Using MPEG-7 Audio Fingerprinting in Real-World Applications

Oliver Hellmuth[1], Eric Allamance[1], Markus Cremer[2], Holger Grossmann[2],
Jürgen Herre[1], Thorsten Kastner[1]

[1] *Fraunhofer Institute for Integrated Circuits IIS, Erlangen, Germany*

[2] *Fraunhofer Institute for Integrated Circuits IIS, AEMT, Ilmenau, Germany*

Correspondence should be addressed to Oliver Hellmuth (`hel@iis.fhg.de`)

ABSTRACT

Finalized in 2001, the MPEG-7 Audio standard provides a universal toolbox for the content-based description of audio material. While the descriptive elements defined in this standard may be used for many purposes, audio fingerprinting (i.e. automatic identification of audio content) was already among the initial set of target applications that were conceived during the design of the standard. This paper reviews the basics of MPEG-7 based audio fingerprinting and explains how the technology has been used in a number of real-world applications, including metadata search engines, database maintenance, broadcast monitoring and audio identification on embedded systems. Appropriate selection of fingerprinting parameters and performance numbers are discussed.

1. INTRODUCTION

A number of specifications have emerged recently aiming at defining a unified interface for description and characterization of multimedia content. Such descriptive data is usually referred to as *metadata* and facilitates efficient content handling by describing its formal, structural and semantic aspects. Among these metadata formats, the recent ISO/MPEG-7 standard [1] probably features the most comprehensive and sophisticated set of tools for multimedia content description. With this specification, the MPEG standards group (ISO/IEC JTC1/SC29/WG11) has extended its traditional scope beyond audiovisual source coding towards a more holistic view of content representation. Specifically, the audio part of MPEG-7 [2][3] provides a number of elements to describe signal-derived low-level features of audio signals and thus serves as a foundation for many conceivable Music Information Retrieval (MIR) applications. It consists of a toolbox of universally usable Low Level Descriptors (LLDs) which contains many well-known features. This basic layer is complemented by a number of application-oriented Description Schemes (DSs) which define content descriptions for specific purposes (e.g. recognition of sound effects, spoken content, and instrument timbre). Among this first set of applications which have initially been conceived at the time of writing the standard, the MPEG-7 Audio provisions for audio fingerprinting are especially attractive as a universal tool for finding metadata for unknown, unlabelled audio content. This paper reviews the basics of MPEG-7 Audio fingerprinting and describes how the underlying technology has been put to work in the context of real-world applications. It discusses important practical issues, such as the appropriate selection of fingerprinting parameters and characterizes the performance achieved in the context of these applications.

2. MPEG-7 AUDIO FINGERPRINTING

In contrast to other metadata standardization efforts, MPEG-7 covers a wide framework, where both high level semantic concepts as well as low level features are contained. The latter can in most cases be extracted directly from the multimedia signal itself, thus allowing numerous possibilities for mechanized processing of large amounts of data.

The basic descriptive entities in MPEG-7 are called Descriptors (Ds) and represent specific content properties or attributes by means of a defined syntax and semantics. Description Schemes (DSs) are intended to combine components with view towards application and may comprise both Descriptors and other Description Schemes. These two entities are syntactically defined by a so-called Description Definition Language (DDL) which also provides the ability for future extension/modification of existing elements. The MPEG-7 DDL [4] is based on XML Schema as the language of choice for the textual representation of content description and for allowing extensibility of description tools.

The audio subpart of the MPEG-7 standard essentially comprises two types of structural elements that support the functionality of robust audio fingerprinting. Firstly, the AudioSpectrumFlatness Low Level Descriptor (LLD) defines the procedure for extracting a feature from the audio signal that proves to be robust against common audio signal distortions as well as very compact with regards to the amount of data necessary to uniquely identify an audio item. Specifically, the feature describes the signal's Spectral Flatness within a number of frequency subbands, which can be interpreted as "tone-likeness vs. noise-likeness" [5]. Secondly, the AudioSignature Description Scheme (DS) is provided, which describes how to instantiate the AudioSpectrumFlatness LLD in a way that an interoperable hierarchy of scalable fingerprints can be established with regard to the following parameters:

- **Temporal scope:** The temporal scope of the fingerprint represents a first degree of freedom and relates to the start position and the length of the audio item for which the feature extraction is carried out. The signal segment used for the fingerprint generation can be chosen freely and depends on the type of the envisaged application. The length of the captured audio item can be decreased to only a few seconds of audio at any random offset into an audio item.
- **Temporal resolution:** The temporal resolution (MPEG-7: "Scaling Ratio") of the fingerprint is an important parameter which - unlike in other systems - can be used to control the trade-off between fingerprint compactness

and its descriptive power (recognition strength and robustness). This is achieved by grouping a variable number (in steps of two to the power of n) of single Spectral Flatness Measure (SFM) vectors over time, and computing a statistical data summarization within these groups. For example, the smallest temporal segment or frame is 30ms long, and thus a grouping of 32 would result in an overall temporal granularity of approximately 1s.

- **Spectral coverage / bandwidth:** Another degree of freedom regarding the scalability of MPEG-7 Audio fingerprints resides in its spectral coverage / bandwidth. In order to introduce a degree of scalability across frequency, the number of frequency bands above a fixed base frequency (250Hz) can be selected as a further parameter of scalability.

In addition to the scaling possibilities enumerated above, another interesting aspect of reducing the size of a fingerprint resides in the numeric representation of feature values. A first obvious choice is to use a floating point representation, such as the IEEE-754 floating point format. However, though not fully specified in the MPEG-7 standard yet, other coded representations are conceivable. Further analysis of the sensitivity of the relevant values to quantization demonstrated that an appropriate 8 bit representation (rather than 32 bits consumed by the floating point representation) and even less yield unimpaired recognition performance. While this type of representation scalability is not yet covered in the recent version of the MPEG-7 standard, it carries a high potential for further data rate reduction and is currently prepared for inclusion into an upcoming extension for the binary representation of metadata (BiM). Some investigations on the performance of quantized features can be found in [6].

The application scenarios for audio fingerprinting described later in this paper benefit through using an MPEG-7 Audio standardized fingerprint in two ways:

- **Scalability:** Temporal scope, temporal resolution and spectral coverage can be adapted to the robustness and storage requirements necessary for the given application scenario. In standard setups, the system will often be exposed to relatively high quality audio at the input. Therefore the "robustness" vs. "fingerprint size" tradeoff can be moved towards smaller fingerprints (=less robustness), i.e. one can store a larger database of fingerprints on a matching server and the classification process becomes faster. The scalable nature of the MPEG-7 Audio fingerprint enables such a reuse of a given fingerprint in different (lower) resolutions.
- **Interoperability:** Due to the interoperability feature of the MPEG-7 standard, fingerprints can be created by one company and used for identification purpose by another one. This makes it much easier to create a comprehensive reference database which is required for some application scenarios. In the near future it is likely that music labels not only produce music for different types of media but also provide the corresponding MPEG-7 Audio fingerprint.

3. SYSTEM DESCRIPTION

Since the scope of the MPEG-7 standard is limited to the definition of interoperable descriptive entities (metadata) rather than describing their usage in actual applications, one more step is required to arrive at a real system. In fact, there are numerous conceivable applications for the set of descriptors and description schemes defined within the MPEG-7 standard which may go substantially beyond the applications that were conceived at the time of the writing of the specification.

For the implementation of an audio identification engine, there are two core components that are common to most systems (see Figure 1). In the first component the audio signal is analyzed and the AudioSignature Descriptor / fingerprint is calculated from the audio signal. This is referred to as the extraction process. In order to define the syntax and semantics of each MPEG-7 descriptor, its extraction process is quite closely defined for most descriptors in order to guarantee interoperability accross different systems. This leaves only limited options for alternative algorithms, for example in the preprocessing of the audio data which may differ among different applications.

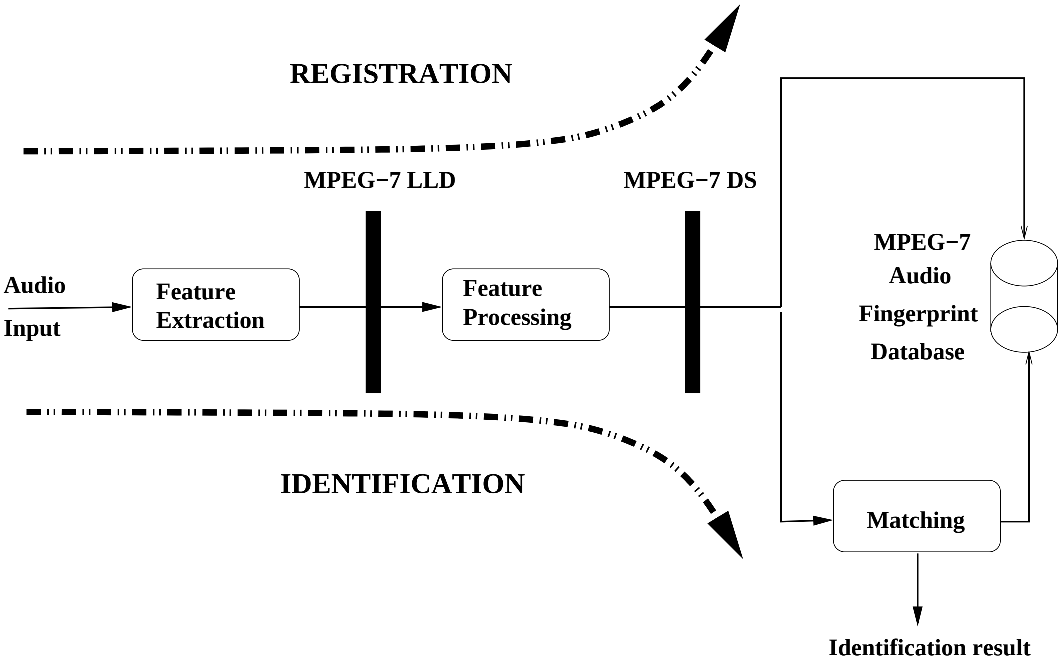

Fig. 1: Audio identification system based on MPEG-7 Audio (principle)

In the second component, the fingerprint is compared to a database of fingerprints, which have already been registered previously. While this part is usually called a classifier in a general pattern recognition context, it will be refered to as a matching engine to reflect the fact that the system is looking for a close match between the query and the reference fingerprints included in the database. The registration of a new fingerprint basically involves the linking of the fingerprint data to relevant metadata, such as artist, album, and track title in a database. This linking is usually done by means of a so called track unique identifier (TUID) which is a number that serves as an index into the metadata database and identifies the metadata set associated with an audio item and its fingerprint. Hence the matching engine is queried with the fingerprint data of an unknown item and returns the TUID of the most likely match together with some measure of confidence for this result. This confidence measure has to provide a good balance between a false positive recognition (i.e. recognition of an unregistered item) and a false rejection of registered items, e.g. due to heavy signal distortions. In general, the discrimination between the individual database items becomes increasingly difficult as the query signal undergoes more and more signal alteration.

The availability of a good measure of confidence is especially important when legal rights and royalty issues are involved, as is the case with some of the applications detailed below. Last but not least, the content of the reference database plays an important role for most applications. While the number of included audio items is often less significant than one might assume, the availability of the items that are queried most frequently is of crucial importance. For many services, the actually queried audio items form only a very small subset of the available published works, suggesting that, for instance, for broadcast monitoring a reasonable service can already be offered with a database of a rather limited size.

4. APPLICATIONS

In this section, several application scenarios of audio fingerprinting are described. Special attention shall be drawn to the way how MPEG-7 is involved in each scenario. Essential fingerpring parameters and achieved performance figures based on an MPEG-7 Audio fingerprinting system are given.

4.1. Searching for related metadata

Today a lot of legacy audio content exists without any link to associated metadata. A huge number of CDs are sold with the minimum amount of metadata possible: artist and title. On the Internet, on a consumer PC or on “homemade” CD-Rs often not even that is available. If the consumer is lucky he has access to a meaningful filename or – in case of a CD-R – to a correct “title-artist” notation somewhere on the CD-R case. If not, how can the consumer gain access to a minimum set of metadata (= artist, title) or – even better – additional information e.g. lyrics, tour dates?

In the case of audio data, a number of solutions for attaching metadata, have been proposed. Examples are the CD-Text formats for CD media or the ID3 tags for MP3 compressed audio files [7][8]. In the general case, however, descriptive information is not necessarily stored together with the content on the same physical medium and thus reliable association / linking of metadata to the audiovisual content often becomes a challenge.

Using an MPEG-7 based fingerprinting technique provides an easy solution to uniquely link the plain audio content to related metadata. Without the existence of any kind of “labelling” (e.g. watermarking) of the audio content itself, a consumer is able to extract a unique identifier from any piece of audio. By means of this fingerprint it is possible to search for related metadata in any kind of database that supports access based on MPEG-7 Audio fingerprints.

A possible scenario depicted in Figure 2 could look like this: A consumer wants to identify an unknown piece of music. He is also interested in some addional information about the artist. From the metadata service provider the user receives an MPEG-7 Audio compliant fingerprint extraction and query tool. Using this tool he extracts the fingerprint from the audio file and queries the service provider over the internet. Based on this unique identifier a database look-up is conducted. If the search is not successful the provider might contact a partner company with a larger archive and obtain the requested information from there. Due to the standardized MPEG-7 Audio fingerprint format, interoperability is guaranteed: The same MPEG-7 Audio fingerprint can be used for querying at the other company. After a successful database look-up the relevant metadata is delivered back to the user.

Parameters and Performance:

The following example configuration describes real parameters and performance of an MPEG-7 based internet audio identification and metadata service scenario. The backend MPEG-7 Audio fingerprint database of such a service may consist of a cluster of 14 standard Intel P4 servers (2 GHz) with a reference fingerprint size of 1 million items. The resulting classification time of this setup is 0.09 seconds per request. Assumptions: The client application on the user's PC extracts the fingerprint with a length of 10 seconds at certain position in the audio file (e.g. at 60 seconds). Therefore only a small part around that position within the reference fingerprints has to be covered (e.g. starting at 30 seconds until 90 seconds). A two stage classification setup is used, as described in [9]. The temporal resolution is 128 for the first and 8 for the second stage.

4.2. Database Clean-Up

Having gathered a lot of audio-related metadata in a large database, from time to time maintenance work like cleaning up the database is necessary. An Audio Fingerprinting system can be used to check for duplicates or incorrect associated metadata. In the following, solutions for removing duplicates and finding incorrectly annotated metadata will be described.

By building up or updating an audio database, it is almost inevitable that the database will contain multiple files of the same content. To perform a self check of a database, i.e. to compare every item in the database to the remaininig items, MPEG-7 Audio fingerprinting can be used. For every audio item, the MPEG-7 Audio fingerprint has to be extracted and will be compared to the rest of the extracted fingerprints from the database. A result list from the matching is obtained for every item, describing the most similar items and therefore also the duplicates in a ranked order. To remove the duplicate items from the database in a final step, it is only necessary to analyze the result lists from the matching which can be done in a fast and efficient manner.

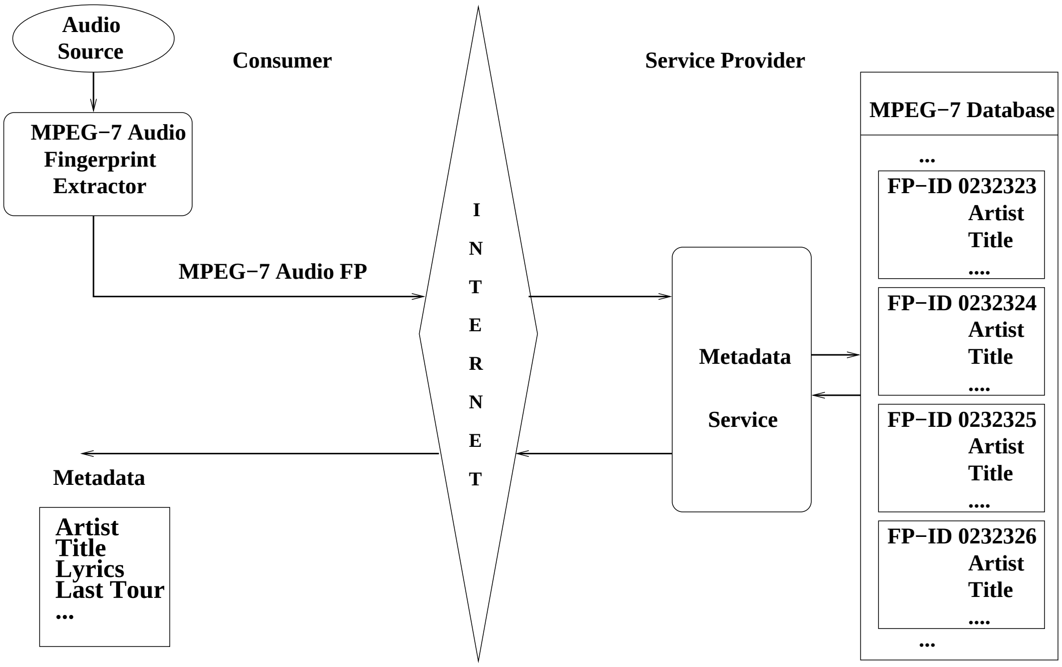
Audio Source
Consumer
MPEG−7 Audio Fingerprint Extractor
MPEG−7 Audio FP
I N T E R N E T
Service Provider
MPEG−7 Database
...
FP−ID 0232323 Artist Title ....
FP−ID 0232324 Artist Title ....
FP−ID 0232325 Artist Title ....
FP−ID 0232326 Artist Title ....
...
Metadata Service
Metadata
Artist
Title
Lyrics
Last Tour
...

Fig. 2: Metadata retrieval based on MPEG-7 Audio

For administrating a large audio or fingerprint database, it is also essential to have a possibility to verify the metadata associated to the corresponding audio file or fingerprint. One way to find incorrect metadata in a database is to compare this database with others. On one hand, the MPEG-7 Audio fingerprint enables, as mentioned before, a fast and efficient way to compare different fingerprints and audio items and is in consequence the instrument to verify the annotated metadata. The MPEG-7 standard on the other hand guarantees the worldwide interoperability of the fingerprint format which allows sharing and trading of fingerprint databases and ensures the availability of databases containing compatible fingerprints to compare with. In practice, the verification of annotated metadata can e.g. be done in two steps: First, search for at least two more entries from the same audio item in different databases by comparing the fingerprints. Then compare the annotated metadata of the fingerprints and make a simple consistency check wether to keep the annotated metadata or to update it.

Parameters and Performance:
As a proof of concept, performance measurements were carried out using a fingerprint database of ca. 1 million items. To decrease the matching time needed per item, a two stage algorithm was implemented. During the first stage a fast fuzzy presearch is done on a coarsely scaled fingerprint database using a temporal scaling of 128 and 8 frequency bands resulting in a list of candidates for a subsequent refinement search. The accurate classification is done during the second stage on a finer scaled fingerprint considering only these similar items. A temporal scaling of 8 was used for this second stage and 8 frequency bands. Performance results were carried out using a cluster of 14 Pentium 4 (2.0GHz) machines. The overall time needed for classifying one million fingerprints was about 4.5 days. The analysis of the result files and the creation of a list with duplicate items took about 3 hours.

4.3. Broadcast Monitoring

Broadcast monitoring is another important application scenario for MPEG-7 based fingerprints. The system described here is capable of handling several independent audio input streams in parallel and produces a playlist-like report for further analysis. The following section always assumes "broadcast monitoring" to be based on *Audio* signals. In the case of monitoring *Audio/Video* signals, the Audio part can be utilized for identification of the whole audiovisual signal.

The purpose of broadcast monitoring is to reliably certify that a transmission over a certain medium took place. Additionally it is often desirable to determine the time of transmission of the audio item, its duration and the received quality. In most cases this verification happens at the receiver side to make sure that the recorded data reflects the material the listener (viewer) actually received.

The field of broadcast monitoring was and will be an extremly important measuring instrument for different parties involved in music production, broadcasting, advertising and music consumption. Automatic broadcast monitoring of arbitrary media channels permits artists and music publishers to determine exact figures on how often a certain title was played. Today this is still frequently done by human listening in spot tests. Based on the data gathered the real market penetration is estimated. In consequence smaller (not well known) artists may earn a smaller share of the royalities than they would deserve because they were not represented adequately in the spot test. For advertising agencies a reliable check if and when commercials were broadcasted is indispensable as well. Independent broadcast monitoring based on fingerprints extracted from the audio part of commercials can easily be utilized for that verification. For the same reason broadcasters are interested in a reliable log file of their transmission schedule as well. This would be an independent proof that the commercial was delivered to the consumer under the predefined conditions. Not only single evaluations if and when a single piece of music was played are interesting, but also statistical calculations which aim at a broader view of the entire market. Artists, music publisher, broadcaster and even music consumers make decisions based on that statistical data. Using a fingerprint based automatic broadcast monitoring solution can produce more reliable data not only for a "mainstream" analysis (like Top10,Top100,..), but also for more complex questions from the data mining research field. An interesting question may be "What is the most popular music when driving home from work?" An answer to this question may be found through a statistical analysis of several broadcast monitoring log files of the most popular radio stations at the relevant time frame.

Figure 3 shows a broadcast monitoring system based on MPEG-7 Audio fingerprints. The input consists of a configurable multi-channel *MPEG-7 Audio Fingerprint Extractor*. A unique channel ID and a timestamp is attached to the fingerprint and sent to the *Matching Server*. After the excerpt is identified, this intermediate result is passed on to the *Postprocessing Unit*. To lower the transfer overhead within the system, an internal index which refers to the MPEG-7 Audio fingerprint is used ("Track Unique ID" (TUID)). In the *Postprocessing Unit*, a special algorithm tries to combine all related excerpts, examines different hypotheses and picks the most likely candidate for the next playlist entry. The playlist is then enhanced with the desired metadata (e.g. artist, title) and written into a log file.

Parameters and Performance:
A standard "high resolution" setup could consists of the following parameters. The excerpt length of the small audio segments is 6 seconds with an overlap of 3 seconds. The temporal resolution is set to 8. This setup is capable of detecting start and end times for each entry with an accuracy of +/- 1.5 seconds. With a minimum length of 6 seconds for an audio item to be detectable this setup is also capable of monitoring short commercials. If these design parameters are not accurate enough, it is possible to use a higher resolution at the expense of computing time for classification. For a "low resolution" setup, one could use an excerpt length of 30 to 60 seconds (without overlap) and a temporal resolution of 256. This setup is basically designed to identify *whether* a particular item was played without an exact start or end time (+/- 30 seconds). The advantage is a fast classification with much lower memory and storage consumption for the reference fingerprints.

4.4. Audio Identification on Embedded Devices

The increasing processing power and storage size available on portable devices, like e.g. MP3-players, enable new applications beyond decoding compressed audio bitstreams. These resources could be partially used to perform an identification of unla-

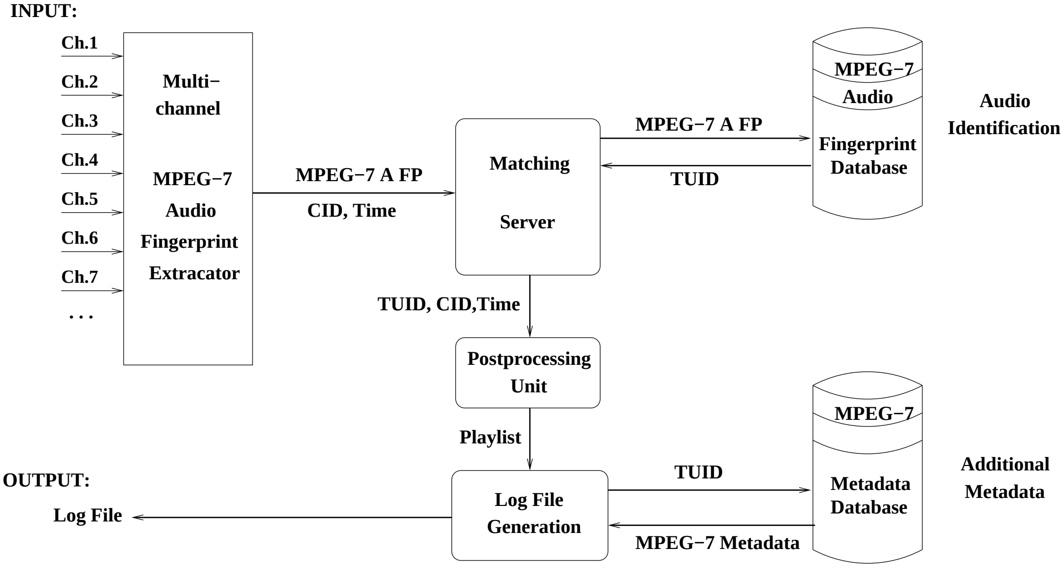

Fig. 3: Broadcast monitoring based on MPEG-7 Audio fingerprints (MPEG-7 A FP)

beled songs or, if a microphone or a line input is provided, to identify audio signals captured by these inputs. Porting a fingerprinting system to embedded devices is not restricted to those player devices. One could imagine common appliances, such as kitchen radios or car-stereos, to be equipped with such a system and thus providing a song identification feature.

In the case that the device has some connectivity capabilities, like the Internet or a GSM channel, the audio identification system will amount to a fingerprint extractor. An extracted fingerprint can then be submitted immediately to a remote server or, if no connection is available at that time, stored on the device for a delayed querying. On the other hand, when such connections cannot be set up regularly, the identification process has to take place on the device itself. Thus, the challenge lies in adapting the matching algorithm and the associated fingerprint database in order to cope with the available resources on the device. Additionally, since the audio identification will not be the primary device functionality but rather some added value, only a small percentage of the resources will be allocated to this task. Furthermore, some means must also be provided for updating the database, but this topic is beyond the scope of this paper and not addressed here.

As described in Section 3, the audio identification system consists of two main entities, namely the extractor and the classifier. The first task is computationally inexpensive and thus can be integrated easily into an embedded environment even on limited integer processing units, like 16 or 32 bit CPUs. The required memory resources are also very moderate consisting only in a few buffers holding the segments of the audio signal and the resulting compact fingerprint. On the other hand, the classifier is more resource demanding by orders of magnitude. Since the entire fingerprint database must be accessible for the matching algorithm, the memory limitations for all storage devices (RAM and hard disk) will dictate the size of the database. This size is roughly linearly dependent on the number of fingerprints, assuming that they have the same size on average. Therefore, the average fingerprint size is a crucial issue in porting the audio identification system to embedded devices and should remain as small as possible. To meet these size requirements, four distinct parameters can be modified independently.

Parameters and Performance:

- Temporal scope: If it can be assumed that the songs to be identifed will be presentend in almost their full length, apart from the begining and the end which are likely to be faded, then it is sufficient to store a fingerprint taken only from an excerpt of a few seconds. To reduce the risk of false negatives (at the expense of increased memory space), additional fingerprints can be extracted at different time locations from the same song.

- Temporal resolution: Typical resolution values are around 64 temporal windows of $30ms$ each. Since this downscaling process is associated with a loss of information, the recognition performance may be affected.

- Spectral coverage: A typical number of bands on embedded systems is 8, covering the frequency range from 250Hz to 1kHz respectively. The resulting fingerprint size is directly proportional to this number.

- Data resolution: As discussed in [6], the audio fingerprint vectors can be quantized with a few bits without loss of recognition performance. Under certain circumstances it has been found that the quantization even improves the recognition rates when the elements were quantized from a 32 bit floating-point representation to an 8 bit integer. Reducing the quantization to 4 bits per component would further reduce the fingerprint size by 50%. A typical quantization level on embedded systems is 4 or 8 bit.

By modifying these parameters, high resolution fingerprints can be transcoded to reach a target size. This downsizing process has a direct impact on the robustness and, consequently, on the recognition performance of the system. Figures on the recognition rates depending on these parameters can be found in [9]. The amount and type of information to be discarded will be chosen depending on the type of distortion which is likely to be encountered, considering some worst case recognition rate scenarios. Nevertheless, a size of few hundred bytes per fingerprint are easily achievable while maintaining good recognition rates. The downscaled versions also have the nice property of still remaining compatible with the MPEG-7 Audio standard and thus maintaining the interoperability between systems which adhere to this standard.

Another consequence of these downscaling operations is a reduced computational cost for the matching algorithm. As was discussed in [9], reducing the temporal resolution by a factor of two increases the classification speed by four. In contrast, the frequency resolution has only a linear influence on the efficiency, as well as the data resolution in some cases. This decrease in the computational cost is necessary due to the limited capabilities of the used low cost processing units.

The MPEG-7 Audio identification system has already been succesfully ported to a *Personal Digital Assistant* (Compaq/HP iPAQ). The build-in microphone of this device allowed realtime capturing and identification from various external audio sources. The results were very promising and proved the feasibility of the presented concepts. More supported platforms will be available shortly.

5. CONCLUSIONS

The MPEG-7 Audio standard provides a generic framework for the descriptive annotation of audio data summarizing essential features of the signal. Two of these descriptive elements provide the ability to extract robust, compact-sized, and scalable descriptions from the audio signal which can be used as unique identifiers ("fingerprints") for the signal. The paper investigated practical issues and results for a set of applications of this fingerprinting technology. By virtue of its scalability, the technology proves to be applicable to a wide range of different applications with diverse requirements. More applications will arise over time.

6. REFERENCES

[1] ISO/IEC JTC1/SC29/WG11 (MPEG). Information technology - multimedia content description interface. International Standard 15938, ISO/IEC, 2001.

[2] ISO/IEC JTC1/SC29/WG11 (MPEG). Multimedia content description interface - part 4: Audio. International Standard 15938-4, ISO/IEC, 2001.

[3] Adam Lindsay and Jürgen Herre. MPEG-7 and MPEG-7 Audio: An Overview. *AES*, 49(7/8):589–594, July/August 2001.

[4] ISO/IEC JTC1/SC29/WG11 (MPEG). Multimedia content description interface - part 2: Description definition language. International Standard 15938-2, ISO/IEC, 2001.

[5] Jürgen Herre, Eric Allamanche, and Oliver Hellmuth. Robust matching of audio signals using spectral flatness features. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, 2001.

[6] Oliver Hellmuth, Eric Allamanche, Jürgen Herre, Thorsten Kastner, Markus Cremer, and Wolfgang Hirsch. Advanced Audio Identification using MPEG-7 Content Description. In *111th AES Convention*, New York, 2001. Preprint 5463.

[7] Red Book. Philips, Sony, May 1999.
http://www.licensing.philips.com/cdsystems.

[8] S. Hacker. *MP3: The Definitive Guide*. O'Reilly, 2000.

[9] Thorsten Kastner, Eric Allamanche, Jürgen Herre, Oliver Hellmuth, Markus Cremer, and Holger Grossmann. MPEG-7 Scalable Robust Audio Fingerprinting. In *112th AES Convention*, Munich, 2002. Preprint 5511.