# Semantic Retrieval of Multimedia Data

Samira Hammiche
LIRIS – University Claude Bernard Lyon 1
43 bld du 11 Novembre 1918
69220 Villeurbanne – France
shammich@bat710.univ-lyon1.fr

Salima Benbernou
LIRIS – University Claude Bernard Lyon 1
43 bld du 11 Novembre 1918
69220 Villeurbanne – France
sbenbern@bat710.univ-lyon1.fr

Mohand-Saïd Hacid
LIRIS – University Claude Bernard Lyon 1
43 bld du 11 Novembre 1918
69220 Villeurbanne – France
mshacid@bat710.univ-lyon1.fr

Athena Vakali
Department of Informatics
Aristole University of Thessaloniki
54006 Thessaloniki – Greece
avakali@csd.auth.gr

## ABSTRACT

This paper deals with the problem of finding multimedia data that fulfill the requirements of user queries. We assume both the user query and the multimedia data are expressed by MPEG-7 standard. The MPEG-7 formalism lacks the semantics and reasoning support in many ways. For example, the search of the implicit data can not be achieved, due to its description based on XML schema. We propose a framework for querying multimedia data based on a tree embedding approximation algorithm, combining the MPEG-7 standard and an ontology.

## Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: *Search Process*

## General Terms

Design

## Keywords

MPEG-7, Multimedia Data, Ontologies, Tree embedding, Approximation

## 1. INTRODUCTION

In the recent years, there has been an important increase of multimedia data available in digital format. However, the extraction of useful information from the multimedia data and the manipulation of this information in practical systems such as multimedia search engines are still open problems. The most important barrier has been the lack of a comprehensive, simple and flexible representation of multimedia data. The MPEG-7 standard [14] was proposed as a first step towards solving this problem by providing a generic library of descriptions to cover almost all multimedia data types. Thus, a multimedia data can be described by an MPEG-7 description which is XML [4] based.

To fully exploit the variety of the MPEG-7 compliant descriptions, a system that can use and reason about the descriptions is required. The MPEG-7 formalism does not allow reasoning mechanism because it is defined solely in XML schema [5, 1]. XML schema is suitable for expressing the syntax, structural, cardinality and typing constraints required by MPEG-7. However, it displays some limits when we need to extract implicit information. This is due to the fact that there is neither formal semantics nor inference capabilities associated with the elements of an XML document.

This paper focuses on the *reasoning* issue to search multimedia data. In this setting, the MPEG-7 standard is extended by an ontology for the domain knowledge representation allowing some reasoning mechanism over the MPEG-7 description. This information retrieval problem from MPEG-7 descriptions can be stated as follows: given a user multimedia data query $\mathcal{Q}$ and an MPEG-7 database $\mathcal{D}$ describing multimedia data extended by an ontology $\mathcal{T}$ of a domain, retrieve multimedia data that could semantically answer the client query. In the proposed framework, both the user query and the database description are specified as *ordered labeled trees*. So, the problem of searching multimedia data against MPEG-7 description collections is regarded as the problem of a well known *tree embedding* augmented by the subsumption reasoning over the ontology. The tree embedding algorithm is used as an effective multimedia data retrieval tool.

The rest of the paper is organized as follows: Section 2 gives an overview of the MPEG-7 standard. Section 3 presents our motivation in using reasoning mechanism over ontologies to semantic search of multimedia data. The representations of query and MPEG-7 description are provided in Section 4. Section 5 outlines our framework for semantic search of multimedia data and illustrates an ontology example. Section 6 presents the algorithm for the semantic search. The algo-


*MMDB'04,* November 13, 2004, Washington, DC, USA.

rithm combines subsumption reasoning and tree embedding approximation mechanism. In Section 7, we survey related work. We conclude in Section 8.

# 2. PRELIMINARIES

## 2.1 MPEG-7 overview

MPEG-7, formally named "*Multimedia Content Description Interface*", is an ISO/IEC [1] standard that aims at describing the multimedia data content by attaching to them metadata [14, 15]. It specifies a standard set of description tools that consists of Descriptors (Ds), Description Schemes (DSs), a Description Definition Language (DDL) and System Tools. Descriptors represent features or attributes of multimedia data such as color, texture, textual annotation and media format. Description Schemes specify the structure and semantic of the relationships between their components which may be both Ds and DSs. The Description Definition Language defines and extends the Ds and DSs. It is XML schema based. Finally, System Tools are used to satisfy the storage and transmission requirements.

The description of multimedia data, which is used for content retrieval comprises the following areas [7, 13]:

- *Description of the storage media:* file and coding format, image size, audio quality, etc.
- *Creation and production information:* creation date and location, title, genre, etc.
- *Content semantic description:* events, concepts, objects, etc.
- *Content structural description:* shot and key frames with color, texture and motion features, etc.
- *Metadata about the description:* author, version, creation date, etc.

All these types of information can be handled by MPEG-7 description schemes.

## 2.2 MPEG-7 Example

Let us consider an extract of the *VideoSegment DS* depicted in Figure 1.

The *VideoSegmentType* extends the *SegmentType*. It is a sequence of child element declarations. The *sequence* group element forces the child elements of *VideoSegmentType* to appear in the instance document exactly in the order which they are declared in the DDL. The first one is *MediaTime* which describes the temporal localization of the video segment. It is then followed by a list of alternative child elements, indicated by a *choice* group element. The element *choice* says that only one of its children has to appear in an instance.

Let us now introduce an example of *VideoSegmentType* describing the segment tree hierarchy shown in figure 2. In this examples [2], the video root represents a video sequence with three scenes: *dribble*, *shoot* and *goal*. The root video is split into three video segments where the shot start time and duration are given. Textual annotations are also associated with each video segment using the *TextAnnotation D*.

Figure 3 shows the MPEG-7 description of the video example given in figure 2.

```
<complexType name="VideoSegmentType">
  <complexContent>
    <extension base="mpeg7:SegmentType">
      <sequence>
        <choice minOccurs="0">
          <element name="MediaTime"  type="mpeg7:MediaTimeType"/>
          <element name="TemporalMask"
                   type="mpeg7:TemporalMaskType"/>
        </choice>
        <choice minOccurs="0" maxOccurs="unbounded">
          <element name="VisualDescriptor"
                   type="mpeg7:VisualDType"/>
          <element name="VisualDescriptionScheme"
                   type="mpeg7:VisualDSType"/>
          <element name="VisualTimeSeriesDescriptor"
                   type="mpeg7:VisualTimeSeriesType"/>
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>
```

**Figure 1: Extract of *VideoSegment DS***

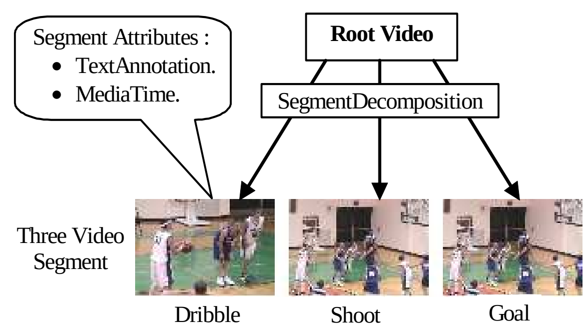


**Figure 2: Example of segment tree**

## 2.3 MPEG-7 restrictions

The MPEG-7 standard aims to standardize tools (Ds and DSs) for describing multimedia data. The extraction of useful information from multimedia data and the usage of this information in practical systems such as multimedia search engines are not considered in MPEG-7. These two technologies, extraction and retrieval, are both variable and competitive.

Moreover, MPEG-7 does not convey a formal semantics since its DDL is XML schema based. XML is mainly used to provide a structure for documents and does not impose any common interpretation of the data contained in the document. Thus, XML schema enables to add structure (new Ds and DSs) to the MPEG-7 standard but it does not express the meaning of the structure. For example, a video segment cannot be indexed according to its audio-visual gender (report, interview, film) or according to its general thematic (sport, science, politic). Additionally, inference mechanisms are not supported by MPEG-7. For example, given two descriptions about the same subject, one cannot say which one

[1] International Standard Organization/International Electro-technical Committee

[2] The video segments are extracted from the material found at http://ghs.pasco.k12.fl.us/clips/clips.html

```
<Mpeg7>
  <Description xsi:type="ContentEntityType">
   <MultimediaContent xsi:type="VideoType">
    <Video id="root video">
            <!-- MediaTime description -->
      <MediaTime>
        <MediaTimePoint>T00:00:00 </MediaTimePoint>
        <MediaDuration> PT0M30S </MediaDuration>
      </MediaTime>
      <!-- Temporal Decomposition description -->
      <TemporalDecomposition gap="false" overlap="false">
        <VideoSegment id="dribble">
            <!-- TextAnnotation description -->
         <TextAnnotation>
             <FreeTextAnnotation>dribble </FreeTextAnnotation>
         </TextAnnotation>
         <MediaTime>
           <MediaTimePoint>T00:00:00 </MediaTimePoint>
           <MediaDuration> PT0M15S</MediaDuration>
        </MediaTime>
      </VideoSegment>
      <VideoSegment id="shoot">
            <!-- TextAnnotation description -->
            <!-- MediaTime description -->
      </VideoSegment>
      <VideoSegment id="goal">
            <!-- TextAnnotation description -->
            <!-- MediaTime description -->
     </VideoSegment>
    </TemporalDecomposition>
   </Video>
  </MultimediaContent>
 </Description>
</Mpeg7>
```

**Figure 3: Example of MPEG-7 description**

is more or less explicit than the other. This is due to the fact that implicit knowledge is not represented. Adding semantic knowledge to MPEG-7 descriptions is needed in order to enable reasoning over multimedia data.

## 3. MOTIVATING EXAMPLE

Assume we have a collection of MPEG-7 descriptions storing metadata about *insect* images. Figure 4 shows a subset of such a collection. The query to retrieve all images regarding "*flying insect*" stands for finding all images that contain the textual annotation in the description. On the contrary, the semantic search of the query will allow to find all images containing a different type of flying insect.

The query is represented as a tree as depicted in figure 5. Obviously, the only answer to the query is the image **a**. Images **b** and **c** do not satisfy the query because they contain instances of *flying insects* that do not match exactly the query requirements. However, if we specify an ontology that represents concepts of the *insect* domain as an ordered relation, as depicted in figure 6, then images **b** and **c** can be returned as answers to the query, provided that we have an appropriate inference mechanism allowing to draw such a deduction. The subsumee concepts (*bees* and *butterfly*) are answers to the query (*flying insect*), then, we obtain a specific answer to a general query.

In the proposed framework, reasoning and embedding tree mechanisms are used in a complementary way.

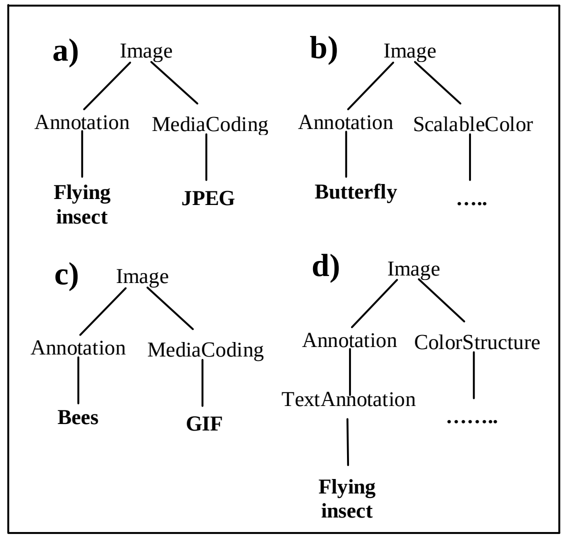


**Figure 4: Extracts of images MPEG-7 descriptions**

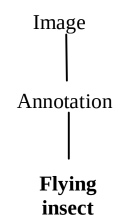


**Figure 5: Query MPEG-7 description**

## 4. REPRESENTING AND QUERYING MPEG-7 DESCRIPTIONS

An MPEG-7 description is an instance of a description scheme, so it represents an XML document. Various models have been proposed to represent XML documents. The W3C [3] proposed a generic model named Document Object Model (DOM) [2]. In this model, an XML document can be represented as an ordered labeled tree.

*Definition 1.* **(Rooted tree)**. A rooted tree is a structure $T = (N, E, root(T))$ which consists of a finite set of nodes $N$, a finite set of edges $E$ and a node $root(T) \in N$ that forms the root node of $T$. $E$ must satisfy the following conditions:

1. the root has no parent;
2. every node of the tree, except the root, has exactly one parent;
3. all nodes are reachable from the root.

*Definition 2.* **(Labeled rooted tree)**. A labeled rooted tree $T = (N, E, root(T))$ is a rooted tree in which a label is defined for every node $u \in N$.

*Definition 3.* **(Path)**. Let $T = (N, E, root(T))$ be a rooted tree. If $n_1, n_2, .., n_m$ is a sequence of nodes in $T$ where $n_1$ is the parent of $n_2$, $n_2$ is the parent of $n_3$, etc., then $n_1, n_2, .., n_m$ is a path of length $m - 1$.

[3]World Wide Web Consortium

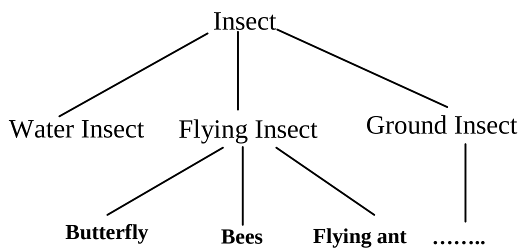


**Figure 6: Ontology domain**

*Definition 4.* **(Child node, Leaf, Internal node, Descendant, Ancestor)**. Let $T = (N, E, root(T))$ be a labeled rooted tree.

The nodes of $T$ with a common parent are *children* of their parent, denoted by: $children(u) = \{v \in N | (u, v) \in E\}$.

A node without children is a leaf. Nodes which are not leaves are internal nodes. Leaves are sometimes called external nodes.

A node $n$ is an ancestor of a node $u$ (alternatively, $u$ is a descendant of $n$) if there is a path from $n$ to $u$.

*Definition 5.* **(Ordered tree)**. An ordered tree is a tree in which both the ancestor (parent-child) relationship and the left-to-right ordering among siblings are significant.

## 4.1 Tree based MPEG-7 description

An MPEG-7 description is an XML document. It can be represented as an ordered labeled tree $T = (N, E, root(T))$ with three kinds of nodes:

- $N_e$: element nodes which represent the non-leaf nodes with one label, the name;
- $N_t$: text nodes which represent the leaf nodes with one label, the value;
- $N_a$: attribute nodes which represent the leaf nodes with two labels, the name and the value.

$root(T)$ represents the described MPEG-7 data . For example, if the MPEG-7 description deals with a video data, then the label of $root(T)$ is *video.*

The elements of the MPEG-7 description tree are:

- $N$=$N_e \cup N_t \cup N_a$;
- $E$: the edges capture elements nesting in the XML document;
- The labels of the intermediate nodes represent the element names corresponding to the MPEG-7 metadata. But, the leaves of the tree are filled with the value of extracted features like the color histogram value or the annotation concepts used in the search process.

To illustrate this formal representation, figure 7 shows an extract of the MPEG-7 tree representation corresponding to the one given in figure 3.

## 4.2 Tree interpretation of MPEG-7 queries

The formulation of a query should be simple even for occasional users and should require only a partial knowledge of the structure and the content of the multimedia data. A

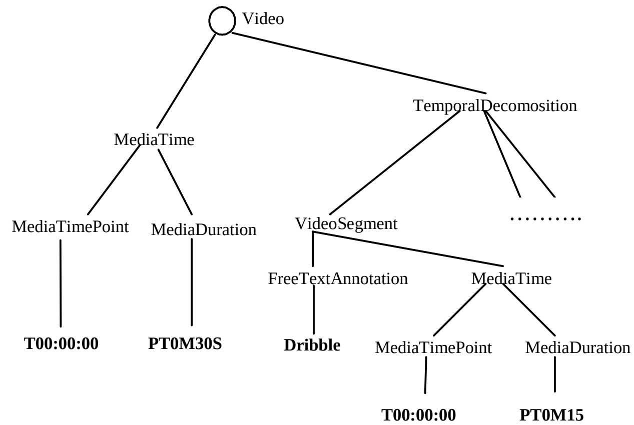


**Figure 7: Example of MPEG-7 description tree**

retrieval process should retrieve not only the results that match exactly the query, but also answers with a structure and content similar to the conditions specified by the query.

Suppose the user searches for multimedia data on the basis of their structure and content. An example of such a query: "find all video composed of two segments and annotated with the words 'dribble goal"'. The query will then be translated into MPEG-7 formalism and then represented as a labeled ordered tree (see figure 8 ).

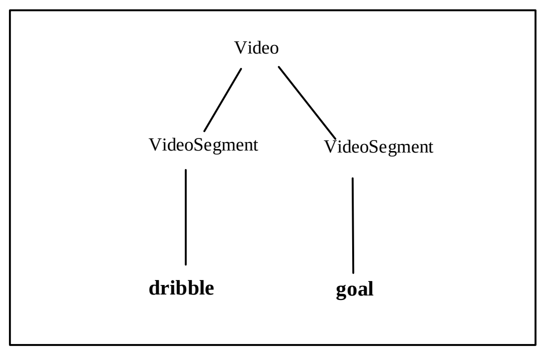


**Figure 8: Example of MPEG-7 query tree**

# 5. SEMANTIC SEARCH

In this section, we provide an outline of the framework for searching multimedia data based on their MPEG-7 description. The architecture of this framework is depicted in figure 9.

In this framework, we define domain specific ontology. We integrate this ontology in the MPEG-7 database during the indexing process. The indexers (people responsible for providing metadata for indexing the multimedia content) use terms and relationships with the precise semantic that the ontology contains for indexing. There are four different components:

- MPEG-7 Metadata Generator: this component is responsible of the metadata creation process which is guided by the appropriate ontology. It receives as input a multimedia data and text annotation and pro-

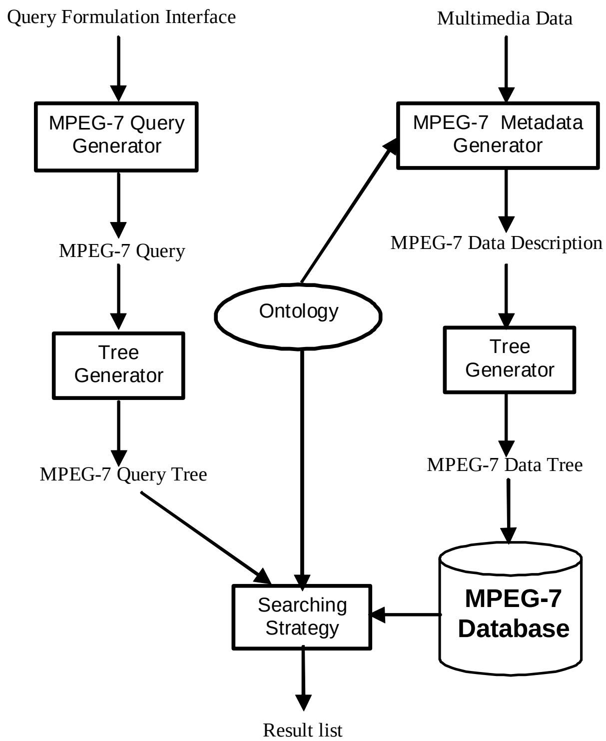


**Figure 9: Framework of the retrieval process**

duces as output the corresponding MPEG-7 metadata description. Metadata can be extracted from the multimedia data automatically or manually. Automatic extraction algorithms fit well for low level features such as colors or file size. For higher semantic features such as objects and their properties, manual annotation are required;

- MPEG-7 Query Generator: the query formulation interface (see figure 9) provides the users with an interactive, easy to use, and graphical interface. The user query is translated into its corresponding form in MPEG-7 by the MPEG-7 Query Generator component. The resulting query is represented as an MPEG-7 description;
- Tree Generator : it models the MPEG-7 description as a labeled ordered tree. The details of this process were given in the previous section;
- Searching Strategy: this component is based on the tree embedding formalism. The MPEG-7 query tree is approximatively embedded into the MPEG-7 data trees, stored in the MPEG-7 database, to find the multimedia data that are answers to the user query. The approximate embedding algorithm makes use of the ontology to reason about MPEG-7 descriptions and deduce implicit information.

Due to the space limitation, we will focus only on the two latter components, namely Tree Generator and Searching Strategy.

In the following, we illustrate an ontology example.

**Ontology example**

We have defined an ontology for the description of basketball matches. The definition of the ontology is based on the rules of FIBA (the international organization responsible for the definition of basketball rules) [3]. We defined our ontology using description logic. We have separated five areas in a basketball match: time periods (first time, second time, timeout, penalty period, etc.), places (stadium), actor roles (player, referee, coach, etc.), teams (home team, visiting team) and events (fault, goal, pass, shoot, etc.).

Let us introduce a fraction of the ontology dealing with actor roles in a basketball match (see figure 10).

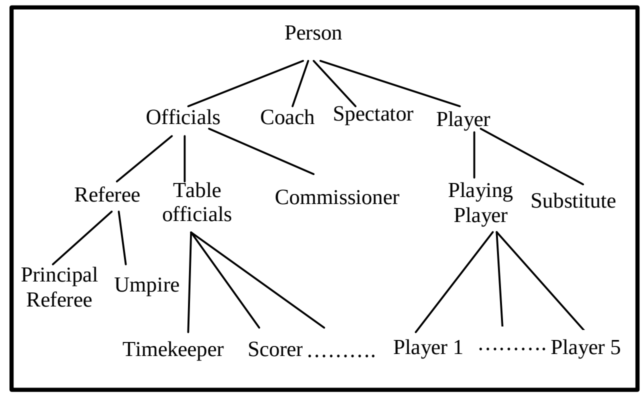


**Figure 10: Ontology concepts for person roles in a basketball match**

According to figure 10, the roles in a basketball game are: Officials, Coach, Player and Spectator. An Official instance may be a Referee, a Table Officials or a Commissioner. A Referee may be a Principal Referee or Umpire. A player instance may be a Playing Player or a Substitute.

# 6. THE TREE EMBEDDING PROBLEM

We have presented a way to interpret both multimedia data and query as ordered labeled trees. With this interpretation, the problem of answering a query can be seen as the problem of embedding a query tree in the data tree with respect to the domain ontology.

## 6.1 The Tree Embedding Approximation Problem

The goal in designing the current framework is to approximately embed the MPEG-7 query tree into the MPEG-7 description tree in such a way that the labels and the ancestorship of the nodes are preserved. Then, we allow the MPEG-7 description tree to have intermediate nodes in those found in the MPEG-7 query tree, but their predecessor-successor relationship must still hold in the tree. The labels are preserved if they are equal or the data node label is subsumed by the query node label. An example of allowable embedding is shown in figure 11. Tree $P$ is approximately embedded in the tree $T$.

*Definition 6.* **(Unordered tree inclusion).** Tree inclusion problems are defined via embeddings between trees. Let $P = (V, E, root(P))$ and $T = (W, F, root(T))$ be trees. An injective function $f$ from the nodes of $T$ is an embedding of

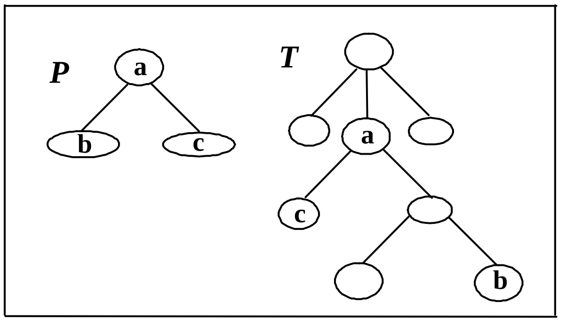


**Figure 11: Example of allowable embedding**

$P$ into $T$, if it preserves labels and ancestorship. That is, for all nodes $u$ and $v$ of $P$ we require that:

1. $f(u) = f(v)$ if and only if $u = v$,
2. $label(u) = label(f(u))$, and
3. $u$ is an ancestor of $v$ in $P$ if and only if $f(u)$ is an ancestor of $f(v)$ in $T$.

We say that $P$ is an *unordered included tree* of $T$ if there is an embedding of $P$ in $T$.

*Definition 7.* **(Tree Embedding Approximation)**. Given a query tree $q$ and a data tree $d$, an approximate embedding of $q$ in $d$ is a partial injective function $f : node(q) \rightarrow node(d)$ such that $\forall q_i, q_j \in q$:

1. $\text{sim}(label(q_i), label(f(q_i))) > 0$, where sim is a similarity function that returns a score normalized in [0,1] stating the similarity between the two given labels.
2. $q_i$ is a parent of $q_j \Rightarrow f(q_i)$ is an ancestor of $f(q_j)$.

*Definition 8.* **(Postorder traversal)**. A postorder traversal is a technique for recursively processing the nodes of a tree in which any subtrees are processed first, then the root is processed.

*Definition 9.* **(Match)**. Let $f$ be an embedding according to the definition 7. Let $u$ be a node of a query. We call the data node $f(u)$ a match of $u$.

### 6.2 Algorithm for the tree embedding approximation $\mathcal{TEA}$

For each query node, data nodes fetched are those that are members of a potential embedding of the query tree in the data tree. The query tree $Q$ is processed in a bottom up fashion following the postorder numbering. The postorder ensures that for each child node of the current query node, the matching set of that child node is calculated.

The query leaves and the other query nodes are processed differently. For each query leaf $q_i$ (lines 5–8), the algorithm calculates its matching set $M(q_i)$. This set includes the data leaves that have the same label as $q_i$ and those for which the label corresponding is subsumed by the label of $q_i$ (condition in line 6). A subsumption algorithm is used to check the subsumption between concepts (labels). Relating the other nodes (internal nodes and the root), the $\mathcal{TEA}$ algorithm incrementally builds an embedding of the current query node. Assume that $q_i$ is a current internal query node. The algorithm fetches all data nodes that have the same label as $q_i$ (line 10), then it tries to embed the query subtree rooted at $q_i$ into all the data subtree rooted at $d$ (lines 12–18). Then, for the same query node, we can obtain several embedding in the data tree.

As the problem of tree embedding is NP-hard [10] and in order to limit data nodes quantity to be considered in the searching process, the $\mathcal{TEA}$ algorithm starts from the bottom, looking for similarity matches between query and data leaves. If there is no match of any query leaf or internal node (line 25), then the data tree doesn't satisfy the query and the search process over this data tree is stopped (the condition of the While loop is false). However, if all query nodes are processed ($i = |Q|$ and $Match = True$), then a match is found. The data tree satisfies the query tree and the multimedia data corresponding to this data tree is returned.

### 6.3 Illustration of the algorithm

The domain ontology is introduced in the framework at two levels. The first is the indexing process. The second is during the search strategy represented by tree embedding algorithm.

To illustrate how the algorithm processes to get answer to a given MPEG-7 query tree, using additional background knowledge represented in the ontology, let us introduce an example (see figure 12).

In this example, we ask for basketball video extract where the referee whistle a fault. The tree representation of such a query is illustrated in figure 12 (a). Figure 12 (b) is a part of an MPEG-7 description stored in the MPEG-7 database. $Q_1, Q_2$ and $Q_3$ are the postorder numbering of the query nodes. $D_1, D_2, D_3$, etc. are the postorder numbering of the MPEG-7 description nodes. The dashed arrows represent the matching function between a query node and the appropriate data node. It preserves the ancestor-child relationships and the concept meaning also. The algorithm calculates the matching of the leaves query. The match of $Q_1$, noted $M(q_1)$ is the set $\{D_4\}$ although $label(Q_1)$ is different from $label(D_4)$. In this case, we check wether the data label node is subsumed by the query label node (ie, the two labels have the same meaning). This verification is done using the defined ontology of figure 10. The process of embedding is continued until processing the root query. Note that we have the same case when processing the query node $Q_3$. The label data node $D_8$ is subsumed by the label query node with respect to the ontology defining the events in a basketball match (a walk is a fault). In the example, $M(Q_4) = \{D_4, D_6, D_8, D_{12}\}$ is not empty, so the query rooted at $Q_4$ is embedded in the data subtree rooted at $D_{12}$ and this subtree constitute an answer to the query. The embedding is approximative since the algorithm preserves the meaning of the query's concepts and the ancestor-child relationship. We do not have an exact embedding.

## 7. RELATED WORK

Our work relates to query language for MPEG-7 description and tree embedding algorithms.

### 7.1 MPEG-7 query languages

The arrival of the MPEG-7 standard was an important evolution in modelling and representing the multimedia data. Such descriptions are however useful only once they can be

**Algorithm 1** Searching embedding between query tree in data tree

```
Input : Q – a list of query nodes sorted in postorder
        D – a list of data nodes
Output : Boolean to indicate if a match is found
Match := True                     // We suppose that there is a match in the beginning
i := 1                  // i is an index representing the postorder numeration
while (Match = True and i ≤ |Q| ) do { // |Q| is the number of query node}
  M(q_i) := {∅}                   // The match set belonging to q_i
  if ( children(q_i) = {∅} ) then {// Process query leaves}
    for all d ∈ leaves(D) such that ( label(d) = label(q_i) or label(d) ⊑ label(q_i) ) do { // Make the subsumption test if label(d) ≠ label(q_i) }
      M(q_i) := M(q_i) ∪ {d}              // Construct the matching set for the leaves query
    end for
  else {// Process internal nodes of the query}
    for all d ∈ D such that (label(d) = label(q_i)) do
      S := {∅}                    // The set of data node combinations
      for all q_j ∈ children(q_i) do
        S' := {∅}                 // The set of child node combinations
        for all d_m ∈ M(q_j) such that (d is an ancestor of d_m) do
          S' := S' ∪ {d_m}
        end for
        S := S ∪ S'
      end for
      if S ≠ {∅} then
        M(q_i) := M(q_i) ∪ {S ∪ {d}}
      end if
    end for
  end if
  if (M(q_i) = {∅}) then
    Match := False          // There is a query node that does not match ⟹ stop the processing
  else
    i := i + 1              // Process the next query node in postorder
  end if
end while
return Match;
```

correctly and easily retrieved via an adapted query langauge.

Currently, the approaches used for querying MPEG-7 descriptions can be classified into two main groups [8]. The first group uses directly XQuery [6], a standard defined by W3C, as the retrieval language. The choice of XQuery is motivated by the fact that it fits as flexible query language to extract data from XML based documents. However, XQuery requires an advanced knowledge of the MPEG-7 details in order to express a precise query. So, the user is getting lost in describing the details of MPEG-7 rather than focusing on the essence of his retrieval requirements.

The second group defines new query languages or adapts the existing ones to support retrieval in the MPEG-7 descriptions. In [9], access methods based on an inference network model that allows deduction of semantic and implicit information, are developed for searching video MPEG-7 descriptions. In [12], a specification of crucial query issues in MPEG-7 XML queries is proposed, which takes into account the implicit information to be extracted from the MPEG-7 descriptions.

The semantic aspects are not directly expressible in XQuery, and therefore each of these two approaches [9, 12] proposes their own specialized query language. However, in [8] an XQuery adaptation for MPEG-7 document retrieval called *SVQL (Sematic Views Query Language)*, was proposed. *SVQL* is a high level query language that allows users to express their requirements following the semantic view model (*PhysicalView, ProductionView, ThematicView, VisualView and AudioView*) in a concise, abstract and precise way.

## 7.2 Tree embedding algorithms

The problem of tree and graph embedding has been studied in both the graph theory community and the formal query answering models community. The work that is close to ours is the one presented in [10]. It introduces the tree inclusion problem. Various algorithms were proposed to solve the different classes of tree inclusion problems. For the most general case, these algorithms have polynomial complexity. The exact ordered tree embedding problem has been studied in [11, 16, 17] and is solvable in polynomial time. However, we are not interested in an ordered inclusion, because ordering in the query data may be inconsistent. Even, if it was consistent, the order might not be known to the user.

Schlieder and Naumann [19] extended the exact embedding problem and studied the approximate embedding problem for unordered trees. An approximate tree embedding algorithm was introduced in [18, 19] to finds all the approximate matches and rank them according to their similarity to the query.

Our approach is similar to the approach presented in [19]. It relates to the tree embedding problem for searching multimedia data based on their MPEG-7 description. It combines

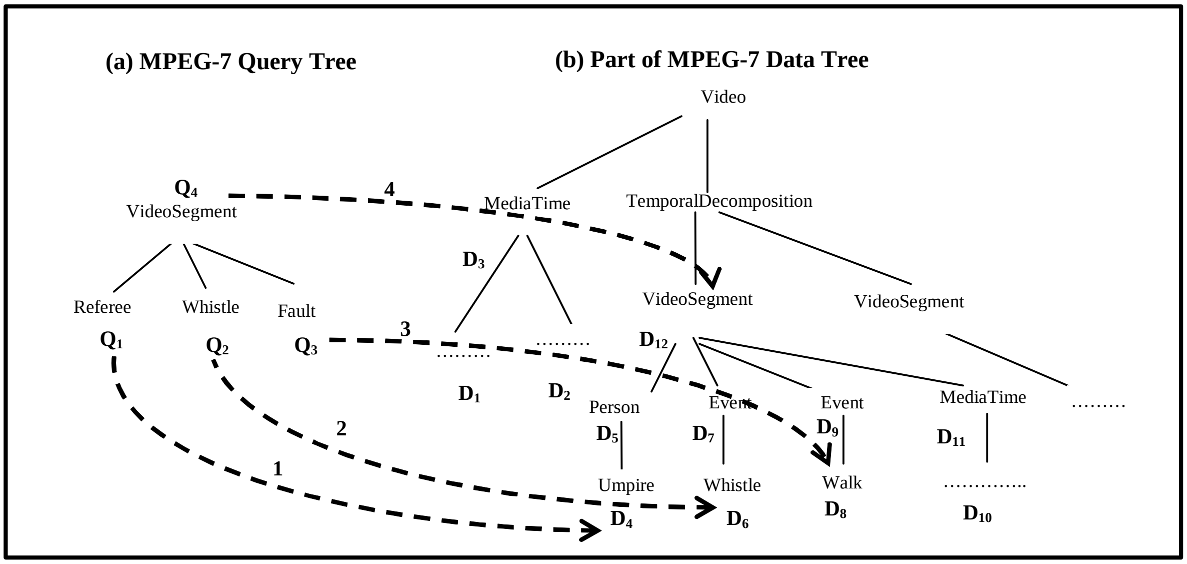


**Figure 12: Embedding of an MPEG-7 query tree in an MPEG-7 data tree**

structural and semantic similarity to get an approximative embedding. The semantic similarity measure is the equality of labels and the subsumption between the query node and the data node, then, we preserve the meaning of concepts.

# 8. CONCLUSION

We have presented a framework for the semantic retrieval of multimedia data. The search strategy is based on a tree embedding algorithm extended with a domain ontology to reason about the multimedia data description, represented by means of MPEG-7 formalism. The algorithm proposed (Tree Embedding Approximation $\mathcal{TEA}$) finds the approximate matches of the query tree nodes and return a boolean. This boolean is set to true if there is a matching between the query tree and the data tree, to false otherwise.

Since the proposed $\mathcal{TEA}$ algorithm finds all the approximate matches of the root query node, we plan to improve the semantic similarity measure to evaluate the results of queries against MPEG-7 descriptions. We also plan to propose a ranking function to rank the result by their relevance to the query.

# 9. REFERENCES


[1] XML Schema Part 2: Datatypes, W3C recommandation, 2 May 2001. http://www.w3.org/TR/xml-schema-2/.

[2] Document Object Model (DOM). http://www.w3.org/DOM/, 2004.

[3] FIBA – International Basketball Federation. Basketball game rules. http://www.fiba.com, 2004.

[4] XML Extensible Markup Language. http://www.w3.org/XML/, 2004.

[5] XML Schema Part 1: Structures, W3C recommandation, 2 May 2001. http://www.w3.org/TR/xml-schema-1/, 2004.

[6] XQuery 1.0 : An XML Query Language. W3C Working Draft 12 November 2003. http://www.w3.org/TR/xquery/, 2004.

[7] A. Benitez, H. Rising, C. Jörgensen, R. Leonardi, A. Bugatti, K. Hasida, R. Mehrotra, A. M. Tekalp, A. Ekin, and T. Walker. Semantic of multimedia in MPEG-7. In *Proceedings of IEEE 2002 Conference on Image Processing (ICIP-2002)*, Rochester, New York, USA, September 2002.

[8] N. Fatemi, O. A. Khaled, and G. Coray. An XQuery adaptation for MPEG-7 documents retrieval. *XML conference and Exposition*, December 2003.

[9] A. Graves and M. Lalmas. Video retrieval using an MPEG-7 based inference network. In *The $25^{th}$ Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 339–346. ACM Press, August 2002.

[10] P. Kilpläinen. *Tree Matching Problems with Applications to Structured Text Databases.* PhD thesis, Dept. of Computer Science, University of Helsinki, 1992.

[11] P. Kilpläinen and H. Mannila. Oredered and unordered tree inclusion. *SIAM Journal of Computing*, 24(2):340– 356, April 1995.

[12] P. Liu and L. Sushu. Queries of digital descriptions in MPEG-7 and MPEG-21 XML documents. *XML Europe 2002, Barcelona, Spain*, May 2002.

[13] M. Lux and J. Becker. XML and MPEG-7 for Interactive Annotation and Retreival using Semantic Metadata. In *Proceedings of I-Know'02*, volume 8, pages 965 – 984. J.UCS, 2002.

[14] J. Martinez. MPEG-7 Overview. Multimedia Content Description Interface. http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm/, 2003.

[15] J. M. Martinez. Overview of MPEG-7 Description Tools, Part 2. *IEEE MultiMedia*, pages 83– 93, 2002.

[16] H. Meuss. Indexed tree matching with complete answer representations. In *Proceedings of the 4th International Workshop on Principles of Digital*

*Document Processing*, pages 104 – 115. Springer–Verlag, 1998.

[17] R. Ramesh and I. Ramakrishnan. Nonlinear pattern matching in trees. *Journal of the ACM*, 39(2):295 – 316, 1992.

[18] T. Schlieder. ApproXQL: Design and implementation of an approximate pattern matching language for XML. Technical report, Freie Universitat Berlin, May 2001.

[19] T. Schlieder and F. Naumann. Approximate tree embedding for querying XML data. In *Proceedings of the ACM SIGIR Workshop on XML and Information Retrieval*, July 2000.