CASAM: Collaborative Human-Machine Annotation of Multimedia

Robert J. Hendley*, Russell Beale, Chris P. Bowers
School of Computer Science, University of Birmingham, UK.

Christos Georgousopoulos, Charalampos Vassiliou
INTRASOFT International S.A.

Eric Karstens
European Journalism Centre, The Netherlands

Ralf Moeller
Software Technology and Systems Institute, TUHH, Germany

Petridis Sergios
Institute of Informatics and Telecommunications, NCSR, Greece.

Dimitris Spiliotopoulos
Athens Technology Center S.A, Greece.

*R.J.Hendley@cs.bham.ac.uk

Abstract The CASAM multimedia annotation system implements a model of cooperative annotation between a human annotator and automated components. The aim is that they work asynchronously but together. The system focuses upon the areas where automated recognition and reasoning are most effective and the user is able to work in the areas where their unique skills are required. The system's reasoning is influenced by the annotations provided by the user and, similarly, the user can see the system's work and modify and, implicitly, direct it.

The CASAM system interacts with the user by providing a window onto the current state of its annotation, and by generating requests for information which are important for the final annotation or to constrain its reasoning. The user can modify the annotation, respond to requests and also add their own annotations. The objective is that the human annotator's time is used more effectively and that the result is an annotation that is both of higher quality and produced more quickly. This can be especially important in circumstances where the annotator has a very restricted amount of time in which to annotate the document.

In this paper we describe our prototype system. We expand upon the techniques used for automatically analysing the multimedia document, for reasoning over the annotations generated and for the generation of an effective interaction with the end-user. We also present the results of evaluations undertaken with media professionals in order to validate the approach and gain feedback to drive further research.

Keywords *video, multimedia, annotation, mixed-initiative*

Introduction

The annotation of multimedia documents is a very important task, common to a wide range of application areas. It is important functionally, but also economically, since the annotation process is critical to the effective retrieval, and hence use and re-use, of these multimedia assets. Within the CASAM project (Computer-Aided Semantic Annotation of Multimedia, an EU-funded initiative), we have focused upon the annotation and retrieval of video news reports for news agencies and it is clear that, in this domain, the potential for financial benefit of a rich annotation is large. However, the techniques developed here have a much wider potential application, not only to other video based repositories, but also to other non-text based domains.

With text-based documents, there are well established and very effective ways to analyse the document's content and extract sufficient knowledge to support high quality retrieval. With documents that are video or image based, this has proved to be extremely difficult and it is still the case that effective annotation of multimedia documents relies upon the skill and expertise of human annotators. This is an expensive and scarce resource, and it takes significant time and resources to produce high quality annotations. Within the typical context of tight deadlines and budget constraints the potential depth and quality of annotations is often limited. This in turn limits the opportunities for retrieval of the material.

Automated analysis of multimedia is making significant progress. Similarly, automated reasoning over these results allows higher-level annotations to be produced and ambiguities to be resolved. It is still the case, however, that the results produced are insufficient to allow them to be used: they are unreliable and also they are incapable of recognising many of the most significant, or more subjective features, that are crucial to providing a rich annotation.

The paper is structured as follows. Firstly we expand further upon the motivation for the approach used within the CASAM system in the context of previous and related work. We then present the overall architecture. The techniques and achievements of each of the major components are then described. First the multimedia analysis (KDMA) component, which identifies low level concepts from the multimedia. This is followed by the reasoning (RMI) component which attempts to form higher level interpretations of the multimedia content based upon input from both the KDMA component and the user. Finally we describe the interface presented to the user (HCI) and the numerous challenges of managing dialogue between the system and the user. We then describe and present the results of evaluation and user studies performed with the system. Finally we discuss the outcomes and resulting conclusions as well as identifying areas that warrant further investigation.

Related Work

Researchers have examined a range of different approaches to enhance workflow and effectiveness of multimedia annotation. In user driven annotation the system only supports the annotation process by providing an appropriate set of tools to support the user. Common issues identified within these systems include managing the various perspectives of annotators [11], relating textual description temporally to the video content [1, 22, 24, 27] and navigating the multimedia content in relation to the annotation through some form of timeline [6, 10]. Clearly a user driven approach to annotation puts the user firmly in control of the process. However, annotation is a time consuming and laborious task. Annotation systems often ignore the typical workflows employed by professional annotators and annotation often needs to be verified by another annotator [24]. There is also the danger that much time could be wasted through repeated refinement and attempts to improve the annotation without adding to the overall quality.

Semi-autonomous annotation systems typically aim to support the system and user working together to annotate some multimedia content. In most cases, this normally involves the system autonomously annotating some aspects of the video, whilst the user annotates everything else they think is appropriate. Automatic annotation systems have attempted to identify emotional context [8], recognise repeated occurrence of identified objects [31] through use of additional sensory information (time, location, camera state) [30, 32]. Semi-autonomous annotation systems work most effectively when the systems and the user play to their strengths and the dialogue between the two is optimally supported. Supporting this optimal dialogue is a challenge. There are typically thousands of annotations automatically generated making it very difficult for the user to check each of these. In addition, machine learning algorithms tend to perform better when identifying low level and tangible content, such as geometry (angles, distances etc.), objects and people, and struggle to identify more abstract or high-level concepts such as emotion and mood.

Collaborative annotation systems enable multiple users to annotate videos either synchronously or asynchronously. Generally, an asynchronous method of collaborative annotation is preferred; there is little desire amongst users to annotate synchronously [25]. However, there are exceptions where the

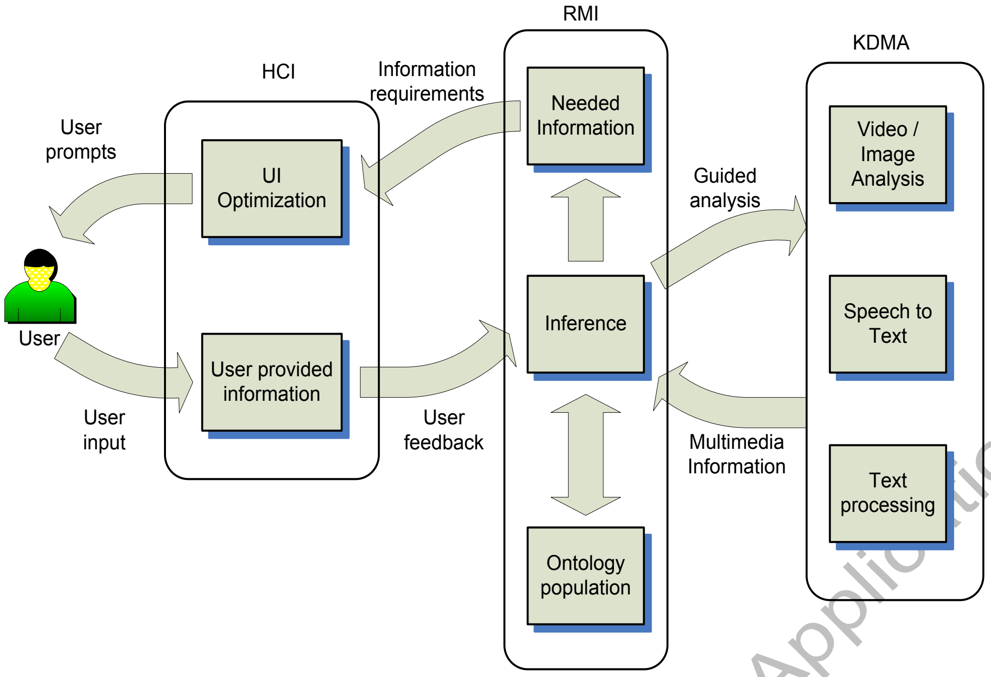


Figure 1: Conceptual architecture of CASAM system

process of annotating synchronously as a group provides opportunity for discussion and critique [9]. Collaborative annotation introduces a range of interaction issues. Real time collaborative annotation requires managed communication between annotators. Solutions have included the use of instant messaging [34, 40] and the dynamic update of visualisation of a shared annotation state [4, 16]. However, little work considers the case of a human and machine are collaboratively annotating.

Overall CASAM Methodology & Architecture

The CASAM system is based on the premise that an optimised dialogue between human annotator and automated analysis and reasoning will results in a system which, when compare with either a human annotator or an automated annotation system acting alone, is able to:

- Reduce the time taken to produce an annotation.
- Improve the quality of the annotation produced.
- Increase the quantity of annotations produced.

CASAM implements a model of cooperative annotation between a human annotator and automated components. The aim is that they work asynchronously, independently, but at the same time. The system focuses upon the areas where automated recognition and reasoning are most effective and the user is freed to work in the areas where their unique skills are required. The system's reasoning is influenced by the annotations provided by the user and, similarly, the user can see the system's work and modify and, implicitly, direct it.

Figure 1 shows a conceptual view of the CASAM system. The three components work asynchronously sharing information as it becomes available. All of the components build upon this information, as well as modifying and deleting it as necessary. This changing information, in turn, can direct and focus the work of each component (both implicitly and explicitly). Users are an integral part of this process and, in particular:

- They are provided with a window onto the current state of the annotation through which they can observe, modify or delete the system's annotations.
- They can add their own annotations which are then analysed by the system, incorporated into the current annotation state and built upon by the system.

- The system can identify important information for the annotation or annotation process and generate explicit requests for information from users in the form of queries.

Internally, CASAM is an ontology based system. It uses a restricted description logic for its internal representations, to communicate between components and to represent the final annotation result. This is, however, transparent to the user.

In order for the CASAM system to successfully support the notion of computer-aided, semantic annotation of multimedia content, aiming at maximizing performance and benefits in a semi-manual annotation scheme, it employs a variety of techniques from the fields of human computer interaction, machine reasoning and multimedia analysis.

The primary objective of the Human Computer Interaction (HCI) component, and more specifically the user interface that this component provides, is to act as an entry point for the human operator of the system. Through this interface a user can feed the system with the multimedia content to be annotated and provide some additional metadata. Those are then passed to the RMI and KDMA components and the workflow of the processing is initiated.

The Knowledge Driven Multimedia Analysis (KDMA) component analyses multimedia content and identifies objects, producing low-level information. This effort is periodically assisted by input provided either from HCI (in the form of structured or unstructured auxiliary information on the document under annotation) or from RMI in the form of reasoned interpretations. The more information is provided to KDMA, the more accurate its results become. When KDMA produces sufficient information, it communicates this information to the RMI component.

The information that is generated by HCI and KDMA is utilised by the Reasoning for Multimedia Interpretation (RMI) component to infer higher-level interpretations of the multimedia content. In the event where there is an ambiguity between the produced interpretations, appropriate queries are generated and forwarded to HCI to enable the user to disambiguate between possible interpretations.

The communication and orchestration of these components is managed by an Integration Platform (IP) module that acts as a central point of reference for the system, capable of coordinating the interactions and flow of information in a seamless manner. Specifically, the IP provides:

- An Integration Wrapper that operates as a central repository for the system, managing and storing the multimedia documents and resulting annotation.
- A Business Process Execution Language (BPEL) Orchestrator, which handles message dispatching among components.
- An Orchestrator Logger, which provides monitoring facilities of message exchanges among CASAM's components.
- A Semantic Search Engine, which performs searches on the repository created by CASAM's annotation session results.
- An authentication and authorization mechanisms for supporting user login and system roles.
- Content management console that supports create, read, update and delete (CRUD) operations within the user and multimedia objects of the platform.
- A Streaming Media Server to provide flexible and responsive media streaming.

A typical workflow that portrays how an annotation process is carried out via the CASAM system is as follows:

1. Initially, HCI authenticates the user. After a successful authentication, a unique session identifier is produced, which will accompany all messages exchanged for the entire session.
2. HCI retrieves a list of multimedia documents available from the IP repository.
3. A multimedia document is selected by the user and submitted for processing.

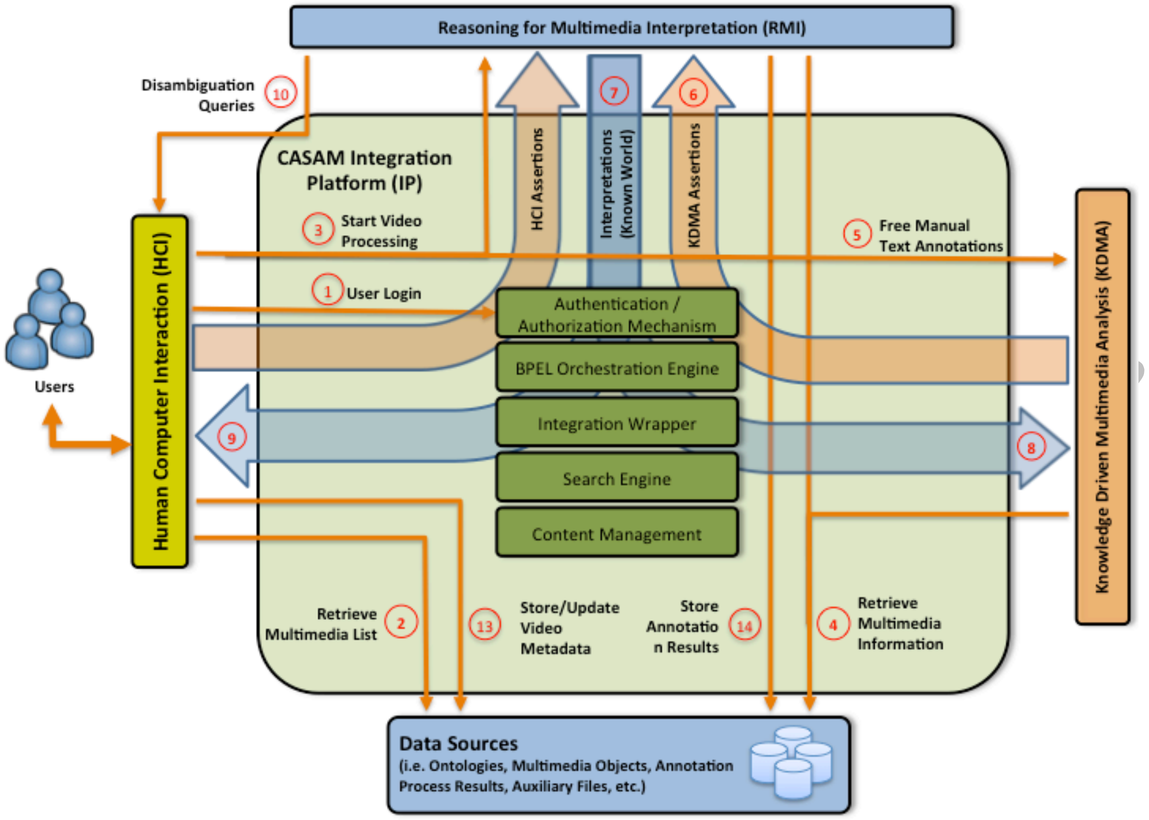

Reasoning for Multimedia Interpretation (RMI)
Disambiguation Queries
10
7
6
CASAM Integration Platform (IP)
HCI Assertions
Interpretations (Known World)
KDMA Assertions
3
Start Video Processing
5
Free Manual Text Annotations
1
User Login
Users
Human Computer Interaction (HCI)
Knowledge Driven Multimedia Analysis (KDMA)
Authentication / Authorization Mechanism
BPEL Orchestration Engine
9
Integration Wrapper
8
Search Engine
Content Management
Retrieve Multimedia List
2
13
Store/Update Video Metadata
Store Annotatio n Results
14
4
Retrieve Multimedia Information
Data Sources
(i.e. Ontologies, Multimedia Objects, Annotation Process Results, Auxiliary Files, etc.)

Figure 2: Interaction between the CASAM components

4. The user's selection triggers HCI, RMI and KDMA to request a *DocumentObject*, which contains access information for the multimedia document and all its related information, from the IP and to begin processing. HCI also retrieves general information about the document in the form of International Press Telecommunications Council (IPTC) metadata from the IP and displays them to the user.

5. HCI displays the multimedia document to the user and the user may begin to annotate. This information enables HCI to produce assertions that are then sent to RMI.

6. KDMA uses the multimedia ontology to process the multimedia content and produces low-level information (assertions). When results are produced, they are sent to RMI and HCI.

7. RMI receives assertions (produced by KDMA and/or HCI) and resolves any possible conflicts; while the "Known World" definition (a logical construct defining what is known about the video) is constantly updated based on the evolving information. Based on that, it performs reasoning to produce new interpretations, which are sent to KDMA and HCI.

8. KDMA uses the interpretations that have been produced by RMI to improve its results.

9. HCI displays information produced by RMI and the user can disambiguate or compensate for wrongly interpreted information.

10. RMI may create queries that are directed to both KDMA and HCI. For the ones targeting HCI, after the user has addressed them, a reply is sent back to RMI. With respect to the ones targeting KDMA, after an analysis is performed, a reply specific to the query is returned.

11. HCI directs KDMA to focus its processing to a special section of the video.

12. At any moment the user may provide structured and/or unstructured information about the multimedia content. This information is submitted to KDMA through HCI.

13. Steps 6 to 12 are repeated until the user decides that the annotation results are satisfactory and the whole process is ended.
14. The user signals the end of the annotation session through the GUI. HCI then request that all processes stop and the results of the annotation session are stored in Web Ontology Language (OWL) format by RMI.

The interactions among the CASAM components, based on the process described above, are illustrated in *Figure 2*.

The CASAM system adheres to the Service Oriented Architecture (SOA) paradigm in order to allow the realisation of a loosely coupled architecture where all constituent components that form the integrated CASAM toolkit are developed in a platform-independent approach, unbound by any distributed limitations. The utilisation of well-accepted standards and common communication protocols (such as SOAP, XML, BPEL etc.) permits the extensibility of the system in terms of seamlessly introducing new software components that encapsulate the state of the art in research areas related with the functionalities that CASAM provides.

In terms of third-party system integration, a basic requirement for CASAM is an ability to seamlessly integrate into existing multimedia content repositories and other systems. The main integration levels that can be supported by CASAM include:

- **Integrating at Content Level:** According to CASAM's design, all information about the documents available for annotation is stored in the database of the Integration Platform. Given that a third party system will provide "live" access to its content catalogue, the Integration Platform application can use this data access (via web service or xml feed) to seamlessly plug in to the catalogue and integrate CASAM with the 3rd party content repository. Alternatively the Integration Platform can provide "write access" to a third party system, via a web service, and replicate the content catalogue in its database.
- **Integrating at Access Control Level:** Access control for the CASAM prototype is managed by the integration platform. The current access control mechanism uses the Integration Platform's database for authenticating and authorizing users. This role of the database can easily be substituted by any LDAP, Active Directory or any type of user management platform currently employed by an organization.
- **Other Integration Possibilities:** Apart from being integrated into an organization's content repository and user base, CASAM's open and pluggable SOA architecture also provides the option of seamlessly enhancing the automated annotation functionality.

Knowledge Driven Multimedia Annotation (KDMA)

KDMA is the back-end component of the CASAM annotation tool responsible for the low-level analysis of multimedia content. It integrates methods that allow extracting information from audio-visual streams and texts, which can ultimately ease the users' annotation task. It includes a large number of methods to deal with particular aspects of multimedia analysis, aiming to provide information in three directions. First, semantically analysing the content of the documents, providing information with respect to particular concepts that pertain to the question "what the video is about". Second, extracting information with respect to people appearing in the video, by means of speaker and person face clustering. Last, locally tagging the video with respect to the audio or video context.

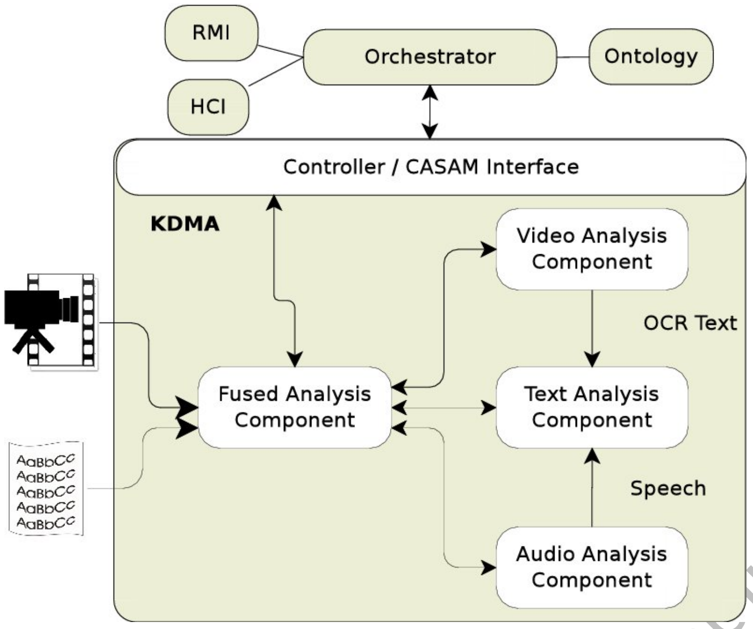


Figure 3: The architecture of the KDMA module

Design overview

Figure 3 depicts the overall architecture of the KDMA component, where separate analysis components communicating with each other through a controller are integrated. KDMA uses CASAM interface methods and objects to receive input and send results according to the CASAM ontology. The requests for analysis are converted into internal structures and dispatched to media analysis components, namely the *Video, Audio* and *Text* components. The media-specific components then produce a series of tags that represent the information detected, which are further combined through the *Fusion* component. All results are sent back to the CASAM system in the form of ontological assertions, represented as RDF-like triples. KDMA also supports the exporting of the extracted information in OWL format in which case the results are validated with a reasoner [23]. A degree of confidence in the interval [0,1] is given for each assertion.

Interactivity

In KDMA, data is processed as soon as it is available and the resulting information is constantly enriched and improved as the user adds information. KDMA employs parallelism to speed up the analysis of documents and reaches almost real time by keeping a good balance between complexity and speed of execution. In particular, KDMA supports incremental communication when receiving requests and sending results. This allows greater flexibility with respect to the order and the level of analysis and is well suited for interactivity with the user, namely:

- *Time focus*. Analysis results are sent in blocks. E.g. if a 5-minute video is given for analysis, the 1st reply of KDMA may concern the 1st minute, the second one the second minute, etc. Importantly, the user drives implicitly the time focus of the analysis, since the time point displayed at the interface gets higher priority and thus is analysed first.
- *Levels of Granularity.* For the same query, easy to extract or approximate information is sent first, and thus fast, while hard to extract/more accurate information is sent in a second step.
- *Adapting to user feedback.* When knowledge regarding the analysed document is changed, either by user-provided feedback or by higher level reasoning, KDMA re-analyses the data and provides updated analysis results.

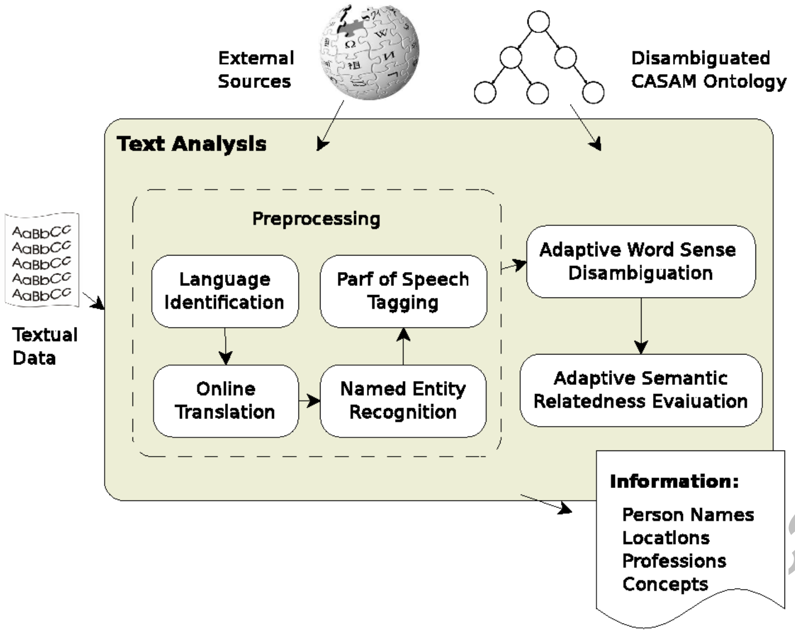


Figure 4: A schematic diagram of the text analysis process

What the video is about

Recognizing relevant topics of discussions in KDMA is primarily guided by textual data. These may stem directly from user annotations or indirectly, through, speech or text detected in video frames. In the latter case, particular methodologies for speech detection and video text detection and enhancements [2] are used to detect and extract the relevant text. In all cases, the text undergoes a semantic analysis that results in suggestions of pertinent concepts found in the working ontology.

Pre-processing Steps

The overall approach of text analysis is depicted in Figure 4. The first step of text analysis is to translate it in English, when needed, to ensure consistency with available lexical resources. Language identification is performed by means of an N-gram-based approach [7] and the translation by the Google Translate Service (http:// translate.google.com). Subsequently, named entities are identified and semantically annotated with the corresponding concepts of the ontology using the OpenCalais service (http://www.opencalais.com). A tagger [36] that assigns parts of speech tags to each word, such as verb, noun and adjective, complements the text analysis results. Finally, a state-of-the-art unsupervised method for word sense disambiguation [38] exploiting lexical resources such as WordNet [29] is applied. The calculation is performed between the specific meaning of a word and an ontology concept, providing a more accurate score.

Semantic Relatedness Calculation

Relevant topics, with respect to the textual content analysed, are suggested through a degree of semantic closeness between text keywords or key-phrases and lexicalizations of ontology concepts. This degree takes a value in the interval [0, 1] with high values indicating close semantic relation. In particular, we have used the *Omiotis* [37] measure which has the advantage of utilizing of all the provided semantic relations by WordNet, and that it is applicable to terms of any part of speech type. This measure has been shown to provide the highest correlation with human judgments among the dictionary-based measures of semantic relatedness [37]. Note that the semantic relatedness calculation of text found within the video is adapted by the text annotations directly provided by the use, though their respective ontology concepts, thus improving the overall accuracy.

People in the video

An important part of the video analysis in CASAM concerns the detection of humans in the audio-visual stream, because we are focussing on the domain of news, and most news content relates to people. We have focused on person *clustering* rather than person *identification*, to allow analysis of content where people appearing are not assumed to be know beforehand. Determining that a particular subset identified faces correspond to a particular individual person, either by their voice or their appearance, may significantly reduce the human annotator's work, by providing fast identification of all occurrences of a given person in a video.

Speaker Clustering

Speaker clustering is the process of grouping the homogeneous speech segments, according to the speaker identity. The methodology includes several steps, such as detection of speech segments, similarity evaluation and clustering based on similarities. The novelty of our approach lies on applying the K-means clustering algorithm to a suitable discriminant subspace, where the Euclidean distance reflects speaker differences. Speaker-conditional statistics are estimated using single-speaker segments statistics. This makes it possible to use Linear Discriminant Analysis to find the optimal discriminative subspace, using unlabelled data [17].

Face clustering

Similarly to speaker clustering, face clustering requires (a) finding face segments, (b) extracting similarity indexes between any two faces and (c) use these to cluster faces into distinct groups. In particular, after detecting video regions of faces using the Viola-Jones methodology [39], the SIFT algorithm is used to provide similarities. SIFT [28] is a widely used algorithm oriented towards finding homographies between image parts. The SIFT features are invariant to image scale and rotation, and have been shown to provide robust matching against distortion, change in viewpoint and change in illumination. The similarity between two faces is then obtained as the minimum number of keypoint matches between them. In the final step, the matches between any two people are assembled into a matching matrix that is considered as the adjacency edge matrix of an undirected weighted graph, where each face instance is a vertex. Thus finding the faces that belong to the same person translates into finding clusters into this graph [33]. Using the maximum-clique approach, our algorithm consistently attributes a person face to exactly one person, favouring clusters with strong interdependencies. The overall approach has been shown to provide good results in CASAM corpora.

Local Annotations

A requirement of the KDMA module is to annotate the video at temporal locations with labels that describe the content. A multi-labelling classification approach has been developed to suggest a number of environmental sounds, possibly co-occurring with speech, as well as video scene-level tags. In particular, for the audio stream, a mid term analyses that uses features such as spectral roll-off, spectral entropy and spectral centroid is conducted [18], whereas, for the video stream, colour and texture features are obtained from video shot key frames. These are then assembled in features vectors used to train and subsequently use a set of classifiers that detect occurrences of sounds, such as wind, engine, water, applause or music and scene-level qualifications, such as indoor/outdoor, urban/ vegetation, mountain, road, water. Resulting local annotations are then obtained using a winner-takes-all scheme.

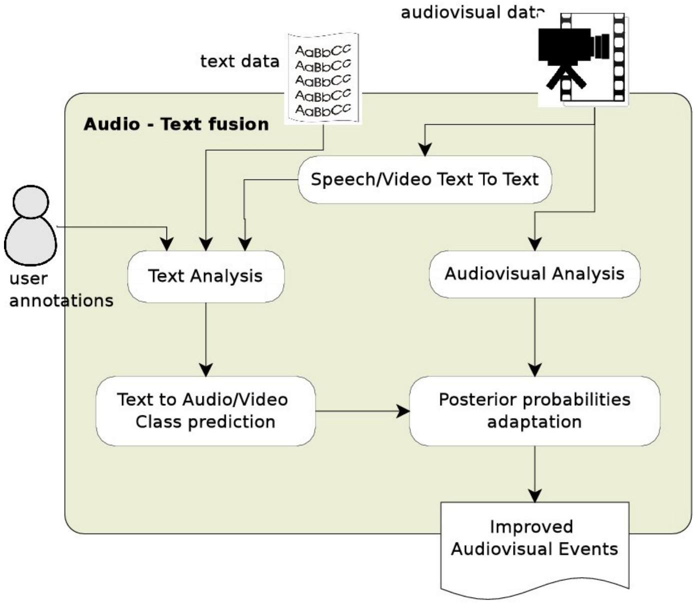


Figure 5: Fusing text with audio-visual information

Fusing audio-visual with text cues

The fusion sub-component uses the probabilities of concepts detected in text to obtain an estimation for the priors probabilities of concepts to be subsequently detected in audio and video, at the *audio-visual document level* (see Figure 5). A supervised training set containing a mapping from text-extracted to audio/video-extracted concepts for each document of a reference corpus is used to train regression models from which the probabilities of audio and video concepts are obtained. For a new document, the results of text analysis is given as an input to the regression model which outputs more accurate *prior* probabilities for audio and video classes for this document. These priors are then taken into account while calculating the corresponding *posterior* probabilities of the audio and video concepts, thus improving the accuracy of the audio-visual analysis results.

Reasoning for Media Interpretation (RMI)

RMI receives input from KDMA and HCI, as described in Figure 2. The output of KDMA provides a structural description of the document that is utilised by RMI and HCI. A multimedia document (first layer in figure 6) is a structured object consisting of (second layer) objects representing different modalities (text, audio, video etc.). For each of these modalities, over time certain phases are determined by KDMA (third layer). For each of the phase objects, possibly of different modalities, temporal or positional information is made explicit (fourth layer).

With phase objects, e.g., audio segments, video shots, or named entities in a piece of text (third layer in figure 6) there are associated domain objects. For instance, in an audio segment KDMA might have detected a person speaking (speech). The audio segment from which the information is extracted overlaps with a certain video segment (see figure 7) that the human annotator might have associated with an object that can be identified as a politician. Given the input from KDMA and HCI, the goal of RMI is to use declarative representations to derive a more abstract description of the situation, a so-called high-level interpretation. Declarative means that an interpretation is based on logical knowledge bases (an ontology and a set of rules) and logic-based decision processes are used as the basis for the generation of interpretations. In particular, RMI is designed in the tradition of abduction-based interpretation systems [15].

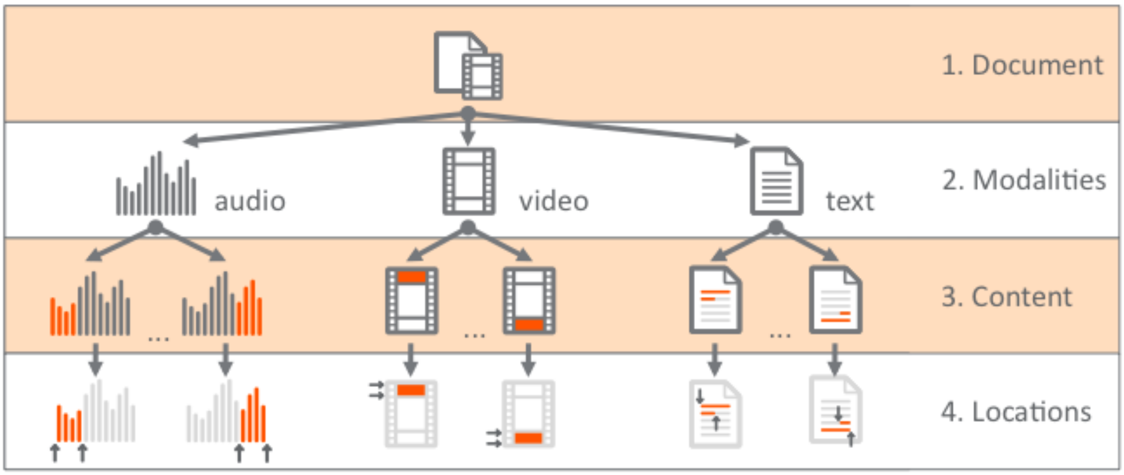


Figure 6: Structure of a multimedia document

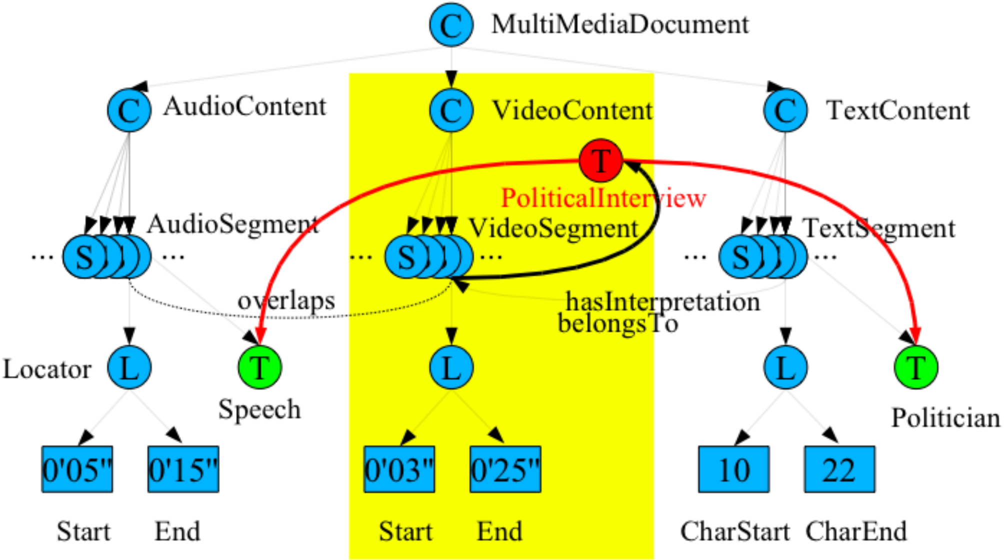


Figure 7: Tags (T) 'Speech' and 'Politician' computed by KDMA are combined to a higher-level interpretations (PoliticalInterview) which is attached to the corresponding video shot (VideoSegment) by HCI after RMI has communicated the information to other modules.

For the abductive approach, RMI uses logic programming rules as a definition for the space of possible interpretations, accompanied by the domain ontology, which, besides defining the vocabulary to be used by all modules in its signature, is used here to reduce the space of possible interpretations to meaningful ones using logical axioms (Tbox). The focus of attention can also be declaratively specified using focus of attention rules. In the example in Figure 7 the temporal constellation of speech and a politician is "aggregated" to a political interview, which is attached to the video shot overlapping the audio shot being the source of the speech. This new object can be the source of further interpretations.

In this sense the lower-level tags, derived from KDMA or provided by the user through the HCI component, plus the document structure are seen as "observations" of the interpretation agent which tries to explain what it receives by constructing a context, in this case the 'political interview' tag, in order to "explain" the observations. Explaining means that the formulas being added to the set of formulas representing the document structure entail the observation formulas (and the formulas derived by focus of attention rules). The set of formulas (assertions) for an interpretation is called an *Abox*.

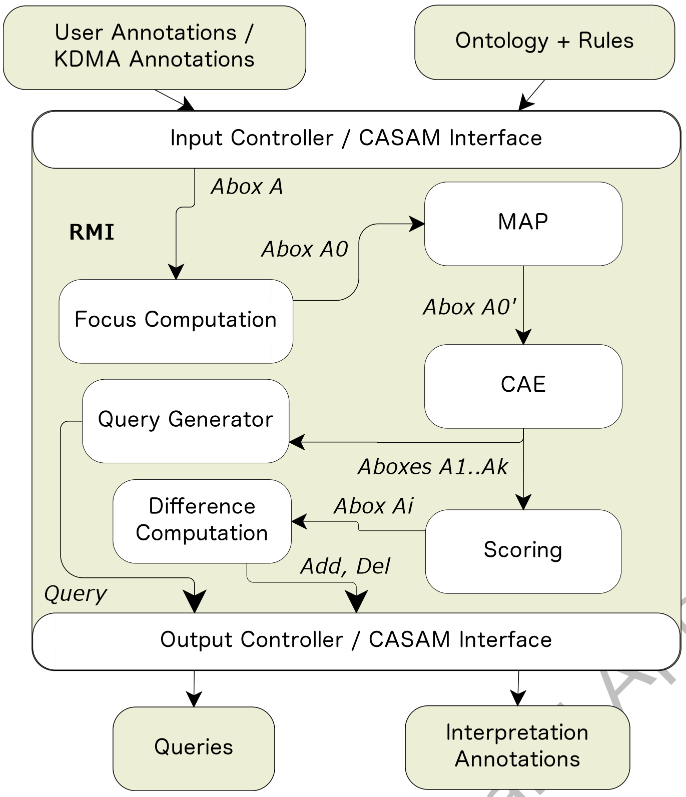


Figure 8: Architecture of the RMI module

Formulas for explaining observations are computed using abduction as an inference service (see [15, 26] for a detailed evaluation and for further applications). All observations and the corresponding explanations constitute an interpretation of the video content. The architecture of RMI is presented in Figure 8. The handling of control signals as shown in figure 2, such as Start Video Processing (label 3), Store Annotation Rules (label 14), and Retrieve Multimedia Information (label 4), is obvious and omitted from Figure 8 for the sake of clarity.

Objects at all layers might be created by KDMA and HCI on the fly. Thus, RMI employs focus rules as a dynamic control regime to cope with incrementally delivered input. Since the input grows considerably over time, RMI applies sound and complete Abox modularisation techniques [20] such that reasoning is applied to subsets of the formulas for a document only. Furthermore, focus rules determine which objects (and which temporally coincident events) are actually explained using abduction-based reasoning with respect to the terminological knowledge in the ontology and a set of interpretation rules (see Figure 8).

Given that the focus (subset of the whole document structure) is determined from the input Abox *A*, the observation to be explained are collected into an Abox (*A0*). In a first process, inconsistencies due to multiple classifications might have to be resolved. Since the observations are associated with certainty factors, which are converted into probabilities indicating that the interpretation agent considers the corresponding Abox formula as true. Using a maximum a posterior operator MAP [14], a maximal consistent subset of the input Abox can be determined (*A0'* in Figure 8).

Depending on the resources available RMI iteratively selects assertions from this Abox and tries to interpret them, i.e. explain them, using the abduction engine (called CASAM abduction engine, CAE, in Figure 8) using the interpretation rules (Rules). Depending on the situation there might be multiple interpretations possible (called *A1…Ak* in Figure 8), and RMI scores the interpretations using Markov logic [14]. The aim is to associate with each of the output Aboxes for CAE the probability that the observations are true, given the interpretations it maintains are true. For observations that, at a certain point of time, are not yet associated with an interpretation, the priors derived from the certainty factors are used [20]. The Abox with the maximum probability value is selected (see the box Scoring). In the spirit of Markov logic, the interpretation rules are associated with weights in order to specify the probability distribution that the RMI agent should assume for ranking interpretations. RMI generates formulae for the Markov logic reasoning system Alchemy, which is used for reasoning. Using sampling techniques and in particular Alchemy's MC-SAT algorithm, acceptable running times could be achieved (see also [19] for an approach using Gibbs sampling). The best interpretation is called *Ai* in Figure 8.

RMI computes the differences between the current best Abox and the previous (or an empty one in the initial case) and communicates these differences to the other modules in two sets of assertions: *Add* - things to be added w.r.t the previous interpretation; and *Del* - things to delete w.r.t. the previous interpretation. RMI stores the current interpretation to be used as the previous one in the next round. Thus, RMI repeatedly informs the other modules about the currently most-probably correct set of Interpretation Assertions ("known world"). RMI computes interpretations (label 7 in Figure 2) in an incremental and asynchronous way. In addition, it generates queries that help other modules to determine which information might be relevant for disambiguation interpretation alternative (label 10 in Figure 2).

Note that CAE might be called multiple times if time permits as there might still be assertions to explain in *A0'*. Note also that, if new assertions arrive, the whole pipeline is started again with the execution of the focus computation process. Internally, RMI maintains a small set of ranked interpretations (*A1…Ak*) on an agenda, only the best of which is extended in the next step. The best interpretation might change by considering more assertions. The pipeline might be restarted if new input arrives from KDMA or HCI. For instance, the human annotator (user) can also invalidate some tags, which could possibly result in RMI switching to another best interpretation if, in the new round, Scoring receives the next set of interpretation Aboxes.

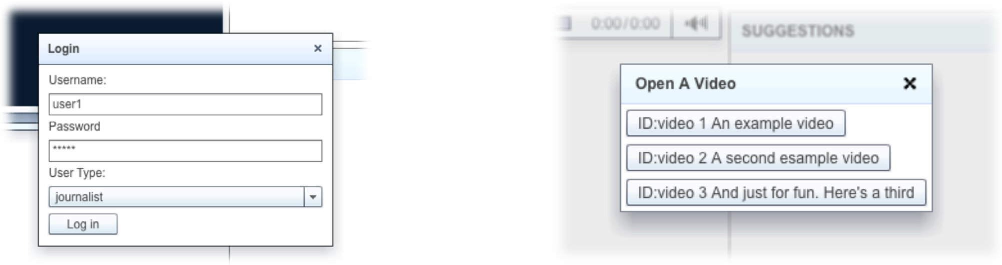
Besides the computation of interpretations the task of RMI is also to give some hints about useful information that might help to discriminate between multiple possible interpretations. Information about this is communicated as so-called queries to HCI and KDMA (see the module Query Generator and [21] for details). The HCI component parses these queries and presents them in the GUI to enable the user to disambiguate between possible interpretations (see the description of HCI). KDMA can use queries to control its data-directed analysis processes.

User Interface and User Interaction (HCI)

The HCI component has to satisfy several requirements. It is, ultimately, the window onto the whole CASAM system and so must provide a user interface that is easy to understand and use. However, the problem is complex with many interactive systems present. The quantity of information produced by the automated components of CASAM is very large with many thousands of assertions being generated. It is unrealistic to expect to present this volume of information to the user and even less realistic to expect them to fully perceive and understand all of it. Similarly, the number of information requests from the system can be very high. Expecting the user to immediately respond to all of these is also not sensible since they are time consuming to address and will distract the user from their own goals. Finally, the representation used internally by CASAM to describe and communicate its annotations is based on description logic. This is not an appropriate representation for the end-user (a media professional) to use and manipulate, since they will not understand it or what it represents.

The end-user will, typically, have a very limited amount of time in which to produce the annotation for the document. In the case of a journalist annotating a news report they will have deadlines to meet, after which the value of the report may be substantially lower. They may, typically, only have a very small number of minutes in which to produce their annotation. The role of the HCI component, therefore, becomes twofold:

1. Provide a user interface that is effective, responsive and easy to use.
2. Manage the dialogue between the system and the user to ensure best use of the user's time.



Login

Username: user1

Password: *****

User Type: journalist

Log in
0:00/0:00
SUGGESTIONS
Open A Video
ID:video 1 An example video
ID:video 2 A second esample video
ID:video 3 And just for fun. Here's a third

(a)

(b)

Figure 9: User login screen (a) followed by video selection popup (b)

We must seek to gain as much value from the interaction as possible whilst minimising the cost (both in terms of the user's time but also as measured by their cognitive load). This means that we must select what information to present to the user and determine the best way to present it. We must also understand what information to explicitly request from the user and when and how to make this request.

CASAM HCI Architecture

The HCI component's architecture is divided into two parts. The front-end is designed to be executed on the end-user's client machine in order to provide a responsive user interface. It is implemented as an Adobe Flash client in order to ensure portability across different hardware and software platforms. The back-end is executed on a remote server in order to offload heavy processing and bandwidth requirements. The two communicate using a (relatively) lightweight protocol in order to allow the user to work with standard network connections. The back-end implements the agreed web service contracts with the other components of the CASAM system. In principle, both parts could be installed on one machine in order to provide a stand-alone implementation.
The division of responsibilities between the two parts of the HCI component can be viewed conceptually as:

1. The back-end working strategically to determine *what* to display.
2. The front-end working tactically to decide *when* and *how* to display it.

CASAM User Interface/HCI Front End

The HCI front-end component has two roles; it provides the user interface to the whole CASAM system and it implements those parts of the HCI component's architecture that are tightly coupled with the interface. It communicates with the back end component using web services.

The user interface has been designed using an iterative user-centred design methodology. In the first phase an understanding of the user requirements is constructed through building user personas, scenarios and early stage prototypes. Representative gold standard annotations were produced by expert end-users to give an insight into the necessary user representations of the final annotation and of the process. The context, abilities and preferences of the end-users were also assessed and used to guide the design of the first stage prototypes.

The final prototype presents a user interface model that is loosely based upon video editing software UI paradigms with which the end-users will be familiar. User initially login with an existing account details (Figure 9a) and then choose an appropriate video to annotate (Figure 9b). The video is then presented to the user with the various interaction components organised around the video.

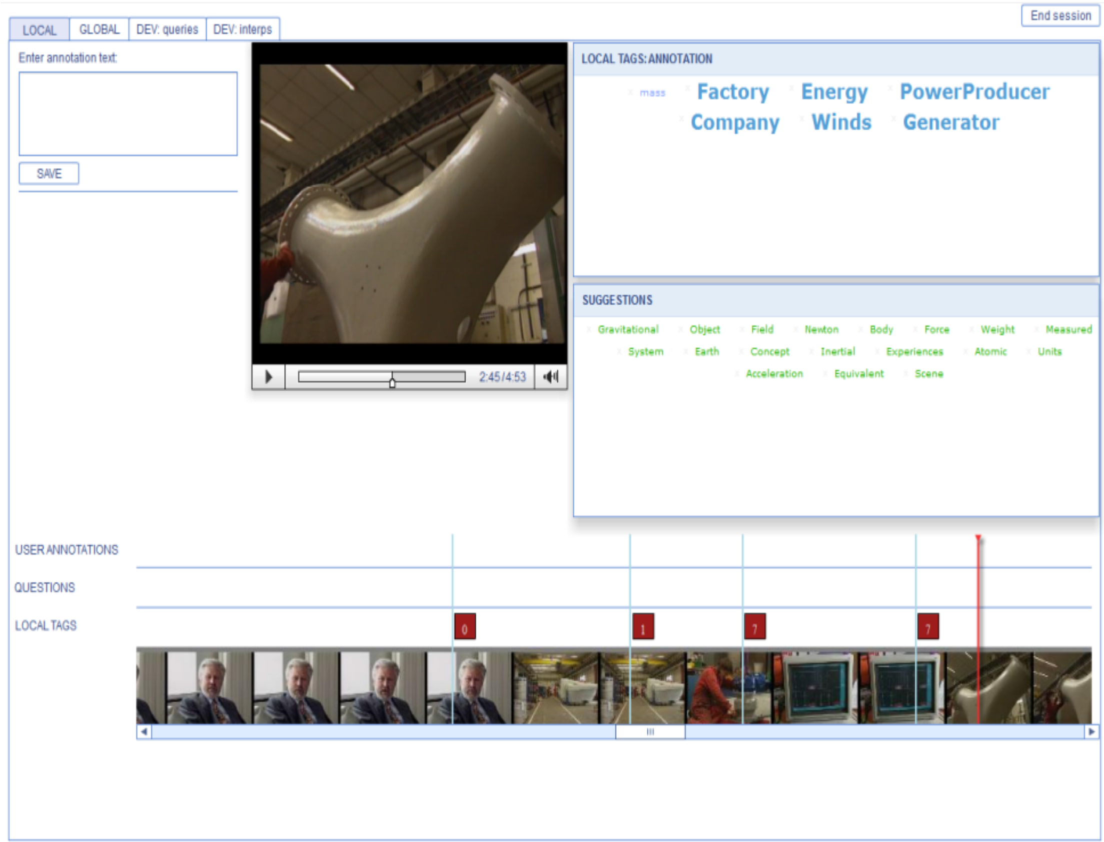


Figure 10: Example instance of CASAM user interface

Figure 10 provides an overview of the entire interface as would be visible to the user. The annotations are organised around shots and also the video as a whole. They are able to navigate around the video using standard video controls and also through the video timeline at the bottom of the screen. There are alternate tabs that allow them to switch between the global video annotation and the annotation for the current shot. The top panel on the right shows the current annotation state while below it are a series of suggested annotations. To the left is a panel for user free-text annotations and below that the area where queries to the user are displayed.

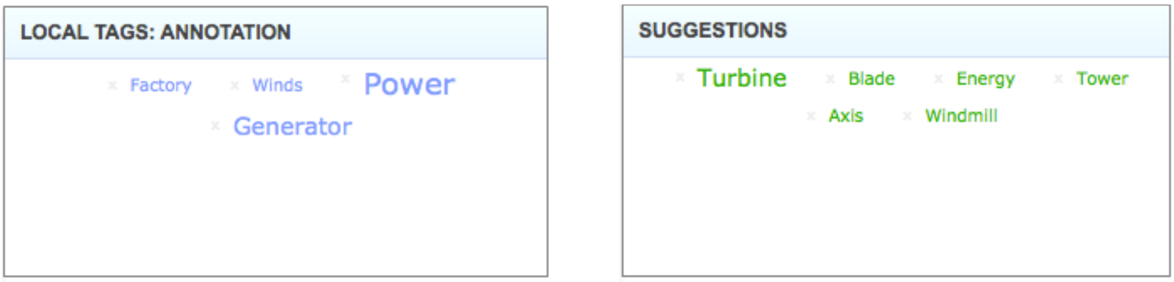
Dialogue Management/HCI Back End

The final part of the HCI component is the HCI back end. This is responsible for the strategic aspects of the management of the interaction with the user. It also implements the interface to the rest of the CASAM system: it receives, analyses and organises the information flowing to and from the other CASAM components.

In this component we also build a number of empirical models that are used to predict the annotations that are likely to be used for this document. These are transmitted to the front end and shown to the user as appropriate. The back end also generates queries for the user, again based upon empirical evidence and the current state of the system's annotation.

Managing the Dialogue

As we have discussed earlier, the RMI and KDMA components of CASAM can generate very large amounts of data (e.g. assertions about the current video) and also requests for information (queries representing information that RMI is requesting in order to restrict the annotation space). Ideally, the user would be able to understand and respond to all of these, in order to assist the machine intelligence components in RMI and KDMA. However, the quantity is so great that this is rendered infeasible. Instead, we need to manage this dialogue so that the limited time and cognitive capacity of the user is utilised most effectively. This means that we have to restrict the amount of information that



(a) (b)

Figure 11: The CASAM tag cloud component showing both the current state of annotation (a) and some suggested tags (b)

is presented to the user. We have to limit the number of explicit requests that the system makes to the user and we need to organise this so that the dialogue is as natural as possible. At the same time we need to maximise the information gain that is made from the dialogue – both to improve the final annotation and to improve the efficiency of the annotation process (for instance, the RMI component will work more effectively if the annotation space is constrained by user input). In addition to these competing requirements, there are also constraints on the amount of time available from the user. This can be extremely limited, perhaps only a few minutes, and so we must extract as much value from the interaction as possible.

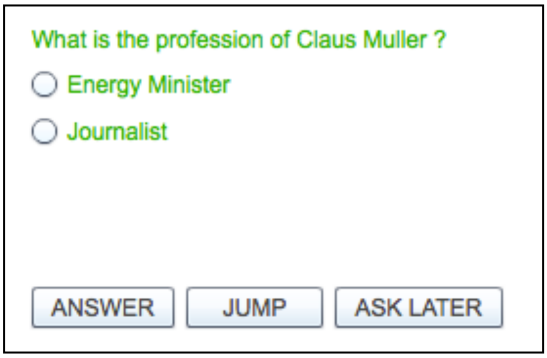
The KDMA and RMI components are able to specify a confidence value with any assertions that they make and so we can use this as one of the measures driving the dialogue. Similarly, the RMI component will specify the importance of the information requested through a query and so we have a measure of the value to the annotation process of requesting a piece of information. In Creed et al [13] we describe a series of experiments which were designed to identify a cost associated with different forms of interruptions within an annotation task. This cost, together with the value to the system, can be used to calculate a balance between cost and benefit that can then inform the dialogue management system.

This cost-benefit analysis has been used as the primary driver of the dialogue management. Because the cost is responsive to the dialogue context this means that a coherent dialogue is an emergent property of the cost-benefit balance rather than something over which the system explicitly reasons. As well as displaying information provided by the other CASAM components, the HCI component generates its own suggested annotations and its own queries. These are intended as a complement to those generated by the RMI component. They are the result of using empirical models of the data and the annotation space rather than being based upon formal reasoning over the current annotation and the ontology. That is, the current annotation is used to generate a prediction for probable annotations or to suggest information that is normally associated with those that are already represented.

In the following sub-sections we explore in more depth how the system manages the dialogue with the user and also we outline how the HCI generated tags and queries are created.

Annotations

Annotations are represented in the form of a tag cloud. These tags allow the system to receive assertions from the user without the time constraints of explicit queries and so represent a more passive aspect of the dialogue. The user can choose which annotations to confirm or reject by either selecting a relevant tag (thereby asserting it as true) or deleting it (thereby asserting it as false). The HCI component receives a large number of annotations from the other CASAM components, often too many to usefully display to a user. Many of the assertions generated by the system from both the KDMA and RMI components are of limited value to a user in terms of describing the content. Similarly, many are asserted with a low confidence. The HCI component is, therefore able to use this in order to limit the quantity of information displayed. An example is show in Figure 11a.



What is the profession of Claus Muller ?

○ Energy Minister

○ Journalist

Figure 12: Example of query presented to the user

Suggested Annotations

The HCI component also adaptively displays some suggested annotations, generated through a process that predicts those annotations that are likely to be associated with current annotation state. This process uses a corpus-based approach. It is, essentially, building an empirical model of the annotation space and using this to drive the prediction process based upon the current annotation state. Each of these words has a Term Frequency-Inverse Document Frequency (TF-IDF) value, which describes how discriminatory the terms are in describing the content of the annotation. An example of some suggested tag and the original tags from which they were derived is show in Figure 11b.

Queries

Queries represent suggestions for information to be obtained from the user. The role of the HCI component is to turn these formal queries into meaningful dialogue elements and optimise their role in that dialogue. There are several factors that affect the efficiency of the dialogue and its impact on the user's experience, and these will depend upon the current context of the system.

Timing and Interruptions

Queries are potentially disruptive, and the extent of this disruption is dependent upon the context of the user and the state of the system. Since queries interrupt the user, requiring the user to form a response, an adaptive system needs to be able to manage these interruptions to the benefit of both the user and the system. There is significant research discussing the impact of interruptions on the user [3]. However, much of this research is related to interruptions in the form of notifications. In the case of the CASAM system the interruptions are questions that have a non-trivial relationship with the underlying annotation task. The impact of context between the interrupted and interrupting task is less well understood.

Determining whether or not a query should be displayed is dependent upon the dynamic state of the system and of the user. This takes the form of a cost-benefit trade-off where the cost comes from the cost of interrupting and asking a question and the benefit comes from the value of the answer. The benefit of answering a query is relatively easy to quantify, especially in the case where a response results in disambiguating between possible known worlds arising in the reasoning system of RMI. In this case the benefit of a user response to a query can be quantified by the magnitude of potential changes to the known worlds. Quantifying the cost of interruption is more difficult. In order to inform the choice of input factors and weightings, and to gain further insight into the impact of interruption within the context of the CASAM system, an experiment was undertaken in the form of a user study [5, 12]. The results of the study indicate that there are two clear factors that define the magnitude of the impact of an interruption on the user:

- *The context of the user:* A significant body of research has shown that interruptions are less costly when the user's cognitive load is lower and that this coincides with boundaries between tasks. In the case of the CASAM system task, boundaries for annotation occur between the video segments.
- *The context of the system:* The results of our study show that the context of the interruption in terms of the relationship between the interruption and interrupted task is important. The cost of interruption is much lower when the interrupting task is related to the interrupted task.

An example showing the presentation of a query to the user is shown in Figure **12**.

HCI Queries

The query generation service operates on the current state of annotation and produces new queries to be posed to the user. The advantages of CASAM HCI being able to generate queries are two-fold:

1. A significant number of the assertions generated by the CASAM KDMA component are not easily associated with the video content directly. This is especially true for assertions generated from auxiliary documents. In order to associate these assertions with concepts that are already identifiable within the video content by position or time we attempt to identify valid relations between an entity that is associated with a video segment and one that is not. If one or more syntactically valid assertions can be formed (subject-relation-object tuple) then a query can be raised.
2. Queries can be generated based on the context of previous questions and the annotation state. For example, if a query response has managed to associate a face recognised in the video with a name mentioned in an auxiliary document then new queries relating to that newly identified person can now be considered, such as questions about their profession.

Queries can also be produced with the aid of query and interpretation schema, which describe which potential annotations we might expect to have available given some confirmed annotations generated by the system. For example, a schema applicable to a sports domain might tell us that if we have a football game and a goal has been scored then a typical annotation set might include:

- The name of the player who scored the goal
- The team who scored the goal
- The goalkeeper who conceded the goal
- The team who conceded the goal

User Models

Different types of user will have different goals and constraints. Therefore their sessions should proceed in different ways, with a dialogue that is adaptive to their particular needs. CASAM supports 3 user classes, each with different requirements that can influence how query-based dialogues should proceed.

- A journalist may have relatively short periods of time in which to work. They are able to manipulate the position of the video play-head however, the playback of the video is paused when a query is presented to provide a clear interruption and to allow the user time to formulate a response.
- An archivist will likely have significant time and will aim to describe the content as comprehensively as possible. To best support this, and in addition to the interaction behaviour experienced by journalist user type, archivist user types will experience a pause both when a query is presented but also at the end of each video segment. This provides an opportunity for more queries to be presented to the user since the paused state reduces the interruption cost as described previously in the Timing and Interruptions section.

- Live users are an exceptional case as the video play head is no longer controllable by the user since the video is assumed to be a live feed. Users are annotating the video directly as it is recorded or as it arrives. Queries are much more costly in this scenario since there is less likelihood of a lull in the users cognitive load. Therefore the presentation of a query is more dependent upon the context of the query with the current content of the video and with the previous dialogue with the user.

The behaviour of these three user models is defined both in the weightings used in the calculation of the cost-benefit model but also in the functionality of the play head. Currently each of these user models has a different predetermined set of weights used in the cost-benefit calculations. Ultimately these weights could be tuned to better suit the user type but also to adapt the interaction behaviour in order to personalise the user interface to better suit a particular user.

Evaluation & User Studies

Since the system was built using user-centred design approaches, the system was evaluated in this context, with both ease and quality of annotation in mind.

- *Ease of annotation* refers to the amount of effort required by the user to annotate a multimedia document. Easiness is related to the time required for the annotation, the number of interactions with the interface, etc.
- *Quality of annotation* refers to the richness of information in the annotation. Richness involves both localisation of information to particular segments of the multimedia document, structure of the information in respect to a particular domain and accuracy of the information.

These factors could also be described as a measure of user satisfaction with the final product, comprising both the actual usability appraised by the professional test users, their objective economisation of work time and effort, and their perceived satisfaction with the results of annotation.

Besides a number of ad-hoc user evaluations and user trial sessions during the entire development process, related in particular to the user interface, key user studies were undertaken on the two intermediate prototypes of the integrated CASAM system as well.

In the course of evaluating the CASAM prototype we employed tried-and-tested techniques to elicit useful feedback from participants. To this end, user evaluations were conducted in face-to-face sessions. We used three main techniques to collect user feedback during practical testing, all complemented with audio documentation for backup and notes made by the test leader:

1. *Thinking aloud and observation*: For this technique, participants were given assignments they had to perform with the CASAM prototype. Participants were encouraged to talk about their impressions and actions during the evaluation session. In such a way, the mental models by which users address a task or try to achieve a goal could be detected and analysed. All the while, the participants' behaviour was also observed in order to detect semi-conscious/habitual interactions with the system or barriers that are not expressly addressed by the user. The benefit of this approach was that user behaviour and user satisfaction became immediately transparent. The need for modifications became apparent, as did the level of need for specific training or introduction to the CASAM tool. At the same time, professional users expressed to what extent CASAM actually caters to their everyday work requirements.
2. *Constructive interaction (teaching back)*: This technique consists of two stages. In the first step, one participant has the opportunity to explore and familiarise themselves with the CASAM system. In the second step, the same participant then explains the functionality of the system to a novice participant. The success rate of this direct user-to-user training tangibly demonstrated the mutual understanding of the system, revealing how deep the actual understanding has become at this point and highlighting features that remain unclear or hard

to grasp. In areas where this “Chinese whispers” test worked well, the system showed very clear and easy usability; where not, the misapprehensions highlight urgent action points.

3. *Collection of express feedback*: Immediately after finishing their hands-on experience with CASAM, participants were asked for their personal evaluation of the system. They filled in a standardised questionnaire and were also given the opportunity to independently express their opinion and possible suggestions. The technique allowed the collection of a wealth of reactions and recommendations. While such information alone, without the abovementioned first two steps would have run the risk of misrepresenting the user experience, since people tend to rationalise or to respond according to pre-existing prejudices, In this case it constituted a useful supplement to the observations made during the practical work with CASAM. However, all user evaluations needed to take into account that users frequently tend to react adversely and insecure to new, unaccustomed software. This is particularly true for those professional users who have long-term experience with other software solutions in the particular field of CASAM.

The first prototype was evaluated with a total of 28 users, and the final one with 34 users located in Germany, Portugal, Greece, and the Netherlands. Participants represented the entire scope of CASAM’s target groups, including archivists, journalists, editors, multimedia producers, and IT experts working in broadcasting and audio-visual production companies, news agencies, audio-visual archives, and as freelancers. Since the first round focused on interface usability and detailed improvement recommendations in the narrower sense, it was conducted primarily with junior to mid level staff involved with production and archiving duties on a practical basis. In contrast, the second round primarily set out to test the advancements and innovations CASAM brings to the business sector and therefore put an emphasis on mid to executive level participants in order to better put the system into perspective.

Participants received a brief explanation of the user interface layout and basic functions of the system and were then asked to initially watch the first few minutes of video to gain some insight into its overall content. At that point, participants were encouraged to proactively navigate the video, to answer and review the system-generated queries that were presented, to enter manual annotations based on their respective professional demands, to select appropriate system-generated annotation suggestions, and to select or delete annotations where appropriate. Unless clarification or further guidance was sought, participants were left to their own devices. Only where it appeared that participants might entirely ignore or miss certain functions within the allotted time frame, were they prompted to look at or try those functions.

After about 15 minutes of interaction with the system, participants were asked to review the annotation results at video level as well as shot level and, where deemed necessary, to manipulate the annotations. In addition to the annotation interface (Figure 10), participants were asked to perform searches using the search interface, consisting of a Google-like query input box, while selected participants (primarily IT professionals) were also confronted with the content management and user administration interface.

User feedback was very rich in detail. The vast majority of participants commented on a limited number of identical issues; at a point roughly two-thirds into first prototype evaluation, few new issues or observations were reported, with the exception of executive-level strategic remarks that only indirectly referred to the qualities of the actual prototype. The first prototype aimed to prove the feasibility of the approach during evaluations and at the same time provide feedback on functionality and usability to help guide the remaining development process. Both these objectives were fully achieved. In addition, participants clearly confirmed that easiness and quality of annotation were already at a significantly improved level over typical approaches used in practical multimedia working environments today.

Observation of uninitiated users interacting with CASAM yielded the overall impression that they very quickly grasped the general purpose of the system and of its main controls. This was not least due to the fact that the interface design was recognised to be inspired by the layout of popular video editing systems. Accordingly, media professionals immediately felt generally “at home”. The proactive, dynamic prompting of user interaction was praised as it was seen as a means to increase and sustain user motivation.

Overall, the participants’ response to the prototype was very favourable. The graphical user interface was generally welcomed, and all subjects recognised and applauded the major time and effort-saving potential of the CASAM system as well as the benefits to be reaped from significantly improved retrieval of archived footage. They stressed that the automatic temporal (or shot-related) allocation of tags to the video alone would already make their lives easier, as they would no longer need to navigate to and record “in” and “out” time codes of relevant shots. The same holds true for speech recognition.

The majority of users were pleasantly surprised by the quality of tags already achieved and conceded that a similar depth and breadth of annotation would have been out of reach with manual annotation, if only for reasons of manpower constraints. The general concept of CASAM met with unanimous approval, as did the overall usability aspects. Basic controls, such as play/pause, sound volume, clicking tags, annotation input, and scrolling through the video were understood immediately, though not necessarily rated fully satisfying or navigated easily. However, it was striking that all subjects experienced difficulties adapting to video behaviour and navigation inside the video and to the interaction with system-generated questions.

In all, user evaluations disclosed general approval of the system and its objectives. Participants clearly saw the demand for and benefits of CASAM in their respective work environments. Interestingly, this held true throughout the broad range of specialisations that were represented. This means that the rather universal approach of CASAM successfully caters to the entire bandwidth of audio-visual activities and is capable of playing a part in all stages of a video’s life cycle.

The more experience participants gained of the working principles of the system, in particular the results of ontology-based semantic analysis and reasoning, the more interested they became in knowing more about CASAM’s inner workings and wished for a “window” into the system that provided indicators from which source and by exactly which method individual annotations were generated. They fully comprehended and appreciated the added value of an integrated semantic annotation tool over mere text, speech, sound and image recognition and over manual systems. However, the still-nascent technology also prompted unrealistically high expectations in some subjects.

In parallel, an extended methodological approach to measuring annotation quality was developed. Building on inspirations from the TRECVID project [35], we essentially suggest an annotation quality metric that gauges search success based on third-party and CASAM-assisted annotations of a controlled set of videos with equally controlled search tasks given to users who have not participated in any annotation process of the videos in question. The speed of retrieval as well as subjective user satisfaction with the search process and results will then serve as a descriptor of the relative annotation quality as a means to an end, i.e., the efficient exploitation of previously untapped-into video material.

Quantitative Evaluation

A quantitative evaluation of the CASAM system was performed using two distinct use scenarios:

Scenario 1: The subject must follow the whole video duration and annotate.

Scenario 2: The subject may use the system to its fullest for fast yet effective annotation.

Seven novice users (of average age of 34) and five expert users (of average age of 31) were selected for the task. The novice subjects had the basic 15-minute training on the CASAM system, giving them insights into the underlying theory and the annotation system demonstration. The expert subjects were journalists and scientists that had a more extensive hands-on experience on multimedia annotation and at least 10 days of familiarisation with the final version of the system as well as earlier CASAM prototypes. Three videos were selected to be annotated by both groups.

Based on the user requirements and expected results of the human-machine synergy methodology, the following hypotheses were formulated:

Hypothesis A: The expert subjects should be noticeably faster than the novice users in both task scenarios. The novice subjects should not exceed 60% of the multimedia document duration for Scenario 1, while they should be more than 20% faster than the video duration for Scenario 2. The expert subjects may be much faster while retaining accuracy.

Hypothesis B: The CASAM-based annotation should be at least 40% more accurate than manual one in terms of inter-annotator agreement and common annotation values (concepts).

To test Hypothesis A, five novice and five experts, in separate groups, were asked to annotate three different videos (the same for each group) using both scenarios in random order. Time, effort (measured in number of clicks) and accuracy were measured.

Table 1 depicts the average values for the above as well as the thresholds derived from Hypothesis A that is grounded. As expected, experts are faster than novices, but we now have an understanding of the margin.

Table 1: CASAM annotation speed results for Hypothesis A grounding. The average video duration (for the three selected videos) was 4:20'.

	Novice group	Expert Group	Threshold
Scenario 1	1.45x	1.23x	1.5x (met)
Scenario 2	0.60x	0.37x	0.8x (met)

For testing Hypothesis B, two novice users were asked to annotate all three videos without using CASAM. They were asked to provide a number of annotations they felt confident they could be used to provide enough information about the video and classify them as GLOBAL or LOCAL, for the latter also providing the time stamp. Apart from that, they were asked to produce ten annotations in addition to their original ones after each video was played at least once. Their efforts were evaluated against a resident expert and between them. Only Scenario 2 was applied. Table 2 shows the comparative results.

Table 2: Measured effort, accuracy, consistency and number of annotations between CASAM and a fully manual approach.

	Effort (# of annotations per 10 clicks)	Accuracy	Agreement (consistency)	# of annotations (average)
CASAM	209.0	>0.9	0.92	69.40
Manual	7.5	0.2-0.5	0.43 (basic) 0.27 (additional)	5.33 (basic) 10.00 (additional)

Both hypotheses were satisfied, showing that the CASAM approach yields a large number of annotations with the user mostly required to approve and disapprove. This leads to high consistency and accuracy, both of paramount importance when retrieving multimedia documents.

Summary & Conclusions

This paper has presented the details of the CASAM architecture and the workings and interrelationships of the components that constitute it. The CASAM system offers a synergistic approach to annotation, using a range of machine intelligence approaches to both detect underlying components of a video, and logical reasoning to construct more complex explanations of the low level features using ontologies and description logics. The orchestrator architecturally coordinates this, whilst the user interacts with a simple-seeming interface that allows free form, user driven annotation and exploration of the multimedia document, whilst reflecting the salient parts of the underlying machine-determined annotation. In addition, the interface implements intelligent dialogue management and user modelling approaches to present appropriate queries to the user, allowing the system to benefit from the user's abilities to comprehend complex scenarios and guide the underlying mechanisms.

The system represents a holistic approach to the annotation of multimedia, and so direct quantitative comparisons with existing systems are relatively meaningless as functionality is very different. Instead the system has been extensively trialled and tested with users throughout its development and at a final evaluation stage, with extensive qualitative and quantitative feedback demonstrating that it offers speed, ease and comprehensiveness advantages.

Whilst the CASAM system has been designed, and tested, for the specific domain of news multimedia, the approach used and lessons learned are applicable across many different domains of annotation. Whether it's the fundamental principle of using computers to undertake complex processing to detect underlying features, or to allow them to create logical worlds from this data, or to work interactively with the user, the system has proven successful. What is important to realise is that, whilst the developments and improvements made in all the contributing areas are important, it is the combination of techniques into an integrated whole that provide the user with a rich and effective experience.

Limitations and Further Work

The success of CASAM is best measured in terms of the quality if annotation it produces. Ultimately this should be achieved by considering the appropriations of retrieved documents as a result of searching a catalogue of CASAM annotated multimedia documents. However, without large scale deployment and a wide catalogue of multimedia documents this is very difficult. In related research this constraints is often overcome using gold standards of annotation. An example annotation, gathered from experts, which represents the idealised human annotation for a given document. However, these are only effective for evaluating the annotation quality of annotation for the document for which the gold standard was originally generated. Additionally, they do not reflect the forms of annotations that are typically generated from automatic analysis. The CASAM system highlights the need to define a non-subjective metric for quality of annotation.

The Performance of the CASAM system is heavily dependent upon auxiliary text documents. Whilst annotation is successfully generated automatically from video material alone, the breadth and depth of annotation improves dramatically when boot strapped with additional text content. Ensuring high levels of performance, without the need for boot strapping with additional text documents, might be achieved using shared empirical information gained for each application domain.

During the evaluation, a commonly raised issue was the difficultly encountered in interpreting queries generated by the system. Queries, produced by the RMI or HCI components are in the form of description logic statements. In order to present a more human readable form each query is parsed to form a complete human readable subject-object question form. However, in many cases this simple parsing is not adequate to form a human comprehendible question in the context of current annotation state.

Acknowledgements

This work was supported by the European commission and partly funded through project FP7-217061. We would like to thank all members of the CASAM project team who contributed to the results of this work, and to all the users who gave their time and comments.

References

1. Abowd, G.D. et al.: The Family Video Archive: an annotation and browsing environment for home movies. Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval - MIR ’03. pp. 1-8 ACM Press, Berkeley, California (2003).

2. Anthimopoulos, M. et al.: A Pixel-Based Evaluation Method for Text Detection in Color Images. 2010 20th International Conference on Pattern Recognition. 1, 2, 3264-3267 (2010).

3. Bailey, B.P. et al.: Measuring the effects of interruptions on task performance in the user interface. SMC 2000 Conference Proceedings 2000 IEEE International Conference on Systems Man and Cybernetics Cybernetics Evolving to Systems Humans Organizations and their Complex Interactions Cat No00CH37166. 2, 757-762 (2000).

4. Bargeron, D. et al.: Annotations for streaming video on the web. CHI 99 extended abstracts on Human factors in computing systems CHI 99. 278 (1999).

5. Bowers, C. et al.: Choosing your moment: interruptions in multimedia annotation. Human-Computer Interaction–INTERACT 2011. pp. 438–453 Springer (2011).

6. Burr, B.: VACA: a tool for qualitative video analysis. Extended Abstracts of CHI: ACM Conference on Human Factors in Computing Systems. pp. 1-6 ACM Press (2006).

7. Cavnar, W.B., Trenkle, J.M.: N-Gram-Based Text Categorization. Ann Arbor MI. 48113, 2, 4001 (1994).

8. Chen, L. et al.: EmoPlayer: A media player for video clips with affective annotations. Interacting with Computers. 20, 17-28 (2007).

9. Cherry, G. et al.: Using a Digital Video Annotation Tool to Teach Dance Composition. Interactive Multimedia Electronic Journal of Computer-Enhanced Learning. 5, 1, (2003).

10. Correia, N., Cabral, D.: Interfaces for Video Based Web Lectures. Sixth IEEE International Conference on Advanced Learning Technologies ICALT06. pp. 634-638 IEEE Computer Society (2006).

11. Costa, M. et al.: Annotations as multiple perspectives of video content. Proceedings of the tenth ACM international conference on Multimedia MULTIMEDIA 02. pp. 283-286 ACM Press (2002).

12. Creed, C. et al.: User perception of interruptions in multimedia annotation tasks. Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries. pp. 619–622 ACM (2010).

13. Creed, C. et al.: Synergistic Annotation of Multimedia Content. Proc. 3rd International Conference on Advances in Computer-Human Interactions. pp. 205-208 IEEE (2010).

14. Domingos, P., Richardson, M.: Markov Logic: A Unifying Framework for Statistical Relational Learning. Engineering. 10, 49-54 (2004).

15. Espinosa, S.: Content Management and Knowledge Management: Two Faces of Ontology-Based Text Interpretation. Hamburg University of Technology (2011).

16. Fagá Jr., R. et al.: A social approach to authoring media annotations. Proceedings of the 10th ACM symposium on Document engineering. pp. 17-26 ACM (2010).

17. Giannakopoulos, T., Petridis, S.: Unsupervised Speaker Clustering in a Linear Discriminant Subspace. Ninth International Conference on Machine Learning and Applications (ICMLA 2010). pp. 1005 - 1009 IEEE Press (2010).

18. Giannakopoulos, T. et al.: User-driven recognition of audio events in news videos. 2010 Fifth International Workshop Semantic Media Adaptation and Personalization. pp. 44-49 IEEE (2010).

19. Gries, O., Möller, R.: Gibbs sampling in probabilistic description logics with deterministic dependencies. Proc. of the First International Workshop on Uncertainty in Description Logics, Edinburgh. (2010).

20. Gries, O. et al.: A probabilistic abduction engine for media interpretation based on ontologies. Web Reasoning and Rule Systems. D, 182–194 (2010).

21. Gries, O. et al.: Media interpretation and companion feedback for multimedia annotation. The 5th International Conference on Semantic and Digital Media Technologies (SAMT 2010), Lecture Notes in Computer Science. Springer. pp. 1-15 (2010).

22. Guimarães, R.L. et al.: Creating and sharing personalized time-based annotations of videos on the web. Proceedings of the 10th ACM symposium on Document engineering DocEng 10. 27-36 (2010).

23. Haarslev, V., Möller, R.: Racer: An owl reasoning agent for the semantic web. Proceedings of the International Workshop on Applications Products and Services of Webbased Support Systems. 91–95 (2003).

24. Hagedorn, J. et al.: VCode and VData: Illustrating a new framework for supporting the video annotation workflow. Proceedings of the working conference on Advanced visual interfaces. pp. 317-321 ACM Press, Napoli, Italy (2008).

25. Hunter, J., Schroeter, R.: Co-Annotea: A system for tagging relationships between multiple mixed-media objects. Multimedia IEEE. 15, 3, 42–53 (2008).

26. Kaye, A.: A Logic-Based Approach to Multimedia Interpretation. Hamburg University of Technology (2011).

27. Kipp, M.: Anvil-a generic annotation tool for multimodal dialogue. Seventh European Conference on Speech Communication and Technology. Citeseer (2001).

28. Lowe, D.G.: Object recognition from local scale-invariant features. Proceedings of the Seventh IEEE International Conference on Computer Vision. 2, [8, 1150-1157 vol.2 (1999).

29. Miller, G.A.: WordNet: a lexical database for English. Communications of the ACM. 38, 11, 39-41 (1995).

30. Nack, F., Putz, W.: Designing annotation before it's needed. Proceedings of the ninth ACM international conference on Multimedia - MULTIMEDIA '01. pp. 251 - 260 ACM Press, New York, New York, USA (2001).

31. Neuschmied, H. et al.: Fast annotation of video objects for interactive TV. Proceedings of the 15th international conference on Multimedia. pp. 158-159 ACM Press, Augsburg, Germany (2007).

32. Patel, S.N., Abowd, G.D.: The ContextCam: Automated point of capture video annotation. UbiComp 2004: Ubiquitous Computing. 301–318 (2004).

33. Schaeffer, S.: Graph clustering. Computer Science Review. 1, 1, 27-64 (2007).

34. Schroeter, R. et al.: A Synchronous Multimedia Annotation System for Secure Collaboratories. 2006 Second IEEE International Conference on eScience and Grid Computing eScience06. 41-41 (2006).

35. Smeaton, A.F. et al.: Evaluation campaigns and TRECVid. In: Wang, J.Z. et al. (eds.) MIR 06 Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval. pp. 321-330 ACM Press (2006).

36. Toutanova, K., Manning, C.D.: Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics. 13, 63-70 (2000).

37. Tsatsaronis, G. et al.: Omiotis: A thesaurus-based measure of text relatedness. Machine Learning and Knowledge Discovery in Databases. 5782, 742-745 (2009).

38. Tsatsaronis, G. et al.: Word Sense Disambiguation with Spreading Activation Networks Generated from Thesauri. Strategy. 1725-1730 (2007).

39. Viola, P., Jones, M.J.: Robust Real-Time Face Detection. International Journal of Computer Vision. 57, 2, 137-154 (2004).

40. Zhai, G. et al.: eSports: Collaborative and Synchronous Video Annotation System in Grid Computing Environment. ISM 05 Proceedings of the Seventh IEEE International Symposium on Multimedia. pp. 95-103 Ieee (2005).