AUDIO CONTENT TRANSMISSION

Xavier Amatriain, Perfecto Herrera

Audiovisual Institute
Pompeu Fabra University, Barcelona (Spain)
{xavier.amatriain,perfecto.herrera}@iua.upf.es

ABSTRACT

Content description has become a topic of interest for many researchers in the audiovisual field [1][2]. While manual annotation has been used for many years in different applications, the focus now is on finding automatic content-extraction and content-navigation tools. An increasing number of projects, in some of which we are actively involved, focus on the extraction of meaningful features from an audio signal. Meanwhile, standards like MPEG7 [3] are trying to find a convenient way of describing audiovisual content. Nevertheless, content description is usually thought of as an additional information stream attached to the 'actual content' and the only envisioned scenario is that of a search and retrieval framework.

However, in this article it will be argued that if there is a suitable content description, the actual content itself may no longer be needed and we can concentrate on transmitting only its description. Thus, the receiver should be able to interpret the information that, in the form of metadata, is available at its inputs, and synthesize new content relying only on this description. It is possibly in the music field where this last step has been further developed, and that fact allows us to think of such a transmission scheme being available on the near future.

1. Introduction

The model proposed is based on an analysis-synthesis process. Therefore, the only data involved in the transmission step will be the content description taking the form of metadata. A multilevel 'content tree' is proposed as an efficient content description representation. Several technologies are available for representing content description, but, taking into account our experience in MPEG-7's standardization process [4], we would encourage an XML-based metadata language such as MPEG-7's DDL.

The model here proposed is depicted in Figure 1. It is interesting enough to note that such a transmission model implies a redefinition of the schemes commonly used to model the communication act itself [5] as it can be seen as a step beyond Shannon and Weaver's traditional communication model [6] (see Figure 2). In our model, the stream to be transmitted is no longer seen as a stream of bits with no abstract meaning, information is an abstraction of the actual content, in other words, a 'stream of meaning'.

In this sense, noise is thought of as anything added to the original piece of information that is likely to change its meaning or make it difficult to understand. Thus, the traditional definition for noise as a change in the bitstream being transmitted would only fit our definition if the change is 'substantial' and can produce a change of meaning.

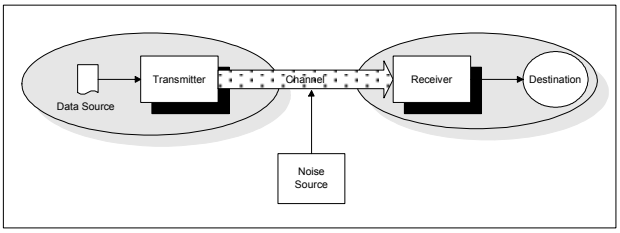

Figure 2. *S&W traditional transmission model*

In the next sections, we will particularize this idea to the case of audio and music content transmission and will give some details and clues on each of the components' functionality.

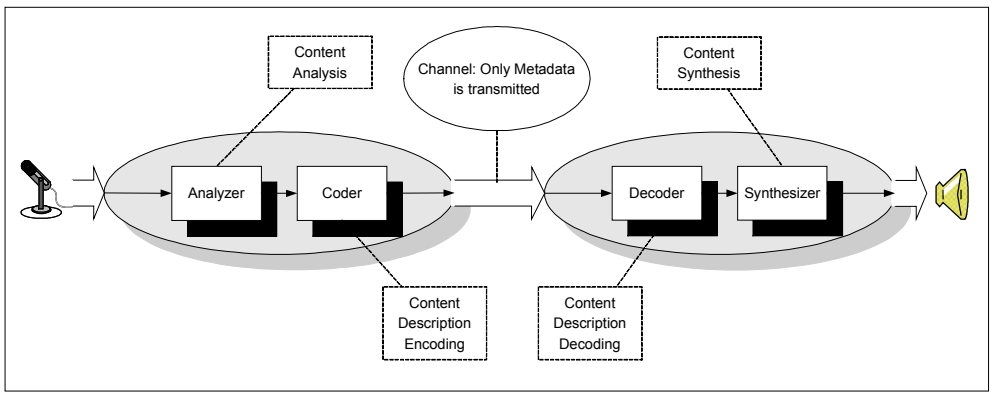

Figure 1. *Content transmission model*

2. The Analysis Step (Content Extraction)

The easiest way to add content description to an audiovisual chunk of information is by means of textual or oral annotation. The extraction process is in that case performed by an 'expert' that can interpret the content and extract some useful information, provided there is an appropriate taxonomy available.

When thinking in terms of automatic content-extraction[7], two levels are usually distinguished: low-level content descriptors and high-level content descriptors. As a first approach, and in the broad sense, low-level descriptors are those related to the signal itself and have little or no meaning to the end-user. In other words, and thinking in terms of our domain, these descriptors cannot be 'heard'. On the other hand, high-level descriptors are meaningful and might be related to semantic or syntactic features of the sound.

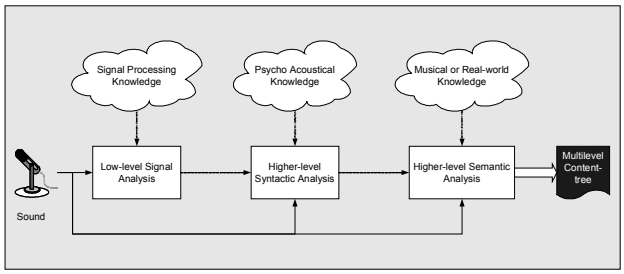

Figure 2. The three levels in the analysis process

It is obvious that the borderline between these categories is thin and not always clear. Some descriptors can be viewed as either low or high-level (or as either syntactic or semantic) depending on the characteristics of the extraction process or the targeted use. Although these categories will be used throughout this paper, we might better think in terms of a multilevel analysis scheme as the one depicted in the following figure.

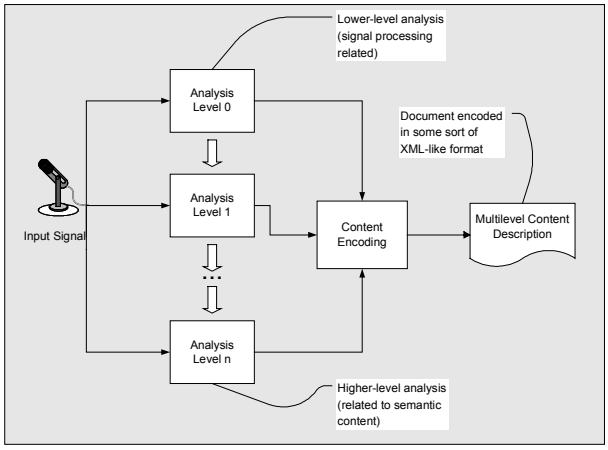

Figure 3. *Multilevel analysis step*

Low-level content descriptors

As mentioned before, low level descriptors are closely related to the signal itself or any of its representations. Any audio signal can be represented as a time-domain signal or as its spectral transform, and following this same idea a first (and yet incomplete) categorization, separates low-level descriptors into two categories: temporal and spectral descriptors.

Temporal descriptors can be immediately computed from the actual signal or may require a previous adaptation stage in order to extract the amplitude or energy envelope of the signal, thus only taking into account the overall behavior of the signal and not its short-time variations. Examples of temporal descriptors are attack time, temporal centroid, zero-crossing rate, etc...

Many other useful descriptors can be extracted from the spectrum of an audio signal. These descriptors can be mapped to higher level attributes. As a matter of fact, of the five basic dimensions of a sound, two of them (pitch and brightness) are more easily interpreted in the frequency domain and a third one (timbre) is also very closely related to the spectral characteristics of a sound. A previous analysis step needs to be accomplished in order to extract the main spectral features. For inharmonic sounds a Fourier analysis (FFT or STFT) can be enough, but a further step (which may include fundamental extraction, peak tracking and some sort of separation of the sinusoidal and residual component of the signal) is useful for the analysis of harmonic features [8]. Descriptors directly derived from the spectrum are, for example: spectral envelope, power spectrum, spectral amplitude, spectral centroid, spectral tilt, spectral irregularity, spectral shape, spectral spread…; derived from the spectral peaks: number of peaks, peak frequencies, peak magnitudes, peak phases, sinusoidality…; derived from a fundamental detection: fundamental frequency, harmonic deviation; etc...[9]

High-level content descriptors

While descriptors presented in the previous section are purely syntactic (that is, they do not carry any information on the actual meaning of the source), high-level descriptors can carry either semantic or syntactic meaning. For example, a region of an audio track can be viewed as a segment (syntactic) or as a musical note (semantic).

Syntactic high-level descriptors can be sometimes computed as a combination of low level descriptors. In [4], for example, we presented a way of describing timbre of isolated monophonic instrument notes (the scheme for computing the descriptors of a harmonic timbre is depicted in Figure 4). Syntactic descriptors usually refer to features that can be understood by an end-user without previous signal processing knowledge as they may refer to psycho acoustical properties of the source, but they do not actually carry any semantic meaning about the content itself. In other words, syntactic descriptors cannot be used to label a piece of sound according to what actually 'is' but rather to describe how it is structured or what is made of. In that sense, the computation of syntactic descriptors (either low or high-leveled) is not dependent on any kind of musical knowledge. In the case of our timbre descriptor, for example, the resulting descriptor is not sufficient to label a note as being 'violin' or 'piano' but rather to compute relative perceptual distances between different instrument samples.

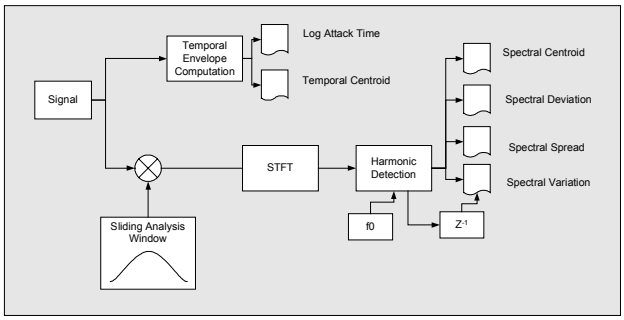

Figure 4. Combining low-level descriptors for creating higher-level syntactic descriptors: MPEG-7's Timbre Descriptor

When trying to label a chunk of audio with a semantic descriptor, more high-level or real world knowledge needs to be applied. The ultimate purpose of a semantic descriptor is to label the piece of sound to which it refers using a commonly accepted concept or term. The degree of abstraction of a semantic descriptor has a wide range. Labels such as 'scary' or more concrete such as 'violin sound' can be considered semantic descriptors. It is obvious that the higher we go on the 'abstraction ladder' the harder it is to automatically extract a description and the more possibilities there are that we end up using manual annotation.

Different proposals have been made in order to create a semantic map or level structure for describing an audio scene, probably the latest being the ten-level map presented in the MPEG Geneva meeting (May, 2000)[10]. This proposal includes four syntactic levels and six semantic levels: Type/Technique, Global Distribution, Local Structure, Global Composition, Generic Objects, Generic Scene, Specific Objects, Specific Scene, Abstract Objects, and Abstract Scene.

While that proposal is quite theoretical and simple and comes from a generalization of a similar structure proposed for video description, other proposals come from years of studies on the specific characteristics of an audio scene and have even had practical applications. One of the most renowned techniques that can fit into this category is CASA (Computer Auditory Scene Analysis)[11]. It is far beyond the scope of this paper to go deep into any of these proposals, but it is interesting to note that CASA has addressed the issue of describing complex sound mixtures that include music, speech and sound effects, also providing techniques for separating these different kind of streams into so-called sound objects (see [12], for example).

3. The Coding Step (Content Description)

In the coding step, all the content information extracted in previous steps needs to be encoded in an appropriate format. Binary and textual based versions of the format should be provided in order to observe both coding and transmission efficiency and readability. It is also important for the coding scheme used to offer support to the way that the output of our analysis block is organized. In that sense, a highly structured language that enables the description of a tree-like data structure (the so-called content tree).

There are many examples of coding schemes used for encoding metadata or, more precisely, audiovisual content description, perhaps the most ambitious being MPEG7. Although MPEG7 is focused on search and retrieval issues, the actual encoding of the audiovisual content description is flexible enough as for being used by a system as the one proposed in this article[13][14]. It is based in an extension of W3's XML-Schema called MPEG7's DDL (Descriptor Definition Language). XML-Schema is a definition language for describing the structure of an XML document using the same XML syntax and it is supposedly bound to replace the existing DTD language. It is thus a tagged textual format but it also includes support for most Object Oriented concepts [15]. Note so, that XML-Schema will be the language used for structuring our content, but the actual output of the analysis will be a regular XML document. Another subject, which will not be dealt with in this paper, is how this textual information could be compressed and transformed into a more efficient binary format suitable for transmission.

On the other hand, the encoding step must also be in charge of deciding the degree of abstraction to be applied to the output of the content extraction step. This decision must be taken on the basis of the application and the user's requirements although it will obviously affect the data transmission rate. The encoder must decide what level of the content tree should indeed be encoded depending on the degree of concreteness demanded to the transmission process, degree that will usually be fixed by the particularities of the receiver. If only high-level semantic information is encoded, the receiver will be forced to use more of its 'artificial imagination' (see next section).

4. The Decoding Step (Content Interpretation)

The main task of the decoder is to interpret the information received through the channel in order to be able to feed the Synthesizer with the correct parameters. Two main processes are expected from the decoder depending if the content description received is high or low level. In next sections we will detail their main characteristics.

If the decoder is input low-leveled descriptions, there are two options, depending on the application requirements. The low level descriptors can be directly fed into the Synthesis engine or there can be an intermediate 'abstraction process'. In the abstraction process, the decoder has to use 'real world' knowledge in order to convert low-level information into mid-level information, more understandable from the synthesizer point of view. If the abstraction process is omitted and the synthesizer receives low-level information but this description is not exhaustive, those parameters not specified should be taken as default. Thus, paradoxically, the synthesizer is granted some degrees of freedom and the result may loose concretion.

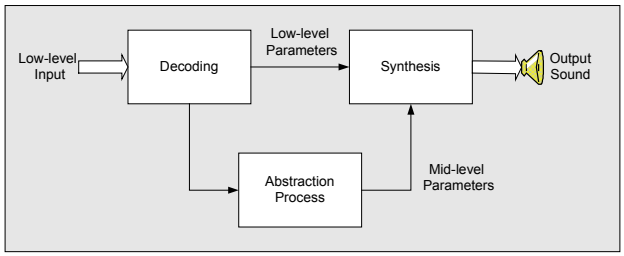

Figure 6. Low-level input to the Decoder: Abstraction Process

If the input to the decoder is high-level semantic information, an intermediate process is always needed in order to make the content description understandable by the synthesis process. It is what we call 'Artificial Imagination'.

I will try to clarify what this term, contrary of the 'abstraction process' earlier mentioned, means by using an example. Imagine the decoder's input is 'violin note'. The synthesizer will be unable to interpret that content description because of its degree of abstraction. The decoder is thus forced to lower the level of abstraction by suppressing degrees of freedom. The output of the decoder should be something like 'violin note, pitch: C4, loudness: mf…' This process is accomplished by means of its 'artificial imagination'.

Artificial imagination is a one-to-many process, that is, the same input should yield a finite set of different outputs. The way the decoding process gets rid of the degrees of freedom should rely on user or application preferences as well as on random processes and context awareness. In the previous example, the decision on the note and loudness to be played could be based on knowledge on the author, the style, the user's likes, previous or future notes, harmony and a final random process to choose one of the best alternatives.

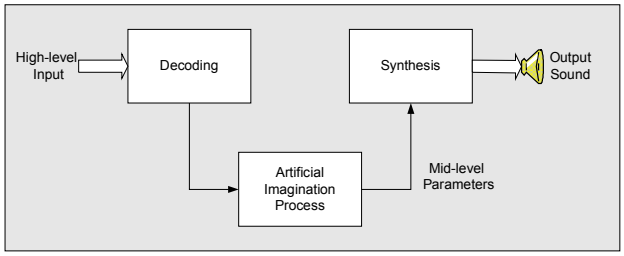

Figure 7. High-level input to the Decoder: Artificial Imagination Process

5. Synthesis Step

The key point of the language used for expressing synthesis parameters is that it must not only meet the requirements of the synthesizer's input but also the needs of the decoder's output.

Many languages have been developed for the purpose of controlling a synthesizer [16][17][18]. Among them, the most extended one is MIDI [19][20] although its limitations make it clearly not sufficient for the system proposed in this paper. Another synthesis language that deserves consideration at this point is MPEG4's SAOL (Structured Audio Orchestra Language)[21][22].

SAOL has been mostly developed at the MIT and has been recently standardized by MPEG and included in MPEG4. SAOL is indeed an evolution of the well known CSound synthesis language. The main advantage of using SAOL at this point of the process is that it should be possibly linked into the parameters coming out from the analysis step, provided that MPEG7 was used at the encoding process[23].

But maybe most interesting of all at this step would be to use an XML-based language that instead of describing the content from a signal analysis point of view (as MPEG7 mainly does) it tried to describe content from a more symbolic approach thus enabling this information to be understood by a synthesis engine. This would prevent the loss of data in an otherwise necessary intermediate conversion process. Many different proposals of such languages are currently being discussed, although none of them seems to have, at the time being, more than an application driven scope [24].

6. A Combined Receiver Scheme: Content-based Synthesis

Although sometimes it may be useful to conceptually separate the receiver into a decoder and a synthesizer, many other times, a combined scheme that treats the receiver as a whole will be more feasible.

In that case, the resulting receiver scheme is what we call a "Content-based Synthesizer", which, at first sight, does not defer much from that of a "traditional" synthesizer.

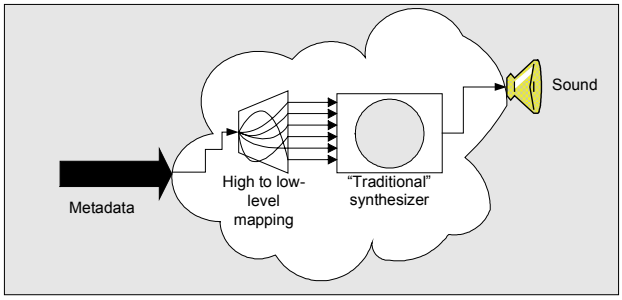

Figure 8. Combined scheme for modeling the receiver of a content transmission system

In a general situation, a simple mapping strategy may be sufficient. But if the level of abstraction of the input metadata is higher, the gap between the information transmitted and the low-level parameters that are to be fed to the synthesis engine might be impossible to fill using conventional techniques. Imagine for example a situation where the transmitted metadata included a content description such as: [*genre: jazz, mood: sad, user_profile: musician*].

The latter example leads to the fact that we are facing a problem of search and retrieval, more than one of finding an appropriate mapping strategy. We could have a database made up of sound files with an attached content description in the form of metadata. The goal of the system is then to find what register in the database fulfils the requirements of the input metadata.

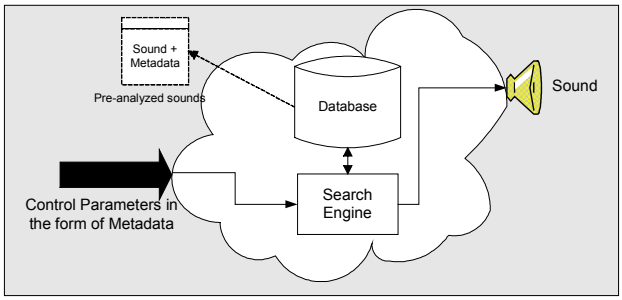

Figure 9. Search and retrieval as a means of synthesizing

A problem we still have to face with such a model is the difficulty to automatically extract from the signal itself parameters with such a level of abstraction. We can find examples of existing applications that implement the system depicted in the previous figure but they always need a previous step of manually annotating the content of the whole database.

A possible solution to this "inconvenience" is the use of machine learning techniques. It is recently becoming usual, in this sort of frameworks, to implement, for example, collaborative filtering engines (classification based on the

analysis of users' preferences: "if most of our users classify item X as being Y, we label it that way"). In that case though, the classification and identification is performed without taking into account any inner property of the sounds. On the other hand, if what we intend to have is a system capable of learning from the sound features, we may favor a Case-Based Reasoning (CBR) engine as the one used in [25].

Anyhow, a first precondition for deciding on the system's viability would be to reduce the size of the resulting database. There is no need to store sounds that could be easily obtained from other already existing in the database. In the case that no sound exactly matched the content description at the input we could then just find the most "similar" one and adapt it in the desired direction. This adaptation step is basically a "content based transformation".

A possible block diagram for the resulting system is depicted in the figure 10. Note how the user input is fed again into the CBR engine.

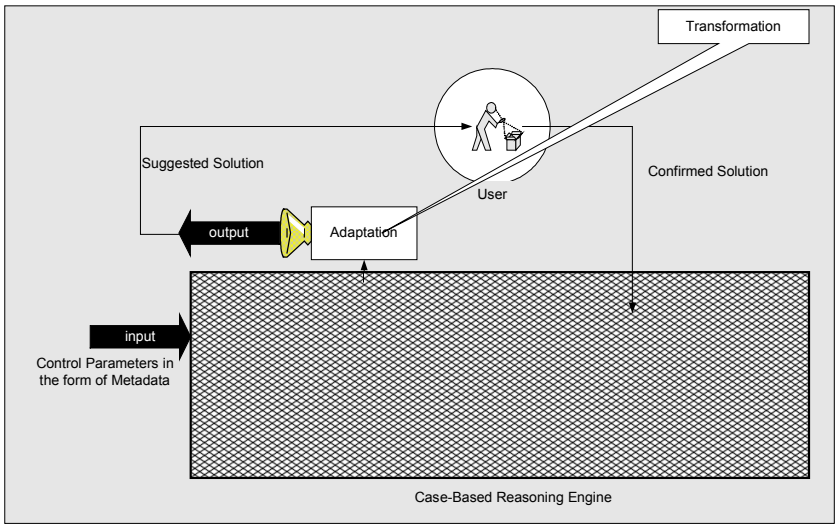

Figure 10. The receiver of a content transmission model as a content-based synthesizer

One of the problems that still remains is what "similarity" measure the system has to deal with. Similarity in sound and music is obviously a many-dimensional measure that can be highly dependent on a particular application. Furthermore, it may turn out that our database has more than one case that is similar to the content description received. All of them may need a further adaptation (transformation) but the problem is how to decide on what transformation is more immediate and effective. In that sense, it may be interesting to identify and classify items for the database not only for what they actually are but for what they may become. A sound can thus be classified as "bright-able", "piano-able", "fast-able" [26]. If a solution is confirmed as accepted by the user we may not only add the resulting sound and its content description to the database but also the knowledge derived from the adaptation process.

7. Conclusions

The purpose of this abstract is to discuss, from a theoretical approach, how new technologies have brought up the opportunity to redefine the model of information transmission that was defined more than fifty years ago. As detailed in the different sections, the model proposed is based on a new paradigm: the transmission of content. The model must be understood rather as a working framework than as a system that should be due in the short term.

Even so, the necessary technologies to implement the different modules are already available or are expected to be in a short term as interest in content-processing grows among different research teams[27][28]. A first analysis of these technologies reveals the fact that none of them are considering compatibility with other modules of the system. Mechanisms to link the content analysis process and the synthesis interfaces must be sought. This key point is likely to be the main difficulty to overcome and our team is already investing efforts that head in that direction.

8. Acknowledgements

The work reported in this paper has been partially funded by the IST European project CUIDADO and by the TIC national project TABASCO.

9. REFERENCES

[1] Camurri, Antonio; "Music Content Processing And Multimedia: Case Studies and Emerging Applications of Intelligent Interactive Systems", Journal of New Music Research, Vol.28 No.4, 1999.

[2] Chiariglione, Leonardo; "The Value of Content", Technology Reviews, March 2000.

[3] Martínez, Jose M., "Overview of the MPEG-7 Standard", document number: ISO/IEC JTC1/SC29/WG11 N4031 http://www.cselt.it/mpeg/standards/mpeg-7/mpeg-7.htm

[4] Peeters, Geoffroy; Herrera, Perfecto; Amatriain, Xavier. "Audio CE for Instrument Description (Timbre Similarity)", Input document for the Maui Meeting of MPEG, November 1999. Doc. num. m5422

[5] Darnell, Donald; Approaches to Human Communication, Richar Budd and Brent Ruben eds, Spartan Books, New York, 1972

[6] Shannon, Claude and Weaver, Warren; The Mathematical Theory of Communication, Univ. of Illinois, Urbana, 1949

[7] Scheirer, Eric D.; Music Listening Systems, Phd Thesis for the MIT, June 2000

[8] Serra, X. 1996. "Musical Sound Modeling with Sinusoids plus Noise", in G. D. Poli, A. Picialli, S. T. Pope, and C. Roads, editors, Musical Signal Processing. Swets & Zeitlinger Publishers.

[9] Serra,X. and Bonada,J. Sound Transformations Based on the SMS High Level Attributes. Proceedings of the Digital Audio Effects Workshop (DAFX98), Barcelona, November 1998.

[10] Jaimes, A. Benitez, A. B. Chang, S.-F. "Multiple Level Classification of Audio Descriptors", Doc num. ISO/IEC JTC1/SC29/WG11 M6114, Geneva, Switzerland, May/June 2000.

[11] Bregman, A. S., Auditory Scene Analysis: the Perceptual Organization of Sound, MIT Press, Cambridge, MA, 1990

[12] Nakatani, Tohomiro and Okuno, Hiroshi G.; "Sound Ontology for Computational Auditory Scene Analysis", Proceeding for the 1998 conference of the American Association for Artificial Intelligence.

[13] Vetro, A.; "MPEG-7 Applications Document v.10", Document number: ISO/IEC JTC1/SC29/WG11 N3934, January 2001/Pisa.

[14] Lindsay, Adam and Kriechbaum, Werner; "There's More Than One Way to Hear It: Multiple Representations of Music in MPEG-7", Journal of New Music Research, Vol.28 No.4, 1999.

[15] W3's XML-Schema homepage, [http://www.w3.org/XML/Schema]

[16] Amatriain, Xavier; Bonada, Jordi; Serra, Xavier. 1998. "METRIX: A Musical Description Language and Class Structure for a Spectral Modeling Based Synthesizer". Proceeding of the Digital Audio Effects Workshop (DAFX98).

[17] Mc Millen, Keith. 1994. "ZIPI: Origins and Motivations". Computer Music Journal 18(4). pp 48-96

[18] Selfridge-Field, Eleanor.1997. Beyond Midi, The Handbook of Musical Codes. MIT Press.

[19] MIDI Manufacturers Association. 1998. MIDI 1.0 Detailed Specification. Los Angeles: The International MIDI Association.

[20] Miles Huber, David.1991. The MIDI manual. USA. Howard W.Sams.

[21] Scheirer, Eric D. ; "SAOL: The MPEG-4 Structured Audio Orchestra Language", Proceeding for the 1998 ICM

[22] Synthetic/Natural Hybrid Coding (SNHC) section of the MPEG-4. 1996. Final Committee Draft Version 1.8. Document num.FCD ISO/IEC 14496-3 Subpart 5. MIT Media Laboratory. [http://sound.media.mit.edu/mpeg4]

[23] Pereira, F. 2001. "MPEG-7 Requirements Document V.14", Document number: ISO/IEC JTC1/SC29/WG11 N4035, March 2001, Singapore.

[24] Cover, R. "XML and Music", in The XML Cover Pages. [http://www.oasis-open.org/cover/xmlMusic.html]

[25] Arcos, J. L.; R. López de Mántaras; X. Serra. 1998. "Saxex: a Case-Based Reasoning System for Generating Expressive Musical Performances", Journal of New Music Research, Vol. 27, N. 3, Sept. 1998.

[26] Rolland, P.; Pachet F. 1995. "Modeling and Applying the Knowledge of Synthesizer Patch Programmers" In G. Widmer (ed.), Proceedings of the IJCAI-95 International Workshop on Artificial Intelligence and Music, 14th International Joint Conference on Artificial Intelligence (IJCAI-95), Montreal, Canada.

[27] Karjalainen, M. "Immersion and content- a framework for audio research", Proceedings of the IEEE Workshop of Applications of Signal Processing to Audio and Acoustics, 1999.

[28] Tolonen, Tero. "Object-Based Source Modeling for Musical Signals", Proceedings of the 109th AES Convention, Los Angeles, 2000.