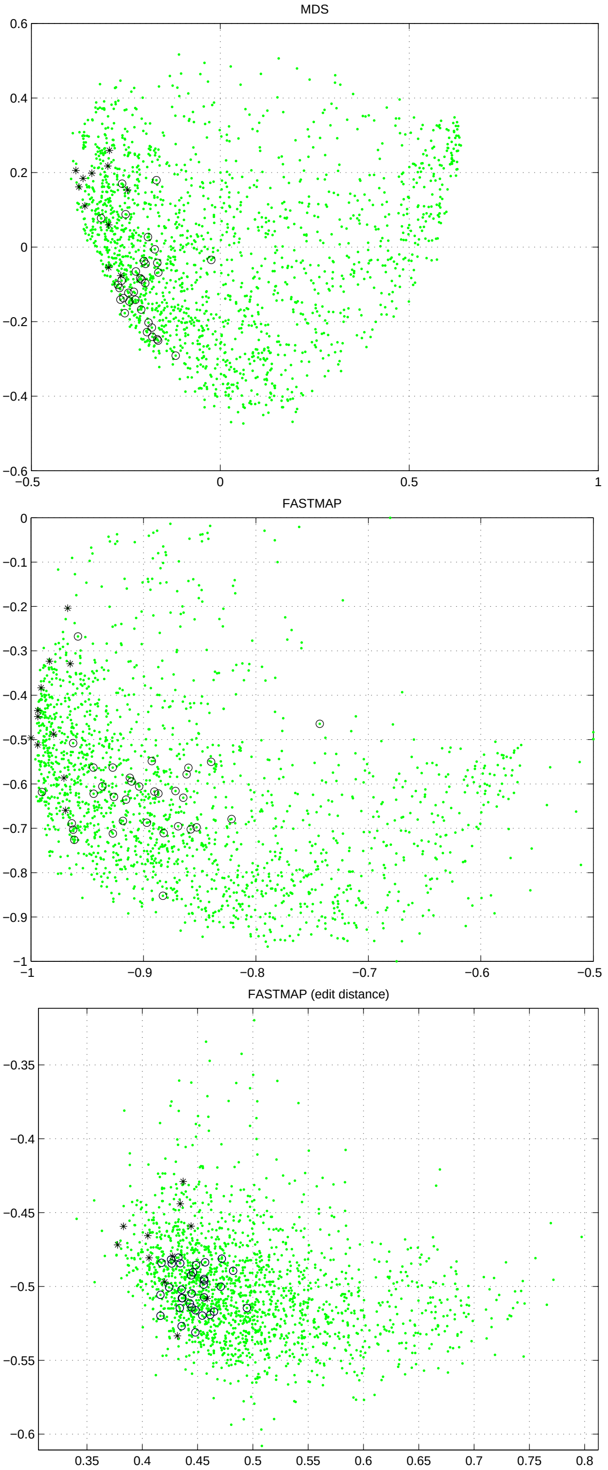


Figure 2.19: Representing 1840 songs as points in a 2-D space by MDS (top) and *FastMap* (bottom). Asterisks and circle correspond to songs by Rex Gyldo and Three Doors Down respectively.

# Chapter 3

# Semantic Audio Management

Traditional IR offers the ability to search and browse large amounts of text documents and presenting results in a “ranked-by-relevance” interface. For the case of multimedia there are two main approaches: The first is to generate textual indices manually, semi automatically or automatically and then use traditional IR. The other approach is to use content-based retrieval, where the query is non textual and a similarity measure is used for searching and retrieval.

In the previous chapter, we have critically reviewed strengths of a low-level audio description technique. While it is able to identify a distorted recordings in huge databases, and even extended to perform similarity searches on huge audio databases and “query by example” type of queries, it is not able to textually describe audio assets like users generally prefer to search for them, i.e. using textual high-level semantic descriptions.

This chapter focuses on bridging the semantic gap when interacting with audio using knowledge management techniques. The semantic gap relates to the difficulty of moving from low-level features that can be automatically extracted from audio toward higher-level features that can be understood by humans. We will validate this approach on the specific problem of sound effects retrieval. During this chapter we will present, implement algorithms and evaluate its performance on top of one of the biggest sound effects providers search engine, the sound-effects-library.com.[1] As we outlined in Chapter 1.2, there is interest in making sound effects metadata easily searchable, less expensive to create and reusable

[1] http://www.sound-effects-library.com

to support possible new users—including computers—and applications.

The main sections of the chapter will include sound ontology management, automatic annotation, a sound effects search engine that integrates both low-level and high-level content-based algorithms and finally an application for intelligent authoring built on top of the search system.

Section 3.1 deals with sound description and ontology management. If one of the goals is to build computational models able to label audio like humans do, the first step is analyzing how users– in this case librarians and audio experts– label sounds. In the section we review some taxonomic proposals for audio description found in the literature, which types of descriptions are actually found in SFX commercial systems and how to encode such descriptions in a format that humans and computers can understand. We will highlight the connections to a multimedia standardization process: MPEG-7 (Manjunath et al., 2002).

Section 3.2 introduces the automatic sound annotation. It starts with an overview current state of the art on sound annotation where we identify some limitations–mainly scalability in the number of classes described and level of detail in the annotations — and discuss the implementation of a general sound annotator that is able to generate high-level detail descriptions. These descriptions can be used as is or presented to a librarian for its validation.

Section 3.3 describes the integration of low and higher level methods to browse and search audio in a commercial SFX provider on line system (Cano et al., 2004f). Low-level content-based techniques are used for "query-by-example" and visualization. High-level and knowledge based are used for semi-automatic labeling the database as well as better control on the search process.

Finally in Section 3.4 we illustrate an intelligent authoring application: Automatic ambiance generation, that builds upon the search engine and leverages the above described techniques.

This chapter relates to the dissertation goals 3 and 4 as presented in Section 1.3.

## 3.1 Sound Ontology Management

Sound effect management systems rely on classical text descriptors to interact with their audio collections. Librarians tag the sounds with textual description and file them under categories. Users can then search for sounds matching keywords as well as navigating through category trees. Audio filing and logging is a labor-intensive error-prone task. Moreover, languages are imprecise, informal and words have several meanings as well as several words for each meaning. Finally, sounds are multi modal, multicultural and multifaceted and there is not an agreement in how to describe them.

Despite the difficulties inherent in creating SFX metadata, there is need to catalog assets so as to reuse afterward. Media assets have value. As Flank and Brinkman (2002) point out, there are many situations where reusing media content is, not only not economically appealing—think of the cost of sending a team to record Emperor penguins in their natural habitat—but sometimes audio cannot be re-recorded—like natural catastrophes or historical events (Flank and Brinkman, 2002). Complete digital media management solutions include media archiving and cataloging, digital right management and collaborative creative environments. This section focuses on the knowledge management aspects of sound effect descriptions with the purpose of making metadata easily searchable, less expensive to create and reusable to support possible new users—including computers—and applications.

MPEG-7 offers a framework for the description of multimedia documents (Manjunath et al., 2002; Consortium, 2001, 2002). The description tools for describing a single multimedia document consider semantic, structure and content management descriptions. MPEG-7 content semantic description tools describe the actions, objects and context of a scene. In sound effects, this correlates to the physical production of the sound in the real world, “1275 cc Mini Cooper Door Closes” , or the context, “Australian Office Atmos Chatter Telephones”. MPEG-7 content structure tools concentrate on the spatial, temporal and media source structure of multimedia content. Indeed, important descriptors are those that describe the perceptual qualities independently of the source and how they are structured on a mix. Content management tools are organized in three areas: Media information—which describes storage format, media quality and so on, e.g: “PCM Wav 44100Hz stereo”—, Creation information—which describes the sound generation process, e.g: who and how created the sound—and finally usage information—which describes the copyrights, availability of the content and so on (Manjunath et al., 2002). Figure 3.1 shows an example on how to describe a SFX inspired on MPEG-7 Multimedia Description Schemes (MDS). The original description is "Golf Swing And Hole" and had been added to the following categories: "Whooshes, Golf, Sports:Golf:Hits:Swings:Swishes".

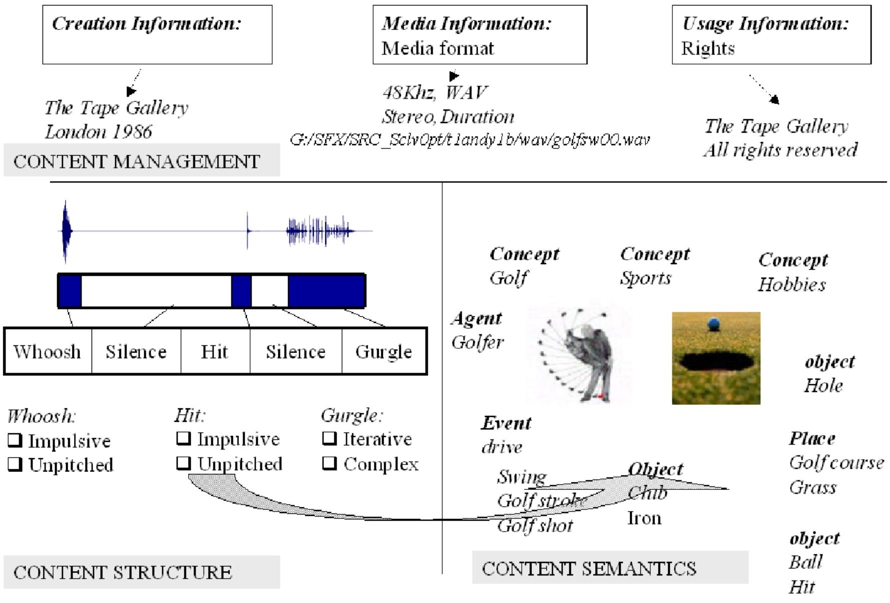


Figure 3.1: Example of a SFX description inspired on MPEG-7 Multimedia Description Scheme.

The use of MPEG-7 description schemes provide a framework suitable for Multimedia description. In order to ensure interoperability, overcome the polysemy in natural languages and allow the description to be machine readable, the terms within the fields need to be standard. It is important to know whether "bike" refers to "bicycle" or to "motorcycle". MPEG-7 classification schemes allow to define a restrained vocabulary that defines a particular domain as categories with semantic relationships, e.g: Broader term, narrow term, related term and so on. Casey (2002) presents an example of using the classification scheme to define a hierarchical sound classification model with 19 leaf nodes. However it is very complicated to devise and maintain taxonomies that account the level of detail needed in a production-size sound effect management system—the categories needed in professional environments exceed the several thousands and they do not follow a hierarchical structure. We have found that it is faster to start developing taxonomies on top on a semantic network such as WordNet rather than starting from scratch. WordNet[2] is an English lexical network designed following psycholinguistic theories of human lexical memory in a long-term collaborative effort (Miller, 1995). We have developed a WordNet editor to expand it with specific concepts from audio domain, such as "close-up"– which refers to recording conditions—and other specific concepts from the real world— e.g.: a Volvo is a type of a car— as well as with the perceptual ontologies. To do so, besides reviewing the existing literature for sound events description, we have mined the concepts associated to sounds by major sound effect library providers and added them to the WordNet. Such a knowledge system is not only useful for retrieval (see Section 3.3) but it is also used as ontology backbone for general sounds classification (see Section 3.2).

### 3.1.1 On Sound Effects Cataloging

One of the most time-demanding and error-prone task when building a library of sound effects is the correct labeling and placement of a sound within a category. The information retrieval model commonly used in commercial search engines is based on keyword indexing. Librarians add descriptions for the audio. The systems match the descriptions against the users' query to retrieve the audio. Sounds are difficult to describe with words. Moreover, the librarian must add the text thinking on the different ways a user may eventually look for the sound, e.g: "dinosaur, monster, growl, roar" and at the same time with the maximum detail. We display in Figure 3.2 some of the fields the librarian could consider when describing a SFX.

The vagueness of the query specification, normally one or two words, together with the ambiguity and informality of natural languages affects the quality of the search: Some relevant sounds are not retrieved and some irrelevant ones are presented to the user. Sound effect management systems also allow browsing for sounds in manually generated categories. It is difficult to manage large category structures. Big corpses may be labeled by different librarians that follow somewhat different conventions and may not remember under which

[2] `http://www.cogsci.princeton.edu/~wn`

category sounds should be placed (e.g: Camera:clicks or clicks:camera). Several ways of describing a sound include: source centered description, perceptual, post-production specific and creation description (See Figure 3.2).

#### 3.1.1.1 Semantic Descriptors

Semantic descriptors usually refer to the source of the sound, that is, what has physically produced the sound, e.g: "car approaching". They also refer to the context, e.g: "Pub atmos". The importance of source-tagging for sound designers is questioned by L.Mott (1990). Mott explains that the sound engineer should concentrate on the sound independently on what actually produced it because in many occasions the natural sounds do not fulfill the expectations and must be replaced with sounds of distinct origin, e.g: "arrow swishes" or "gun bangs". There are, however, cases where having the true sound can add quality to a production, e.g: Using the real atmosphere of a Marrakesh market tea house. Besides, describing the source of a sound is sometimes easier than describing the sound itself. It is difficult to describe the "moo of a cow" without mentioning "moo or cow" but just perceptual attributes.

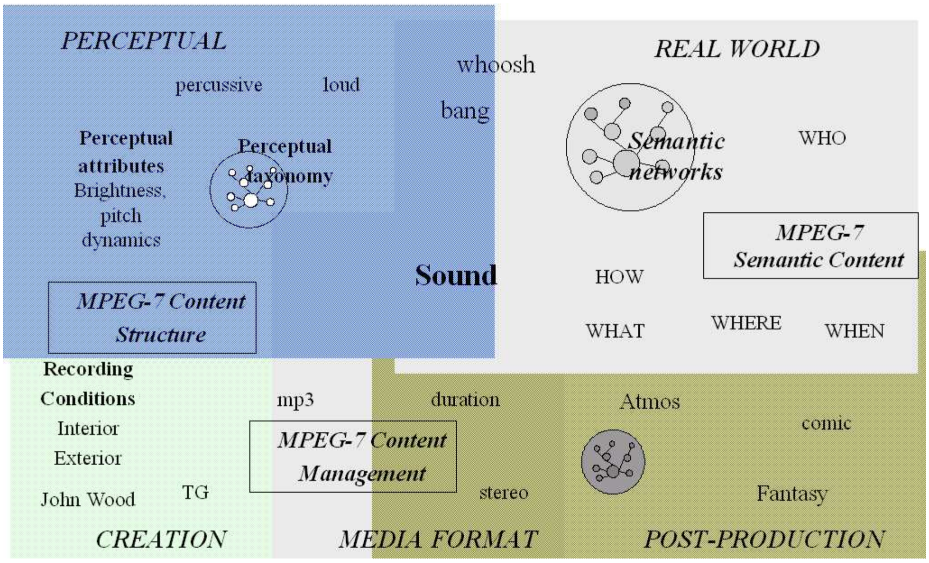


Figure 3.2: Block Diagram of the System.

| | |
|---|---|
| rumbles, roars, explosions, crashes, splashes, booms | Whistles Hisses Puffing |
| Snorts, Whispers, Murmers, Mumbles, Grumbles, Gurgles | Screeches, Creaks, Rustles, Buzzes, Crackles, Scrapes |
| *Noises make by percussion on:* Metal, Wood, Skin, Stone, Pottery, etc. | *Voices of Animals and Men:* Shouts, Screams, Groans, Shrieks, Howls |

Table 3.1: Russolo' Sound-Noise Categories

#### 3.1.1.2 Perceptual Descriptors

Perceptual descriptors describe the perceptual qualities independently of the source that actually created the sound. Classical research on auditory perception has studied the world of sounds within a multidimensional space with dimensions such as pitch, loudness, duration, timbral brightness, and so on (SOB, 2003). Since they refer to the properties of sound, sometimes there is a mapping between sound descriptions to perceptual measurable features of the sound.

Another possibility to describe sounds is the use of onomatopoeia, words that imitate sounds and are extensively used in comics— e.g: roar, mmm, ring. The futurist painter Russolo (1986) proposed in 1913 a categorization of noises in six separate groups: Rumbles, whistles, whispers, screeches, noises obtained by percussion and voices of animals and men (see Table 3.1).

Schaeffer (1966), in the search of a lexicon to describe sounds, introduced the reduced listening (*écoute réduite*) which consists in the disposition of the listener to focus on the sound object itself with no reference to the source causing its production. His *solfège* of sound objects (see Table 3.2) considered attributes such as mass (perception of "pitchiness") or harmonic timbre (bright/dull, round/sharp). The original aim of such a classification scheme was to come up with a standarised symbolic notation (such as western music notation) for electro acoustic music.

<table>
<tr><th colspan="6"><i>MATTER CRITERIA</i></th></tr>
<tr><td colspan="2">MASS<br>Perception of “noiseness”</td><td colspan="2">HARMONIC TIMBRE<br>Bright/Dull</td><td colspan="2">GRAIN<br>Microstructure of the sound</td></tr>
<tr><th colspan="6">SHAPE CRITERIA</th></tr>
<tr><td colspan="3">DYNAMICS<br>Intensity evolution</td><td colspan="3">ALLURE<br>Amplitude or Frequency Modulation</td></tr>
<tr><th colspan="6">VARIATION CRITERIA</th></tr>
<tr><td colspan="3">MELODIC PROFILE: pitch variation type</td><td colspan="3">MASS PROFILE<br>Mass variation type</td></tr>
</table>

Table 3.2: Schaeffer’ *Solfege* of sound objects

Gaver (1993) introduced a taxonomy of environmental sounds on the assertion that sounds are produced by interaction of materials. The hierarchical description of basic sonic events include those produced by vibrating objects (impacts, scraping and others), aerodynamic sounds (explosions, continuous) and liquid sounds (dripping and splashing). The ecological approach to perception distinguishes two types of invariants (i.e.: High-order acoustical properties) in the sound generation: structural and transformational. Structural refer to the objects properties meanwhile transformational refer to the change they undergo (Gibson, 1979).

Schafer (1977) classifies sounds according to their physical characteristics (acoustics), by the way they are perceived (psychoacoustics), according to their function and meaning (semiotics and semantics); or according to their emotional or affective qualities (aesthetics). Since he is interested in analyzing the sonic environment—soundscape—he adds to Schaeffer sound object description information on the recording settings, e.g: estimated distance from the observer, estimated intensity of the original sound, whether it stands clear out of the background, environmental factors: short reverb, echo.

#### 3.1.1.3 Post-production Specific Descriptors

Other important searchable metadata are Post-production specific. According to L.Mott (1990), the categories of sounds according are: Natural sounds (actual source sound), characteristic sounds (what a sound should be according to someone), comedy, cartoon, fantasy.

#### 3.1.1.4 Creation Information

Creation metadata describes relevant information on the creation or recording conditions of the sound. Creation terms that we have found mining SFX descriptions are the library that produced the sound and the engineer that recorded it. Most of the terms we have found refer to the recording conditions of the sound, e.g: to record a "car door closing" one can place the microphone in the interior or in the exterior. Some examples of such descriptors are: interior, exterior, close-up, live recording, programmed sound, studio sound, treated sound. These terms have been added to the taxonomies.

### 3.1.2 Ontology Management

The use of taxonomies or classification schemes alleviates some of the ambiguity problems inherent to natural languages, yet they pose others. It is very complicated to devise and maintain classification schemes that account for the level of detail needed in a production-size sound effect management system. The MPEG-7 standard provides description mechanisms and ontology management tools for multimedia documents (Manjunath et al., 2002). Celma *et al.* built a flexible search engine for opera works using classification schemes of the MPEG-7 framework (Celma and Mieza, 2004). Even though, powerful, the approach would require a huge human effort to extend it for SFX. SFX many times are described referring to the source that produced it. It is not trivial to put terms that describe the world in classification schemes. According to the latest version of WordNet (WordNet 2.0), the number of distinct terms is 152,059 and the number of concepts 115,424. WordNet is well suited as starting point for ontology-backbone.

Standard dictionaries organize words alphabetically. WordNet organizes concepts in synonym sets, *synsets*, with links between the concepts like: broad sense, narrow sense, part of, made of and so on. It knows for instance that the word piano as a noun has two senses, the musical attribute that refers to “low loudness” and the musical instrument. It also encodes the information that a grand piano is a type of piano, and that it has parts such us a keyboard, a loud pedal and so on. Such a knowledge system is useful for retrieval. It can for instance display the results of a query “car” in types of cars, parts of car, actions of a car (approaching, departing, turning off). The usefulness of using WordNet in Information Retrieval has been proved useful in the case of image retrieval for example in (Aslandogan et al., 1997) and in general multimedia asset management (Flank, 2002). Other available general-purpose ontologies are Cyc [3] which attempts to create an ontology and database of everyday common-sense knowledge, with the goal of enabling AI applications to perform human-like reasoning and SUMO, created as part of the IEEE Standard Upper Ontology Working Group [4], to develop a standard upper ontology that promotes data interoperability, information search and retrieval, automated inferencing, and natural language processing. Even though WordNet already organizes plenty of terms, it sometimes lacks specific knowledge of the specific domain of sound effects and audio in general, such as “close-up” when referring to the recording technique or that a “747” is an airplane (this property is known as lack of “granularity”). We have developed an editor to expand WordNet with terms and concepts from perceptual taxonomies to describe sounds and mining legacy metadata from sound effects libraries. We have based the editor on the work by Brian (Brian, 2002). The addition of a new term, for instance Jaguar—referring to the car—would go as follows: We would first search for the term in WordNet to make sure it does not exist. There is one concept associated with jaguar:

```
1. jaguar, panther, Panthera onca, Felis onca --
   a large spotted feline of tropical America
   similar to the leopard;
```

Since this synset does not apply, the second step is finding a node in the taxonomy to which attach the new concept, in our case if ask the synsets for “car”:

```
1. car, auto, automobile, machine, motorcar --
   4-wheeled motor vehicle; usually propelled
   by an internal combustion engine;
2. car, railcar, railway car, railroad car --
   a wheeled vehicle adapted to the rails of
```

[3] http://www.cyc.com

[4] http://suo.ieee.org/

```
   railroad; "three cars had jumped the rails"
3. cable car, car -- conveyance for passengers
   or freight on a cable railway;
4. car, gondola -- car suspended from an airship
   and carrying personnel and cargo and power
   plant)
5. car, elevator car --where passengers ride up
   and down;
```

In this case we would choose connecting Jaguar with the first synset: “car, auto, ...”. The type of relationship between Jaguar and car is of hyponim/hypernim (Narrower/Broader). If we ask for the list of hypernims of the “jaguar” we get:

```
1. jaguar, panther, Panthera onca, Felis onca --
   a large spotted feline of tropical America
   similar to the leopard;
   => big cat, cat
     => feline, felid
       => carnivore
         => mammal
           ...
              => animal, animate being, beast
                => organism, being
                  => living thing, animate thing
                    => object, physical object
                      => entity, physical thing

2. Jaguar -- British car ...
 => car, auto, automobile, machine, motorcar
   => motor vehicle, automotive vehicle
     => wheeled vehicle
       => vehicle
         => conveyance, transport
           => instrumentality, instrumentation
             => artifact, artefact
```

```
               => object, physical object
                 => entity, physical thing
```

A hyponim acquires all the features of its hypernim. Therefore if we ask the meronyms—relation of being part of—of Jaguar in its second new meaning we get:

```
Sense 2
Jaguar
 HAS PART: accelerator, gas pedal, gas, throttle
 HAS PART: air bag
 HAS PART: automobile engine
 HAS PART: car horn, motor horn, horn, hooter
 HAS PART: bumper
 HAS PART: car door
 ...
```

To put another example, the sound "Thrush And Nightingale Various Calls" becomes labeled with the following set of concepts:

```
  01234719%n thrush -- (songbirds having
   brownish upper plumage with a spotted breast)
  01237641%n nightingale, Luscinia megarhynchos
    -- (European songbird noted for its melodious
   nocturnal song)
  05680983%n birdcall, call, birdsong, song --
   (the characteristic sound produced by a bird)
```

The numbers before the definitions correspond to the unique identifiers, *offsets*, of the concepts, or synonym sets, *synsets* as referred in the WordNet literature Miller (1995).

After the disambiguation of the terms used to label a database of 60,857 sounds from over 30 libraries of sound effects, music and music samples, we have 3,028 different concepts. The histogram of number of synsets assigned per sound sample is depicted in Figure 3.3. The higher the number of synsets, the more detailed is the description of the sound. Table 3.3

| # of Sounds | Synset | Terms and Glossary |
|---|---|---|
| 5653 | 03131431%n | drum, membranophone, tympan – (a musical percussion instrument; usually consists of a hollow cylinder with a membrane stretch across each end ) |
| 4799 | 13697183%n | atmosphere, ambiance, ambience – (a particular environment or surrounding influence; "there was an atmosphere of excitement" ) |
| 4009 | 06651357%n | rhythm, beat, musical rhythm – (the basic rhythmic unit in a piece of music; "the piece has a fast rhythm"; "the conductor set the beat" ) |
| 3784 | 07719788%n | percussion section, percussion, rhythm section – (the section of a band or orchestra that plays percussion instruments ) |
| 3619 | 14421098%n | beats per minute, bpm, metronome marking, M.M. |
| 3168 | 00006026%n | person, individual, someone, somebody, mortal, human, soul |

Table 3.3: Appearance number of most popular concepts (synsets).

shows the most commonly used concepts. The first column indicates the number of sounds that have been labeled with the synset, the second column, the offset (WordNet Synset-ID) and the third the glossary. The distribution of 3,028 synsets with respect its syntactic function is as follows: 2,381 nouns, 380 verbs, 251 adjectives and 16 adverbs (see Figure 3.4). The following are examples of disambiguation of captions into synsets:

```
Dalmatian Dog Bark Interior ->
   01778031%n dalmatian,  ...
   01752990%n dog, domestic dog, ...
   00826603%v bark -- make barking sounds
   00915868%a interior -- (situated ...

Cello pizzicato ->
   02605020%n cello, violoncello
         => bowed stringed instrument, string
           => stringed instrument
               => musical instrument, instrument

   00908432%a pizzicato -- ((of instruments in
      the violin family) to be plucked with the
      finger)
```

```
00422634%r pizzicato -- ((music) with a light
   plucking staccato sound)
```

The extended semantic network includes the semantic, perceptual and sound effects specific terms in an unambiguous way, easing the task for the librarian and providing higher control on the search and retrieval for the user. Further work needs to deal with concepts that appear on different parts-of-speech—pizzicato is both an adjective and an adverb—but are equivalent for retrieval purposes.

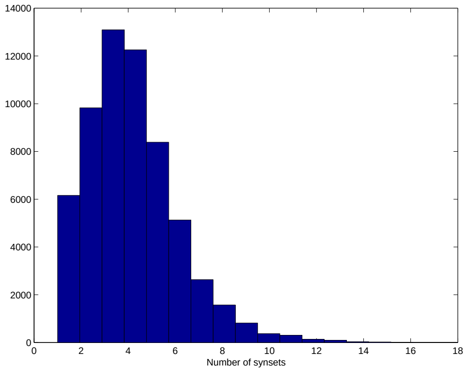


Figure 3.3: Histogram of the number of concepts assigned to each SFX. The higher the number of concepts the most detailed the specification of the SFX.

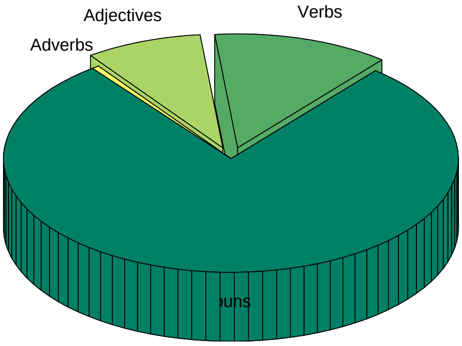


Figure 3.4: Distribution of nouns, verbs, adjectives and adverbs after disambiguating the tags associated to a SFX collection.

### 3.1.3 Summary

In this section, we have presented some of the problems in describing SFX. Specifically, how to store searchable and machine readable SFX description metadata. We have reviewed some of the literature for audio taxonomic classification as well as mined legacy SFX metadata. We have implemented a knowledge management system inspired on the MPEG-7 framework for Multimedia and relying on WordNet as taxonomy-backbone. This is convenient for librarians because they do not need to add many terms since many relations are given by the lexicon. Categories can be created dynamically allowing user can search and navigate through taxonomies based on psycholinguistic and cognitive theories. The terms—even though described externally as plain English—are machine readable, unambiguous and can be used for concept-based retrieval. Specific SFX terms as well as external taxonomies can be added to the lexicon. In the next Section, we will present a computational method for automatic generation of metadata.

## 3.2 Sound Annotation

Sound effects providers rely on classical text retrieval techniques to give access to manually labeled audio collections. The manual annotation is a labor-intensive and error-prone task. There are attempts toward metadata generation by automatic classification. State of the art of audio classification methods, except for reduced-domain tasks, is not mature enough for real world applications. Audio classification methods cannot currently provide the level of detail needed in a sound effects management system, e.g: "fast female footsteps on wood", "violin pizzicato with natural open strings" or "mid tom with loose skin bend at end". In audio classification, researchers normally assume the existence of a well defined hierarchical classification scheme of a few categories. On-line sound effects and music sample providers have several thousand categories. This makes the idea of generating a model for each category quite unfeasible, as several thousand classifiers would be needed. As a knowledgeable reader will agree, the usual number of categories appearing in academic research papers is not greater than a few tenths.

In this context, we present an all-purpose sound recognition system based on nearest-neighbor classification rule, which labels a given sound sample with the descriptions corresponding to the similar sounding examples of an annotated database. The terms borrowed from the closest match are unambiguous due to the use of WordNet[5] (Miller, 1995) as the taxonomy back-end. The tagging is unambiguous because the system assigns concepts and not just terms to sounds. For instance, the sound of a "bar" is ambiguous, the system will return "bar" as "rigid piece of metal or wood" or as "establishment where alcoholic drinks are served".

The rest of the Section is organized as follows: In Subsection 3.2.1 we briefly enumerate some approaches to the problem of automatic identification and we discuss the difficulties inherent in automatically describing any isolated sound with a high level of detail. In Subsection 3.2.2, we present a taxonomy that represents the real world extended for sound effects description. From Subsection 3.2.3 to 3.3.3 we describe the system setup as well as the main results of our evaluation of the system. We conclude the Section discussing

[5] `http://www.cogsci.princeton.edu/~wn/`

possible continuations of the approach.

### 3.2.1 Related work

Existing classification methods are normally finely tuned to small domains, such as musical instrument classification (Kostek and Czyzewski, 2001; Herrera et al., 2003), simplified sound effects taxonomies (Wold et al., 1996; Zhang and Kuo, 1999; Casey, 2002) or sonic environments, e.g: "street, pub, office, church" (Peltonen et al., 2002). Different audio classification systems differ mainly on the acoustic features derived from the sound and the type of classifier. Independently of the feature extraction and selection method and the type of classifier used, content-based classification systems need a reduced set of classes (e.g: less than 20) and a large number (e.g: 30 or more) of audio samples for each class to train the system.

Classification methods cannot currently offer the detail needed in commercial sound effects management. It would require to develop thousands of classifiers, each specialized in distinguishing little details and a taxonomy that represents the real world. Dubnov and Ben-Shalom (2003) point out that one of the main problems faced by natural sounds and sound effects classifiers is the lack of clear taxonomy. In musical instrument classification, the taxonomies more or less follow perceptual-related hierarchical structures (Lakatos, 2000). The assumption is that there is a parallelism between semantic and perceptual taxonomies in musical instruments. Accordingly, in such problems one can devise hierarchical classification approaches such as (Martin, 1999; Peeters and Rodet, 2003) in which the system distinguishes first between sustained and non-sustained sounds, and then among strings, woodwinds and so on. In every-day sound classification, there is no such parallelism between semantic and perceptual categories. On the contrary one can find hissing sounds in categories of "cat", "tea boilers", "snakes". Foley artists exploit this ambiguity and create the illusion of "crackling fire" by recording "twisting cellophane".

Moreover, the design and implementation of a taxonomy or classification scheme that include the concepts of the real world is a daunting task. The MPEG-7 standard provides mechanisms and taxonomy management tools for describing multimedia documents. Casey (2002) shows an example on how to build such a classification scheme using MPEG-7. However, it is very complicated to devise and maintain classification schemes that account for the level of detail needed in a production-size sound effects management system. We have found that it is much faster to start developing taxonomies on top of a semantic network such as WordNet rather than starting from scratch (see Section 3.1).

Slaney describes in (Slaney, 2002) a method of connecting words to sounds. He avoids the needs of taxonomy design when bridging the gap between perceptual and semantic spaces searching for hierarchies in an unsupervised mode. Barnard et al. (2003) describe a similar approach for matching words and images.

### 3.2.2 Taxonomy management

As we have presented in Section 3.1, WordNet is a lexical network designed following psycholinguistic theories of human lexical memory. Standard dictionaries organize words alphabetically. WordNet organizes concepts in synonym sets, called *synsets*, with links between the concepts. Such a knowledge system is useful for supporting retrieval functionalities in a a music and sfx search engine (Cano et al., 2004c). It can for instance display the results of a query "car" in types of cars, parts of car, and actions of a car (approaching, departing, turning off).

### 3.2.3 Experimental setup

The dataset used in the following experiments consists of 54,799 sounds from over 30 different libraries of sound effects, music and music samples. These sounds have been unambiguously tagged with the concepts of an enhanced WordNet. Thus a violin sound with the following caption:"violin pizzicato D#" has the following synsets:

- violin, fiddle – (bowed stringed instrument that is the highest member of the violin family; this instrument has four strings and a hollow body and an unfretted fingerboard and is played with a bow)
- pizzicato – ((of instruments in the violin family) to be plucked with the finger)
- re, ray – (the syllable naming the second (supertonic) note of any major scale in solmization)
- sharp – ((music) raised in pitch by one chromatic semitone; "C sharp")

In Figure 3.5, we show a histogram with the number of synsets the sounds have been labeled with after disambiguation. It should be clear that the higher the number of synsets, the higher the detail with which a sound is described. In average, a sound is labeled with 3.88 synsets. In Figure 3.6 we plot the rank-frequency analysis of the synsets. For this analysis we counted the occurrence of different synsets and then sorted them according to descending frequency. The plot is repeated for various parts of speech, specifically: noun, verb, adjective and adverb. The distribution of 3,028 synsets with respect its syntactic function is as follows: 2,381 nouns, 380 verbs, 251 adjectives and 16 adverbs. The number of synsets for which there are ten or more examples sounds is 1,645.

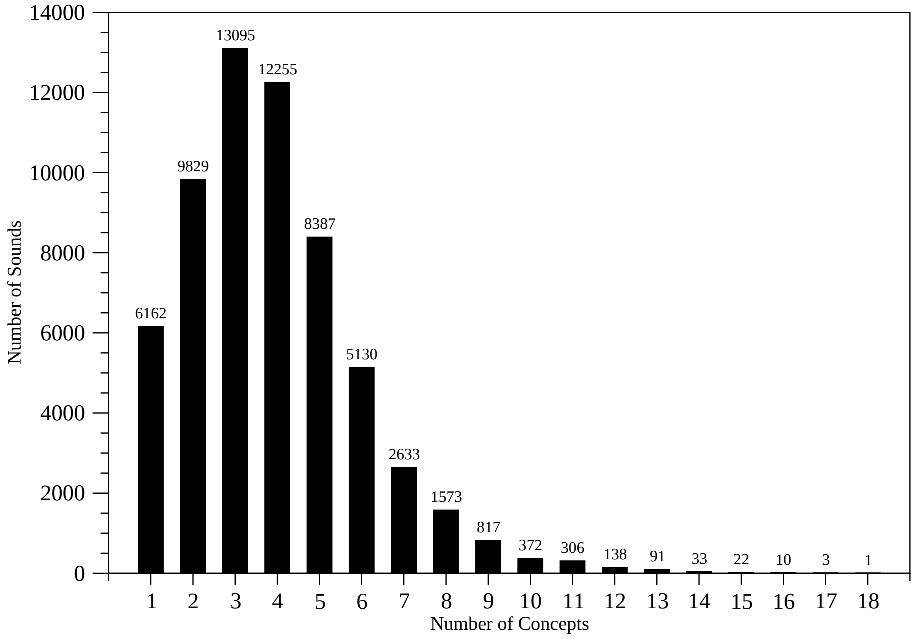


Figure 3.5: Histogram of number of synsets (concepts) per sound

The classifier uses a set of 89 features and a nearest-neighbor classifier (Cover and Hart, 1967) using a database of sounds with WordNet as a taxonomy backbone. We refer to Section 3.2.4 for the features to Section 3.2.5 for the classifier.

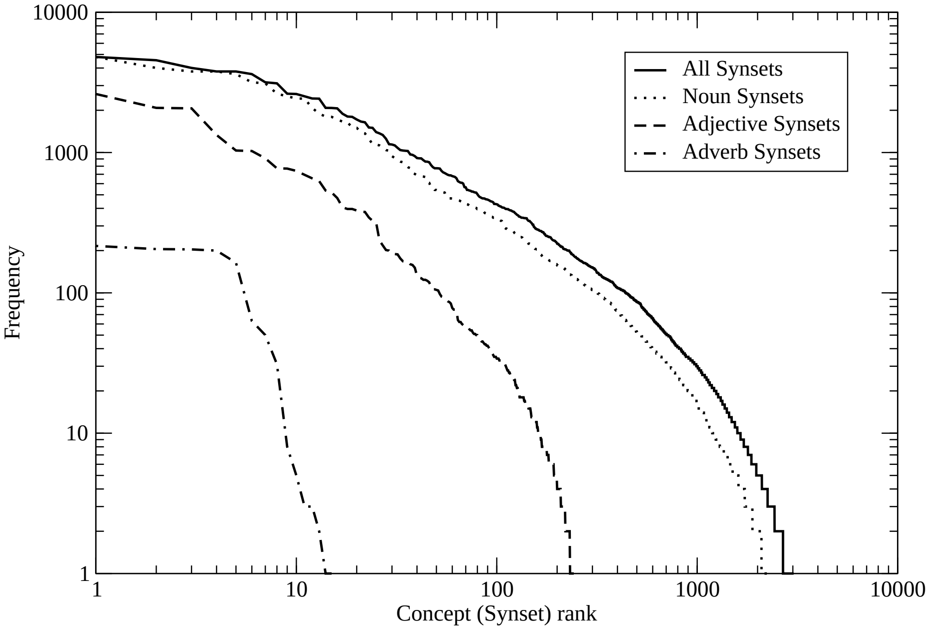


Figure 3.6: Loglog plot of the number of sounds described per synset as a function of the synset rank. The frequency rank is plotted for the different parts of speech: noun, verb, adjective and adverbs. The synsets (concepts) on the left have been chosen by the editors to describe a high number of sounds, (e.g: “ambiance” is used to describe 4,799 sounds of our test database). The concepts on the right describe a smaller number of sounds of our test database, (e.g: the concept “jaguar” as a “feline” is used to describe 19 sounds).

### 3.2.4 Low-level Features

Every audio sample to be classified is converted to 22.05 KHz mono and then passed through a noise gate in order to determine its beginning and its end. After a frame-by-frame analysis we extract features belonging to three different groups: a first group gathering spectral as well as temporal descriptors included in the MPEG-7 standard; a second one built on the acoustic spectrum division according to Bark bands outputs the mean and variance of relative energies for each band; and finally a third one, consisting of Mel-Frequency Cepstral Coefficients and their corresponding variances (see (Herrera et al., 2002) for details).

#### 3.2.4.1 Spectro-temporal descriptors

*Spectral Flatness* is the ratio between the geometrical mean and the arithmetical mean of the spectrum magnitude.

$$SFM = 10. \log \frac{(\prod_{k=1}^{N/2} S_p(e^{j\frac{2\pi k}{N}}))^{\frac{1}{N/2}}}{\frac{1}{N/2}\sum_{k=1}^{N/2} S_p(e^{j\frac{2\pi k}{N}})}$$

where $S_p(e^{j\frac{2\pi k}{N}})$ is the spectral power density calculated on the basis of an N-point Fast Fourier Transform.

*Spectral Centroid* is a concept adapted from psychoacoustics and music cognition. It measures the average frequency, weighted by amplitude, of a spectrum. The standard formula for the (average) spectral centroid of a sound is:

$$c = \frac{\sum_j c_j}{J}$$

where $c_j$ is the centroid for one spectral frame, and $J$ is the number of frames for the sound. The (individual) centroid of a spectral frame is defined as the average frequency weighted by amplitudes, divided by the sum of the amplitudes.

$$c_j = \frac{\sum f_j a_j}{\sum a_j}$$

*Strong Peak* intends to reveal whether the spectrum presents a very pronounced peak.

*Spectral Kurtosis* is the spectrum 4th order central moment and measures whether the data are peaked or flat relative to a normal (Gaussian) distribution.

$$kurtosis = \frac{\sum_{i=1}^{N}(Y_i - \overline{Y})^4}{(N-1)s^4}$$

where $\overline{Y}$ is the sample mean, $s$ is the sample standard deviation and $N$ is the number of observations.

*Zero-Crossing Rate* (ZCR), is defined as the number of time-domain zero-crossings within a defined region of signal, divided by the number of samples of that region.

*Spectrum Zero-Crossing Rate* (SCR) gives an idea of the spectral density of peaks by computing ZCR at a frame level over the spectrum whose mean has previously been subtracted.

*Skewness* is the 3rd order central moment, it gives indication about the shape of the spectrum in the sense that asymmetrical spectra tend to have large Skewness values.

$$skewness = \frac{\sum_{i=1}^{N}(Y_i - \overline{Y})^3}{(N-1)s^3}$$

where $\overline{Y}$ is the mean, $s$ is the standard deviation, and $N$ is the number of data points.

#### 3.2.4.2 Bark-band energy

Bark-band energy are the energies after dividing the spectrum into the 24 Bark bands, corresponding to the first 24 critical bands of hearing (Zwicker and Fastl, 1990). The published Bark band edges are given in Hertz as [0, 100, 200, 300, 400, 510, 630, 770, 920, 1080, 1270, 1480, 1720, 2000, 2320, 2700, 3150, 3700, 4400, 5300, 6400, 7700, 9500, 12000, 15500]. The published band centers in Hertz are [50, 150, 250, 350, 450, 570, 700, 840, 1000, 1170, 1370, 1600, 1850, 2150, 2500, 2900, 3400, 4000, 4800, 5800, 7000, 8500, 10500, 13500]. These bands are perception-related and have been chosen to enable systematic, instead of database-dependent, division of the spectrum. In order to cope with some low-frequency information, the two lowest bands have been split into two halves (Herrera et al., 2002).

#### 3.2.4.3 Mel-Frequency Cepstrum Coefficients

Mel-Frequency Cepstrum Coefficients (MFCCs) are widely used in speech recognition applications. They have been proved useful in music applications as well (Logan, 2000). They are calculated as follows:

1. Divide signal into frames.

2. For each frame, obtain the amplitude spectrum.

3. Take the logarithm.

4. Convert to Mel spectrum.

5. Take the discrete cosine transform (DCT).

Step 4 calculates the log amplitude spectrum on the so-called Mel scale. The Mel transformation is based on human perception experiments. Step 5 takes the DCT of the Mel spectra. For speech, this approximates principal components analysis (PCA) which decorrelates the components of the feature vectors. Logan (2000) proved that this decorrelation applies to music signals as well. As they can be used as a compact representation of the spectral envelope, their variance was also recorded in order to keep some time-varying information. 13 MFCCs are computed frame by frame, and their means and variances are used as descriptors.

### 3.2.5 Nearest-neighbor classifier

We use the k=1 nearest neighbor decision rule (1-NN)(Jain et al., 2000) for classification. The choice of a memory-based nearest neighbor classifier avoids the design and training of every possible class of sounds (in the order of several thousands). Another advantage of using a NN classifier is that it does not need to be redesigned nor trained whenever a new class of sounds is subsequently added to the system. The NN classifier needs a database of labeled instances and a similarity distance to compare them. An unknown sample will borrow the metadata associated with the most similar registered sound. The similarity measure of the system is a normalized Manhattan distance of the above enumerated features:

$$d\,(x,y) = \sum_{k=1}^{N} \frac{|x_k - y_k|}{(max_k - min_k)}$$

where $x$ and $y$ are the vectors of features, $N$ the dimensionality of the feature space, and $max_k$ and $min_k$ the maximum and minimum values of the $k$th feature.

In some of our experiments the standard deviation-normalized Euclidean distance did not perform well. Specially harmful was the normalization with standard deviation. Changing the normalization from the standard deviation to the difference between maximum and minimum boosted classification accuracy. For example the percussive instrument classification (see Section 3.3.3) raised from 64% to 82% correct identification. Changing the distance from Euclidean to Manhattan provided an extra 3% of improvement (85% correct identification).

### 3.2.6 Experimental results

The first experiment that we report consisted in finding a best-match for all the sounds in the database. Table 3.4 shows some result examples: the left column listers the original caption of the sound and the right column lists the caption of the nearest neighbor. The caption on the right would be assigned to the query sound in an automatic annotation system. As can be inferred from Table 3.4, it is not trivial to quantitatively evaluate the performance of the system. An intersection of the terms of the captions would not yield a reasonable evaluation metric. The WordNet based taxonomy can inform us that both “Trabant” and “Mini Cooper” are narrow terms for the concept “car, automobile”. Thus, the comparison of number of the common synsets on both query and nearest-neighbor could be used as a better evaluation. As was shown in more detail in (Cano et al., 2004d), the intersection of synsets between query and best-match is 1.5 in average, while 50% of the times the best-match did not share a single common synset (see Figure 3.7). The intersection of source descriptions can be zero for very similar sounding sounds. The closest-match for a “paper bag” turns out to be a “eating toast”. These sounds are semantically different but perceptually similar. This situation is very common, foley artists take advantage of the ambiguity and use “coconut half-shells” to create the sound of a “horse’s hoof-beats” (L.Mott, 1990). This ambiguity is a disadvantage when designing and assessing perceptual similarity distances. Figure 3.7 plots the number of correctly identified concepts (Perceptual Distance line) together with the perfect score (Concepts of SFX). The results of the system using a textual distance, a cosine distance over the caption terms are also displayed for comparison. The perfect score in the concept prediction scheme is achieving the same concept probability distribution as the labeled database. The textual distance provides, much better results and approximates reasonably well the database concept/sound distribution.

The second experiment consisted in the prediction of synsets, that is, how well a particular concept, say “cat miaow”, will retrieve “miaow” sounds. The methodology is as follows. For each synset, we retrieved the sounds that had been labeled with that particular synset. For each sound its nearest-neighbor was calculated. We finally computed how many best-matching sounds were also labeled with that synset. From the total of 3,028 synsets we restricted the experiment to the ones that had been used to label 10 or more sounds. There

| Query Sound Caption | Nearest-neighbor Caption |
|---|---|
| Mini Cooper Door Closes Interior Persp. | Trabant Car Door Close |
| Waterfall Medium Constant | Extremely Heavy Rain Storm Short Loop |
| M-domestic Cat- Harsh Meow | A1v:Solo violin (looped) |
| Auto Pull Up Shut Off Oldsmobile | Ferrari - Hard Take Off Away - Fast |
| Animal-dog-snarl-growl-bark-vicious | Dinosaur Monster Growl Roar |

Table 3.4: The classifier assigns the metadata of the sounds of the second column to the sounds of the first.

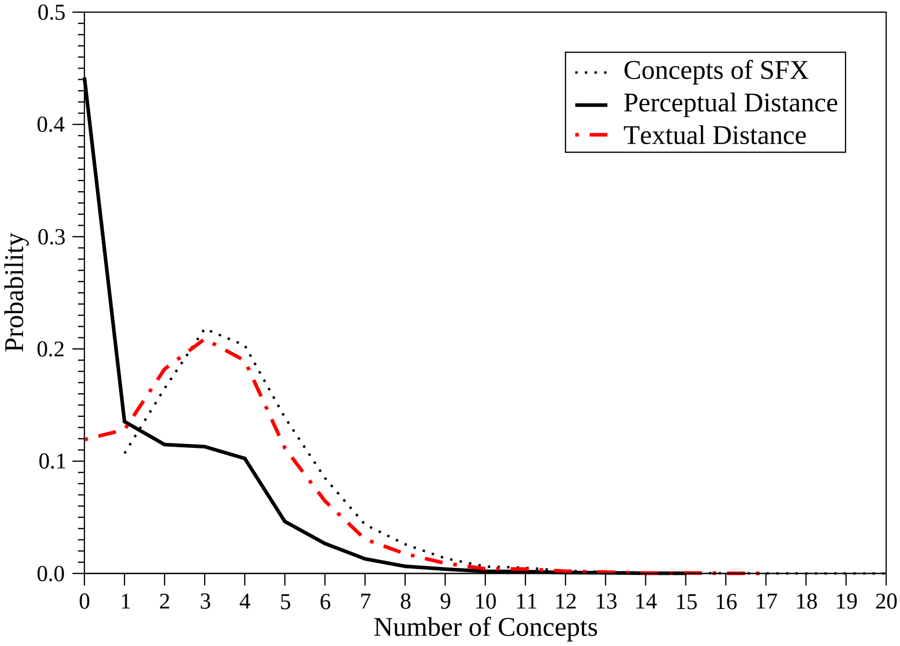


Figure 3.7: Probability distribution of correctly identified synsets. For each sound we count the intersection of concepts correctly predicted. The Concepts of SFX is the distribution of the number of concepts assigned to every sound effect (SFX). Concepts of SFX represents the perfect score. The perceptual distance prediction plot indicates the prediction accuracy using 1-NN and the perceptual similarity distance. The textual distance line indicates the prediction using the textual captions and a cosine distance and it is shown for comparison.

were 1,645 synsets tagging at least ten sounds. Figure 3.2.6 displays the results. The top figure displays how often a synset retrieved sounds whose best-matches were also labeled

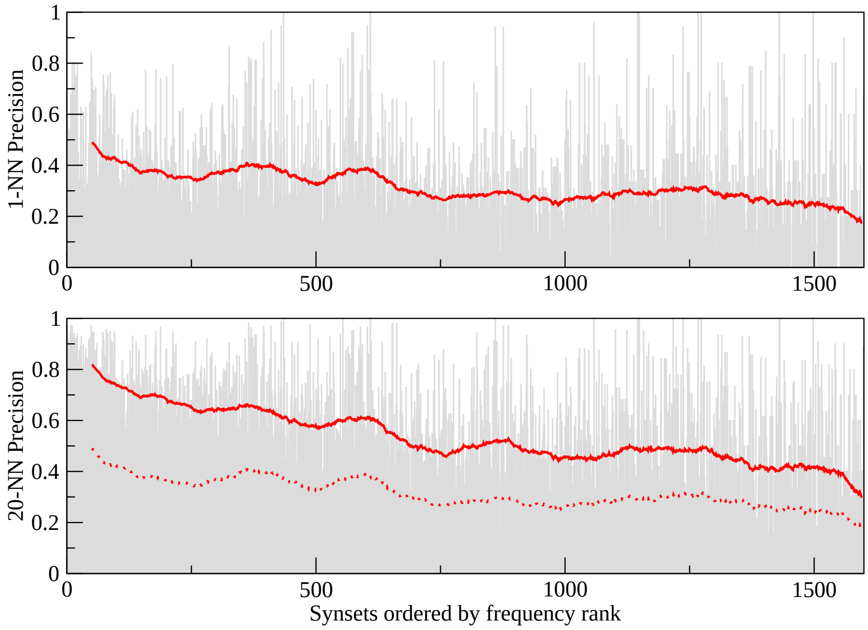


Figure 3.8: Synset precision using the 1-NN perceptual distance. The X axis corresponds to the synsets ordered by its frequency rank. The graph at the top shows the precision of the 1-NN. The bottom graph displays how often at least on the 20 best retrieved sounds was labeled with the synset. The plots have been smoothed with an average filter. The dotted line of the bottom graph reproduces the precision of the 1-NN of the top graph.

with that synset. The bottom figure, on the other hand, shows the probability that at least one of the best 20 retrieved sounds was labeled with the particular synset. The ordering of synsets on the x-axis corresponds to their frequency rank as displayed in Figure 3.6. It is interesting to see that there is not a strong correlation between the synset frequency and the precision. On a random guess one would expect some synsets predicted much better only because they are very frequent.

In a third experiment we tested the general approach in reduced domain classification regime mode: percussive instruments and harmonic instruments. The performance is comparable to that of state-of-the-art classifiers. We refer to (Herrera et al., 2003) for a review.

| | AF | AS | BF | BT | BA | BC | CE | DB | EC | FL | HO | OB | PI | SS | TT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AF | 7 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| AS | 0 | 18 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| BF | 0 | 0 | 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| BT | 0 | 0 | 0 | 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| BA | 0 | 0 | 0 | 0 | 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| BC | 0 | 0 | 0 | 1 | 0 | 10 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| CE | 0 | 1 | 0 | 0 | 0 | 1 | 74 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DB | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 72 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| EC | 0 | 1 | 1 | 0 | 0 | 2 | 0 | 0 | 5 | 1 | 0 | 1 | 0 | 2 | 1 |
| FL | 1 | 2 | 0 | 3 | 0 | 1 | 0 | 0 | 0 | 11 | 0 | 4 | 0 | 0 | 0 |
| HO | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 10 | 0 | 0 | 0 | 0 |
| OB | 0 | 1 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 1 | 7 | 0 | 0 | 1 |
| PI | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 87 | 0 | 0 |
| SS | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 24 | 0 |
| TT | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 7 |

Table 3.5: Harmonic instruments confusion matrix where AF:AltoFlute, AS:AltoSax, BF:BassFlute, BT:BassTrombone, BA:Bassoon, BC:BbClarinet, CE:Cello, DB:DoubleBass, EC:EbClarinet, FL:Flute, HO:Horn, OB:Oboe, PI:Piano, SS:SopranoSax, TT:TenorTrombone.

Table 3.5 depicts the confusion matrix of a 15 class harmonic instrument classification which corresponds to a 91% (261 audio files). In the 6 class percussive instrument classification an 85% accuracy was observed (955 audio files) using 10 fold validation (see Table 3.6).

In the last experiment we report here we tested the robustness of the NN classification framework to audio distortions. The sounds of the instruments appearing in Table 3.5 were transcoded and resampled into WAV PCM format and Ogg Vorbis format.[6] Ogg Vorbis is a lossy audio compression format. It would be advantageous for the classifier to be robust

[6] http://www.vorbis.com

| | SN | TO | HH | CR | KI | RI |
|---|---|---|---|---|---|---|
| SN | 150 | 1 | 2 | 2 | 1 | 20 |
| TO | 1 | 148 | 2 | 0 | 19 | 0 |
| HH | 5 | 7 | 153 | 0 | 1 | 4 |
| CR | 21 | 0 | 2 | 45 | 0 | 12 |
| KI | 1 | 17 | 0 | 0 | 182 | 0 |
| RI | 15 | 0 | 5 | 4 | 0 | 135 |

Table 3.6: Percussive instruments confusion matrix where SN:Snare, To:Tom, HH:Hihat, CR:Crash, KI:Kick, RI:Ride

| | Wav 44kHz | Ogg 44kHz | Ogg 11kHz |
|---|---|---|---|
| Wav 44kHz | 91.5% | 92.0% | 75.0% |
| Wav 22kHz | 86.4% | 85.6% | 82.0% |
| Wav 11kHz | 71.8% | 73.1% | 89.3% |
| Ogg 44kHz | 90.3% | 91.5% | 76.0% |
| Ogg 11kHz | 74.0% | 74.8% | 91.5% |

Table 3.7: Accuracy robustness to different distortions on the harmonic instruments classification. The columns indicate the reference audio quality and the rows the performance with the different distortions. Wav: PCM Microsoft WAV format, Ogg: Ogg Vorbis encoding, #kHz: Sampling rate

to content-preserving distortions introduced by this compression. The classification results of the distorted instrument sounds are depicted in Table 3.7. The percentages indicate the classification accuracy using different audio qualities. The columns are the audio qualities used as reference. The rows indicate the audio qualities used in the queries. The loss of accuracy, of up to 15% on some cases, suggest that better results can be obtained by designing more robust features to these "content-preserving" transformations.

### 3.2.7 Discussion

A major issue when building sound classification systems is the need of a taxonomy that organizes concepts and terms unambiguously. When classifying any possible sound, the taxonomy design is a complicated task. Yet, a taxonomy or classification scheme that encodes the common sense knowledge of the world is required. WordNet can be be used as a starting taxonomy. Classifiers are trained to learn certain concepts: "cars" , "laughs", "piano". Sound samples are gathered and are tagged with those concepts and finally a classifier is trained to learn those concepts. The number of concepts and its possible combinations in the real world makes this approach unfeasible, as one would need to train tens of thousands of classifiers and new ones would have to be trained for new concepts. We have presented an alternative approach that uses an unambiguously labeled big audio database. The classifier uses nearest-neighbor rule and a database of sounds with WordNet as taxonomy backbone. Resulting from the NN-based concept attachment, a list of possible sources is presented to the user: this sound could be a "paper bag" or "toast"+"eating". Information from text or images can additionally be used to disambiguate the possibilities.

We acknowledge that the use a single set of features and a single distance for all possible sound classes is rather primitive. However, and as Figure 3.2.6 indicates, there is room for improvement. The NN rule can be combined with other classifiers: If the system returns that a particular sound could be a violin pizzicato or a guitar, we can then retrieve pizzicato violin and guitar sounds of the same pitch and decide which is more likely. In order to perform this classification on real-time a lazy classifier could be chosen. Another example is "car approaches", where we can look for other "cars" and other "motor vehicle" "approaches" or "departs" to decide which is the right action. This same rationale applies to adjective type of modifiers, something can be described as "loud", "bright" or "fast". The concept "fast" means something different if we talk of "footsteps" or "typing".

## 3.3 Sound effects search engine

In this Section we present a SFX retrieval system that incorporates content-based audio techniques and semantic knowledge tools implemented on top of one of the biggest sound effects providers database. From Subsection 3.3.1 to 3.3.2 we describe the implemented enhancements of the system.

### 3.3.1 System overview

Sound FX providers rely on text descriptions to manage internally and sell their audio collections. Sound engineers search for sounds by matching a query against the descriptive keywords that a librarian has attached to each sound. There are several professional providers that offer SFX using keyword-matching as well as navigating through categories that organize the sounds in classes such as Animal, Cars, Human and so on (e.g.: www.sound-effects-library.com, www.sounddogs.com, www.sonomic.com). Web search engines such as www.altavista.com or www.singingfish.com offer audio search using standard text-based web retrieval indexing the words that appear near audio content in the HTML page.

**Limitations of text-based approach**

Discussion on the inadequacy of using text descriptors to describe sound is frequent in the literature (L.Mott, 1990). It is pointed out that sounds are too difficult to describe with words. Perceptual descriptions are too subjective and may vary for different people. Source descriptions convey sometimes more descriptive power and are objective. However, sound may have been synthesized and have no clear origin. Other cons on current text-based approaches include:

- Library construction, that is, tagging of sounds with textual description, is a labour-consuming, error-prone task and yet the number of sound samples is constantly increasing.
- It is difficult for a librarian to add keywords that would match the ways users may eventually query a sound, e.g.: see Figure 3.1 for possible keywords to label a "golf drive".
- The sounds without caption are invisible to the users.
- Big corpuses may be labeled by different librarians that follow somewhat different conventions.
- The vagueness of the query specification, normally one or two words, together with the ambiguity and informality of natural languages affects the quality of the search: Some relevant sounds are not retrieved and some irrelevant ones are presented to the user.
- Sound effect management systems allow browsing for sounds in manually generated categories. The design and maintenance of category trees is complicated. It is very time consuming for a librarian to place a sound in the corresponding categories. Finally, It is difficult for users to navigate through somebody else's hierarchy.

In order to overcome the above shortcomings, solutions have been proposed to manage media assets from a content-based audio perspective, both from the academia and the industry (FindSounds, 2003; Wold et al., 1996). However, even though text-search has some shortcomings, content-based functionality should only complement and not substitute the text search approach for several reasons: first, because the production systems work, second, because there is a great deal of legacy meta-data and new sound effects are released by the major vendors with captions, third, because text-retrieval is generally faster than content-based, and finally because users are familiar with using words, i.e. high-level descriptions, to search for media assets.

In this Section, we present how to construct a SFX management system that incorporates content-based audio techniques as well as knowledge based tools built on top of one of the biggest sound effects providers database. The new sounds added into the system will be labeled automatically by the automatic annotator described in 3.2 to be validated by the librarian.

In the implemented system we aim at combining the best of two worlds to offer tools for the users to refine and explore a huge collection of audio. Similar work on integrating perceptual and semantic information in a more general multimedia framework is MediaNet (Benitez et al., 2000). The system we present is specialized for SFX. The current prototype uses a collection of sounds from a major on-line sound effects provider: www.sound-effects-library.com. The sounds where described with captions that mapped to concepts of the WordNet ontology as described in Section 3.1.

#### 3.3.1.1 Functional blocks

The system has been designed to ease the use of different tools to interact with the audio collection and with speed as a major design issue. On top of these premises we have implemented the blocks of Fig. 3.9. The sound analysis, audio retrieval and metadata generation blocks were described in Section 3.2. The text retrieval, text processor and knowledge manager blocks are described in Section 3.3.2.

#### 3.3.1.2 System architecture

The audio processing engines is implemented in C++. The ontology management and integration of different parts is done with Perl and a standard relational database management system. The functionality is available via a web interface and exported via SOAP (http://www.w3.org/TR/soap). The SOAP interface provides some functionalities—such as interaction with special applications, e.g.: Sound editors and annotators—which are not available via the web interface. See Figure 3.10 for a diagram of the architecture.

Content-based audio tools ease the work of the librarian and enhance the search possibilities for the user. It simplifies the labeling of new sounds because many keywords are

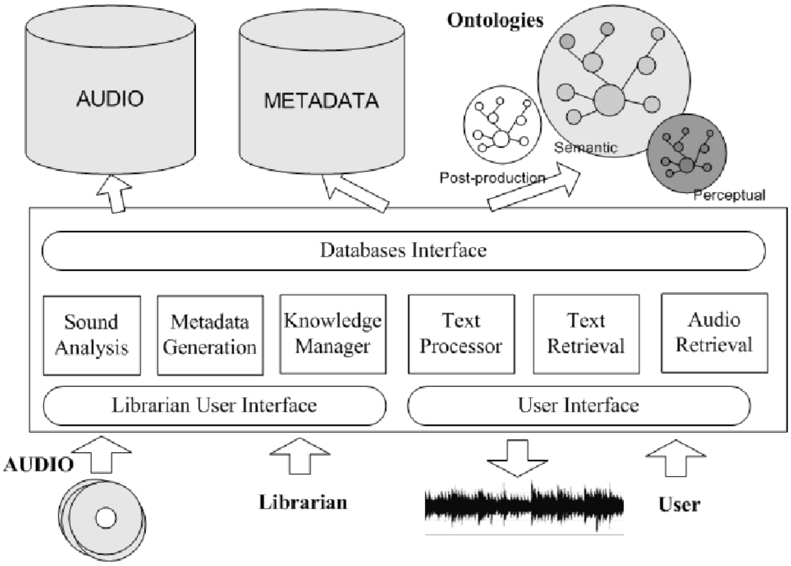


Figure 3.9: Functional Block Diagram of the System.

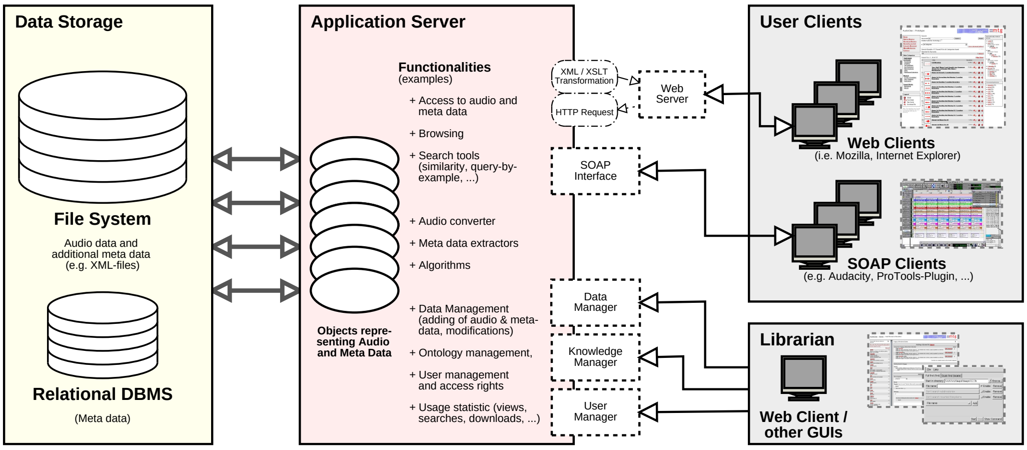

Data Storage
File System
Audio data and additional meta data (e.g. XML-files)
Relational DBMS
(Meta data)
Application Server
Functionalities
(examples)
+ Access to audio and meta data
+ Browsing
+ Search tools (similarity, query-by-example, ...)
+ Audio converter
+ Meta data extractors
+ Algorithms
+ Data Management (adding of audio & meta-data, modifications)
+ Ontology management,
+ User management and access rights
+ Usage statistic (views, searches, downloads, ...)
Objects representing Audio and Meta Data
XML / XSLT Transformation
HTTP Request
Web Server
SOAP Interface
Data Manager
Knowledge Manager
User Manager
User Clients
Web Clients
(i.e. Mozilla, Internet Explorer)
SOAP Clients
(e.g. Audacity, ProTools-Plugin, ...)
Librarian
Web Client / other GUIs

Figure 3.10: System Architecture

automatically presented to the librarian.

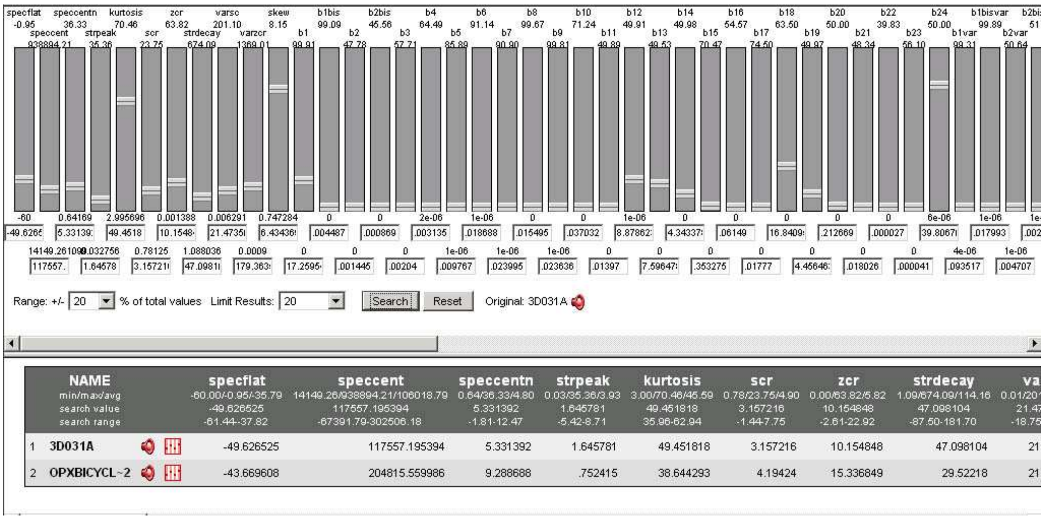
To achieve it, the new sound is compared to the collection with Nearest Neighbor search and the text associated with the similar matches is presented to the librarian (see Section 3.2). The sound analysis module (see Figure 3.9), besides extracting low-level sound descriptors used for the similarity search (see Subsection 3.2.4 for details), generates mid-level searchable descriptors as those detailed in Subsection 3.3.1.4 (crescendo, noisy, iterative, percussive, and so on). Content-based tools offer the user functionalities such as:

Virtual Foley Mode: Find perceptually similar sounds. A user may be interested in a glass crash sound. If none of the retrieved sounds suits him, he can still browse the collection for similar sounds even if produced by different sources, even if unlabeled.

Clustering of sounds: Typically a query like "whoosh" may retrieve several hundred results. These results are clustered and only one representative of each class is displayed to the user. The user can then refine the search more easily.

Morphological Descriptors: Another option when the list of results is too large to listen to is filtering the results using morphological descriptors (see Section 3.3.1.4).

Query by example: The user can provide an example sound or utter himself one as a query to the system, possibly restricting the search to a semantic subspace, such as "mammals".



-60 -49.6265 | 14149.26109 117557. | 0.64169 5.33139 | 0.032756 1.64578 | 2.995696 49.4518 | 0.78125 3.15721 | 0.001388 10.1548 | 1.088036 47.0981 | 0.006291 21.4735 | 0.0009 179.363 | 0.747284 6.43436 | 0 17.2595 | 0 .004487 | 0 .001445 | 0 .000869 | 0 .00204 | 2e-06 .003135 | 1e-06 .009767 | 1e-06 .018688 | 1e-06 .023995 | 0 .015495 | 1e-06 .023636 | 0 .037032 | 0 .01397 | 1e-06 8.87862 | 0 7.59647 | 0 4.34337 | 0 .353275 | 0 .06149 | 0 .01777 | 0 16.8409 | 0 4.45646 | 0 .212669 | 0 .018026 | 0 .000027 | 0 .000041 | 6e-06 39.8067 | 4e-06 .093517 | 1e-06 .017993 | 1e-06 .004707 | 1e .002

Range: +/- 20 % of total values Limit Results: 20 Search Reset Original: 3D031A

| | NAME | specflat | speccent | speccentn | strpeak | kurtosis | scr | zcr | strdecay | va |
|---|---|---|---|---|---|---|---|---|---|---|
| | min/max/avg | -60.00/-0.95/-35.79 | 14149.26/938894.21/106018.79 | 0.64/36.33/4.80 | 0.03/35.36/3.93 | 3.00/70.46/45.59 | 0.78/23.75/4.90 | 0.00/63.82/5.82 | 1.09/674.09/114.16 | 0.01/20 |
| | search value | -49.626525 | 117557.195394 | 5.331392 | 1.645781 | 49.451818 | 3.157216 | 10.154848 | 47.098104 | 21.47 |
| | search range | -61.44-37.82 | -67391.79-302506.18 | -1.81-12.47 | -5.42-8.71 | 35.96-62.94 | -1.44-7.75 | -2.61-22.92 | -87.50-181.70 | -18.75 |
| 1 | 3D031A | -49.626525 | 117557.195394 | 5.331392 | 1.645781 | 49.451818 | 3.157216 | 10.154848 | 47.098104 | 21 |
| 2 | OPXBICYCL~2 | -43.669608 | 204815.559986 | 9.288688 | .752415 | 38.644293 | 4.19424 | 15.336849 | 29.52218 | 21 |

Figure 3.11: Slider web interface

#### 3.3.1.3 Similarity Distance

The similarity measure is a normalized Manhattan distance of features belonging to three different groups: a first group gathering spectral as well as temporal descriptors included in the MPEG-7 standard; a second one built on Bark Bands perceptual division of the acoustic spectrum and which outputs the mean and variance of relative energies for each band; and, finally a third one, composed of Mel-Frequency Cepstral Coefficients and their corresponding variances (see Subsection 3.2.4 for details):

$$d\left(x,y\right)=\sum_{k=1}^{N}\frac{|x_k-y_k|}{(max_k-min_k)}$$

where $x$ and $y$ are the vectors of features, $N$ the dimensionality of the feature space, and $max_k$ and $min_k$ the maximum and minimum values of the $k$th feature.

The similarity measure is used for metadata generation: a sound sample will be labeled with the descriptions from the similar sounding examples of the annotated database. This type of classification is known as one-nearest neighbor decision rule (1-NN)(Jain et al., 2000). The choice of a memory-based nearest neighbor classifier avoids the design and training of every possible class of sound which is of the order of several thousands. Besides, it does not need redesign or training whenever a new class of sounds is added to the system. The NN classifier needs a database of labeled instances and a similarity distance to compare them. An unknown sample will borrow the metadata associated with the most similar registered sample. We refer to Section 3.2 for further details on sound annotation.

The similarity measure is also used for the query-by-example and to browse through "perceptually" generated hyperlinks (see Subsubsection 3.3.3).

#### 3.3.1.4 Morphological Sound Description

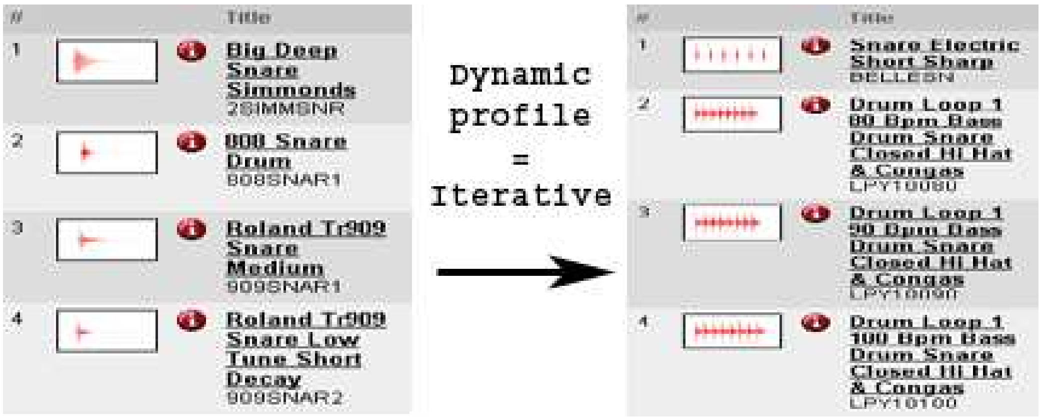
The morphological sounds descriptor module extracts a set of descriptors that focused on intrinsic perceptual qualities of sound based on Schaeffer's research on *sound objects* (Schaeffer, 1966). The rationale behind Schaeffer's work is that all possible sounds can be described in terms of its qualities—e.g.: pitchness, iterativeness, dynamic profile and so on—regardless of their source of creation. The extractor of morphological descriptors (Ricard and Herrera, 2004) currently generates the following metadata automatically:

Pitchness: (Organization within the spectral dimensions) Pitch, Complex and Noisy.

Dynamic Profile: (Intensity description) Unvarying, Crescendo, Decrescendo, Delta, Impulsive, Iterative, Other.

Pitchness Profile: (Temporal evolution of the internal spectral components) Varying and Unvarying

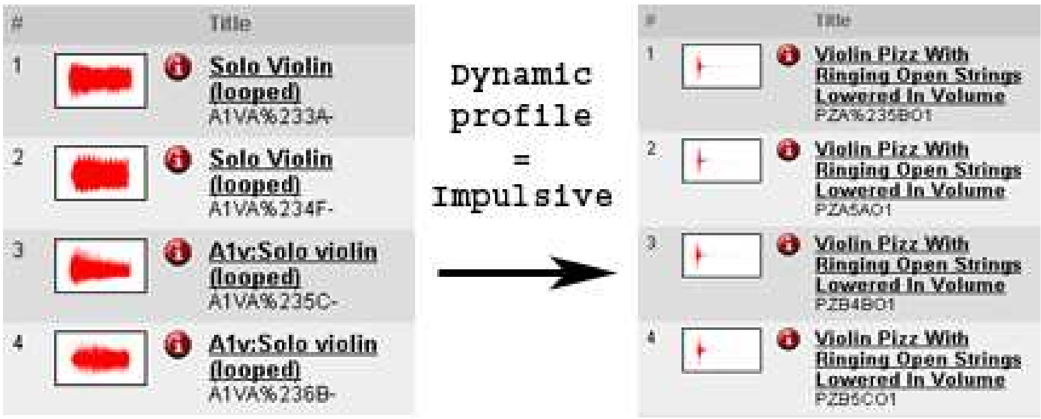
Pitch Profile: (Temporal evolution of the global spectrum) Undefined, Unvarying, Varying Continuous, Varying Stepped.



| # | Title |
|---|---|
| 1 | Big Deep Snare Simmonds<br>2SIMMSNR |
| 2 | 808 Snare Drum<br>808SNAR1 |
| 3 | Roland Tr909 Snare Medium<br>909SNAR1 |
| 4 | Roland Tr909 Snare Low Tune Short Decay<br>909SNAR2 |

| # | Title |
|---|---|
| 1 | Snare Electric Short Sharp<br>BELLESN |
| 2 | Drum Loop 1 80 Bpm Bass Drum Snare Closed Hi Hat & Congas<br>LPY10080 |
| 3 | Drum Loop 1 90 Bpm Bass Drum Snare Closed Hi Hat & Congas<br>LPY10090 |
| 4 | Drum Loop 1 100 Bpm Bass Drum Snare Closed Hi Hat & Congas<br>LPY10100 |

Figure 3.12: Morphological descriptor filtering. The iterative dynamic profile allows to discriminate between snare samples and loops

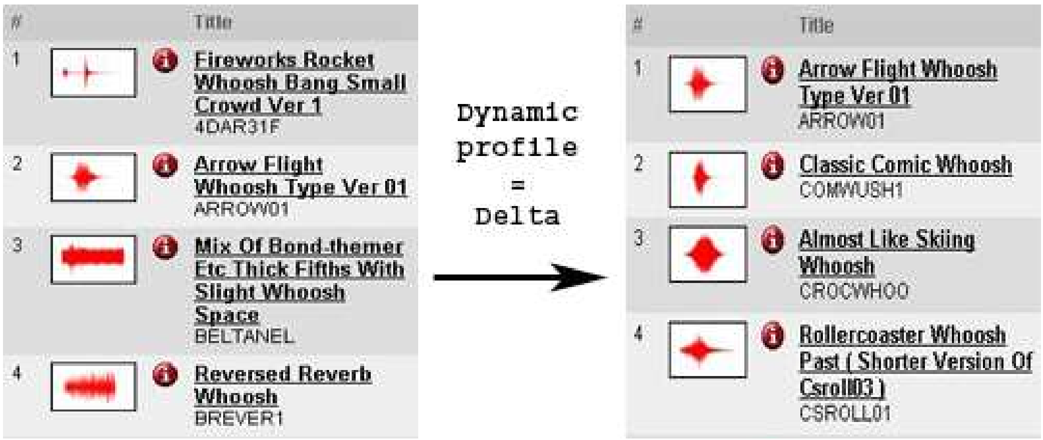


| # | Title |
|---|---|
| 1 | Solo Violin (looped)<br>A1VA%233A- |
| 2 | Solo Violin (looped)<br>A1VA%234F- |
| 3 | A1v:Solo violin (looped)<br>A1VA%235C- |
| 4 | A1v:Solo violin (looped)<br>A1VA%236B- |

| # | Title |
|---|---|
| 1 | Violin Pizz With Ringing Open Strings Lowered In Volume<br>PZA%235BO1 |
| 2 | Violin Pizz With Ringing Open Strings Lowered In Volume<br>PZA5AO1 |
| 3 | Violin Pizz With Ringing Open Strings Lowered In Volume<br>PZB4BO1 |
| 4 | Violin Pizz With Ringing Open Strings Lowered In Volume<br>PZB5CO1 |

Figure 3.13: Morphological description filtering. The impulsive dynamic profile allows to discriminate violin pizzicati.

These descriptors can be used to retrieve abstract sounds as well as refine other types of searches. Besides applying to all types of sounds, the use of an automatic extractor avoids expensive human labeling while it assures consistency. For details on the construction and



| # | Title |
|---|---|
| 1 | Fireworks Rocket Whoosh Bang Small Crowd Ver 1<br>4DAR31F |
| 2 | Arrow Flight Whoosh Type Ver 01<br>ARROW01 |
| 3 | Mix Of Bond-themer Etc Thick Fifths With Slight Whoosh Space<br>BELTANEL |
| 4 | Reversed Reverb Whoosh<br>BREVER1 |

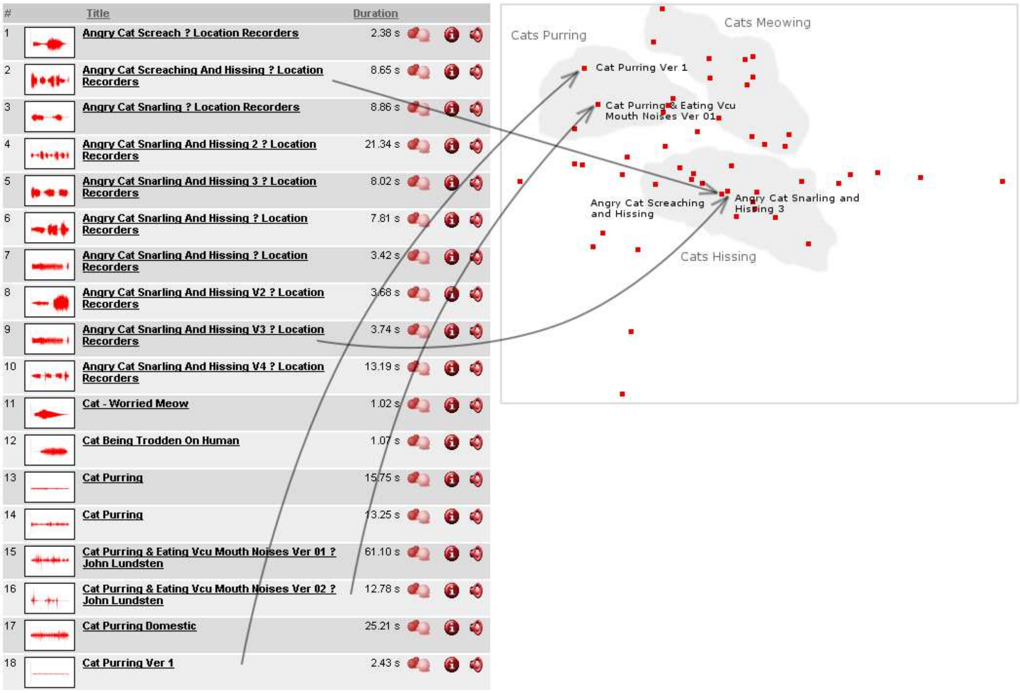
| # | Title |
|---|---|
| 1 | Arrow Flight Whoosh Type Ver 01<br>ARROW01 |
| 2 | Classic Comic Whoosh<br>COMWUSH1 |
| 3 | Almost Like Skiing Whoosh<br>CROCWHOO |
| 4 | Rollercoaster Whoosh Past ( Shorter Version Of Csroll03 )<br>CSROLL01 |

Figure 3.14: Morphological description filtering. The delta dynamic profile example.

usability evaluation of the morphological sound description we refer to (Ricard and Herrera, 2004).

#### 3.3.1.5 Clustering and Visualization Tools

Usually, systems for content-based retrieval of similar sounds output a list of similar sounds ordered by increasing similarity distance. The list of retrieved sounds can rapidly grow and the search of the appropriate sound becomes tedious. There is a need for a user-friendly type of interface for browsing through similar sounds. One possibility for avoiding having to go over, say 400 gunshots, is via clustering sounds into perceptually meaningful subsets, so that the user can choose what perceptual category of sound he or she wishes to explore. We used a hierarchical tree clustering with average linkage algorithm and the above mentioned similarity distance (Jain et al., 2000). Another possibility of interaction with the sounds is using visualization techniques, specifically Multidimensional scaling (MDS) (Shepard, 1962; Kruskal, 1964), self-organizing maps (SOM) (Kohonen, 1997; Honkela et al., 1997; Pampalk et al., 2002b) or FastMap (Faloutsos and Lin, 1995; Cano et al., 2002b), to map the audio samples into points of an Euclidean space. Figure 3.15 displays a mapping of the audio samples to a 2D space. In the example it is possible to distinguish different classes of cat sounds, e.g.: “purring”, “hissing” and “miaow” sounds.



| # | Title | Duration |
|---|---|---|
| 1 | Angry Cat Screach ? Location Recorders | 2.38 s |
| 2 | Angry Cat Screaching And Hissing ? Location Recorders | 8.65 s |
| 3 | Angry Cat Snarling ? Location Recorders | 8.86 s |
| 4 | Angry Cat Snarling And Hissing 2 ? Location Recorders | 21.34 s |
| 5 | Angry Cat Snarling And Hissing 3 ? Location Recorders | 8.02 s |
| 6 | Angry Cat Snarling And Hissing ? Location Recorders | 7.81 s |
| 7 | Angry Cat Snarling And Hissing ? Location Recorders | 3.42 s |
| 8 | Angry Cat Snarling And Hissing V2 ? Location Recorders | 3.68 s |
| 9 | Angry Cat Snarling And Hissing V3 ? Location Recorders | 3.74 s |
| 10 | Angry Cat Snarling And Hissing V4 ? Location Recorders | 13.19 s |
| 11 | Cat - Worried Meow | 1.02 s |
| 12 | Cat Being Trodden On Human | 1.07 s |
| 13 | Cat Purring | 15.75 s |
| 14 | Cat Purring | 13.25 s |
| 15 | Cat Purring & Eating Vcu Mouth Noises Ver 01 ? John Lundsten | 61.10 s |
| 16 | Cat Purring & Eating Vcu Mouth Noises Ver 02 ? John Lundsten | 12.78 s |
| 17 | Cat Purring Domestic | 25.21 s |
| 18 | Cat Purring Ver 1 | 2.43 s |

Cats Purring
Cats Meowing
Cat Purring Ver 1
Cat Purring & Eating Vcu Mouth Noises Ver 01
Angry Cat Screaching and Hissing
Angry Cat Snarling and Hissing 3
Cats Hissing

Figure 3.15: FastMap visualization screenshot. The points of the 2D map refer to different audio samples. The distances on the euclidean space try to preserve distances in the hyper-dimensional perceptual space defined by the similarity distance of subsection 3.3.1.3

### 3.3.2 WordNet-based knowledge manager

The use of a WordNet-based taxonomy management together with Natural Language Processing tools enhances text-search engines used in sound effects retrieval systems by going from keyword to concept-based search. At the same time it eases the librarian task when describing sounds and it simplifies the management of the categories. Some of the benefits of the knowledge management system are listed below:

- Higher control on the precision and recall of the results using WordNet concepts. The query "bike" returns both "bicycle" and "motorcycle" sounds and the user is given the option to refine the search.

- Common sense "intelligent" navigation: The concept relations encoded in WordNet can be used to propose related terms. It is generally accepted that recognition is stronger than recall and a user may not know how the librarian tagged a sound. If a user asks for the sound of a “Jaguar”, the system presents results of Jaguar as an automobile as well as a feline. Moreover, once the right concepts is specified, say jaguar meaning the feline, it proposes also sounds by other big cats, such as lions or tigers.

- Proposal of higher level related term not included in the lexical network. WordNet does not have all possible relations. For instance, “footsteps in mud”, “tractor”, “cow bells” and “hens” may seem related in our minds when we think of farm sounds but do not have direct links within WordNet. It is possible to recover this type of relations because there are many sounds that have been labeled with the concept “farm”. The analysis of co-occurrence of synsets, “tractor” and “farm” co-occur significantly, allows the system to infer related terms (Banerjee and Pedersen, 2003).

- Building on WordNet, it is possible to construct a lemmatizer which can convert, say “bikes” becomes “bike”, an inflecter that allows to expand it to “bike, bikes and biking”, and a name entity recognition module, that is able to identify “Grand piano” as a specific type of piano.

- Module for the phonetic matching, e.g: “whoooassh” retrieves “whoosh”. Phonetic matching is used in information retrieval to account for the typo errors in a query and thus aims at reducing the frustration of a user. In sound effects retrieval, it is even more important since it is common practice to describe sounds as they sound if one reads them. WordNet has a very complete onomatopoeia ontology.

### 3.3.3 Similarity evaluation

We have used 54,799 sounds from the Sound-Effects-Library (http://www.sound-effects-library.com) for the experiments. These sounds have been unambiguously tagged with concepts of an enhanced WordNet. Thus a piano sound with the following caption:“Concert Grand Piano - piano” may have the following synsets (the numbers on the left are the unique WordNet synset identifiers):

- 02974665%n concert grand, concert piano – (a grand piano suitable for concert performances)
- 04729552%n piano, pianissimo – ((music) low loudness)

#### 3.3.3.1 Experimental setup

The evaluation of similarity distances is a tricky subject. Perceptual listening tests are expensive. Another possibility is to evaluate the goodness of the similarity measure examining the performance in a 1-Nearest Neighbor (NN) classification task. As we will see in Section 3.3.3.3, the overlap between semantic and perceptual taxonomies complicates the evaluation, e.g. a "cat miaow", a "a cello" and an "old door opening" may sound very similar and yet they were originated by very different sources. There are cases where there is a "decent" relationship between semantic and perceptual taxonomies. For instance, in musical instruments, the semantic taxonomy more or less follows an acoustic classification scheme, basically due to the physical construction, and so instruments are wind (wood and brass), string (plucked or bowed) and so on (Lakatos, 2000; Herrera et al., 2003). Another example where there is a decent mapping between semantic description and it perceptual description is the case of "onomatopoeia".

We have experimented three ways of assessing the perceptual similarity between sounds:

- Perceptual listening experiments
- Accuracy on classification or metadata generation using a similarity measure
- Consistency on the ranking and robustness to perceptual preserving distortions such as resampling, transcoding (converting to MP3 format at different compression rates and back).

We discuss with more detail the first two items.

#### 3.3.3.2 Perceptual listening tests

In order to test the relevance of our similarity measures, we asked users of our system to give a personal perceptual evaluation on the retrieval of sounds by similarity. This experiment was accomplished on 20 users who chose 41 different queries, and produced 568 evaluations on the relevance of the similar sound retrieved. During the evaluation, the users were presented with a grading scale from 1—not similar at all—to 5—closely similar. The average grade was 2.6 which slightly above more or less similar. We have at our disposal the semantic concepts associated with the 54,799 sounds used in the experiment. It turned out that the semantic class of a sound is crucial in the user's sensation of similarity. Although more experiments and thorough analysis should be undertaken, our informal experiment seems to hint that the users gave better grades to retrieved sounds that are from the same semantic class as the query sound (40% of the best graded sounds belonged to the same semantic class). In the prototype, in addition to the purely content-based retrieval tools, the use of the knowledge-based tools allows searches for similar sounds inside a specific semantic family.

#### 3.3.3.3 Metadata annotation performance

We have done two experiments in metadata annotation. The first experiment consisted in finding a best-match for all the sounds in the database. This experiments is described in detail in Section 3.2.

As we discuss in Section 3.2, the number of concepts (synsets) that the sound in the database and their best match have in common was bigger than one—at least one synset—half of the time. Yet we must remind the reader that there are many occasions when this evaluation metric is not appropriate for similarity assessment: The intersection of source descriptions can be zero for very similar sounding sounds. The closest-match for a "waterfall medium constant" turns out to be a "extremely heavy rain". These sounds are semantically different but perceptually equivalent. The ambiguity is a disadvantage when designing and assessing perceptual similarity distances because it makes it difficult to quantitatively evaluate the performance of a similarity measure.

In a second experiment we have tested the general approach in reduced domain classification regime mode: percussive instruments, harmonic instruments and we achieve state-of-the art results. The assumption when we assess the similarity in such a reduced domain is that there is a parallelism between semantic and perceptual taxonomies in musical instruments. The psychoacoustic studies of Lakatos (2000) revealed groupings based on the similarities in the physical structure of instruments. We have therefore evaluated the similarity with classification on the musical instruments space, a subspace of the universe of sounds.

As reported in 3.2, in the 6 class percussive instrument classification we achieve a 85% recognition (955 audio files) using 10 fold validation. The results for a 8 class classification of harmonic instruments is a 77.3% (261 audio files).

### 3.3.4 Summary

After introducing some difficulties inherent in interacting with sound effect repositories— both for the librarian who designs such content repositories and for potential users who access this content— We presented several technologies that enhance and fit smoothly into professional sound effects providers working processes. Several low-level content-based audio tools have been integrated providing possibilities of accessing sounds which are unrelated from the text caption but sound the same—even if they are unlabeled. The automatic annotator of Section 3.2 is used as an aid for the librarian, e.g. the librarian is presented with the list of relevant concepts generated by the computer and validates or rejects the hypothesis.

The system can be accessed and evaluated at http://audioclas.iua.upf.es

In the next section, an intelligent authoring system is implemented on top of the search engine. The authoring system will allow for seamless creation of ambiances using both knowledge and perceptual tools.

## 3.4 Intelligent authoring

This section describes an intelligent authoring system for ambiance generation. Ambiances are background recordings used in audiovisual productions to make listeners feel they are in places like a pub or a farm. Accessing to commercially available atmosphere libraries is a convenient alternative to sending teams to record ambiances yet they limit the creation in different ways. First, they are already mixed, which reduces the flexibility to add, remove individual sounds or change its panning. Secondly, the number of ambient libraries is limited. We propose a semi-automatic system for ambiance generation. The system creates ambiances on demand given text queries by fetching relevant sounds from a large sound effect database and importing them into a sequencer multi track project. Ambiances of diverse nature can be created easily. Several controls are provided to the users to refine the type of samples and the sound arrangement.

### 3.4.1 Introduction

Traditionally, from the film production process point of view, sound is broken into a series of layers: dialog, music and sound effects L.Mott (1990). SFX can be broken further into *hard SFX* (car doors opening and closing, and other foreground sound material) and *Foley* (sound made by humans, e.g: footsteps) on the one hand, and *ambiances* on the other hand. Ambiances—also known as atmospheres—are the background recordings which identify scenes aurally. They make the listener really feel like they are in places like an airport, a church, a subway station, or the jungle. Ambiances have two components: The *ambient loop*, which is a long, streaming, stereo recording, and *specifics* or *stingers*, which are separate, short elements (e.g: dog barks,car horns, etc) that trigger randomly to break up repetition Peck (2001).

Sound engineers need to access sound libraries for their video and film productions, multimedia and audio-visual presentations, web sites, computer games and music. Access to libraries is a convenient alternative to sending a team to record a particular ambiances (consider for instance “a Rain forest” or “a Vesuvian eruption”. However, the approach has some drawbacks:

1. Accessing the right ambiances is not easy due to the information retrieval models, currently based mainly on keyword search Cano et al. (2004e).

2. The number of libraries is large but limited. Everybody has access to the same content although sound designers can use them as starting point and make them unrecognizable and unique.

3. Ambiances offered by SFX library providers are already mixed. There may be a particular SFX in the mix that the sound engineer does not want in that position or may be does not want at all. Because the ambiances are already mixed, it is a hassle for the sound engineer to tailor the ambiance.

In this context, we present a system for the automatic generation of ambiances. In short, the system works as follows: the user specifies his need with a standard textual query, e.g: "farm ambiance". The ambiance is created on-the-fly combining SFX related to the query. For example,the query "farm ambiance" may return "chicken", "tractors", "footsteps on mud" or "cowbells" sounds. A subset of retrieved sounds is randomly chosen. After listening to the ambiance, the user may decide to refine the query—e.g: to remove the "cowbells" and add more "chickens"—, ask another random ambiance—with a "shuffle-type" option—or decide that the ambiance is good enough to start working with. The system outputs the individual SFX samples in a multi track project.

The intended goals of the approach can be summarized as follows:

Enhance creativity: Sound engineers have access to a huge ever-changing variety of ambiances instead of a fix set of ambiances. The combination of individual SFX provides a substantially larger number of ambiances.

Enhance productivity: Engineers can have several possible sonifications in a short time.

Enhance flexibility: Having different SFX of the ambiance separately in a Multi Track gives more flexibility to the ambiance specification process, some sounds–a bird singing in a forest ambiance–can be removed or their location in the time line changed. It also allows for spatialization using 5.1.

Enhance quality: With a very low overhead—basically clicking on a “shuffle” button and adjusting some sliders, sound engineers can obtain several ambiance templates. Hence, the production cycle reduces. The producers can give their feedback faster and their opinions be incorporated earlier in the production improving the overall quality.

### 3.4.2 System Description

The system is based on a concept-based SFX search engine developed within the AudioClas project (www.audioclas.org). The objectives of the project were to go beyond current professional SFX provider information retrieval model, based on keyword-matching, mainly through two approaches Cano et al. (2004c):

Semantically-enhanced management of SFX using a general ontology, WordNet Miller (1995).[7]

Content-based audio technologies which allow automatic generation of perceptual meta data (such as prominent pitch, dynamics, beat, noisiness).

These two approaches are the building blocks of the semi-automatic ambiance generation. Current prototype uses 80.000 sounds from a major on-line SFX provider.[8] Sounds come with textual descriptions which have been disambiguated with the augmented WordNet ontology Cano et al. (2004e). As we detailed in Section 3.1 WordNet is a lexical database that, unlike standard dictionaries which index terms alphabetically, indexes concepts with relations among them.

There are two main functional blocks in the system. The first one retrieves the relevant sounds of the SFX Database and a second one organizes the sounds in a multi track according to some heuristic rules (see figure 3.16).

### 3.4.3 Sound selection and retrieval

The first step has been mining ambiance sounds to learn the type of sources used. We use a database of SFX that has been labeled with concepts rather than with words (see Cano

[7] http://www.cogsci.princeton.edu/~wn/

[8] http://www.sound-effects-library.com

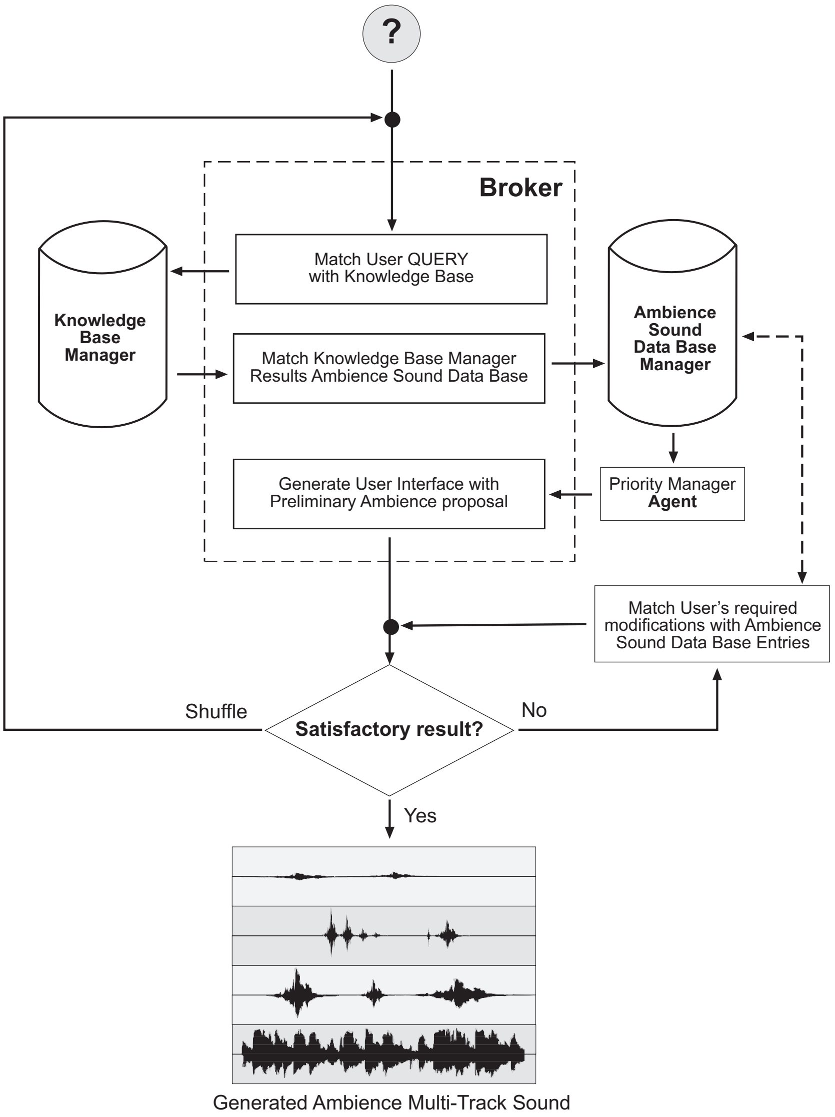

?
Broker
Knowledge Base Manager
Match User QUERY with Knowledge Base
Match Knowledge Base Manager Results Ambience Sound Data Base
Ambience Sound Data Base Manager
Priority Manager Agent
Generate User Interface with Preliminary Ambience proposal
Match User's required modifications with Ambience Sound Data Base Entries
Shuffle
Satisfactory result?
No
Yes
Generated Ambience Multi-Track Sound

Figure 3.16: Flow diagram of the ambiance generation system.

et al. (2004e) for details). We are therefore able to study the co-occurrence of concepts in sounds. For example, the ambiance “Farm Ambiance Of Rooster And Hen With Wagtail In Background” has been converted to:

```
01466271%n hen, biddy -- (adult female chicken)
01206115%n wagtail -- (Old World bird having a very
  long tail that jerks up and down as it walks)
 02893950%n farm -- (workplace consisting of farm
  buildings and cultivated land as a unit)
```

By mining this information we learn that farm is related to the concept hen and the concept wagtail. Moreover, there are relations encoded in WordNet, which knows that hen and chicken are related. Whenever a user asks for farm sounds we can retrieve a set of sounds where farm appears. Besides we can also search for the sounds of the related concepts, such as chicken. A random subset of the relevant sounds is forwarded to the subsequent block, the sound sequencing.

### 3.4.4 Sound sequencing

Besides the definition and selection of the suitable SFX, a significant part of the work of the sound designer is setting up parameters and time lines in a multi track project, such as volumes or panoramic envelopes. This section details some of the rules used to mix all fetched tracks and compose the synthetic atmosphere. The SFX retrieval module returns mono and dry (no effect has been applied) tracks. Whenever available, the module differentiates between two types of tracks: long ambient tracks and several short isolated effects. One long track is selected to serve as a ambient loop on which the short sounds, or specifics, are added. With such picture of the workspace we hint some rules on how to place the tracks in the mix, how to adjust channel controls (gain, panning and equalization), and which effects (echo, reverb) can be applied to each track.

The systems automatically distributes the tracks along the mix, placing first the ambient loop and inserting sequentially the specifics, with a probabilistic criterion. This probabilistic criterion is based on the inverse of a frame-based energy computation. This means that the more energetic regions of the mix will have less probability to receive the following effect track. This process is depicted in figure 3.17.

It is a cunning feature to keep a certain degree of randomness. Again, a shuffle button can remix the atmosphere as many times as desired. Also, further implementations of the model may take into account other parameters such as energy variation (in order to avoid

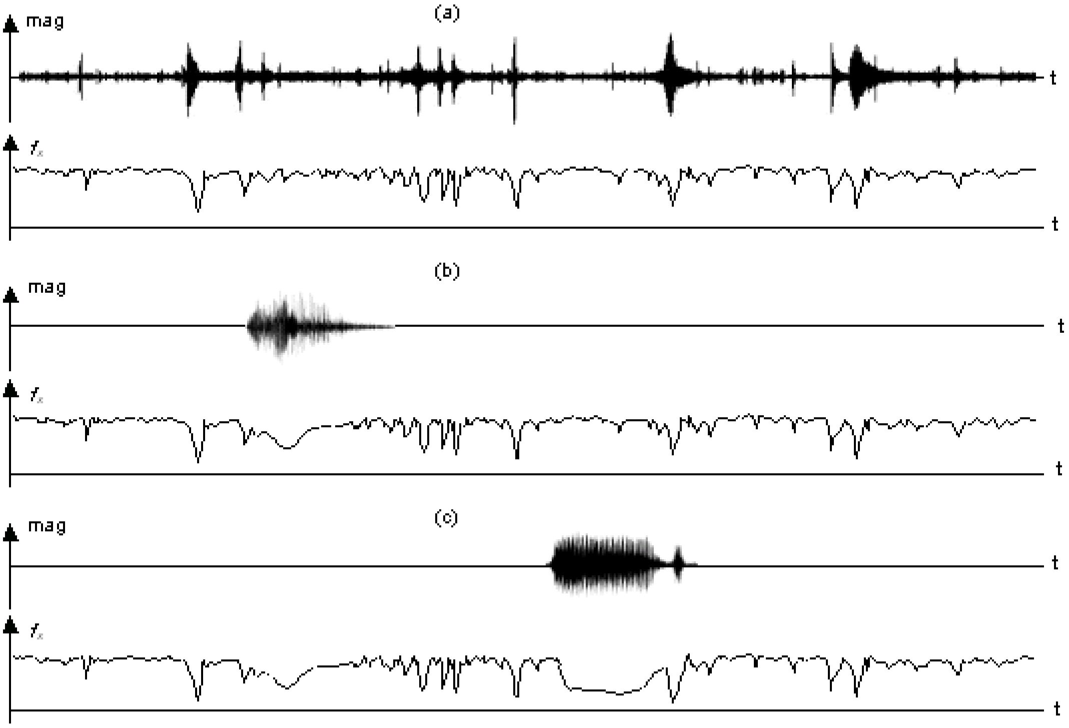


Figure 3.17: Mix example. a. long ambient sound and the corresponding probability density function. b and c SFX added and the corresponding recalculated probability density function

two transients happening at the same time), such as spectrum centroid (in order to avoid as much as possible the frequency content overlap), or others.

Another important feature is the automatic adjustment of channel controls: gain, panning and equalization. Regarding the levels, these are set so that the track maximum levels are 3 dB above the long ambient mean level and that no saturation / clipping problems appear. Regarding the stereo panning the ambient sound is centered and the isolated tracks are panned one left one right along time in order to minimize time overlap. The amount of panning depends on how close are two consecutive tracks, the closer, the more panned. Equalizing is only applied to those tracks that overlap significantly in frequency domain with the adjacent tracks or with the ambient loop sound. In these cases the effect track is 6-band equalized to flatten down to -12 dB the overlapping frequency region.

Finally, the strategy for the automation of the effects we propose is based on rules.

These rules are mainly related with the context of the ambiance. Say we are reconstructing an office atmosphere, we will apply a medium room reverb to whatever effect track we drop to the mix; if we are reconstructing a mountain atmosphere, we can apply some echo to the tracks.

#### 3.4.4.1 Integration in professional environments

The advent of high quality audio and spatialization surround setups (e.g: 5.1), first in the film industry, and more recently in home entertainment with DVD, offers the possibilities to create more engaging and immersive ambient sound. It is now possible to have ambient loops that take advantage of very low pitch sound (using subwoofers). It is possible to simulate movement in a tri-dimensional space or specific sound elements that pan in every direction we wish. On the other hand the complexity of setting up a multi track project for a surround scenario increases a lot. It would be extremely useful for a sound designer to specify at a higher level which surround characteristics are desirable for the ambiance, so that the system can provide him a multi track project file, and respective sound files, already configured to be integrated in his main project.

### 3.4.5 Results and discussion

Let us now give critical comments on some typical examples on ambiance generation:

Some of the ambiances created had too many events in it. The "jungle" ambiance had plenty of tropical birds, elephants and monkeys and sounded more like a zoo than a jungle.

Some of the ambiances need greater detail in the specification. A "war" ambiance query returned war sounds of different epochs, e.g: bombs, machine guns, swords and laser guns.

The sex ambiance retrieved sounds produced by too many people to be realistic.

These experiences lead us to the conception of a refinement control to add/remove specific sound classes or another control for the density of specifics.

As a multi track application, we have used the free editor Audacity.[9] In addition to common sound editing functionalities, Audacity allows to mix several tracks together and apply effects to tracks. Audacity allows to save multi track sessions yet it does not read sessions created by external programs. We have therefore tweaked the application in order to load our automatically generated ambiance multi track sessions.

We have presented a system for semi-automatic ambiance generation. The ambiances generated by textual query can be further refined by the user. The user controls the number of sounds that should be returned and can add and remove types of sounds, e.g: "more penguin sounds". Furthermore the ambiance is delivered to the user as a multi track project, providing thus flexibility to fine tune the results. We plan to extend this work to semi-automatic sonifications of audiovisual productions given scripts (or briefings) and some information of the timing.

[9] `http://audacity.sourceforge.net/`

# Chapter 4

# Discussion and future work

This dissertation addressed several fundamental issues regarding audio content management. We have surveyed a low-level description audio content technique: audio fingerprinting and its applications. We have justified low-level content-based usefulness as well as its limitations with audio fingerprinting example applications: e.g. they provide identification of distorted recordings, they are even able to provide "query-by-example" type of queries but are not able to bridge the semantic gap in the sense that they do not produce human readable metadata. In the context of sound effects, we have underlined open issues in state-of-the-art automatic sound annotation as a method to bridge the semantic gap, mainly its limitations to working conditions in limited domains: a few musical instruments or backgrounds, and proposed both a general scalable memory-based method together with a real-world taxonomy: WordNet. Finally we have implemented and evaluated content and concept-based search in a production size sound effects search engine as well as provide an example of an application that uses the framework: Automatic background generation.

In this last chapter, we briefly summarize the contributions we believe this dissertation makes to the state-of-the-art in automatic content-based audio retrieval. We also outline some promising lines of research that we believe are worth exploring yet are not part of the thesis because of time constraints or because they go beyond the scope of the dissertation.

## 4.1 Summary of contributions

### 4.1.1 Low-level audio description: Audio Fingerprinting

The field of audio fingerprinting was established as a research area at around the same time that this work begun. As a consequence, at the time there were not surveys on the state-of-the-art nor clear boundaries of what was fingerprinting and was not. As of today, audio fingerprinting is probably the most successful audio content-based technology from the industrial point of view in the sense that even though young—its establishment as research area coincided chronologically with the beginning of this doctoral work at the end of the 90s—over the years it has consolidated as a solid technology with several implementations in production monitoring radios and TV channels in several countries and millions of users worldwide. Among the contributions of this dissertation to audio fingerprinting we highlight:

**Audio fingerprinting definition and uses**

In Chapter 2 we introduced the concepts and applications of fingerprinting. Fingerprinting technologies allow the monitoring of audio content without the need of metadata or watermark embedding. However, additional uses exist for audio fingerprinting. The rationale is presented along with the differences with respect to watermarking. The main requirements of fingerprinting systems are described. The basic modes of employing audio fingerprints, namely identification, authentication, content-based secret key generation for watermarking and content-based audio retrieval and processing are depicted. Some concrete scenarios and business models where the technology is used were described.

**Audio fingerprinting functional framework and review of algorithms**

At the beginning of fingerprinting research, the different approaches were described with different rationales and terminology depending on the background of the researchers. Accordingly fingerprinting was presented as a Pattern matching, Multimedia (Music) Information Retrieval or Cryptography (Robust Hashing) problem. In Section 2.2, we presented a unifying functional framework for audio fingerprinting which we believe permits to explain existing systems as different instances of the same general model.

**Fingerprinting system for broadcast audio**

In the context of the AudioDNA fingerprinting system for broadcast audio described in Section 2.3, we contributed with a similarity metric and approximate search methods with a system based on FASTA.

**Integrity verification method**

In Chapter 2.4 we introduced a novel method for audio-integrity verification based on a combination of watermarking and fingerprinting. The fingerprint is a sequence of symbols ("AudioDNA") that enables one to identify an audio signal. Integrity verification is performed by embedding the fingerprint into the audio signal itself by means of a watermark. The original fingerprint is reconstructed from the watermark and compared with a new fingerprint extracted from the watermarked signal. If they are identical, the signal has not been modified; if not, the system is able to determine the approximate locations where the signal has been corrupted.

**From fingerprinting toward music similarity search**

We experimented how fingerprinting methods can be relaxed for similarity search. Specifically, we have proposed the use of *FastMap* method for improving a content-based audio identification system. The tool proves to be interesting, not only for audio fingerprinting research, but also as a component of a search-enabled audio browser.

### 4.1.2 Semantic Audio Management: Sound Effects

Several contributions for bridging the semantic gap in the context of managing sound effects were introduced.

**Sound description: Ontology Management**

We have pointed out some of the problems in cataloging sound. Specifically, how to generate searchable and machine readable SFX description metadata. We have reviewed some of the literature for audio classification as well as mined legacy SFX metadata from professional SFX libraries. We have implemented a knowledge management system inspired on the MPEG-7 framework for Multimedia and relying on WordNet as taxonomy-backbone. The proposed framework has several advantages. The librarian, does not need to add many terms—e.g. this is the sound of "car, automobile, vehicle"— since many relations are already encoded in the ontology and hence they do not need to be explicitly entered. For the user, categories can be created dynamically allowing to search and navigate through taxonomies based on psycholinguistic and cognitive theories. The terms—even though described externally as plain English—are machine readable, unambiguous and can be used for concept-based retrieval as well as serve as a classification scheme for a general sound annotator.

**Generic Sound Annotation**

We have contributed with a general sound annotator that overcomes some of the limitation of current annotation methods. Automatic annotation methods, normally fine-tuned to reduced domains such as musical instruments or reduced sound effects taxonomies. Usually, in identification a classifier is build to identify certain concepts: “cars” , “laughs”, “piano”. Sound samples are gathered and are tagged with those concepts and finally a classifier is trained to learn those concepts. The number of concepts and its possible combinations in the real world makes this approach unfeasible, as one would need to train tens of thousands of classifiers and new ones would have to be trained for new concepts. We have contributed with an all-purpose sound recognition system based on nearest-neighbor classification rule (Cano et al., 2005a). A sound sample will be labeled with the descriptions from the similar sounding examples of a annotated database. The terms borrowed from the closest match are unambiguous due to the use of WordNet as the taxonomy back-end. With unambiguous tagging, we refer to assigning concepts and not just terms to sounds. For instance, using the term “bar” for describing a sound is ambiguous, it could be “bar” as “rigid piece of metal or wood” or as “establishment where alcoholic drinks are served” where each concept has a unique identifier. This solution enables the sound annotator to use and export cues from/to video recognition systems or the inclusion of context knowledge, e.g: “Recognize this sound, the sound is from a mammal”. In the evaluation, the automatic annotator yielded a 30% concept prediction on a database of over 50,000 sounds and over 1,600 classes.

**Sound effects Search Engine**

We have presented several technologies that enhance and fit smoothly into professional sound effects providers working processes. Main sound effects search engines use standard text-retrieval technologies. The vagueness of the query specification, normally one or two words, together with the ambiguity and informality of natural languages affects the quality of the search: Some relevant sounds are not retrieved and some irrelevant ones are presented to the user. Moreover, sounds without caption are invisible to the users. Content-based audio tools offer perceptual ways of navigating the audio collections, like “find similar sounds", even if unlabeled, or query-by-example. We have described the integration of semantically-enhanced management of metadata using WordNet together with content-based methods in a commercial sound effect management system. The audio annotation engine can propose descriptions for new unlabeled audio. The descriptions can be used as generated or, given the limited recognition accuracy, be presented as options to an editor for validation.

**Intelligent Authoring: Ambiance generation**

We have proposed a semi-automatic system for ambiance generation. The system creates ambiances on demand given text queries by fetching relevant sounds from a large sound effect database and importing them into a sequencer multi track project. Ambiances of diverse nature can be created easily. Several controls are provided to the users to refine the type of samples and the sound arrangement.

## 4.2 Future perspectives

This section outlines promising future research lines in the different topics which are covered by this dissertation, namely: audio fingerprinting, semantic sound retrieval as well as some miscellaneous areas such as evaluation or complex networks.

### 4.2.1 Audio fingerprinting

**Compact and robust fingerprints** Even though successful implementations of audio fingerprinting systems exist in the market, there is room for improvement. Specifically more discriminative, compact and robust features would allow for smaller fingerprints. Together with distance and scalable search methods, this would allow for more accurate and computationally efficient systems.

**Extensions to content-based similarity search** The presented framework shares many components with other content-based search engines— e.g.: compact yet informative representations of media assets together with efficient similarity methods. It remains to be seen how the efficient fingerprinting systems can be extended from signal-based pure identification tasks toward higher-level navigation of audio assets. One example into that direction is the MusicSurfer (Cano et al., 2005b), a system for the interaction with massive collections of music. MusicSurfer automatically extracts descriptions related to instrumentation, rhythm and harmony from music audio signals. Together with efficient similarity metrics, the descriptions allow navigation of multi million track music collections in a flexible and efficient way without the need of external metadata or human ratings.

### 4.2.2 Semantic sound retrieval

**Sound similarity measure.** The similarity metrics are at the core of many information retrieval applications, such as clustering, query by example or visualization. For the sound effects prototype, we have presented a very simple similarity measure, the Manhattan distance (L1 distance). Further research should be devoted toward understanding and evaluating different metrics (Santini and Jain, 1999), possibly more related to psychological findings Tversky (1977).

**Flexible distances depending on query point.** The use of a single distance for the whole sound space seems counter intuitive. We would like to explore adaptive functions that weight dimensions depending on the query sound (Hastie and Tibshirani, 1996).

**Reasoning and ontology.** The results returned by the general annotator can be further refined. The NN rule can be combined with other classifiers: If the system returns that a particular sound could be a violin pizzicato or a guitar, we can then retrieve pizzicato violin and guitar sounds of the same pitch and decide which is more likely. In order to perform this classification on real-time a lazy classifier could be chosen. Another example is "car approaches", where we can look for other examples of "cars" and other "motor vehicle" "approaches" or "departs" to decide which is the right action. This same rationale applies to adjective type of modifiers, something can be described as "loud", "bright" or "fast". The concept "fast" means something different if we talk of "footsteps" or "typing".

**Fusion with other sources information: Video, images, context.** A promising direction is researching new methods for generating media annotations by combining audiovisual content analysis, as well as textual information. The output of the different annotators focusing on different aspects of a multimedia object can be aggregated to provide a better accuracy. It might be so that a unimodal algorithm is unable to clearly annotate multimedia content, e.g. a "cat meow" can be confused with an "old door opening" when we concentrate on the audio only. Hints coming from the visual analysis can possibly break this ambiguity. Conversely, in event detection for use in sports, such as finding the significant events in a football game, it has been found difficult to achieve good results with visual (video) analysis alone, but much simpler by detecting increases in the crowd cheering and clapping in the audio domain. This includes developing joint analysis methods, ontology and data fusion techniques for connecting automatically extracted metadata with semantic labels and social networks information. The sound annotator considered the sounds in isolation when many times they occur in a stream. The fact that certain events occur together in a sequence provides cues of what is going on: a sequence of whoosh, silence, hit and a gurgle sounds, separately are impossible to unambiguously identify, however, when they happen in a sequence, that is very indicative of golf events.

### 4.2.3 Miscellaneous

**On evaluation**

Among the vast number of disciplines and approaches to MIR (an overview of which can be found in Downie (2003b)), automatic description of audio signals in terms of musically-meaningful concepts plays an important role. As in other scientific endeavor, long-term improvements are bounded to systematic evaluation of models. For instance, text retrieval techniques significantly improved over the years thanks to the TREC initiative (see `trec.nist.org`) and the standardization of databases and evaluation metrics greatly facilitated progress in the fields of Speech Recognition (Przybocki and Martin, 1989), Machine Learning (Guyon et al., 2005) or Video Retrieval (see `http://www-nlpir.nist.gov/projects/trecvid/`). Systematic evaluations permit to measure but also to guide progresses in a specific field. Since a few years, the MIR community has recognized the necessity to conduct rigorous and comprehensive evaluations (Downie, 2003b; Berenzweig et al., 2004). An Audio Description Contest took place during the 5th edition of the ISMIR in Barcelona, Spain, in October 2004. The goal of this Contest was to compare state-of-the-art audio algorithms and systems relevant for MIR. It represents the first world-wide competition on algorithms for audio description for MIR. The original idea to organize such an event emerged from the research infrastructure in place at the Music Technology Group of the Pompeu Fabra University (who hosted ISMIR 2004) where around 50 researchers work on tasks related to musical audio analysis and synthesis (audio fingerprinting, singing voice synthesis, music content processing, etc., see www.iua.upf.es/mtg). A massive storage and computer cluster facility hosts a common repository of audio data and provides computing functionalities, thus permitting evaluation of developed algorithms (Cano et al., 2004b). Several audio description tasks were proposed to the MIR community in advance and the contest organizers gave full support for other potential tasks that would emerge from the community. Participation was open and all aspects of the several contests (data, evaluation methods, etc.) were publicly discussed and agreed. Finally, a total of 20 participants (from 12 research laboratories) took part in one or several of the following tasks: Melody extraction, Tempo induction, Genre Identification, Artist Identification and Rhythm classification (Cano et al., 2006b; Gouyon et al., 2006). The contest has successfully continued as an annual evaluation under the name of MIREX (Downie et al., 2005). This kind of events already greatly improve the quality of the research as well as big advances in a certain field. A promising, although resource consuming, future line would be to establish a certification entity. Such an organization would be in charge of certifying music and audio content-based techniques. We believe that such a certification would greatly enhance wide-spread adoption of audio and music description algorithms by the industry.

**Complex networks and music evaluation**

Complex network analysis is used to describe a wide variety of systems with interacting parts: networks of collaborating movie actors, the WWW, neural networks, metabolic pathways of numerous living organisms, to name a few. New insights can be unveiled by considering musical works and musical artists as parts of a huge structure of interconnecting parts that influence each other (Newman, 2003). A network is a collection of items, named vertices or nodes, with connections between them. The study of the networks' underlying complex systems is easier than studying the full dynamics of the systems. Yet, this analysis can provide insights on the design principles, the functions and the evolution of complex systems (Newman, 2003; Dorogovtsev and Mendes, 2003). Significant amount of multidisciplinary research on social, biological, information and technological networks has uncovered that complex systems of different nature do share certain topological characteristics. Indeed, the spread of certain ideas and religions, the success of companies, the spreading of sexually transmitted diseases such as the AIDS epidemic or computer viruses can be better understood by studying the topologies of the systems where they interact (Newman, 2003).

In preliminary work (Cano et al., 2006a; Cano and Koppenberger, 2004), complex network measurements (Barabási, 2002; Newman, 2003) were used to analyze the topology of networks underlying main music recommendation systems. The properties that emerge raise a discussion on the underlying forces driving the creation of such information systems. We can also obtain some hints about how much of the network structure is due to content similarity and how much to the self-organization of the network. Therefore, it can shed new light on the design and validation of music similarity measures and its evaluation (Logan et al., 2003; Berenzweig et al., 2004). Furthermore, it uncovers possible optimizations when designing music information systems, such as the optimal number of links between artists or the shortest path from artist to artist. In this sense, recommendation networks can be optimized by adding (or removing) links to facilitate navigating from artist to artist in a short number of *clicks*. Finally, we can obtain information about which artist has more links or which genres are more extended. This kind of information may help to understand the dynamics of certain aspects of music evolution, e.g: how did an artist get popular or how the music genres emerged.

Needless to say that there are more open venues than answers in the dissertation but we hope that some of the leads can be taken over by other researchers when designing music information systems that combine the strengths of humans and machines.

# Bibliography

Aigrain, P. (1999). New applications of content processing of music. *Journal of New Music Research*, 28(4):271–280.

Allamanche, E., Herre, J., Helmuth, O., Fröba, B., Kasten, T., and Cremer, M. (2001). Content-based identification of audio material using mpeg-7 low level description. In *Proc. of the Int. Symp. of Music Information Retrieval*, Indiana, USA.

Amatriain, X., Bonada, J., Loscos, A., and Serra, X. (2002). Spectral processing. In Zölzer, U., editor, *DAFX Digital Audio Effects*. J. Wiley and Sons.

Aslandogan, Y. A., Thier, C., Yu, C. T., Zou, J., and Rishe, N. (1997). Using semantic contents and WordNet in image retrieval. In *Proc. of the SIGIR*, Philadelphia, PA.

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison Wesley.

Banerjee, S. and Pedersen, T. (2003). The design, implementation, and use of the Ngram Statistic Package. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, Mexico City.

Barabási, A.-L. (2002). *Linked: The new science of networks*. Perseus.

Barnard, K., Duygulu, P., de Freitas, N., Forsyth, D., Blei, D., and Jordan, M. I. (2003). Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135.

Basalaj, W. (2001). Proximity visualization of abstract data. In *Technical Report 509, University of Cambridge Computer Laboratory*.

Batlle, E. and Cano, P. (2000). Automatic segmentation for music classification using competitive hidden markov models. In *Proceedings of the International Symposium on Music Information Retrieval*, Boston.

Batlle, E., Masip, J., and Cano, P. (2003). System analysis and performance tuning for broadcast audio fingerprinting. In *Proceedings of 6th International Conference on Digital Audio Effects*, London, UK.

Batlle, E., Masip, J., and Cano, P. (2004). Scalability issues in hmm-based audio fingerprinting. In *Proc. of IEEE International Conference on Multimedia and Expo*.

Batlle, E., Masip, J., and Guaus, E. (2002). Automatic song identification in noisy broadcast audio. In *Proc. of the SIP*.

Batlle, E., Nadeu, C., and Fonollosa, J. (1998). Feature decorrelation methods in speech recognition. a comparative study. In *Proceedings of International Conference on Speech and Language Processing*.

Bender, W., Gruhl, D., Morimoto, N., and Lu, A. (1996). Techniques for data hiding. *IBM System Journal*, 35:313–336.

Benitez, A. B., Smith, J. R., and Chang, S.-F. (2000). Medianet: A multimedia information network for knowledge representation. In *Proceedings of the SPIE 2000 Conference on Internet Multimedia Management Systems*, volume 4210.

Berenzweig, A., Logan, B., Ellis, D. P. W., and Whitman, B. (2004). A large-scale evaluation of acoustic and subjective music-similarity measures. *Computer Music Journal*, 28(2):63–76.

Blum, A. and Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97(1-2):245–271.

Blum, T., Keislar, D., Wheaton, J., and Wold, E. (1999). Method and article of manufacture for content-based analysis, storage, retrieval and segmentation of audio information.

Boeuf, J. and Stern, J. (2001). An analysis of one of the sdmi candidates. In *http://www.julienstern.org/sdmi/files/sdmiF/sdmiF.html*.

Boneh, D. (1999). Twenty years of attacks on the rsa cryptosystem. *American Mathematical Society.*

Boney, L., Tewfik, A., and Hamdy, K. (1996a). Digital watermarks for audio signals. In *IEEE Proceedings Multimedia*, pages 473–480.

Boney, L., Tewfik, A., and Hamdy, K. (1996b). Digital watermarks for audio signals. *IEEE Proceedings Multimedia.*

Bregman, A. (1998). Psychological data and computational ASA. In Rosenthal, D. and Okuno, H., editors, *Computational Auditory Scene Analysis.* Lawrence Erlbaum Associates, New Jersey.

Brian, D. (Summer 2002). Lingua:WordNet. *The Perl Journal*, 5(2):12–17.

Burges, C., Platt, J., and Jana, S. (2002). Extracting noise-robust features from audio data. In *Proc. of the ICASSP*, Florida, USA.

Burges, C., Platt, J., and Jana, S. (2003). Distortion discriminant analysis for audio fingerprinting. *IEEE Transactions on Speech and Audio Processing*, 11(3):165–174.

Cano, P. (1998). Fundamental Frequency Estimation In The SMS Analysis. *Proceedings of the Digital Audio Effects.*

Cano, P., Batlle, E., Kalker, T., and Haitsma, J. (2004a). A review of audio fingerprinting. *Journal of VLSI Signal Processing Systems*, To appear.

Cano, P., Batlle, E., Mayer, H., and Neuschmied, H. (2002a). Robust sound modeling for song detection in broadcast audio. In *Proc. AES 112th Int. Conv.*, Munich, Germany.

Cano, P., Celma, O., Koppenberger, M., and Martin-Buldú, J. (2006a). Topology of music recommendation networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 16(013107).

Cano, P., Gomez, E., Gouyon, F., Koppenberger, M., Ong, B., Streich, S., and Wack, N. (2006b). Ismir 2004 audio description contest. *MTG-Technical Report-2006-2.*

Cano, P., Kaltenbrunner, M., Gouyon, F., and Batlle, E. (2002b). On the use of FastMap for audio information retrieval. In *Proceedings of the International Symposium on Music Information Retrieval*, Paris, France.

Cano, P., Kaltenbrunner, M., Mayor, O., and Batlle, E. (2001). Statistical significance in song-spotting in audio. In *Proceedings of the International Symposium on Music Information Retrieval*.

Cano, P. and Koppenberger, M. (2004). The emergence of complex network patterns in music networks. In *Proceedings of Fifth International Conference on Music Information Retrieval*, Barcelona.

Cano, P., Koppenberger, M., Ferradans, S., Martinez, A., Gouyon, F., Sandvold, V., Tarasov, V., and Wack, N. (2004b). Mtg-db: A repository for music audio processing. In *Proceedings of 4th International Conference on Web Delivering of Music*, Barcelona, Spain.

Cano, P., Koppenberger, M., Groux, S. L., Herrera, P., and Wack, N. (2004c). Perceptual and semantic management of sound effects with a WordNet-based taxonomy. In *Proc. of the ICETE*, Setúbal, Portugal.

Cano, P., Koppenberger, M., Groux, S. L., Ricard, J., Herrera, P., and Wack, N. (2004d). Nearest-neighbor generic sound classification with a wordnet-based taxonomy. In *Proc.116th AES Convention*, Berlin, Germany.

Cano, P., Koppenberger, M., Herrera, P., and Celma, O. (2004e). Sound effects taxonomy management in production environments. In *Proc. AES 25th Int. Conf.*, London, UK.

Cano, P., Koppenberger, M., Le Groux, S., Ricard, J., and Wack, N. (2004f). Semantic and perceptual management of sound effects in production systems. In *Proceedings of International Broadcasting Conference*, Amsterdam, The Netherlands.

Cano, P., Koppenberger, M., Le Groux, S., Ricard, J., Wack, N., and Herrera, P. (2005a). Nearest-neighbor automatic sound classification with a wordnet taxonomy. *Journal of Intelligent Information Systems*, 24(2):99–111.

Cano, P., Koppenberger, M., Wack, N., G. Mahedero, J., Aussenac, T., Marxer, R., Masip, J., Celma, O., Garcia, D., Gómez, E., Gouyon, F., Guaus, E., Herrera, P., Massaguer, J., Ong, B., Ramírez, M., Streich, S., and Serra, X. (2005b). Content-based music audio recommendation. In *Proceedings of ACM Multimedia*, Singapore, Singapore.

Casey, M. (2002). Generalized sound classification and similarity in MPEG-7. *Organized Sound*, 6(2).

Casey M. A. and Westner A. (2000). Separation of Mixed Audio Sources By Independent Subspace Analysis. *International Computer Music Conference*, pages 154–161.

CBID (2002). Audio identification technology overview.

Celma, O. and Mieza, E. (2004). An opera information system based on MPEG-7. In *Proc. AES 25th Int. Conf.*, London, UK.

Chávez, E., Navarro, G., Baeza-Yates, R. A., and Marroquin, J. L. (2001). Searching in metric spaces. *ACM Computing Surveys*, 33(3):273–321.

Consortium, M. (2001). ISO working draft - information technology - multimedia content description interface - part4: Audio. Report ISO/IEC 15938-4:2001.

Consortium, M. (2002). MPEG-7 schema and description examples, final draft international standard. Electronic Citation.

Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Trans. Information Theory*, 13:21–27.

Craver, S., M, W., and Liu, B. (2001). What can we reasonably expect from watermarks? In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY.

Dannenberg, R., Foote, J., Tzanetakis, G., and Weare, C. (2001). Panel: New directions in music information retrieval. In *Proceedings of the International Computer Music Conference.*

de Boer, M., Bonada, J., Cano, P., Loscos, A., and Serra, X. (2000). Singing Voice Impersonator Application for PC. *Proceedings of the International Computer Music Conference*.

de C. T. Gomes, L., Gómez, E., and Moreau, N. (2001). Resynchronization methods for audio watermarking. In *Proceedings of 111th AES Convention*.

Dietterich, T. (2002). Ensemble learning. In Arbib, M., editor, *The Handbook of Brain Theory and Neural Networks*. MIT Press, Cambridge MA, 2nd edition.

Dittmann, J. (2001). Content-fragile watermarking for image authentication. In *Proceedings of SPIE, vol. 4314*, Bellingham.

Dittmann, J., Steinmetz, A., and Steinmetz, R. (1999). Content-based digital signature for motion pictures authentication and content-fragile watermarking. In *Proceedings of the International Conference on Multimedia Computing and Systems*, Florence, Italy.

Dixon, R. (1976). *Spread-spectrum systems*. John Wiley & Sons.

Dorai, C. and Venkatesh, S. (2001). Bridging the semantic gap in content management systems: Computational media aesthetics.

Dorogovtsev, S. N. and Mendes, J. F. F. (2003). *Evolution of Networks: From Biological Nets to the Internet and WWW*. Oxford University Press, Oxford.

Downie, J. S. (2003a). Music information retrieval. *Annual Review of Information Science and Technology*, 37:295–343.

Downie, J. S., West, K., Ehmann, A. F., and Vincent, E. (2005). The 2005 music information retrieval evaluation exchange (mirex 2005): Preliminary overview. In *ISMIR*, pages 320–323.

Downie, S. (2003b). The scientific evaluation of music information retrieval systems: Foundations and future. *Computer Music Journal*, 28(2):12–22.

Dubnov, S. and Ben-Shalom, A. (2003). Review of ICA and HOS methods for retrieval of natural sounds and sound effects. In *4th International Symposium on Independent Component Analysis and Blind Signal Separation*, Japan.

Duda, R., Hart, P., and Stork, D. (2001). *Pattern classification.* J. Wiley and Sons, New York, 2nd edition.

Etantrum (2002). Etantrum.

Faloutsos, C. and Lin, K. (1995). Fastmap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets. In *Proceedings of the 1995 ACM SIGMOD*, pages 163–174.

Faloutsos, C., Ranganathan, M., and Manolopoulos, Y. (1994). Fast subsequence matching in time-series databases. In *Proc. of the ACM SIGMOD*, pages 419–429, Mineapolis, MN.

Fernández-Cid P. (1998). *Transcripción Automática de Señales Polifónicas.* PhD thesis, Polythecnical University Madrid.

FindSounds (2003). FindSounds, www.findsounds.com.

Flank, S. (July-September 2002). Multimedia technology in context. *IEEE Multimedia*, pages 12–17.

Flank, S. and Brinkman, S. (2002). Drinking from the fire hose: How to manage all the metadata. In *Proceedings of the International Broadcasting Convention.*

García, R. A. (1999). Digital watermarking of audio signals using a psychoacoustic auditory model and spread-spectrum theory. In *Proceedings of the 107th AES Convention.*

Gaver, W. (1993). What in the world do we hear? an ecological approach to auditory event perception. *Ecological Psychology*, 5(1):1–29.

Gennaro, R. and Rohatgi, P. (1997). How to sign digital streams. *Advances in Cryptology.*

Gibson, J. J. (1979). *The ecological approach to visual perception.* Houghton Mifflin, Boston.

Gómez, E. (2000). *Tatouage de signaux de musique: méthodes de synchronisation.* DEA ATIAM thesis, ENST - IRCAM, Paris.

Gómez, E., Cano, P., de C.T. Gomes, L., Batlle, E., and Bonnet, M. (2002). Mixed watermarking-fingerprinting approach for integrity verification of audio recordings. In *Proceedings of the International Telecommunications Symposium*, Natal, Brazil.

Gómez, E. (2005). Tonal description of polyphonic audio for music content processing. *INFORMS Journal on Computing*, in press.

Gómez, E. and Herrera, P. (2004). Automatic extraction of tonal metadata from polyphonic audio recordings. In *Proc. 25th International AES Conference*, pages 74–80.

Gómez, E., Klapuri, A., and Meudic, B. (2003). Melody description and extraction in the context of music content processing. *Journal of New Music Research*, 32(1):23–41.

Gouyon, F. and Dixon, S. (2005). A review of automatic rhythm description systems. *Computer Music Journal*, 29(1):34–54.

Gouyon, F., Dixon, S., Pampalk, E., and Widmer, G. (2004). Evaluating rhythmic descriptors for musical genre classification. In *Proc. 25th International AES Conference*, pages 196–204, London. Audio Engineering Society.

Gouyon, F., Herrera, P., and Cano, P. (2002). Pulse-dependent analyses of percussive music. In *Proc. 22nd International AES Conference*, pages 396–401. Audio Engineering Society.

Gouyon, F., Klapuri, A., Dixon, S., Alonso, M., Tzanetakis, G., Uhle, C., and Cano, P. (2006). An experimental comparison of audio tempo induction algorithms. *IEEE Trans. Speech and Audio Processing*, in press.

Gouyon, F. and Meudic, B. (2003). Towards rhythmic content processing of musical signals - Fostering complementary approaches. *Journal of New Music Research*, 32(1):41–65.

Guaus, E. and Batlle, E. (2003). Visualization of metre and other rhythm features. In *Proc. IEEE Symposium on Signal Processing and Information Technology*, Darmstadt.

Gusfield, D. (1997). *Algorithms on Strings, Trees and Sequences.* Cambridge University Press.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182.

Guyon, I., Gunn, S., Ben-Hur, A., and Dror, G. (2005). Result analysis of the NIPS 2003 feature selection challenge. In Saul, L. K., Weiss, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 17*, pages 545–552, Cambridge, MA. MIT Press.

Gygi, B. (2001). *Factors in the identification of environmental sounds.* Ph.D. Thesis, Indiana University.

Haitsma, J. and Kalker, T. (2002a). An efficient database search strategy for audio fingerprinting. In *5th IEEE Int. Workshop on Multimedia Signal Processing: special session on Media Recognition*, US Virgin Islands, USA.

Haitsma, J. and Kalker, T. (2002b). A highly robust audio fingerprinting system. In *Proceedings of the International Symposium on Music Information Retrieval*, Paris, France.

Haitsma, J., Kalker, T., and Oostveen, J. (2001). Robust audio hashing for content identification. In *Proc. of the Content-Based Multimedia Indexing.*

Hastie, T. and Tibshirani, R. (1996). Discriminant adaptive nearest neighbor classification and regression. In Touretzky, D. S., Mozer, M. C., and Hasselmo, M. E., editors, *Advances in Neural Information Processing Systems*, volume 8, pages 409–415. The MIT Press.

Haykin, S. (1988). *Digital Communications.* Prentice Hall.

Haykin, S. (1996). *Adaptive Filter Theory.* Prentice Hall.

Herrera, P., Peeters, G., and Dubnov, S. (2003). Automatic classification of musical instrument sounds. *Journal of New Music Research*, 32(1).

Herrera, P., Yeterian, A., and Gouyon, F. (2002). Automatic classification of drum sounds: A comparison of feature selection methods and classification techniques. In Anagnostopoulou, C., Ferrand, M., and Smaill, A., editors, *Music and Artificial Intelligence.* Springer.

Honkela, T., Kaski, S., Lagus, K., and Kohonen, T. (1997). WEBSOM—self-organizing maps of document collections. In *Proceedings of WSOM'97, Workshop on Self-Organizing Maps, Espoo, Finland, June 4-6*, pages 310–315. Helsinki University of Technology, Neural Networks Research Centre, Espoo, Finland.

Jain, A. K., Duin, R. P., and Mao, J. (2000). Statistical pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–37.

Jonathan Foote (1999). An Overview Of Audio Information Retrieval. *ACM Multimedia Systems*, 7:2–10.

Kalker, T. (2001). Applications and challenges for audio fingerprinting. In *presentation at the 111th AES Convention*, New York.

Karlin, S. and Altschul, S. (1990). Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proceedings National Academy Sciences USA*, 87 (1990), 2264-2268.OoO(87):2264–2268.

Kashino, K., Kurozumi, T., and Murase, H. (2003). A quick search method for audio and video signals based on histogram pruning. *IEEE Transactions on Multimedia*, 5(3):348–357.

Kastner, T., Allamanche, E., Herre, J., Hellmuth, O., Cremer, M., and Grossmann, H. (2002). MPEG-7 scalable robust audio fingerprinting. In *Proc. AES 112th Int. Conv.*, Munich, Germany.

Kenyon, S. (1993). Signal recognition system and method. US 5,210,820.

Kimura, A., Kashino, K., Kurozumi, T., and Murase, H. (2001). Very quick audio searching: introducing global pruning to the time-series active search. In *Proc. of Int. Conf. on Computational Intelligence and Multimedia Applications*, Salt Lake City, Utah.

Kirovski, D. and Attias, H. (2002). Beat-id: Identifying music via beat analysis. In *5th IEEE Int. Workshop on Multimedia Signal Processing: special session on Media Recognition*, US Virgin Islands, USA.

Kohonen, T. (1997). *Self-Organizing Maps*. Springer-Verlag, $2^{nd}$ edition.

Kostek, B. and Czyzewski, A. (2001). Representing musical instrument sounds for their automatic classification. *J. Audio Eng. Soc.*, 49(9):768–785.

Kruskal, J. (1964). Nonmetric multidimensional scaling. *Psychometrika*, 29:115–129.

Kurth, F., Ribbrock, A., and Clausen, M. (2002). Identification of highly distorted audio material for querying large scale databases. In *Proc. AES 112th Int. Conv.*, Munich, Germany.

Lacy, J., Quackenbush, S., Reibman, A., Shur, D., and Snyder, J. (1998). On combining watermarking with perceptual coding.

Lakatos, S. (2000). A common perceptual space for harmonic and percussive timbres. *Perception & Psychoacoustics*, (62):1426–1439.

Lemström, K. (2000). *String Matching Techniques for Music Retrieval.* PhD thesis, University of Helsinky. /home/pcano/data/doc/old.doc/MusicInformationRetrieval.

Lemström K. and Perttu S. (2000). SEMEX - An Efficient Music Retrieval Prototype. *Proceedings Of International Symposium on Music Information Retrieval.*

Lindsay, A. and Kriechbaum, W. (1999). There's more than one way to hear it: Multiples representations of music in MPEG-7. *Journal of New Music Research*, 28(4):364–372.

L.Mott, R. (1990). *Sound Effects: Radio, TV, and Film.* Focal Press.

Logan, B. (2000). Mel frequency cepstral coefficients for music modeling. In *Proc. of the ISMIR*, Plymouth, MA.

Logan, B., Ellis, D., and Berenzweig, A. (2003). Toward evaluation techniques for music similarity. In *Proceedings of the 4th International Symposium on Music Information Retrieval.*

Loscos, A., Cano, P., and Bonada, J. (1999). Low-delay singing voice alignment to text. In *Proceedings of International Computer Music Conference 1999*, Beijing, China.

Lourens, J. (1990). Detection and logging advertisements using its sound. In *Proc. of the COMSIG*, Johannesburg.

Maidin, D. O. and Fernström, M. (2000). The best of two worlds: Retrieving and browsing. In *Proceedings of the Conference on Digital Audio Effects.*

Manjunath, B. S., Salembier, P., and Sikora, T. . (2002). *Introduction to MPEG-7. Multimedia Content Description Interface.* John Wiley & Sons, LTD.

Martin, K. D. (1999). *Sound-Source Recognition: A Theory and Computational Model.* Ph.D. Thesis, M.I.T.

Melucci, M., Orio, N., and Gambalunga, M. (2000). An evaluation study on music perception for music content-based information retrieval. In M., M., N., O., and M., G., editors, *Proc. International Computer Music Conference.*

Mihçak, M. and Venkatesan, R. (2001). A perceptual audio hashing algorithm: a tool for robust audio identification and information hiding. In *4th Int. Information Hiding Workshop*, Pittsburg, PA.

Miller, G. A. (November 1995). WordNet: A lexical database for english. *Communications of the ACM*, pages 39–45.

Miller, M. L., Cox, I. J., Linnartz, J.-P. M. G., and Kalker, T. (1999). A review of of watermarking principles and practices. In Parhi, K. K. and Nishitani, T., editors, *Digital Signal Processing for Multimedia Systems*, pages 461–485. IEEE.

Miller, M. L., Rodríguez, M. A., and Cox, I. J. (2005). Audio fingerprinting: Nearest neighbor search in high dimensional binary spaces. *Journal of VLSI Signal Processing Systems*, 41(3):285–291.

Needleman, S. and Wunsch, C. (1970). A General Method Applicable to the Search for Similarities in the Amino Acid Sequences of Two Proteins. *Journal of Molecular Biology*, 48:443–453.

Neuschmied, H., Mayer, H., and Batlle, E. (2001). Identification of audio titles on the internet. In *International Conference on Web Delivering of Music*, Florence, Italy.

Newman, M. (2003). The structure and function of complex networks. *SIAM Review*, 45(2):167–256.

Nicholas, H., Deerfield D. W., and Ropelewski, A. (1997). A Tutorial on Searching Sequences Databases and Sequence Scoring Methods. *no journal.*

Oppenheim, A. and Schafer, R. (2004). From frequency to quefrency: A history of the cepstrum. *IEEE Signal Processing Magazine*, 21(5):95–106.

Oppenheim, A. V. and Schafer, R. W. (1989). *Discrete-Time Signal Processing.* Prentice Hall.

Pampalk, E., Dixon, S., and G., W. (2003). Exploring music collections by browsing different views. In *Proc. International Conference on Music Information Retrieval*, pages 201–208.

Pampalk, E., Rauber, A., and Merkl, D. (2002a). Content-based Organization and Visualization of Music Archives. In *Proceedings of the ACM Multimedia*, pages 570–579, Juan les Pins, France. ACM.

Pampalk, E., Rauber, A., and Merkl, D. (2002b). Content-based organization and visualization of music archives. In *Proc. ACM International Conference on Multimedia*, pages 570–579.

Papaodysseus, C., Roussopoulos, G., Fragoulis, D., Panagopoulos, T., and Alexiou, C. (2001). A new approach to the automatic recognition of musical recordings. *J. Audio Eng. Soc.*, 49(1/2):23–35.

Park, M., Kim, H., and Yang, S. (2006). Frequency-temporal filtering for a robust audio fingerprinting schemes in real-noise environments. *ETRI Journal*, 28(4):509–512.

Pearson, W. and Lipman, D. (1988). Improved tools for Biological Sequence Comparison. *Proceedings of the National Academy of Sciences*, (85):2444–2448.

Peck, N. (2001). Beyond the library: Applying film post-production techniques to game sound design. In *Proc. of Game Developers Conference*, San Jose CA, USA.

Peeters, G. and Rodet, X. (2003). Hierarchical gaussian tree with inertia ratio maximization for the classification of large musical instruments databases. In *Proc. of the 6th Int. Conf. on Digital Audio Effects*, London.

Peltonen, V., Tuomi, J., Klapuri, A., Huopaniemi, J., and Sorsa, T. (2002). Computational auditory scene recognition. In *Proc. of ICASSP*, Florida, USA.

Picone, J. (1993). Signal modeling techniques in speech recognition. In *Proc. of the ICASSP*, volume 81, pages 1215–1247.

Plaschzug, W., Meier, S., Weinert, M., Wagner, F., and Bachmaier, G. (2000). User requirements. In *RAAT21- HSA-000614 Internal Report*, Firenze, Italy.

Przybocki, M. and Martin, A. (1989). Nist speaker recognition evaluations. In *Proc. International Conference on Language Resources and Evaluations*, pages 331–335.

Rabiner, L. R. (1990). A tutorial on hidden markov models and selected apllications in speech recognition. In Waibel, A. and Lee, K.-F., editors, *Readings in Speech Recognition*, pages 267–296. Kaufmann, San Mateo, CA.

Rabiner L.R. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2):257–286.

RIAA (2001). Request for information on audio fingerprinting technologies.

Ricard, J. and Herrera, P. (2004). Morphological sound description: Computational model and usability evaluation. In *Proc.116th AES Convention*, Berlin, Germany.

Richly, G., Varga, L., Kovàcs, F., and Hosszú, G. (2000). Short-term sound stream characterisation for reliable, real-time occurrence monitoring of given sound-prints. In *Proc. 10th Mediterranean Electrotechnical Conference, MEleCon.*

Ruggero, M. A. (1992). Physiology and coding of sounds in the auditory nerve. In *The Mammalian Auditory Pathway: Neurophysiology.* Springer-Verlag.

Russolo, L. (1986). *The Art of Noises.* Pendragon Press.

Sandvold, V., Gouyon, F., and Herrera, P. (2004). Percussion classification in polyphonic audio recordings using localized sound models. In *Proc. International Conference on Music Information Retrieval*, pages 537–540, Barcelona. Audiovisual Institute, Universitat Pompeu Fabra.

Santini, S. and Jain, R. (1999). Similarity measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):871–883.

Saunders J. (1996). Real Time Discrimination of Broadcast Speech/Music. *ICASSP.*

Schaeffer, P. (1966). *Traité des Objets Musicaux.* Editions du Seuil.

Schafer, R. M. (1977). *The Soundscape: Our Sonic Environment and the Tuning of the World.* Alfred Knopf, Inc.

Scheirer E. and Slaney M. (1997). Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator. *ICASSP*, pages 1331–1334.

SDMI (2001). Secure digital music initiative (sdmi).

Seo, J. S., Jin, M., Lee, S., Jang, D., Lee, S., , and Yoo, C. D. (2005). Audio fingerprinting based on normalized spectral subband centroids. In *Proc. of IEEE ICASSP*, Philadelphia, PA.

Serra, X. (1989). *A system for sound analysis/transformation/synthesis based on a deterministic plus stochastic decomposition.* PhD Thesis, CCRMA Stanford University, Palo Alto.

Serra X. (1997). Musical sound modeling wiht sinusoids plus noise. In C. Roads, S. Pope, A., editor, *Musical Signal Processing*. Swets & Zeitlinger Publishers.

Serra X. and Bonada J. (1998). Sound Transformations based on the SMS High Level Attributes. In *Proceedings of the Digital Audio Effects Workshop.*

Shaw, G. (2000). Digital document integrity. In *Proceedings of the 8th ACM Multimedia Conference*, Los Angeles, CA.

Shepard, R. N. (1962). The analysis of proximities: multidimensional scaling with an unknown distance function. *Psychometrika*, I and II(27):125–139 and 219–246.

Sillanpää, J., Klapuri, A., Seppänen, J., and Virtanen, T. (2000). Recognition of acoustic noise mixtures by combined bottom-up and top-down processing. In *Proc. European Signal Processing Conference.*

Slaney, M. (2002). Mixture of probability experts for audio retrieval and indexing. In *IEEE International Conference on Multimedia and Expo.*

Smith, T. and Waterman, M. (1981). Identification of Common Molecular Subsequences. *Journal of Molecular Biology*, pages 195–197.

SOB (2003). *The Sounding Object.* Mondo Estremo.

Soltau H., Schultz T., and Westphal M. (1998). Recognition of Music Types. *IEEE International Conference on Acoustics, Speech and Signal Processing.*

Subramanya, S., R.Simha, Narahari, B., and Youssef, A. (1999). Transform-based indexing of audio data for multimedia databases. In *Proc. of Int. Conf. on Computational Intelligence and Multimedia Applications*, New Delhi, India.

Sukittanon, S. and Atlas, L. (2002). Modulation frequency features for audio fingerprinting. In *Proc. of the ICASSP*.

Sukittanon, S., Atlas, L., and Pitton, J. (2004). Modulation scale analysis for content identification. *IEEE Transactions on Signal Processing*, 52(10):3023–3035.

Theodoris, S. and Koutroumbas, K. (1999). *Pattern Recognition.* Academic Press.

TRM (2002). Musicbrainz trm. musicbrainz-1.1.0.tar.gz.

Tversky, A. (1977). Features of similarity. *Psychological Review*, 84(4):327–352.

Tzanetakis, G. and Cook, P. (1999). Multifeature Audio Segmentation for Browsing and Annotation. *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics.*

Tzanetakis, G. and Cook, P. (2000). Audio Information Retrieval (AIR) tools. *Proceedings International Symposium on Music Information Retrieval.*

Tzanetakis, G. and Cook, P. (2001). Marsyas3d: A prototype audio browser-editor using a large scale immersive visual and audio display. In *Proceedings of the International Conference on Auditory Display*, pages 250–254.

Tzanetakis, G. and Cook, P. (2002). Musical genre classification of audio signals. *IEEE Trans. Speech and Audio Processing*, 10(5):293–302.

Tzanetakis, G., Essl, G., and Cook, P. (2001). Automatic musical genre classification of audio signals. In *Proc. International Symposium for Audio Information Retrieval.*

Venkatachalam, V., Cazzanti, L., Dhillon, N., and Wells, M. (2004). Automatic identification of sound recordings. *IEEE Signal Processing Magazine*, 21(2):92–99.

Vinet, H., Herrera, P., and Pachet, F. (2002). The cuidado project. In H., V., P., H., and F., P., editors, *Proc. International Symposium on Music Information Retrieval*.

Viterbi, A. (1970). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Informational Theory*, 13(2):260–269.

Wang, A. L.-C. and SmithII, J. (2002). System and methods for recognizing sound and music signals in high noise and distortion.

Weis, E. (1995). Sync tanks: The art and technique of postproduction sound. *Cineaste*, 21(1):56.

William R. Pearson (1998). Flexible sequence similarity searching with the FASTA program package.

Witten, I. and Frank, E. (2000). *Data Mining: Practical machine learning tools with Java implementations*. Morgan Kaufmann, San Francisco.

Wold, E., Blum, T., Keislar, D., and Wheaton, J. (1996). Content-based classification, search, and retrieval of audio. *IEEE Multimedia*, 3(3):27–36.

Wold, E., Blum, T., Keislar, D., and Wheaton, J. (1999). Classification, search, and retrieval of audio. In Wold, E., Blum, T., Keislar, D., and J, W., editors, *CRC Handbook of Multimedia Computing*. unknown.

Wu, C. and Kuo, C. J. (2001). Speech content integrity verification integrated with itu g.723.1 speech coding. In *Proceedings of IEEE International Conference on Information Technology: Coding and Computing*, pages 680–684.

Wu, M., Craver, S., Felten, E., and Liu, B. (2001). Analysis of attacks on sdmi audio watermarks. In *Proceedings of the ICASSP*.

Zhang, T. and Kuo, C.-C. J. (1999). Classification and retrieval of sound effects in audiovisual data management. In *Proceedings of the 33rd Asilomar Conference on Signals, Systems and Computers*.

Zwicker, E. and Fastl, H. (1990). *Psychoacoustics, Facts and Models*. Springer-Verlag.

# Appendix A

# Relevant Publications by the Author

In this annex we provide a list of publications related to the dissertation. The updated list of publications can be consulted at http://mtg.upf.edu.

## A.1 Journal Articles

- **Cano, P.** Koppenberger, M. Le Groux, S. Ricard, J. Wack, N. Herrera, P. 2005. 'Nearest-Neighbor Automatic Sound Classification with a WordNet Taxonomy' Journal of Intelligent Information Systems; Vol.24 .2 99-111

  **Relevant to:** Chapter 3

  Sound engineers need to access vast collections of sound effects for their film and video productions. Sound effects providers rely on text-retrieval techniques to offer their collections. Currently, annotation of audio content is done manually, which is an arduous task. Automatic annotation methods, normally fine-tuned to reduced domains such as musical instruments or reduced sound effects taxonomies, are not mature enough for labeling with great detail any possible sound. A general sound recognition tool would require: first, a taxonomy that represents the world and, second, thousands of classifiers, each specialized in distinguishing little details. We report experimental results on a general sound annotator. To tackle the taxonomy definition problem we use WordNet, a semantic network that organizes real world knowledge. In order to overcome the need of a huge number of classifiers to distinguish many different sound classes, we use a nearest-neighbor classifier with a database of isolated sounds unambiguously linked to WordNet concepts. A 30% concept prediction is achieved on a database of over 50,000 sounds and over 1,600 concepts.

- Gouyon, F. Klapuri, A. Dixon, S. Alonso, M. Tzanetakis, G. Uhle, C. **Cano, P.** 2005. 'An experimental comparison of audio tempo induction algorithms' IEEE Transactions on Speech and Audio Processing; Vol.14 .5 .

  **Relevant to:** Chapter 4

  We report on the tempo induction contest organized during the International Conference on Music Information Retrieval (ISMIR 2004) held at the University Pompeu Fabra in Barcelona in October 2004. The goal of this contest was to evaluate some state-of-the-art algorithms in the task of inducing the basic tempo (as a scalar, in beats per minute) from musical audio signals. To our knowledge, this is the first published large scale cross-validation of audio tempo induction algorithms. In order to stimulate further research, the contest results, annotations, evaluation software and part of the data are available at `http://ismir2004.ismir.net/ISMIR_Contest.html`.

- Gomes, L. **Cano, P.** Gómez, E. Bonnet, M. Batlle, E. 2003. 'Audio Watermarking and Fingerprinting: For Which Applications?' Journal of New Music Research; Vol.32 .1

  **Relevant to:** Chapter 2

  Although not a new issue, music piracy has acquired a new status in the digital era, as recordings can be easily copied and distributed. Watermarking has been proposed as a solution to this problem. It consists in embedding into the audio signal an inaudible mark containing copyright information. A different approach, called fingerprinting, consists in extracting a fingerprint from the audio signal. In association with a database, this fingerprint can be used to identify a recording, which is useful, for example, to monitor audio excerpts played by broadcasters and webcasters. There are far more applications to watermarking and fingerprinting. After a brief technical review, this article describes potential applications of both methodologies, showing which one is more suitable for each application.

- **Cano, P.** Celma, O. Koppenberger, M. Martin-Buldú, J. 2006. 'Topology of music recommendation networks'. Chaos: An Interdisciplinary Journal of Nonlinear Science Vol.16 .013107

  **Relevant to:** Chapter 4

  We study the topology of several music recommendation networks, which arise from relationships between artist, co-occurrence of songs in play lists or experts' recommendation. The analysis uncovers the emergence of complex network phenomena in these kinds of recommendation networks, built considering artists as nodes and their resemblance as links. We observe structural properties that provide some hints on navigation and possible optimizations on the design of music recommendation systems. Finally, the analysis derived from existing music knowledge sources provides a deeper understanding of the human music similarity perception.

- Park, J. Celma, O. Koppenberger, M. **Cano, P.**, Martin-Buldú, J. 2007. 'The Social Network of Contemporary Popular Musicians'. International Journal of Bifurcation and Chaos (in press)

  **Relevant to:** Chapter 4

  In this paper we analyze two social network datasets of contemporary musicians collected from the web site allmusic.com (AMG), an on-line browser for music recommendation. One is the collaboration network where two musicians are connected if they ever performed or produced an album together, and the other is the similarity network where musicians have been linked by music experts according to their musical similarity. Both networks exhibit typical characteristics of social networks such as high transitivity. However, differences in link patterns suggest different mechanisms of their formations, and possibly those of the human perception of music compared to automated recommendation systems, which are commonly implemented by many commercial music information providers.

## A.2 Book Chapters

- **Cano, P.** Batlle, E. Gómez, E. Gomes, L. Bonnet, M. 2005. 'Audio Fingerprinting: Concepts and Applications.' Halgamuge, Saman K.; Wang, Lipo Ed., Computational Intelligence for Modelling and Prediction, p.233-245 Springer-Verlag. ;

  **Relevant to:** Chapter 2

  An audio fingerprint is a unique and compact digest derived from perceptually relevant aspects of a recording. Fingerprinting technologies allow the monitoring of audio content without the need of metadata or watermark embedding. However, additional uses exist for audio fingerprinting. This paper aims to give a vision on Audio Fingerprinting. The rationale is presented along with the differences with respect to watermarking. The main requirements of fingerprinting systems are described. The basic modes of employing audio fingerprints, namely identification, authentication, content-based secret key generation for watermarking and content-based audio retrieval and processing are depicted. Some concrete scenarios and business models where the technology is used are presented, as well as an example of an audio fingerprinting extraction algorithm which has been proposed for both identification and verification.

## A.3 Peer-reviewed International Conferences

- **Cano, P.** Koppenberger, M. Wack, N. G. Mahedero, J. Masip, J. Celma, O. Garcia, D. Gómez, E. Gouyon, F. Guaus, E. Herrera, P. Massaguer, J. Ong, B. Ramírez, M. Streich, S. Serra, X. 2005. 'An Industrial-Strength Content-based Music Recommendation System' Proceedings of 28th Annual International ACM SIGIR Conference; Salvador, Brazil

  **Relevant to:** Chapter 4

  We present a metadata free system for the interaction with massive collections of music, the MusicSurfer. MusicSurfer automatically extracts descriptions related to instrumentation, rhythm and harmony from music audio signals. Together with efficient similarity metrics, the descriptions allow navigation of multimillion track music collections in a flexible and efficient way without the need of metadata or human ratings.

- Herrera, P. Celma, O. Massaguer, J. **Cano, P.** Gómez, E. Gouyon, F. Koppenberger, M. Garcia, D. G. Mahedero, J. Wack, N. 2005. 'Mucosa: a music content semantic annotator' Proceedings of 6th International Conference on Music Information Retrieval; London, UK

  **Relevant to:** Chapter 4

  MUCOSA (Music Content Semantic Annotator) is an environment for the annotation and generation of music metadata at different levels of abstraction. It is composed of three tiers: an annotation client that deals with micro-annotations (i.e. within-file annotations), a collection tagger, which deals with macro-annotations (i.e. across-files annotations), and a collaborative annotation subsystem, which manages large-scale annotation tasks that can be shared among different research centers. The annotation client is an enhanced version of WaveSurfer, a speech annotation tool. The collection tagger includes tools for automatic generation of unary descriptors, invention of new descriptors, and propagation of descriptors across sub-collections or playlists. Finally, the collaborative annotation subsystem, based on Plone, makes possible to share the annotation chores and results between several research institutions. A collection of annotated songs is available, as a starter pack to all the individuals or institutions that are eager to join this initiative.

- G. Mahedero, J. **Cano, P.** Martinez, A. Gouyon, F. Koppenberger, M. 2005. 'Natural language processing of lyrics' Proceedings of ACM Multimedia 2005; Singapore, Singapore

  **Relevant to:** Chapter 4

  We report experiments on the use of standard natural language processing (NLP) tools for the analysis of music lyrics. A significant amount of music audio has lyrics. Lyrics encode an important part of the semantics of a song, therefore their analysis complements that of acoustic and cultural metadata and is fundamental for the development of complete music information retrieval systems. Moreover, a textual analysis of a song can generate ground truth data that can be used to validate results from purely acoustic methods. Preliminary results on language identification, structure extraction, categorization and similarity searches suggests that a lot of profit can be gained from the analysis of lyrics.

- **Cano, P.** Koppenberger, M. Wack, N. G. Mahedero, J. Aussenac, T. Marxer, R. Masip, J. Celma, O. Garcia, D. Gómez, E. Gouyon, F. Guaus, E. Herrera, P. Massaguer, J. Ong, B. Ramírez, M. Streich, S. Serra, X. 2005. 'Content-based Music Audio Recommendation' Proceedings of ACM Multimedia 2005; Singapore, Singapore

  **Relevant to:** Chapter 4

  We present the MusicSurfer, a metadata free system for the interaction with massive collections of music. MusicSurfer automatically extracts descriptions related to instrumentation, rhythm and harmony from music audio signals. Together with efficient similarity metrics, the descriptions allow navigation of multimillion track music collections in a flexible and efficient way without the need for metadata nor human ratings.

- **Cano, P.** Koppenberger, M. Herrera, P. Celma, O. Tarasov, V. 2004. 'Sound Effect Taxonomy Management in Production Environments' Proceedings of 25th International AES Conference; London, UK

  **Relevant to:** Chapter 3

  Categories or classification schemes offer ways of navigating and having higher control over the search and retrieval of audio content. The MPEG7 standard provides description mechanisms and ontology management tools for multimedia documents. We have implemented a classification scheme for sound effects management inspired by the MPEG7 standard on top of an existing lexical network, WordNet. WordNet is a semantic network that organizes over 100,000 concepts of the real world with links between them. We show how to extend WordNet with the concepts of the specific domain of sound effects. We review some of the taxonomies to acoustically describe sounds. Mining legacy metadata from sound effects libraries further supplies us with terms. The extended semantic network includes the semantic, perceptual, and sound effects specific terms in an unambiguous way. We show the usefulness of the approach easing the task for the librarian and providing higher control on the search and retrieval for the user.

- **Cano, P.** Koppenberger, M. Ferradans, S. Martinez, A. Gouyon, F. Sandvold, V. Tarasov, V. Wack, N. 2004. 'MTG-DB: A Repository for Music Audio Processing' Proceedings of 4th International Conference on Web Delivering of Music; Barcelona, Spain

  **Relevant to:** Chapter 4

  Content-based audio processing researchers need audio and its related metadata to develop and test algorithms. The MTGDB is common repository of audio, metadata, ontologies and algorithms. The project includes hardware implementation, in the form of massive storage and computation cluster, the software and databases design and the ontology management. The repository, as far as copyright licenses allow, is open to researchers outside out lab to test and evaluate their algorithms.

- **Cano, P.** Koppenberger, M. Le Groux, S. Ricard, J. Wack, N. 2004. 'Knowledge and Perceptual Sound Effects Asset Management' Proceedings of 1st International Conference on E-business and Telecommunication Networks; Setubal, Portugal

  **Relevant to:** Chapter 3

  Sound producers create the sound that goes along the image in cinema and video productions, as well as spots and documentaries. Some sounds are recorded for the occasion. Many occasions, however, require the engineer to have access to massive libraries of music and sound effects. Of the three major facets of audio in post-production: music, speech and sound effects, this document focuses on sound effects (Sound FX or SFX). Main professional on-line sound-fx providers offer their collections using standard text-retrieval technologies. Library construction is an error-prone and labor consuming task. Moreover, the ambiguity and informality of natural languages affects the quality of the search. The use of ontologies alleviates some of the ambiguity problems inherent to natural languages, yet it is very complicated to devise and maintain an ontology that account for the level of detail needed in a production-size sound effect management system. To address this problem we use WordNet, an ontology that organizes over 100,000 concepts of real world knowledge: e.g: it relates doors to locks, to wood and to the actions of opening, closing or knocking. However a fundamental issue remains: sounds without caption are invisible to the users. Content-based audio tools offer perceptual ways of navigating the audio collections, like "find similar sound", even if unlabeled, or query-by-example, possibly restricting the search to a semantic subspace, such as "vehicles". The proposed content-based technologies also allow semi-automatic sound annotation. We describe the integration of semantically-enhanced management of metadata using WordNet together with content-based methods in a commercial sound effect management system.

- **Cano, P.** Fabig, L. Gouyon, F. Koppenberger, M. Loscos, A. Barbosa, A. 2004. 'Semi-Automatic Ambiance Generation' Proceedings of 7th International Conference on Digital Audio Effects; Naples, Italy

  **Relevant to:** Chapter 3

  Ambiances are background recordings used in audiovisual productions to make listeners feel they are in places like a pub or a farm. Accessing to commercially available atmosphere libraries is a convenient alternative to sending teams to record ambiances yet they limit the creation in different ways. First, they are already mixed, which reduces the flexibility to add, remove individual sounds or change its panning. Secondly, the number of ambient libraries is limited. We propose a semi-automatic system for ambiance generation. The system creates ambiances on demand given text queries by fetching relevant sounds from a large sound effect database and importing them into a sequencer multi track project. Ambiances of diverse nature can be created easily. Several controls are provided to the users to refine the type of samples and the sound arrangement.

- **Cano, P.** Koppenberger, M. Le Groux, S. Ricard, J. Herrera, P. Wack, N. 2004. 'Nearest-neighbor generic sound classification with a WordNet-based taxonomy' Proceedings of AES 116th Convention; Berlin, Germany

**Relevant to:** Chapter 3

Audio classification methods work well when fine-tuned to reduced domains, such as musical instrument classification or simplified sound effects taxonomies. Classification methods cannot currently offer the detail needed in general sound recognition. A real-world-sound recognition tool would require a taxonomy that represents the real world and thousands of classifiers, each specialized in distinguishing little details. To tackle the taxonomy definition problem we use WordNet, a semantic network that organizes real world knowledge. In order to overcome the second problem, that is the need of a huge number of classifiers to distinguish a huge number of sound classes, we use a nearest-neighbor classifier with a database of isolated sounds unambiguously linked to WordNet concepts.

- **Cano, P.** Koppenberger, M. Le Groux, S. Ricard, J. Wack, N. 2004. 'Semantic and Perceptual Management of Sound Effects in Production Systems' Proceedings of International Broadcasting Conference; Amsterdam, The Netherlands

  **Relevant to:** Chapter 3

  Main professional sound effects (SFX) providers offer their collections using standard text-retrieval technologies. SFX cataloging is an error-prone and labor consuming task. The vagueness of the query specification, normally one or two words, together with the ambiguity and informality of natural languages affects the quality of the search: Some relevant sounds are not retrieved and some irrelevant ones are presented to the user. The use of ontologies alleviates some of the ambiguity problems inherent to natural languages, yet they pose others. It is very complicated to devise and maintain an ontology that account for the level of detail needed in a production-size sound effect management system. To address this problem we use WordNet, an ontology that organizes real world knowledge: e.g.: it relates doors to locks, to wood and to the actions of knocking. However a fundamental issue remains: sounds without caption are invisible to the users. Content-based audio tools offer perceptual ways of navigating the audio collections, like "find similar sounds", even if unlabeled, or query-by-example. We describe the integration of semantically-enhanced management of metadata using WordNet together with content-based methods in a commercial sound effect management system.

- **Cano, P.** Koppenberger, M. 2004. 'Automatic sound annotation' Proceedings of 14th IEEE workshop on Machine Learning for Signal Processing; São Luís, Brazil

  **Relevant to:** Chapter 3

  Sound engineers need to access vast collections of sound effects for their film and video productions. Sound effects providers rely on text-retrieval techniques to offer their collections. Currently, annotation of audio content is done manually, which is an arduous task. Automatic annotation methods, normally fine-tuned to reduced domains such as musical instruments or reduced sound effects taxonomies, are not mature enough for labeling with great detail any possible sound. A general sound recognition tool would require: first, a taxonomy that represents the world and, second, thousands of classifiers, each specialized in distinguishing little de- tails. We report experimental results on a general sound annotator. To tackle the taxonomy definition problem we use WordNet, a semantic network that organizes real world knowledge. In order to overcome the need of a huge number of classifiers to distinguish many different sound classes, we use a nearest-neighbor classifier with a database of isolated sounds unambiguously linked to Word- Net concepts. A 30% concept prediction is achieved on a database of over 50,000 sounds and over 1,600 concepts.

- **Cano, P.** Koppenberger, M. 2004. 'The emergence of complex network patterns in music networks' Proceedings of Fifth International Conference on Music Information Retrieval; Barcelona

  **Relevant to:** Chapter 4

  Viewing biological, social or technological systems as networks formed by nodes and connections between them can help better understand them. We study the topology of several music networks, namely citation in allmusic.com and co-occurrence of artists in playlists. The analysis uncovers the emergence of complex network phenomena in music information networks built considering artists as nodes and its relations as links. The properties provide some hints on searchability and possible optimizations in the design of music recommendation systems. It may also provide a deeper understanding on the similarity measures that can be derived from existing music knowledge sources.

- Batlle, E. Masip, J. **Cano, P.** 2004. 'Scalability issues in HMM-based Audio Fingerprinting' Proceedings of IEEE International Conference on Multimedia and Expo; Taipei, Taiwan **Relevant to:** Chapter 2

- Batlle, E. Masip, J. **Cano, P.** 2003. 'System analysis and performance tuning for broadcast audio fingerprinting' Proceedings of 6th International Conference on Digital Audio Effects; London, UK **Relevant to:** Chapter 2 An audio fingerprint is a content-based compact signature that summarizes an audio recording. Audio Fingerprinting technologies have recently attracted attention since they allow the monitoring of audio independently of its format and without the need of meta-data or watermark embedding. These technologies need to face channel robustness as well as system accuracy and scalability to succeed on real audio broadcasting environments. This paper presents a complete audio fingerprinting system for audio broadcasting monitoring that satisfies the above system requirements. The system performance is enhanced with four proposals that required detailed analysis of the system blocks as well as extense system tuning experiments.

- **Cano, P.** Batlle, E. Kalker, T. Haitsma, J. 2002. 'A Review of Algorithms for Audio Fingerprinting' Proceedings of 2002 IEEE International Workshop on Multimedia Signal Processing; St. Thomas, Virgin Islands

  **Relevant to:** Chapter 2

  An audio fingerprint is a content-based compact signature that summarizes an audio recording. Audio Fingerprinting technologies have recently attracted attention since they allow the monitoring of audio independently of its format and without the need of meta-data or watermark embedding. The different approaches to fingerprinting are usually described with different rationales and terminology depending on the background: Pattern matching, Multimedia (Music) Information Retrieval or Cryptography (Robust Hashing). In this paper, we review different techniques mapping functional parts to blocks of a unified framework.

- Gómez, E. **Cano, P.** Gomes, L. Batlle, E. Bonnet, M. 2002. 'Mixed Watermarking-Fingerprinting Approach for Integrity Verification of Audio Recordings' Proceedings of IEEE International Telecommunications Symposium; Natal, Brazil

  **Relevant to:** Chapter 3

  We introduce a method for audio-integrity verification based on a combination of watermarking and fingerprinting. The fingerprint is a sequence of symbols ("audio descriptor units") that enables one to identify an audio signal. Integrity verification is performed by embedding the fingerprint into the audio signal itself by means of a watermark. The original fingerprint is reconstructed from the watermark and compared with a new fingerprint extracted from the watermarked signal. If they are identical, the signal has not been modified; if not, the system is able to determine the approximate locations where the signal has been corrupted.

- **Cano, P.** Kaltenbrunner, M. Gouyon, F. Batlle, E. 2002. 'On the use of Fastmap for audio information retrieval and browsing' Proceedings of ISMIR 2002 - 3rd International Conference on Music Information Retrieval; Ircam - Centre Pompidou, Paris, France

  **Relevant to:** Chapter 2,3

  In this article, a heuristic version of Multidimensional Scaling (MDS) named FastMap is used for audio retrieval and browsing. FastMap, like MDS, maps objects into an Euclidean space, such that similarities are preserved. In addition of being more efficient than MDS it allows query-by-example type of query, which makes it suitable for a content-based retrieval purposes.

- **Cano, P.** Batlle, E. Mayer, H. Neuschmied, H. 2002. 'Robust Sound Modeling for Song Detection in Broadcast Audio' Proceedings of 112th AES Convention, 2002; Munich, Germany

  **Relevant to:** Chapter 3

  This paper describes the development of an audio fingerprint called AudioDNA designed to be robust against several distortions including those related to radio broadcasting. A complete system, covering also a fast and efficient method for comparing observed fingerprints against a huge database with reference fingerprints is described. The promising results achieved with the first prototype system observing music titles as well as commercials are presented.

- Gouyon, F. Herrera, P. **Cano, P.** 2002. 'Pulse-dependent analysis of percussive music' Proceedings of AES22 International Conference on Virtual, Synthetic and Entertainment Audio; Espoo, Finland

  **Relevant to:** Chapter 1,4

  With the increase of digital audio dissemination, generated by the popularization of personal computers and worldwide low-latency networks, many entertaining applications can easily be imagined to rhythmic analysis of audio. We herein report on a method of automatic extraction of a rhythmic attribute from percussive music audio signals: the smallest rhythmic pulse, called the *tick*. Evaluations of the proposed scheme yielded quite good results. We then discuss the relevance of use of the tick as the basic feature of rhythmic analysis.

- **Cano, P.** Gómez, E. Batlle, E. Gomes, L. Bonnet, M. 2002. 'Audio Fingerprinting: Concepts and Applications' Proceedings of 2002 International Conference on Fuzzy Systems Knowledge Discovery; Singapore

  **Relevant to:** Chapter 2

  An audio fingerprint is a compact digest derived from perceptually relevant aspects of a recording. Fingerprinting technologies allow monitoring of audio content without the need of meta-data or watermark embedding. However, additional uses exist for audio fingerprinting and some are reviewed in this article.

- Bonada, J. Loscos, A. **Cano, P.** Serra, X. 2001. 'Spectral Approach to the Modeling of the Singing Voice' Proceedings of 111th AES Convention; New York, USA

  **Relevant to:** Chapter 1,4

  In this paper we will present an adaptation of the SMS (Spectral Modeling Synthesis) model for the case of the singing voice. SMS is a synthesis by analysis technique based on the decomposition of the sound into sinusoidal and residual components from which high-level spectral features can be extracted. We will detail how the original SMS model has been expanded due to the requirements of an impersonating applications and a voice synthesizer. The impersonating application can be described as a real-time system for morphing two voices in the context of a karaoke application. The singing synthesis application we have developed generates a performance of an artificial singer out of the musical score and the phonetic transcription of a song. These two applications have been implemented as software to run on the PC platform and can be used to illustrate the results of all the modifications done to the initial SMS spectral model for the singing voice case.

- Batlle, E. **Cano, P.** 2000. 'Automatic Segmentation for Music Classification using Competitive Hidden Markov Models', Proceedings of International Symposium on Music Information Retrieval; Plymouth, MA (USA)

  **Relevant to:** Chapter 2,3

  Music information retrieval has become a major topic in the last few years and we can find a wide range of applications that use it. For this reason, audio databases start growing in size as more and more digital audio resources have become available. However, the usefulness of an audio database relies not only on its size but also on its organization and structure. Therefore, much effort must be spent in the labeling process whose complexity grows with database size and diversity.

  In this paper we introduce a new audio classification tool and we use its properties to develop an automatic system to segment audio material in a fully unsupervised way. The audio segments obtained with this process are automatically labeled in a way that two segments with similar psychoacoustics properties get the same label. By doing so, the audio signal is automatically segmented into a sequence of abstract acoustic events. This is specially useful to classify huge multimedia databases where a human driven segmentation is not practicable. This automatic classification allow a fast indexing and retrieval of audio fragments. This audio segmentation is done using competitive hidden Markov models as the main classification engine and, thus, no previous classified or hand-labeled data is needed. This powerful classification tool also has a great flexibility and offers the possibility to customize the matching criterion as well as the average segment length according to the application needs.

- Loscos, A. **Cano, P.** Bonada, J. 1999. 'Low-Delay Singing Voice Alignment to Text' Proceedings of International Computer Music Conference 1999; Beijing, China

  **Relevant to:** Chapter 1,2

  In this paper we present some ideas and preliminary results on how to move phoneme recognition techniques from speech to the singing voice to solve the low-delay alignment problem. The work focus mainly on searching the most appropriate Hidden Markov Model (HMM) architecture and suitable input features for the singing voice, and reducing the delay of the phonetic aligner without reducing its accuracy.

- **Cano, P.** Loscos, A. Bonada, J. 1999. 'Score-Performance Matching using HMMs' Proceedings of International Computer Music Conference 1999; Beijing, China

  In this paper we will describe an implementation of a score-performance matching, capable of score following, based on a stochastic approach using Hidden Markov Models.

  **Relevant to:** Chapter 1,4

- **Cano, P.** 1998. 'Fundamental Frequency Estimation in the SMS analysis' Proceedings of COST G6 Conference on Digital Audio Effects 1998; Barcelona

  **Relevant to:** Chapter 1,2,3

  This paper deals with the fundamental frequency estimation for monophonic sounds in the SMS analysis environment. The importance of the fundamental frequency as well as some uses in SMS is commented. The particular method of F0 estimation based on a two-way mismatched measure is described as well as some modifications. Finally we explain how pitch-unpitched decision is performed.

## A.4 Technical Reports

- **Cano, P.** Gómez, E. Gouyon, F. Herrera, P. Koppenberger, M. Ong, B. Serra, X. Streich, S. Wack, N. 2006. 'ISMIR 2004 Audio Description Contest' MTG Technical Report: MTG-TR-2006-02

**Relevant to:** Chapter 4

In this paper we report on the ISMIR 2004 Audio Description Contest. We first detail the contest organization, evaluation metrics, data and infrastructure. We then provide the details and results of each contest in turn. Published papers and algorithm source codes are given when originally available. We finally discuss some aspects of these contests and propose ways to organize future, improved, audio description contests.