

# Cross Document Ontology based Information Extraction for Multimedia Retrieval

Dennis Reidsma<sup>1</sup>, Jan Kuper<sup>1</sup>, Thierry Declerck<sup>2</sup>, Horacio Saggion<sup>3</sup>, and Hamish Cunningham<sup>3</sup>

<sup>1</sup> University of Twente, Dept. of Computer Science, Parlevink Group, P.O. Box 217, 7500 AE Enschede, the Netherlands,

{dennisr,jankuper}@ewi.utwente.nl

<sup>2</sup> DFKI GmbH, Language Technology Lab, Saarbruecken

declerck@dfki.de

<sup>3</sup> University of Sheffield, Dept. of Computer Science, NLP Group

{h.saggion,hamish}@dcs.shef.ac.uk

**Abstract.** This paper describes the MUMIS project, which applies ontology based Information Extraction to improve the results of Information Retrieval in multimedia archives. It makes use of a domain specific ontology, multilingual lexicons and reasoning algorithms to automatically create a semantic annotation of sources. The innovative aspect is the use of a cross document merging algorithm that combines the information extracted from separate textual sources to produce an integrated, more complete, annotation of the material. This merging and unification process uses ontology based reasoning and scenarios which are extracted from annotated sources.

The algorithms presented here have been implemented in a working demonstration prototype and have been applied on material from the Euro 2000 Soccer Championships.

## 1 Introduction

The fast growth of the Web makes it increasingly hard to find the right information based on measures using keywords, vector models, site popularity, etc. Therefore new techniques are needed, to access content based on its meaning. There are several initiatives that aim at semantic access of Web content.

Nicolas et al. [10] describe an Information Retrieval system in which both text documents and queries are translated to a CG representation, Zhong et al. [19] use CG's to retrieve online descriptions of garments, Ounis and Pasca [11], Myaeng [9] and Montes-y-Gomez [18] also use CG's for information retrieval purposes.

A lot of work on this subject is being done in the Semantic Web community. The development of annotation formalisms and reasoning in Web environments (e.g. Motta et al., [7, 6]), language technology for the Semantic web (most notably the OntoWeb SIG-5<sup>4</sup>), the development of tools to support manual or

---

<sup>4</sup> For more information on the OntoWeb cf <http://ontoweb.aifb.uni-karlsruhe.de/>

semiautomatic annotation of content (centered around the Semantic Web Annotation group [1]) or Information Retrieval for the semantic web (for example Van Zwol [15], working on manual annotation techniques for web content) are areas attracting many publications.

The Web already contains a massive amount of information that will not be rewritten to fit a knowledge encoding formalism. Furthermore the majority of people writing new texts are probably either not willing, or possibly not able, to enrich these with formal annotations. For this reason several of the projects mentioned above concern techniques for *automatic* semantic annotation of natural language material.

This paper presents the MUMIS (Multi-Media Indexing and Searching) project<sup>5</sup>, which addresses the problems of semantic Information Retrieval in the multimedia domain. It addresses techniques such as Information Extraction, automatic speech recognition and keyframe extraction from video content, to facilitate multilingual access to multimedia archives. In addition the MUMIS project contains a module that combines annotations extracted from separate sources into one integrated, more complete, formal description of their content. This so-called cross-document merging of annotations is one of the main issues in this paper.

The rest of the paper is organized as follows: Section 2 gives an overview of the MUMIS project; Section 3 discusses the ontology in the project; Section 4 presents the Information Extraction; Section 5 presents the cross document merging algorithm; Section 6 discusses the retrieval aspect and the paper ends with the conclusions in Section 7.

## 2 Overview of the MUMIS Project

In the MUMIS project, ontology based Information Extraction is used to improve the results of Information Retrieval in multimedia archives. The content used as test case is a collection of video recordings of soccer matches in three different languages (Dutch, English and German).

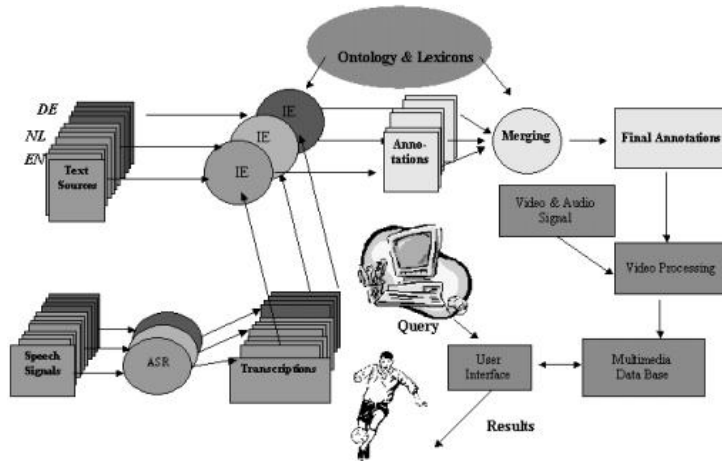
Though progress has been achieved in the content detection of salient objects and events in a sequence of images, such as goals in a soccer game (cf the IST project ASSAVID, [8], [3] and [2]), automatic indexing and retrieval of image and video fragments solely on the basis of analysis of their visual features is still not really feasible. Many research projects therefore have explored the use of *parallel textual descriptions* of the multimedia information for automatic tasks such as indexing, classifying, or understanding.

Within the MUMIS project these textual descriptions are obtained from newspapers, so-called “tickers” and transcriptions of the speech in video record-

---

<sup>5</sup> MUMIS is an EU-funded project within the Information Society Program (IST) of the European Union, section Human Language Technology (HLT). Project participants are: University of Twente/CTIT, University of Sheffield, University of Nijmegen, Deutsches Forschungszentrum für Künstliche Intelligenz, Max-Planck-Institut für Psycholinguistik, ESTEAM AB, and VDA.

ings. These texts are annotated with semantic markup, which in turn is used to disclose the content in the multimedia archive.



**Fig. 1.** Overview of the MUMIS architecture

The global architecture of the MUMIS demonstrator system is shown in Figure 1. The flow through this architecture is described below.

1. Multilingual Information Extraction is applied to sources of different types and in different languages, using an XML-encoded ontology. Each source contains partial (incomplete) information about a soccer match. The resulting knowledge is encoded and passed to the merging module.
2. The separate annotations for one match are merged into one cross-document annotation using a merging algorithm developed in this project. This merged annotation provides a more complete view of the events that took place during the match.
3. This cross-document annotation must be matched to the video recordings of the soccer match, using the time codes and information contained in transcriptions of the speech signal. This results effectively in semantic markup of video fragments.
4. Users can query the system using an interface tailored to the kind of information that has been annotated. The fact that the system can search in the knowledge representation of a soccer match makes the relevance of the returned fragments much more reliable.
5. After determining which fragments are relevant, the corresponding video sequences can be retrieved for the user.

This paper focusses on the points two and three: creating a cross document annotation for Information Retrieval.

### 3 Ontology

In the initial phase of the project there was no full scale ontology. A *multilingual term list* was used instead, listing the relevant concepts together with separate words in all three languages expressing that concept. These lists were used in the Information Extraction modules to detect the occurrence of certain concepts in the text. During the development of the cross document merging component a real ontology was needed, for flexible mapping of events and for reasoning purposes. Therefore the original term list was restructured into an ISA hierarchy, adding extra intermediate superconcepts and identifying attribute and part-of relations.

This section discusses the development of the ontology and its relation to the natural language processing modules.

#### 3.1 Construction of the Ontology

When constructing an ontology and a knowledge base, decisions have to be made about the level of detail, selection of concepts and the number of relations that should be defined between different concepts. It is neither feasible nor useful to try to encode all possible knowledge about the information domain in every detail. An important consideration is the *application* for which the ontology is needed. In MUMIS this application is retrieval of video fragments from recordings of soccer matches.

The basic unit of information in the MUMIS project is an *event*, such as KICK-OFF, GOAL, FOUL or SUBSTITUTION, since it is supposed that the user will want to query for such events. Elements that are directly related to those events, such as players, teams and time stamps, result in a second class of concepts. Other concepts are added when it appears that they are needed for reasoning purposes or when they can be part of a user query.

The ontology is encoded in a simple custom XML format. It can be completely expressed in RDF or OWL, though the mapping has not been finished yet. Since almost any other formalism has a mapping defined from RDF it will not be hard to switch to any other formalism of knowledge encoding that is more commonly used.

#### 3.2 Relation to Language

The link between the ontology and the three languages consists of a flexible XML format which maps concepts to lexical expressions. In this format, every concept can have several children of the class `<term-lang>`. Such a term entry describes a possible lexical realization of a concept. These terms allow several constructions.

- A CAT attribute indicates the part-of-speech class of a lexicon entry. This allows for phrases as well as words to be included as expressing a concept and makes it possible to describe words that belong to distinct categories as

synonyms for one concept. In this sense the lexicon reflects the EuroWordNet strategy to allow for cross-category listing of synonyms [16].

- Regular expression-like syntax can be used to combine entries with the same structure that differ, for example, only in the choice of a word (such as alternative prepositions expressing the same meaning in a certain context).
- The possibility to refer to another *concept* in relation with a lexical entry, indicating that any lexical instantiation of that concept is allowed at that point in the pattern. E.g. the German verb stem “geh” (goes) with some instantiation of the concepts GOAL and BALL together represent a goal-event.

Such a pragmatic lexicon structure makes it possible to design new mappings between ontology and language independent of specific implementations of natural language processing modules.

As an example, consider the following XML-fragment:

```
Example 1. <lex-element
  id="2.4.1.17"
  concept="Out-of-field">
  <term-lang lang="NL" type="synonym" cat="EXCL">
    uit!
  </term-lang>
  <term-lang lang="NL" type="main" cat="PP">
    {uit|buiten}[field-of-play:2.4.1.7]
  </term-lang>
</lex-element>
```

This lexical element describes two possible ways of expressing an OUT-OF-FIELD event in Dutch. The first is a simple exclamation “uit!” (out); the second is a prepositional phrase with either “uit” (out) or ”buiten” (outside) as preposition and any lexical realization of the concept FIELD-OF-PLAY as complement.

## 4 Information Extraction

The first step in the process of disclosing the multimedia archives is single-document information extraction on the different textual sources. Each text is processed separately: any reasoning that is done does not cross document boundaries. This section discusses this ontology based Information Extraction process.

### 4.1 The Natural Language Sources

In Section 2 it was already mentioned that there are several sources of textual input. It is important to note that for each textual source it is known *a priori* which soccer match is being described. Formal texts on the soccer matches are supposed to provide accurate information on properties such as which players were in the teams, who scored at what time and to whom red or yellow cards

were given. Newspaper reports contain little temporal information at all and comments combine information from the actual match with references to related matches (e.g. how a particular player performed in the previous match). Automatic speech transcriptions from the video recordings contain more errors than the other sources and relatively few events. Instead they contain a larger number of player names that are mentioned without an indication of the event they took place in. Furthermore the time codes in transcriptions are very exact but have a different base (the beginning of the recording instead of the beginning of the match). But since they are the only link with the video recordings they are needed to map the annotations to the video recordings.

The tickers are the most detailed and informative source of information about what happened during the match. They are written during the matches as a minute-by-minute account of most of the important events that take place. Every fragment in a ticker is marked with a time stamp, though this temporal information is not very exact (variations of minutes have been found). The next subsection introduces *scenes* as an effective way of grouping events.

## 4.2 Scenes

When examining the ticker texts and annotations of them, some observations can be made. Ticker texts consist of small fragments of text separated by time markers indicating the minute in which the described events took place. Ticker writers might be considered as a kind of semantic filter, writing short fragments describing interesting scenes on the field. There seems to be a limited number of different scenes that are most often used and it seems that different ticker writers group events in the same way. That does not mean that they all consider the same situations to be important, but if they consider a situation important, they mention more or less the same aspects.

An important observation is the fact that one ticker fragment often contains only one scene of interdependent events. This leads to the notion of *scenarios*, much like the scripts introduced by Schank and Abelson [14] (See Section 5.4).

With respect to the order in which scenes and events are mentioned in the texts the following can be said: most *scenes* (fragments) took place in the order in which they are described in the text, but the events *within a scene* are not necessarily mentioned in the order in which they happened on the field.

As a result of these observations, the algorithms of the Information Extraction and merging modules are based on scenes rather than separate events.

## 4.3 The Information Extraction Modules

The Information Extraction modules use natural language processing techniques to annotate the texts. Several other publications deal with the details of these modules [5, 4, 13]. Here it will suffice to say that they analyse the language on morphological and syntactical properties and detect patterns that indicate certain events. Two of the modules also perform coreference and anaphore analysis

to determine the proper subjects of events when those are mentioned in different parts of text.

## 5 The Merging Module

The main topic of this paper is the cross document merging module of the MUMIS project. This module improves the annotations for single documents by using information available in other documents. This results in annotations for the events in a single match that are more complete and more correct, which, in turn, leads to better retrieval results on the multimedia content.

The techniques developed for this merging module are not necessarily dependent on the exact knowledge encoding format used, as long as it supports first order predicate logic-style reasoning. The algorithms have been implemented in a lambda-calculus based environment. The rules have been defined in first order predicate logic style.

As a simple example of the effect of the merging algorithm, consider the following situation, taken from the results on the Euro 2000 match Netherlands vs. Yugoslavia:

*Example 2.* One of the Information Extraction components extracted from document A that in the 30th minute of the match a FREE-KICK was taken, but did not discover who took it. It did find the names of two players though: Mihajlovic (a Yugoslavian player) and Van der Sar (the Dutch keeper). From document B a SAVE in the 31st minute was extracted by the Information Extraction component, and the names of the same two players were recognized. From these two results it now can be concluded that it was Mihajlovic who took the freekick, and that Van der Sar made the save, thus giving a more complete picture of what happened in the 30-31st minute of the match.

The fact that all sources in the project have an (often explicit) timeline related to a known soccer match gives a starting point for creating this merged representation. The rest of this section discusses how this merged representation is obtained.

### 5.1 Overview of the Merging Process

Information extraction and merging from multiple sources has been tried in the past [12], but only for single events. The approach used in MUMIS consists of applying merging to multiple events extracted from multiple sources.

The MUMIS merging process can be separated into three subproblems, which will be explained in greater detail in the following subsections.

*Alignment:* Annotation fragments from different sources describing the same events should be aligned.

*Unification:* Fragments of annotations selected as describing the same events should be unified into one integrated annotation containing all information from all sources about these events. It is possible that ambiguities or conflicts need to be resolved.

*Reordering:* Many events in the ticker source text fragments are mentioned in the wrong order. The merging of unrelated scenes from different sources also introduces some ambiguity with respect to the order of events. In the final annotation all events should be present in the order in which they took place.

## 5.2 Alignment

Bi-document alignment: given two source documents A and B, every scene from A is checked for compatibility with every scene from B. In determining the strength of a possible connection between two scenes, various aspects play a role: number of common players' names, separation in time, etc. From this set of bindings a consistent set of alignments is created through a rule based algorithm. The program calculates the strength of all bindings between all pairs of scenes from documents A and B respectively. The strongest binding is selected and made permanent. Choosing that combination rules out certain other combinations from the two documents A and B, e.g. combinations between scenes from document A which are before scene SA and scenes from document B which are after scene SB are eliminated (see the observation on ordering of scenes in source texts in section 4.2). This process is repeated recursively until only selected bindings between scenes from documents A and B are left.

Multi-document alignment: the above process is performed on every pair of documents. The next step is to build sets of all scenes over all documents, where each of those sets is one of the connected subgraphs of aligned scenes. Notice that the graph is not necessarily complete, i.e. not every pair of scenes in the subgraph needs to be connected. In fact, examples have been found where scenes are incompatible and nevertheless occur in the same subgraph through a sequence of intermediate scenes. However, the aim of this process is to unify the aligned scenes, which cannot be done if incompatible scenes stay connected. In order to exclude such incompatibilities another rule based algorithm splits a graph into complete subgraphs. This splitting-up is also based on the binding strength in a given graph.

## 5.3 Unification

The previously described process results in a collection of fully connected graphs, each containing scenes describing the same set events on the field as derived from different sources. The events in these scenes may still be *partial*: events may not be parametrised with the corresponding players' names, or players' names may still be mentioned without the event in which the player took place. These partial

events from the various scenes in a given graph are combined and empty slots are filled in. At this point several (semantical) rules expressing domain knowledge are used. As a result more completely filled in events are produced. The following list contains some examples of rules that have been defined for this stage.

- The ontology allows for matching two events which are not the same, if one of them is ancestor to the other in the ontology.
- The names of the players from the two teams that are involved in a certain scene give a good indication of which team is attacking or defending at the time. This information can be used to unify events and names that have not yet been combined.
- Certain combinations of events rule out certain unifications of player names with these events. For example, if a scene contains one SHOT-ON-GOAL and one GOAL, they probably involve the same player.
- Other kinds of rules state aspects such as the types of events certain players may be involved in. For example, if the scene contains as unattached players and events a keeper, an attacker, a SAVE and a SHOT-ON-GOAL the keeper is preferably connected to the SAVE and the attacker to the SHOT-ON-GOAL.

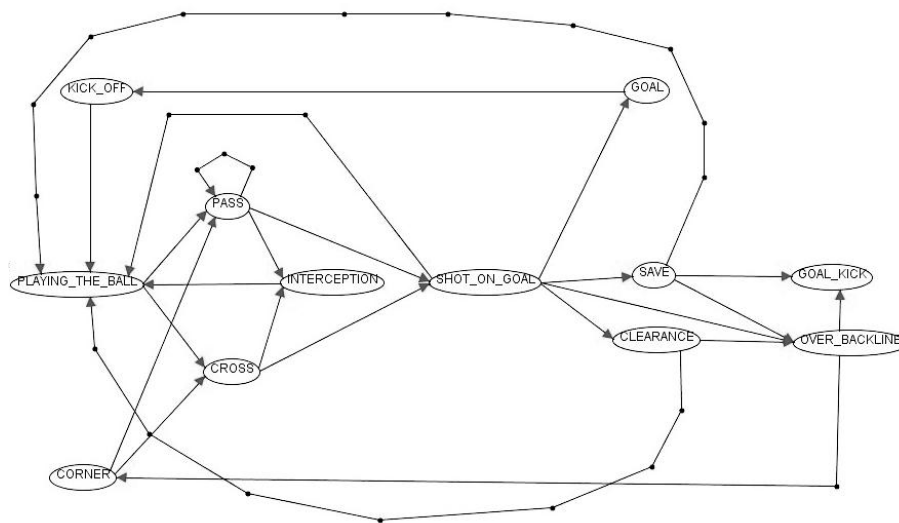
#### 5.4 Reordering

Finally, the events inside such a scene have to be put into the correct order. For example, a shot on goal in the same scene as a goal will typically take place before that goal and not after. But most tickers first mention the goal and then the shot on goal, so their order in the merged annotation is still wrong after the unification phase. Within one scene the time codes are not reliable for ordering, since different ticker texts sometimes describe the same scenes as occurring at times which are minutes apart. Besides that, all events in a scene from one source document have the same timestamp. To solve this ordering problem, the merging module uses scenarios. These scenarios describe “usual sequences of events” in soccer matches, more or less like the scripts of Schank [14].

In the MUMIS project these scenarios are defined as a graph with event types as nodes and directed edges (see Figure 2). Any path through the graph can serve as a scenario. Given a set of events in a scene, their order will be determined by the shortest directed path through the scenario graph covering at least all event types in the scene. Example 3 shows how this works in a simple situation. The edges in the scenario graph could even be conditional, meaning that a certain path is only possible if all conditions are met when the given events are assigned to nodes on that path.

*Example 3.* The unification procedure might result in a scene containing a goal by Kluivert, a pass from Bergkamp and a shot-on-goal by Kluivert. The shortest path through these events would be starting from “pass” through “shot-on-goal” to “goal”. Therefore the ordering of these events in the final merged representation would be (pass, shot-on-goal,goal).

Figure 2 shows a fragment of the scenario graph that was constructed from analysis of the corpus of ticker texts. For each scene in a set of ticker texts, a list of all events described in that scene was constructed manually, putting the events in the right order. Using these manual annotations as scenarios the full scenario graph was constructed. Doing this for four match reports resulted in a graph containing 25 nodes (event types) and 40 edges, which covers all scenes in the reports. The relatively small number of edges that was produced suggests that this is a viable approach, since this confirms the expectation that there are only few “scenarios” used by the ticker writers when they interpret events on the field. Unfortunately there was no time to construct the scenario graph for a really large portion of the corpus, but we expect that that graph would not be much larger.



**Fig. 2.** A fragment of the scenario graph.

## 6 Retrieval of Multimedia Content

The previous sections described the process through which the textual content of the corpus is annotated with semantic markup. Since the aim of the project is to disclose a multimedia archive of soccer matches, these annotations should still be matched to fragments of the video recordings of the soccer matches. Furthermore the video fragments should be made available to the user, with the help of the semantic annotations. These two aspects are described further in this section.

## 6.1 Annotation of Video Recordings

For every match, the merging process results in an extensive annotation of everything that happened. The scenes from this annotation should be matched to fragments of the video recordings of that match. A user query can then result in retrieval of video fragments through searching in the annotations.

The only textual sources that have a time line which can be directly matched to that of the video recordings are the transcripts of the speech in those recordings [17]. Information Extraction for these transcripts is possible, but for several reasons the resulting annotation is not easily merged with the rest of merged annotation:

- The transcripts themselves contain more errors than the other sources.
- The transcripts have a different type of timeline. Instead of annotating the kick-off as minute 0, the transcripts annotate minute 0 as the beginning of the video sequence which is usually some time before the kick-off. Since the kick-off is usually not even mentioned in the transcripts, it is hard to compensate for this difference in time lines.
- There are relatively few events mentioned in the speech accompanying a video recording of a match, since the actual events are supposed to be seen on the video.
- The events that can be extracted are not ordered in scenes as it is with the tickers. Aligning the single events to the scenes from the tickers is a very hard problem.

The challenge is to find a mapping between the time codes in the merged annotation and the time codes in the transcripts. This problem has not yet been solved. A possible way to do this might be to use the occurrence frequency of names of players that were substituted during the game. The name of that player probably has a higher frequency of occurrence in the time span in which he actually played. Using the “jump” in frequency around the time of a substitution on the field a prediction algorithm might be trained to calculate the most probable time shift. Some other events, such as goals and yellow and red cards might also be used as indication of the timeshift, since they might be mentioned more often in the transcripts than other events.

## 6.2 Retrieval of Fragments

To make the annotated content accessible to the user, queries should be formulated in the annotation formalism to match them to the annotated content. In the MUMIS project the focus is on information extraction rather than on retrieval. Therefore the retrieval module does not process natural language queries to obtain the formalized query. An interface has been developed instead that allows the user to enter queries directly in the event-format. The interface makes use of the lexica in the three target languages and the domain ontology to assist the user while entering his or her query. The mapping of annotation to video recordings described in the previous section makes it possible to search in the

annotations but return video fragments corresponding to the annotation fragments found as result for the query. The hits of the query are indicated to the user as thumbnails in the storyboard together with extra information about each of the retrieved events. The user can select a particular fragment and play it (see Figure 3).



**Fig. 3.** Screenshot of the User Interface, results of the query for all fouls in a certain match.

## 7 Conclusions

MUMIS is one of the first multimedia indexing projects which carries out indexing by applying information extraction to multimedia and multilingual information sources. It is an example of a practical approach to semantic access of multimedia web content which yields good results in the first test cases. Especially the focus on the techniques for *cross document* information extraction, using evidence from different sources to improve the quality of the information extraction, seems to be a promising approach for semantic content access. Future work will include extending this approach to a larger and more complicated domain.

## References

- [1] Semantic web page on annotation and authoring.  
<http://annotation.semanticweb.org/>.

- [2] J. Assfalg, M. Bertini, A. Del Bimbo, W. Nunziati, and P. Pala. Soccer highlights detection and recognition using HMMs. *Proceedings of the International Conference on Multimedia and Expo (ICME2002)*, to appear.
- [3] J. Assfalg, M. Bertini, C. Colombo, and A. Del Bimbo. Semantic annotation of sports videos. *IEEE Multimedia*, 9(2):52–60, 2002.
- [4] H. Cunningham. GATE, a general architecture for text engineering. *Computers and the Humanities*, 36:223–254, 2002.
- [5] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics.*, 2002.
- [6] J.B. Domingue and E. Motta. Planet-onto: From news publishing to integrated knowledge management support. *IEEE Intelligent Systems*, pages 26–32, 2000.
- [7] E. Motta, S. Buckingham-Shum, and J. Domingue. Ontology-driven document enrichment: Principles, tools and applications. *International Journal of Human-Computer Studies*, 52:1071–1109, 2000.
- [8] M. Mukunoki, M. Bertini, J. Assfalg, and A. Del Bimbo. Classification of raw material sports videos for broadcasting using color and edge features. *Proc. of Int'l Conf. on Multimedia and Expo (ICME2001)*, 2001.
- [9] S.H. Myaeng. Using conceptual graphs for information retrieval: A framework for adequate representation and flexible inferencing. In *Proc. of Symposium on Document Analysis and Information Retrieval*, 1992.
- [10] S. Nicolas, G.W. Mineau, and B. Moulin. Extracting conceptual structures from English texts using a lexical ontology and a grammatical parser. *Foundations and Applications of Conceptual Structures, supplementary proceedings of the 10th International Conference on Conceptual Structures*, 2002.
- [11] I. Ounis and M. Pasca. A promising retrieval algorithm for systems based on the conceptual graphs formalism. In B. Eaglestone, B.C. Desai, and Jianhua Shao, editors, *Proceedings of the International Database Engineering & Application Symposium (IDEAS'98)*, pages 121–130, 1998.
- [12] R. Radev and K.R. McKeown. Generating natural language summaries from multiple on-line sources. *Computational Linguistics*, 24(3):469–500, 1998.
- [13] H. Saggion, H. Cunningham, K. Bontcheva, D. Maynard, O. Hamza, and Y. Wilks. Multimedia indexing through multi-source and multi-language information extraction: The MUMIS project. *Data & Knowledge Engineering Journal*, 2002.
- [14] R.C. Schank and R.P. Abelson. *Scripts, Plans, Goals and Understanding: an Inquiry into Human Knowledge Structures*. L. Erlbaum, 1977.
- [15] R. van Zwol. *Modelling and searching web-based document collections*. PhD thesis, Centre for Telematics and Information Technology (CTIT), Enschede, the Netherlands., 2002.
- [16] P. Vossen. Eurowordnet: a multilingual database for information retrieval. In *Proceedings of the DELOS workshop on Cross-language Information Retrieval*, 1997.
- [17] M. Wester, J.M. Kessens, and H. Strik. Goal-directed ASR in a multimedia indexing and searching environment (MUMIS). In *7th International Conference on Spoken Language Processing, ICSLP 2002, INTERSPEECH 2002*, 2002.
- [18] Manuel Montes y Gomez, Aurelio Lpez, and Alexander F. Gelbukh. Information retrieval with conceptual graph matching. In *Database and Expert Systems Applications*, pages 312–321, 2000.

- [19] Jiwei Zhong, Haiping Zhu, Jianming Li, and Yong Yu. Conceptual graph matching for semantic search. In U. Priss, D. Corbett, and G. Angelova, editors, *Conceptual Structures: Integration and Interfaces. Proceedings of the 10th International Conference on Conceptual Structures*, 2002. LNAI 2393.