

Representation and linking mechanisms for audio in MPEG-7

Adam T. Lindsay^{a,*}, Savitha Srinivasan^b, Jason P. A. Charlesworth^c,
Philip N. Garner^c, Werner Kriechbaum^{d,1}

^aStarlab NV/SA, Blvd St-Michel 47, B-1040 Brussels, Belgium

^bIBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120, USA

^cCanon Research (Europe) Ltd., 1 Occam Court, Surrey Research park, Guildford, Surrey GU2 5YJ, UK

^dIBM Development Lab Böblingen, Digital Media Solution Centre, Schönaicher Straße 220, D-71032 Böblingen, Germany

Abstract

This paper proposes a general framework for the description of audio within audiovisual sequences for MPEG-7. These related descriptors and description schemes² were initially defined during the first phase of MPEG-7 and then evaluated during the Lancaster Meeting held in February 1999. These proposals are based on the underlying premise that audio content can be expressed by a combination of two synergistic representations, both of which are necessary to represent audio content accurately. The first is a structured or semantic representation of audio such as a sentence, paragraph, score, or class. The second is an unstructured representation of the audio simply represented as a continuous stream of data. Since it is not possible to express all aspects of audio in a structured representation, powerful linking mechanisms are required between these two representations. We propose an audio description scheme as a basic *structure and representation* for audio based on hierarchical, temporal segments. Such a description scheme is essential for both ease of description and to support content based indexing and retrieval of audio. We also propose a description scheme for the representation of larger structures such as *spoken content* in audio, where the annotation is generated using automatic speech recognition. Finally, we propose *linking mechanisms* between structured descriptions and unstructured audio content, as an example facility that would add great power to both of the previously mentioned description frameworks. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: MPEG-7 audio; Audio structure descriptions; Spoken content; Speech recognition transcriptions; Linking mechanisms

* Corresponding author. Contact address: Computing Department, Lancaster University, Lancaster LA1 4YR, UK.

E-mail addresses: adam@starlab.net, atl@comp.lancs.ac.uk (A.T. Lindsay), savitha@almaden.ibm.com (S. Srinivasan), jasonc@cre.canon.co.uk (J.P.A. Charlesworth), philg@cre.canon.co.uk (P.N. Garner), kriechba@de.ibm.com (W. Kriechbaum).

¹ Some of the achievements described in this document were carried out within the ACTS project DICEMAN. The work was part-funded by the European Commission. The views expressed are those of the authors and should not be taken to represent, in any way, the views of the European Commission or its services.

² Throughout this paper, data structures will be referred to as *descriptors* (D) and *description schemes* (DS). A descriptor represents a quantitative measure of audiovisual features. A description scheme describes structures of descriptors and their relationship.

1. Introduction

MPEG-7 aims at standardizing a core set of descriptors and description schemes to enable indexing and retrieval of audiovisual data [8]. It is expected that this standard core set will facilitate those classes of applications that have wide-spread usage and will provide interoperability. To this end, higher level representations of the data contained in multimedia documents will enable a wider range of indexing and retrieval actions performed on the content. It will enable the capture of the rich, expressive structure in audio. A collection of low-level descriptors focussing on fine time–frequency structure will be important for identifying specific content, but one must also consider larger structures on the audio document scale. Description schemes for larger structures such as spoken content derived from audio will be important for identifying audio content at a semantic level that can be expressed in terms of words.

In this paper we emphasize three aspects related to audio structure representation in MPEG-7, each originating from a different proposal. As such, there were no a priori connections between proposals, although we do attempt to note certain relations. The first proposal, outlined in Section 2, is a description scheme (DS) that provides a general framework for the description of audio within audiovisual sequences. This description scheme does not mandate any particular extraction scheme for the audio data encoded, however, we believe that automatic methods will be used to the extent permitted by technological advances. The second proposal, outlined in Section 3, emphasises the representation of larger structures in audio, i.e., a description scheme for representing the spoken content and summary where the content is derived using automatic speech recognition (ASR). The description scheme is based on an automatic extraction method as opposed to manual annotation, and therefore encompasses the output of a typical speech recognition system. The third proposal, presented in Section 4, is a general linking mechanism, relevant to linking within a description, but also extremely useful for creating links between descriptions, which would serve the purposes of the above description schemes well. In

sum, we provide an overview of the larger components at each level of the proposed audio representation and the connections between the different levels.

2. Audio structure

2.1. Motivation and overview

The audio description scheme presented below was proposed and conceived as a parallel structure to visual description schemes proposed at the same time [12] for the purpose of having a unified structure for audiovisual documents. What we present is a first step towards that. The ongoing work in MPEG-7 is bringing the audio, and visual, description schemes even closer together. We give a few brief comments on the latest developments after describing the proposal.

The purpose of the Distributed Internet Content Exchange with MPEG-7 and Agent Negotiations (DICEMAN) [17] audio DS and audio object DS is to provide a general framework for description of audio within audio-visual sequences. It provides a general, hierarchical table of contents, a basic “template” DS for main segments called audio scenes, and an index into relevant content in the form of audio objects [7].

The goal of the main DS is to concentrate primarily on the division of the described media into *temporal segments*, both for considerations of ease of description and ease of retrieval. It does not attempt to address frequency-oriented segmentation, though it does not preclude such a thing, allowing for integration once the right linking and representation mechanisms are found. The audio scene DS is intended for general audio use: for both audio alone, and audio in conjunction with video. It was conceived with representing dramatic audiovisual material as the foremost goal, and is similarly very easily applicable to other narrative or other structured forms (e.g. news). It is applicable to all kinds of audio material with the exception of atomic (indivisible, short) sound effects. However, with this flexibility, we sacrifice specificity, and as such, do not attempt to prescribe any descriptors. Section 3 gives a detailed example of a description

scheme and descriptors with more “bottom-up” descriptive power.

The overall structure of the audio DS is of a general identifier and descriptor that locates and labels the entity to be described, and a hierarchical structure dividing the Audio entity into progressively smaller logical segments. We give some illustrations of general concepts pertaining to this hierarchy, and then define the Audio TOC’s atomic unit (leaf node), the audio scene DS. The audio object DS is then given an overview, along with its relationship to the rest of the audio DS.

2.2. Audio TOC

The audio table of contents (TOC) plays the same role as a table of contents in a book. A traditional TOC divides the book into sections, chapters, and sub-chapters. No part of the book is unaccounted-for. It allows for quick browsing and access to large sections – or progressively smaller sections – of the book. All of this is provided by the audio TOC. Here, the division units are audio segments and audio scenes.

The audio TOC is intended to have a strict, clear hierarchy of segments for easy access to conceptually related and temporally contiguous audio content. Examples may include acts and scenes in a play, scenes in a movie, announcements and music in a radio program, or different news stories. The continuous nature of the segments is to facilitate access by division into manageable units. More flexible entities may be contained in an index within a segment, as described below.

Fig. 1 illustrates a typical temporal segmentation into segments and sub-segments. No part of the

audio stream is unaccounted-for in the first level of segmentation. Similarly, no part of the segments which have further segmentation is unaccounted-for in the sub-segment level. Conversely, there are no times that are connected to more than one segment on a given level.

The audio TOC thus forms a strict hierarchy of segments, each level further down segmenting completely, without gaps or overlaps. When the segments cannot be sub-segmented any further, these leaves are called audio scenes. Fig. 2 gives a simplified view of the class hierarchies in the Audio TOC DSs in UML notation [3]. We outline the most relevant DSs. Note that nearly every DS has temporal descriptors and textual labels attached, beyond the descriptors that carry the semantics indicated by the DS.

Media reference DS. This simple description scheme is a place that contains two different types of descriptors, the time references for the beginning and the end, and the descriptor allowing for identification of the described media.

Audio table of contents (TOC) DS. This hierarchical DS rests as the root of a tree (with audio segments as branches, and audio scenes as leaves). It contains from one to any number of Audio Segment DSs, and a label for the TOC, should a description be needed.

Audio segment DS. Any group of audio segments (or audio scenes) lying beneath a single audio segment (or audio TOC) at the same hierarchical level, must have contiguous time references with all of the others at that level, and span exactly the time spanned by the parent segment (or entire described entity). An audio segment may contain any number of child segment DSs and audio scenes (segments

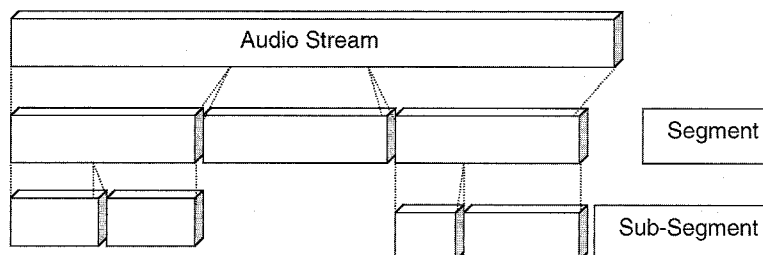


Fig. 1. A plausible segmentation of an audio stream for an audio DS table of contents.

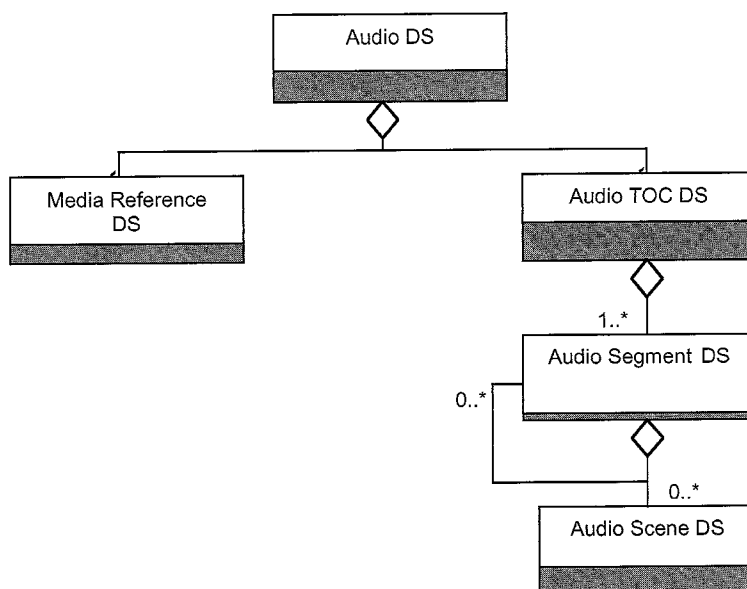


Fig. 2. A simplified view of the audio TOC DS, showing the sub-description schemes.

without children), as long as the children support the time (begin and end) parameters. There also exists a string label for segment identification.

Audio scene DS. If an audio segment contains no child segments, then it is called an audio scene, which has its own properties and its own DS, described in the following section.

2.3. Audio scene

An audio scene has two definitions, one functional, and one conceptual. The functional definition is that of a segment (of an audio stream) that has no sub-segments. That is, an audio scene is a leaf node in the tree formed by the TOC. Conceptually, an audio scene is a temporal section of the audio stream that is unified by one or more characteristics across the entire stream, and presumably perceptually different from adjacent scenes in the stream.

It should be noted that due to the characteristics of the TOC tree, all temporal units (e.g. frames, samples) within the described audio stream are contained in exactly one audio scene. Audio scenes contain a list of component audio Objects, the

characteristic ambience DS, and relationships (i.e. transitions and links) with other audio scenes.

When distinguishing audio scenes, significant characteristics may include background noise (whether ambient or due to recording characteristics), reverberation, or the presence or absence of background music. The distinguishing characteristics of an audio scene may be summarized in an ambience DS, detailed below. Fig. 3 illustrates the subclasses of the audio scene DS. Again, each of the sub-segments has at least a textual label to supplement the usual semantic descriptors that lie therein.

Ambience DS. This optional description scheme is essentially a stub to allow users to insert their own description pertinent to the entirety of the audio scene. The description contained by this DS is defined as being a description of the common feature(s) to the entire audio scene, and distinguishing it from other, adjacent (in time) description schemes. Example descriptions may be of a constant “colored” noise, music, or recording characteristics throughout the scene.

Audio object DS. There may be any number of audio objects within an audio scene, and they are not subject to the same time reference restrictions

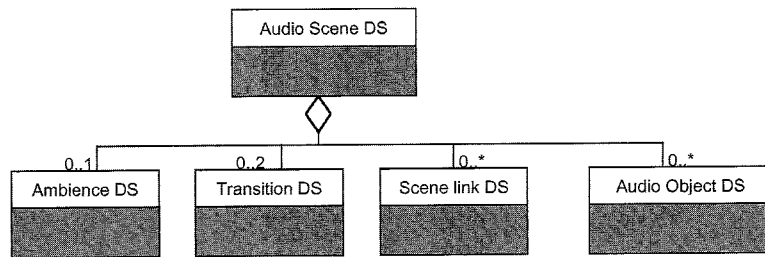


Fig. 3. A simplified view of the audio scene DS.

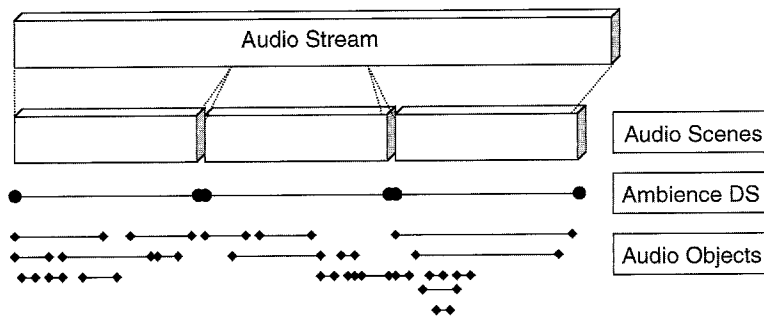


Fig. 4. An example of audio scenes containing ambiances and unlimited numbers of audio objects.

as segments or scenes (i.e. they may freely overlap). The audio scene DS is detailed in its own section, below.

Transition DS. This is a simple description scheme providing a basic description framework for the transitions into and out of scenes, represented by an enumerated list of possibilities.

Scene link DS. This is a light DS capturing relationships between audio scenes. It consists of a link, such as one of the type described in Section 4, and the type of relationship the link represents. There may be any number of scene links in a scene, as any scene has a potentially unlimited number of relationships.

2.4. Audio object DS

Audio objects are time segments which describe a continuous audio event or process, generally from one “source”. Examples could be a section of

dialogue from one speaker, a piece of music, or a sound effect. Each of these different types of audio objects may have labels, links, and their own DSs attached to it.

The audio object DS is the other major part of the DICEMAN audio DS. It exists beneath the audio scene level to provide some structure to an essentially unordered list of audio objects. The list of all audio objects may also conceptually serve on the top-level as an index, akin to the index in a book. Each audio scene enumerates the contained (in whole or part) audio objects, and the top-level index is the union of all of the component indices in the audio scenes. A conceptual diagram of audio scenes and audio objects is shown in Fig. 4.

If one wishes to describe the temporal structure of a fugue, for example, it probably would lie in a single audio scene. The overlapping motives do not (and cannot, given our strict definitions of hierarchies without overlaps or gaps) form hierarchical

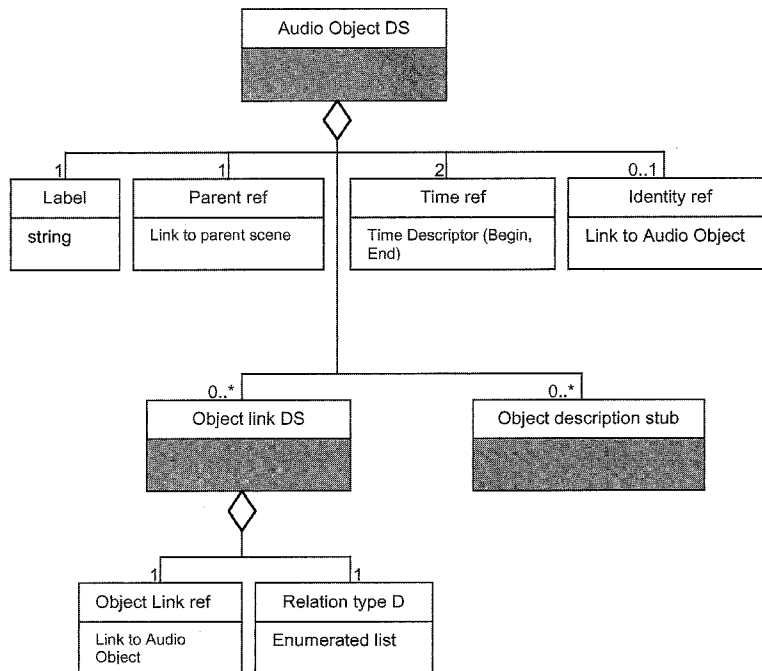


Fig. 5. The class hierarchy for the audio object.

segments, but rather fit into the model of overlapping objects.

Fig. 5 illustrates a simplified view of the class hierarchy under the audio object DS. There is more detail to give a sense of the different links and semantics that may be attached to an individual object.

Audio object DS: parent ref. An audio object contains a link to its parent. In many cases it may be redundant, but it is to accommodate situations in which an object is separated from the rest of the audio hierarchy. Depending on the as-yet-unspecified DDL, or other, binary, representations, it may be entirely unnecessary.

Audio object DS: time ref. An audio object has a beginning and an end. The two times must fall within the range set out by the parent audio scene, but otherwise, there are no restrictions.

Audio object DS: identity ref. The identity reference link is used within an audio object if and only if the object is a direct continuation of the same entity from a prior scene. In the case of any other relationships, use the object link DS.

Audio object DS: object link DS. This is a basic DS capturing relationships between audio objects. It consists of a link, and the type of relationship the link represents. There may be any number of object links for an object, as any object has a potentially unlimited number of relationships.

Object link DS: scene link ref. This is a link to a related audio object. The link mechanism is determined by the DDL.

Object link DS: relation type. This descriptor is an enumerated list representing different types of relationships between audio objects. When instantiated, one value is chosen. Possible values in this list include (non-exhaustive)

- identity (a literal, even waveform-exact, repetition),
- variation (a non-literal repetition, such as a person saying a word twice in succession),
- identical source (the sound is generated from the same source as the other),
- continued phrase (the audio object is the continuation, whether through a phrase boundary or a pause, of a previous audio object).

2.5. Future development and refinement

The reason for having the above-described bi-level structure, with a strictly hierarchical TOC and unstructured objects was to reach a compromise between the power of a strict structure, and flexibility to accommodate real sounds, and different styles of description. It is clear that reality does not always fit into a tree structure, but enough structures, such as transition segments and audio objects, exist to help accommodate description.

On further collaboration after the initial proposal it was clear that these structures are not flexible enough to accommodate all conceivable ones. In particular, the need for overlapping temporal segments was revisited by the proposer. Also, a greater degree of abstraction is needed in order to unify with the video description scheme. As a result, current work is going on to generate a generic segment DS that will be used across media types. The TOC structure will be streamlined to use the segment DS, and there will be other “views” on the data than the existing audio scene-oriented one.

3. Spoken content and summary derived from an automatic speech recognition (ASR) system

3.1. Motivation and overview

In this section, we focus on representation structures for one particular form of an audio object as defined in the previous section, namely that of spoken content. Information retrieval from audio data is sharply different from information retrieval from text because of the linear nature of audio. Advances in audio information retrieval technology are being used to provide insight into the content of the audio, thereby making it more searchable. The available methods can be roughly divided into those that assume speech content in the audio and those that do not. Multimedia retrieval techniques have been classified into “expression” and “semantic” approaches [14,2]. Expression-based retrieval either requires a physical description of the object to be retrieved, or

a sample search query such as “Give me more like this”. Semantic retrieval is based on knowledge about the object to be retrieved which may be conveyed by annotations or multimedia captions. Given the proliferation of audio databases today, systems that can automatically extract information about the audio content are becoming indispensable in comparison with systems that require human-generated annotations or captions. In this context, speech recognition technology is well positioned to support semantic retrieval in audio that assumes speech content.

Having said this, the caveats associated with the use of speech recognition technology must be noted. The primary limitation of the use of speech recognition for retrieval of audio lies in its limited accuracy [1,2,13]. While the technology is proving to be increasingly usable, it does not produce perfect text transcriptions. One aspect of this lies in the difference in recognition word error rates (an accuracy measure for speech recognition systems) between *read-speech*, and *spontaneous, conversational* or *real-world* speech. Read-speech produces a more accurate transcript as opposed to real-world speech. The implications of this to MPEG-7 audio retrieval based on speech recognition is that almost all the audio is likely to be spontaneous, real-world speech. The accuracy of the transcript can vary dramatically between 30% and 70% [6] depending on a variety of factors such as quality of audio, vocabulary or domain match, non-native speakers, audio that includes music and other non-speech sounds, etc. The performance of the information retrieval system is directly limited by the recognition accuracy [6]. The MPEG-7 proposal [4] incorporating other proposals [15] addresses exactly this issue by proposing a description scheme that consists of several descriptors to support the output of a speech recognition system intended to yield acceptable information retrieval performance despite recognition errors.

Ensuring that sufficient data is stored in the descriptor to support accurate later retrieval is thus the prime criteria for evaluating any spoken content. Were the spoken content to be based solely on idealized ‘perfect’ transcriptions, e.g., hand annotations, and created with a specific retrieval task

in mind this would be a trivial criteria. In practice, the annotations will be created using an automatic speech recognition (ASR) system and must support retrieval for a variety of tasks and languages. In this section, we describe the representation of an ASR description scheme, and also describe the linkage between the semantic descriptors and the expression based descriptors (psycho-acoustic spectral and temporal features of audio). We show that the expression and semantic-based audio retrieval approaches are not mutually exclusive, but can complement one another in approaching the ultimate semantic ideal.

Storing the ASR produced transcription is important to MPEG-7 as the spoken content and summary of a speech signal reduces the data storage requirements, and, more significantly, provides words as index terms for the audio that contains speech. This enables leveraging mature text information retrieval technology using standard keyword-based textual queries for audio/video retrieval. Appropriate MPEG-7 applications include storage and retrieval of video databases, retrieval of historical speech databases, movie scene retrieval by memorable auditory events such as keywords or catch phrases, user agent-driven media selection and filtering, personalized television services to support selection of services based on spoken content, semi-automated multimedia editing and educational applications [9].

3.2. *Context of the spoken content*

Unlike most audio transformations, the output of an ASR engine may be considered a stochastic corruption of the idealised transformation (transcription). As such, in the design of a spoken content description scheme, especial attention must be paid to both the limitations of current ASR systems and the retrieval methods that may be applied to the content. In this section we will outline the essential features of ASR outputs and retrieval techniques which have significant design implications on the DS.

The majority of ASR engines are based on an efficient algorithm for matching unknown speech data to a statistical representation of words (or

sub-word units, e.g., phones³). Due to the difficulty in associating a unique transcription to an utterance, ASR engines typically decode employing a lattice containing multiple hypotheses giving the probabilities of the most likely few words, or phones, at each time [11]. The most probable path through the lattice is usually output as the ASR result, giving rise to the generated annotation.

To illustrate the requirements of the spoken content description scheme, we may envisage a hierarchy of retrieval tasks one may wish to perform. At the lowest level would be searching for an entry which 'sounds like' a given retrieval key. Above that are words or phrases and, higher still, is retrieval employing previously extracted metadata. The first category may be illustrated by considering the task of searching for a word, present in the original audio but not in the ASR dictionary. Often these are words of low frequency, e.g. a person's or place name, but which are crucial for discrimination in retrieval. The out of vocabulary (OOV) word will typically be replaced in the decoding by a similar sounding word or words (see, e.g., Fig. 6). Evidently, while the accuracy and vocabulary of ASRs is significantly below that of human capability secondary access methods will be required. To provide this the DS must be capable of retaining phonetic information.⁴ Without retaining phonetic information it is possible for users to create annotations which are not retrievable. Although the accuracy of phonetic retrieval may be limited, especially in the cases of short words [10,18], it does provide an essential safety net.

In the second category we may search an annotation for related words or word stems. Where the actual word is not present in the best path decoding (but present in lower ranked paths), such searching

³ Within this paper we will refer to phones rather than phonemes. Whereas phonemes are defined by human perception, phones are data derived. In general, the sound units defined by ASR systems are data derived. The effect of the difference between phones and phonemes is not significant for the arguments presented here.

⁴ The quantity of phonetic information to retain is up to the annotator. For example, it may depend on the confidence level of the words; more where there is a low confidence and less or none when the confidence in the word decoding is high.

methods. It consists of a number of non-intersecting combined word and phone lattices representing the different parses of the stream of audio for each of the identified speakers. These may temporally overlap. It is not simply a sequence of words with alternatives. The proposed lattice structure will be capable of storing the contents of any of the common formats: *N*-Best word lists, *N*-Best phone lists, the output of Microsoft and Java speech APIs, predefined taxonomy and combinations of these. The particular level of detail is up to the annotator. In the rest of this section we will cover the implementational details of the structure, which are shown diagrammatically in Figs. 8–11. Full details on the data structure may be found in [4].

Briefly, the contents of the descriptor are designed to support multiple speakers (possibly with

overlapping speech), multiple languages/dialects, and to support retrieval by phones and words and metadata. The content of the audio descriptor may be broadly split into two sections: the header and the lattice proper.

The header, shown in Fig. 8, contains all the static information essential for efficiently accessing the contents of the lattice, i.e. phonetic, word and speaker details.

The phonetic information, shown in Fig. 9, consists of a phone dictionary, confusion matrix and (optional) index for each of the phone sets used in the lattice. Different languages, dialects and accents use different phones. Consequently, for large annotations produced by a number of speakers of mixed nationalities, several different phone sets must be stored. In addition, to permit fuzzy matching

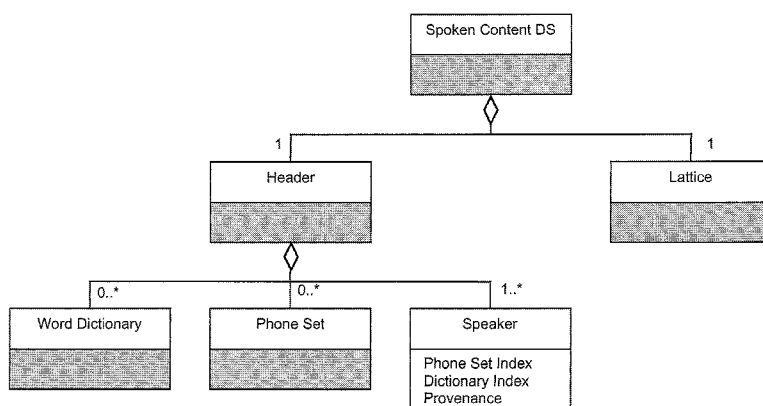


Fig. 8. The class hierarchy of the spoken content description scheme (DS).

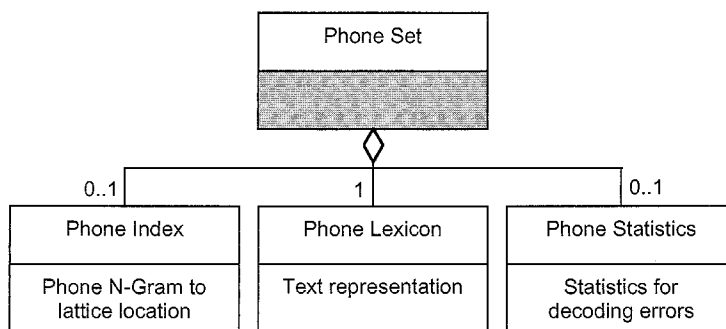


Fig. 9. Class hierarchy for the phone set class of the spoken content DS.

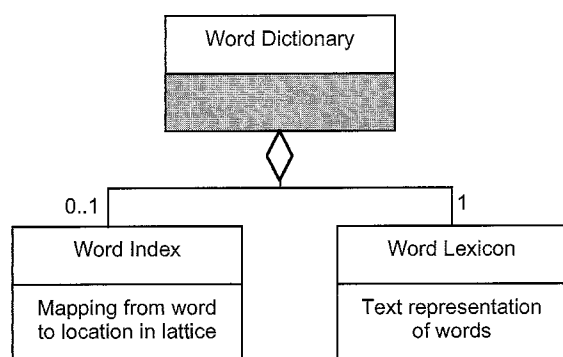


Fig. 10. Class hierarchy for the word dictionary class of the spoken content DS.

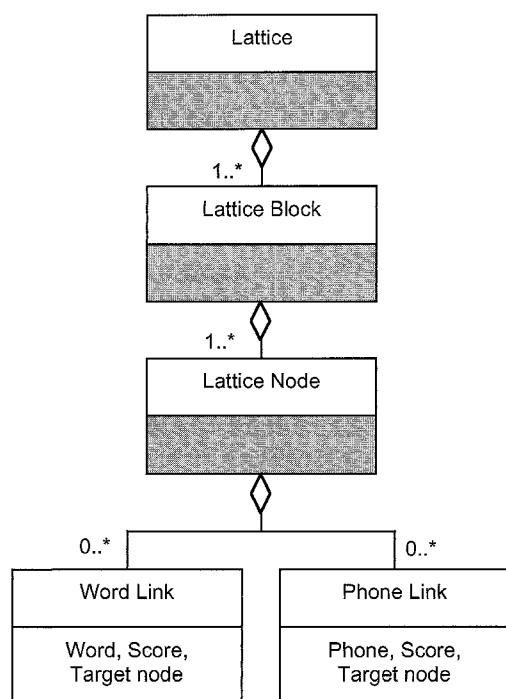


Fig. 11. Class hierarchy for the combined word and phone lattice in the spoken content DS.

between a phonetic query and the annotation, the phone statistics are also stored. These represent likely confusions or corruptions in the phonetic decoding of speech. A holder for an optional index is included to permit fast phonetic searching, e.g.,

where the lattice is largely word based but with phonetic links in regions of low confidence, it would be highly inefficient to search throughout the lattice for phonetic representations.

The word information is similar to the phonetic details. Different dictionaries may be supplied for different speakers either because of different languages, dialects or accents or because of different vocabularies spoken. Indexes are, optionally, provided to allow rapid retrieval at the word or keyword level. It may be noted that metadata is considered to be simply a particular form of a word lattice and so retrieval using metadata falls naturally within the scope of the word retrieval.

The speaker details supplies information about each individual lattice. Multiple speakers may be represented by individual, non-intersecting lattices, for each speaker. Within any lattice the word links are all of one dictionary and the phone links of one phone set. The speaker detail gives the identity of the dictionary and phone set used for the decoding as well as an optional textual description/label of the speaker. In addition to the speaker details there is also a provenance flag which indicates the data source. Typically this will be an ASR transcription or, less frequently, a hand annotation. However, one possibility supported is that of metadata obtained from processing of the ASR output. The metadata is stored in a lattice in the same manner as any other spoken content but the provenance flag indicates its source, e.g., there may be a lattice representing the ASR output for Alice and additionally a lattice representing the topics in Alice's conversation.

The lattice, shown diagrammatically in Fig. 11, contains the ASR output and support data for this task. One critical issue is that the duration of the annotation and the capability of the retrieval device cannot be assumed. As a result, we must permit the lattice to be manipulated in a piecewise fashion, i.e., rather than have absolute indexing we must employ relative indexing. This permits, e.g., the data streaming necessary to support video on demand, but naturally results in what is effectively a linked list data structure. The potential traversal inefficiency of this is widely known. To avoid this problem, the lattice is arranged into a series of

time-ordered blocks of nodes and their timing information provided in a coarse time index. This, combined with the indexes, permits constant time access to the lattice contents.

Each individual block comprises a number of nodes, information about which nodes belong to which speakers and an indication of the audio quality. The last of these permits a level of confidence to be ascribed to the ASR output: when the audio is that of corrupted, garbled or noisy speech the ASR output may be, essentially, random.

Because we are using a heterogeneous lattice, the nodes simply represent points in time while the details reside within the links. Each node possesses a timestamp (relative to the start of the block) and may have multiple phone or word links originating from it. The timestamp is accurate to $\frac{1}{100}$ th of a second to allow the MPEG-7 standard to be used for, e.g., speech corpora.⁵ The word (and phone) links indicate the word (or phone) represented along with the node index offset to the target node and the score for this transition. By employing an offset rather than an absolute index we may reduce memory requirements as well as permit block-wise loading of the lattice.

The proposal outlined in this section is capable of storing a wider range of data than is currently available from any one system. Topics such as blind source separation, speaker identification, language identification, multilingual decoding, topic spotting and linguistic parsing/keyword clustering are all extremely active areas of research and research systems capable of performing any one of these tasks are available. In addition, the current capabilities of ASRs in noisy environments leave much to be desired. Nevertheless, the pace of advance in both algorithms and computational power over the last decade indicates that to ensure that the MPEG-7 standard is useful for a long time, these capabilities must be included.

⁵ Many applications will require an accurate temporal alignment, e.g., were one to employ the MPEG-7 annotation as training speech corpora for speech recognition or synthesis systems, alignment to better than phone resolution is required.

3.4. *Linking of spoken content/summary descriptor with audio object DS*

Real-world audio/audiovisual data typically consists of spontaneous speech as opposed to read speech. Speech recognition accuracy for such audio can be quite poor. Since the objective of the MPEG-7 descriptors and description schemes is to arrive at a set of general descriptors that do not assume any knowledge of the audio content (e.g., mainly speech versus music/non-speech), the descriptors must generate reasonable representations for any content. Although speech recognition systems may be trained for silence detection, they typically generate spoken content for any audio including music, background noise and noisy speech. The annotation corresponding to the non-speech segments in the audio is unpredictable and may contain several inaccurate keywords likely to be used erroneously as index terms for the retrieval. Further, the summary based on the spoken content will be inaccurate too. Indexing and retrieval on such highly inaccurate spoken content will definitely result in markedly lowered precision in the retrieval system.

The linking of the spoken content/summary descriptors to the audio object DS reduces this problem. As seen in Section 2, audio objects are time segments that describe a continuous audio event such as music, background noise or speech. The actual method used for the classification of the audio event may be based on analysis of the audio descriptors. This basic audio classification supported by the audio object DS together with the spoken content output from the speech recognition system results in a more accurate representation of the audio stream. The indexing and retrieval techniques based on the spoken content when combined with the audio classification can achieve higher precision performance in the retrieval system [15]. The benefits of this approach were found to be incremental when performing queries within a given audio/video segment since the inaccurate index terms are highly unlikely query terms given the metadata associated with the audio/video stream. However, the benefits of this approach when searching across large unknown audio/video databases are likely to be compelling because the query terms are not constrained in any manner.

4. Generic and specific links for audio content

4.1. Introduction

Almost all of the proposals dealing with MPEG-7 description schemes, including the one described in Section 2, make use of hyperlinks in one form or another. In most cases linking mechanisms have been implied and not detailed. The following section uses the interplay between representation and realization of audio signals to give a rationale for the need of a powerful linking mechanism in describing contents, tries to unify proposed and implied linking mechanisms, and attempts to summarize the current state of the discussion after the MPEG Seoul meeting, March 1999.

Most acoustic multimedia data have a common property which distinguishes them from visual data: They can be expressed in two equivalent forms, either as an unambiguous textual representation that establishes a “ground truth” (score, script, book) or as a realization (sound recording).⁶ In most cases, the symbolic representation captures nearly all of the important information, but misses emotional and other nuanced aspects of a realization. On the other hand, the symbolic representation either contains additional structural information or makes it comparatively easy to structure this information.

4.2. Describing links

Consider, for example, the audio or video recording of the staging of a theatre play. The symbolic representation of the play has a rich structure built from acts, scenes, and dialogues, but the audio recording is just an unstructured continuous stream of data. The same holds true for an audio book where the printed representation has both a surface structure (pages) and a deep structure built from parts, chapters, sub-chapters, footnotes and cross-references. Finding and addressing speci-

fic structures (pages, chapters, etc.) in the realization (audio recording) are either time consuming, complex, or impossible. This picture changes immediately when the audio stream is time-aligned with its symbolic representation, allowing for indexing of the surface (signal) structure. Furthermore, when aligned using a DS such as that described in Section 2, one may access the deep (content-driven) structure. For recorded speech, standard text-mining technologies can be used to locate sequences of interest and the structure information can be used to segment the audio signal into meaningful units like sentences, paragraphs or chapters. When using the above DS for deep structure of non-textual content, enhanced browsing may still be supported. Even if the transcript is unstructured in the sense that it does not contain structural information like acts, scenes, speaker changes and the like, either the individual characters or words of the text can serve as means to *partition* the document and thus create a very basic kind of structure. Nevertheless, it makes sense to differentiate between *unstructured* and *structured* documents and handle them slightly differently. Fig. 12 shows links between a structured representation of SHAKESPEARE's play Hamlet and the audio recording of a performance.

With two slight modifications the same techniques can be applied to speech material for which no transcript is available: Since the text is not available the transcript has to be generated by speech recognition and the structure has to be deduced from the audio data. The intrinsic problems with this have been detailed in Section 3.

The same is true for annotating a musical performance with its score. However, in this case besides technicalities like the need for a different synchronization engine a new problem arises. For almost all music a single point in the audio recording will correspond to many “texts” in the symbolic representation: As soon as two instruments play together notes in two different voices in the score have to be linked with the audio recording, i.e. one-to-many links are needed.

The problem of linking audio with text can be seen as a “hyperlinking” problem and can be addressed with one of the available hyper linking standards. However, since often neither the

⁶The one visual analogue, the story board, is much more rarely used, and therefore less recognized as an indisputable ground truth. Additionally, it is often a translation between two visual media, and not between signal and symbol.

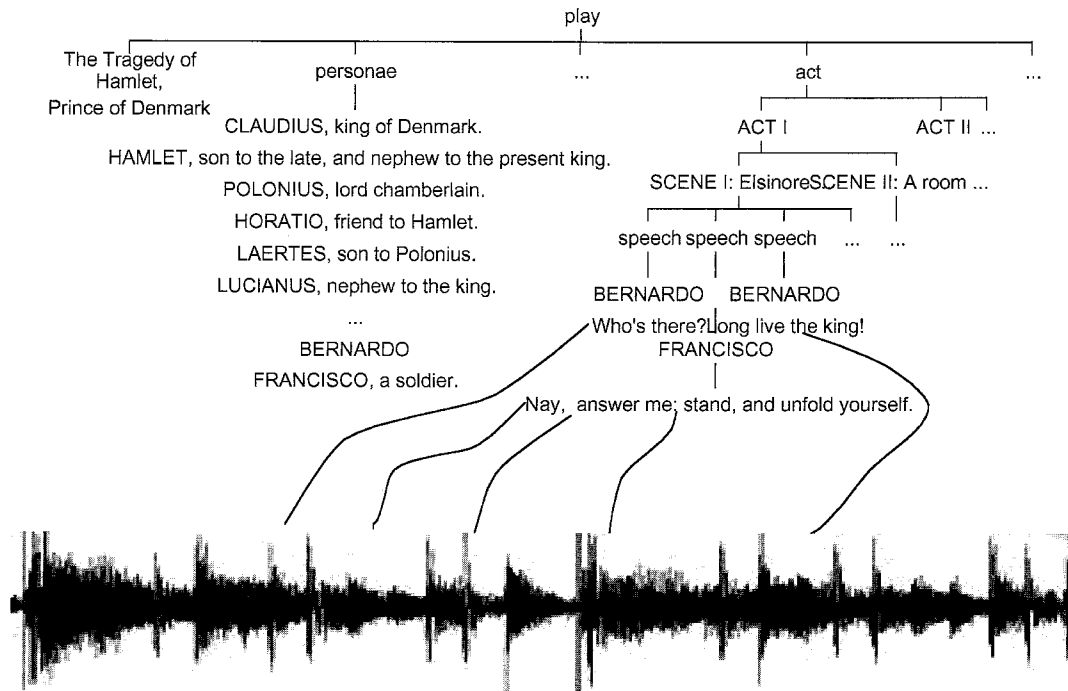


Fig. 12. A hypothetical structured text representation of a play, with curved lines representing links to audio material.

recording nor the text can or should be changed, a simple hyper-link scheme as provided within HTML is not sufficient. Instead, it is necessary to use so-called 'independent' hyper links, which are external of the files they link. In addition, these hyper links must be bi-directional (text to audio and vice versa) to allow both, applications like querying the text and playing back the speech, as well as playing the audio and switching on a display of the score at an arbitrary point in time.

4.3. Representing links

One model for independent hyperlinks is the HyTime ilink [5] scheme. Similar functionality can be provided by other implementations and will be provided by XML linking mechanisms. In the following discussion SGML and HyTime will be used as examples. An independent link consists of three components.

- Anchors, which are regions or points in a text or audio document.

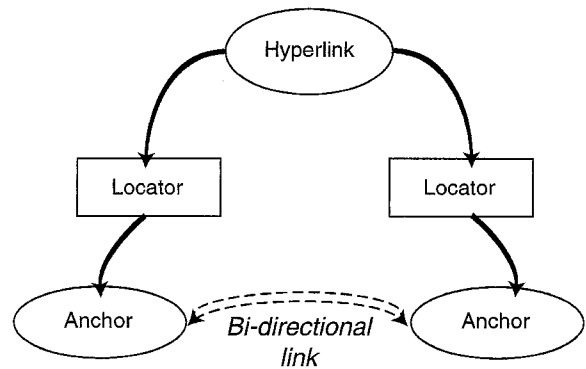


Fig. 13. The canonical bi-directional link provided by HyTime.

- Locators, which locate or address anchors.
- Links, which link or connect locators (see Fig. 13).

The ISO/IEC HyTime standard offers locators to uniquely address elements (subtrees) within SGML documents. This mechanism assigns a list of integer values to each node of an SGML tree. The list of

integer values is the ‘road map’ to get from the root of the SGML document (tree locator ‘1’) to the specific SGML element using several ‘traffic rules’ to generate the tree locator integer list. These rules are

- The ‘journey’ starts at the root element and adds one integer for each horizontal level below on the way down the SGML tree.
- The root element of the SGML tree has the tree locator ‘1’.
- Each integer stands for one horizontal level of the SGML tree.
- Each integer value is generated by counting the number of nodes from left to right. Only the children of the node above are taken into account.
- The leftmost node (left most child) of a node above is assigned the integer value ‘1’.

Starting at the root element (tree locator ‘1’) and taking all the above rules into account to generate the tree locator, the nodes (elements) of the following abstract tree will be addressed by the tree locators listed in Table 1.

4.4. Links and their applications

Tree locators provide a powerful addressing mechanism that covers a wide range of special cases like lattice structures or matrices. Whether additional location mechanisms to point into audio (and video) streams are needed is currently still discussed among the MPEG community.

Besides being essential in synchronizing the different modalities of multimedia data, independent

links can be used to express relations between elements. This capability of links allows an approach to audio structure which is complementary to the one described in the preceding chapter: Not segments or objects determine the organisation of the data but a web of links where the link ends may be spatio-temporal units or almost anything, like e.g. an instrument, a composer, etc. Whether MPEG-7 will favour one of the two possible approaches, provide two alternative mechanisms, or offer a unified approach is still discussed among those working on description schemes.

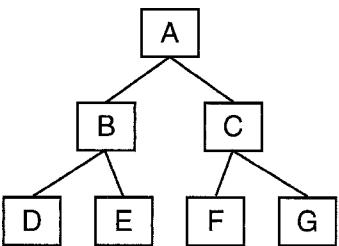
5. Conclusions

This paper summarizes some MPEG-7 audio description schemes and descriptors necessary to capture the expressive structure in audio. The audio description schemes parallel the visual description schemes towards arriving at a unified structure for audiovisual documents. The description schemes and linking mechanisms are intended to facilitate temporal description of structure in audio. A collection of low-level audio descriptors that support expression-based retrieval is appropriate for capturing the spectral and temporal features in audio. Higher level descriptors are necessary to provide semantic retrieval from a richer representation of the structure in audio expressed as a textual content and summary descriptor. A mechanism to solve a need most commonly felt in audio (linking descriptions and transcripts with audio content) is generalized to form a very powerful generic method for use across MPEG-7. Combined low-level and high-level descriptors are intended to facilitate retrieval of audio content. Thus, the proposed description schemes and descriptors are intended to simplify description and retrieval of audio content.

References

- [1] S. Dharanipragada, M. Franz, S. Roukos, Audio indexing for broadcast news, in: Proceedings of the Seventh Text Retrieval Conference (TREC-7), NIST special publication, 1998.

Table 1
Tree locators for the abstract SGML tree

	Element	Tree locator
 <pre> graph TD A[A] --> B[B] A --> C[C] B --> D[D] B --> E[E] C --> F[F] C --> G[G] </pre>	A	1
	B	1 1
	C	1 2
	D	1 1 1
	E	1 1 2
	F	1 2 1
	G	1 2 2

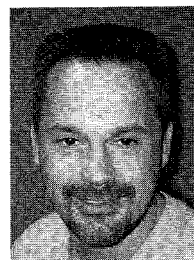
- [2] J. Foote, An overview of audio information retrieval, in: L. Wyse (Ed.), *Multimedia Systems*, Vol. 7, Springer, Berlin, 1999, pp. 2–10.
- [3] M. Fowler, K. Scott, I. Jacobson, *Uml Distilled: Applying the Standard Object Modeling Language*, Addison-Wesley, Reading, MA, 1997.
- [4] P. Garner, J. Charlesworth, MPEG-7 Seoul meeting, Proposal M4458, March 1999.
- [5] ISO/IEC 10744, ISO/IEC Copyright Office, Geneva.
- [6] S.E. Johnson et al., Spoken document retrieval for TREC-7 at Cambridge University, in: *Proceedings of the Text Retrieval Conference (TREC-7)*, NIST special publication, 1998.
- [7] A. Lindsay, On behalf of the DICEMAN consortium, "DICEMAN Audio DS", MPEG-7 Lancaster Meeting, Proposal P187, P188, February 1999.
- [8] ISO/IEC JTC1/SC29/WG11/N2727, MPEG-7 Requirements Document V.8, Seoul meeting, March 1999.
- [9] ISO/IEC JTC1/SC29/WG11/N2728, MPEG-7 Applications Document V.8, Seoul meeting, March 1999.
- [10] K. Ng, V.W. Zue, Subword unit representations for spoken document retrieval, in: *Proceedings Eurospeech*, 1997.
- [11] L. Rabiner, B.H. Juang, *Fundamentals of Speech Recognition*, Wiley, New York, 1993.
- [12] P. Salembier et al., Video DS, MPEG-7 Lancaster Meeting, Proposal P185, P186, February 1999.
- [13] M.A. Siegler, M.J. Witbrock, S.T. Slatery, K. Seymore, R.E. Jones, A.G. Hauptmann, Experiments in spoken document retrieval at CMU, in: *Proceedings of the sixth Text Retrieval Conference (TREC-6)*, NIST special publication, 1998.
- [14] S.W. Smoliar, J.D. Baker, T. Nakayama, L. Wilcox, Multimedia search: an authoring perspective, in: *Proceedings of the First International Workshop on Image Databases and Multimedia Search*, IAPR, August 1996, pp. 1–8.
- [15] S. Srinivasan, D. Petkovic, Phonetic confusion matrix based spoken document retrieval, in: *Proc. SIGIR2000*, Athens, Greece, July 2000, ACM, New York.
- [16] S. Srinivasan, D. Petkovic, MPEG-7 Lancaster Meeting, Proposal P148, P150, February 1999.
- [17] L. Ward, N. O'Connor, Using MPEG-7 content and the internet to globalise the archive content business, in: *IBC99 Proceedings*, IBC, London, 1999, pp. 380–385.
- [18] M. Wechsler, Spoken document retrieval based on Phoneme Recognition, Thesis, Swiss Federal Institute of Technology, Zurich, 1998.



Adam T. Lindsay is the principal investigator of the multimedia research division at Starlab, a research firm based in Brussels, Belgium. He joined Starlab after obtaining his M.S. at the Massachusetts Institute of Technology Media Lab. His current research is on applying MPEG-7-style metadata to multimedia to make it more intelligent about itself. Adam is an active member of the MPEG-7 standardization group, where he serves as the editor of the MPEG-7 Applications Document and as the leader in MPEG-7 Audio activities.



Savitha Srinivasan is a researcher at the Visual Media Group at IBM's Almaden Research Center, in San Jose, California. She joined the Speech Applications Group at IBM's T.J. Watson Research Center, New York after receiving an MS degree in computer science from Pace University. Her current research interests include speech recognition and other synergistic techniques for knowledge mining in audio, with the practice of object-oriented techniques.

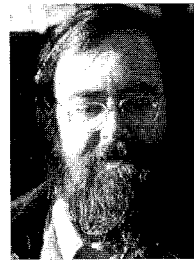


Jason P.A. Charlesworth is a researcher in the Retrieval group at the Canon Research Center in Guildford, England. His current research interests include speech recognition, spoken document retrieval and statistical information retrieval. Jason received a PhD in theoretical physics at Cambridge University and did postdoctoral research at London and Cornell.



Philip N. Garner is a researcher at Canon Research Centre Europe Ltd in the UK. His research interests include speech recognition, pattern recognition and statistics. Before joining CRE, Philip was at the Defense Evaluation and Research Agency in Malvern, UK, and studied Elec-

tronic Engineering at Southampton University. He is a Chartered Engineer.



Werner Kriechbaum, after being trained as a neurobiologist, spent some time in an earlier life to work on the processing of auditory signals in insects, visually guided behavior of flies and the electrophysiology of human cognition. Almost 10 years ago he joined IBM Germany to work

on real-time Unix operating systems and their performance, the description and visualization of large and complex data sets, and the perception-based description of multi-media data. The last topic led to his involvement in the MPEG-7 standardization effort. His main research interests are human perception and the description of art.