Integrated Computer-Aided Engineering 18 (2011) 15–28
DOI 10.3233/ICA-2011-0360
IOS Press

Multi-domain framework for multimedia archiving using multimodal interaction

Hildeberto Mendonça[a], Olga Vybornova[a,b], Jean-Yves Lionel Lawson[a] and Benoit Macq[a,*]
[a]*Laboratoire de Télécommunications et Télédétection* (*TELE*), *Université catholique de Louvain* (*UCL*), *Louvain-la-Neuve, Belgium*
[b]*Dynamic Intelligent Systems Lab, Institute for System Analysis of the Russian Academy of Sciences, Moscow, Russia*

Abstract. Multimedia content is very rich in terms of meaning, and archiving systems need to be improved to consider such richness. This research proposes archiving improvements to extend the ways of describing content, and enhance user interaction with multimedia archiving systems beyond the traditional text typing and mouse pointing. These improvements consider a set of techniques to segment different kinds of media, a set of indexes to annotate the supported segmentation techniques and an extensible multimodal interaction to make multimedia archiving tasks more user friendly.

Keywords: Multimedia archival, multimodal interaction, segmentation, annotation

1. Introduction

The Internet became a widely accessible network with a distributed infrastructure that allowed it to scale exponentially. This scalability has enabled the availability of several on-line services that produced substantial impact on people's daily lives. A study made by International Data Corporation (IDC) [11] shows that 487 billion gigabytes of data were added to the digital universe in 2008. By 2012, this number will be 5 times bigger. 70% of this number was produced by individuals. In 2007, the amount of digital information created in a year surpassed, for the first time, the amount of storage for it, resulting in significant data loss.

Unknown people have become active content providers, captivating other people's interests and needs. They have been more active than ever, driving a wave of creativity all over the web. Social networks have influenced this process, by giving visibility to users' content, mainly for those with strong friendship links. Friends are seen as potential propellants of content from those they consider reliable sources. The production of new content continues to increase with new content being published every day on web services such as youtube.com, vimeo.com, flickr.com and several other press websites, such as nytimes.com, lemonde.fr, globo.com and so on. Actually, media companies have been dealing with a large amount of data for years; it only became a global challenge and concern with the popularity of the Internet.

An important factor of attractiveness is the way and format in which the creation was conceived. At this point, multimedia content plays a relevant role on the general understanding of the message or expression. The affordable price of broadband services empowered the access and availability of large size content, such as videos on demand, live show video streams, music, podcasts, live radio streams, graphics, photos, bidimensional (2D) and tridimensional (3D) rendering, and so on.

Multimedia content is a rich resource that should be treated appropriately. Pictures, videos, audios, 3D models, etc., are full of meaning, subjectivity, expressiveness, and other influences, represented visually, aurally, spatially, and interpreted in many different ways. Some messages are understood by everyone while oth-

*Corresponding author: Benoit Macq, Head of Unit Laboratoire de Télécommunications et Télédétection – TELE, Université catholique de Louvain – UCL, Bâtiment Stévin – 2, Place du Levant B – 1348 Louvain-la-Neuve, Belgium. Tel.: +32 10 47 2271; Fax: +32 10 47 2089; E-mail: Benoit.macq@uclouvain.be.

ISSN 1069-2509/11/$27.50

ers are understood by target groups. Therefore, it is primordial that the evolution of multimedia archiving systems goes in the direction of rich representation of content and endows users with modern user interaction techniques.

Nowadays, devices with special interaction features have gained popularity. Computers and mobile devices are now endowed with multi-touch surfaces that allow small gesture recognition, accelerometers for movement detection and wider gesture recognition, microphones enabling speech recognition, and cameras for image processing. Therefore, hardware is not a limitation to empower user interaction anymore. However, applications should be developed to profit from these new resources and multimedia applications are potential candidates to maximize their use, bringing accessible entertaining experiences for media producers, editors, consumers, librarians, archivists and others.

Following this tendency, we present a framework conceived and initially developed in the context of the IRMA (from French: Multimodal Research Interface for Audiovisual Content) Project [8]. The framework started aiming to create an economically viable interface for multimodal search and retrieval in indexed multimedia libraries for audiovisual companies, such as television channels, radio stations, surveillance companies and others. Nowadays, the system has been extended in the context of the 3D Media Project [21], adding segmentation and annotation of 3D models.

In this article we will explore the state of the art on multimedia archiving systems for the purpose of content description, investigating their techniques for segmentation and annotation. Then, describe the multimedia archiving framework developed in the context of this research and continue exploring its support for interaction modalities to enhance user experience. In the end, we summarize our contributions, further potentialities of the framework and future works as a consequence of the evolution of this investigation.

2. Related works

Exploring the literature and observing other initiatives that are calling attention of the market and communities of specialists, we could identify systems that can be compared with our work. In order to drive this comparative study we have defined some requirements to explore the evolution of such systems until the current state of multimedia storage, description and retrieval. These requirements have helped to refine the list of systems and to delimit the scope of the research. They are:

- *Multiple media support*: a fundamental requirement of every multimedia system;
- *Distributed architecture*: accessible through the network and scalable for multiple users;
- *Segmentation*: users can excerpt media to delimit relevant content;
- *Annotation*: implemented with segmentation to allow the description of segments' content;
- *Features recognition*: recognition of patterns on media content to assist on the generation of segments and annotations;
- *Multi-domain support:* support for multiple domains to address several user specialties.

These requirements are ordered from basic features to advanced ones. From a standalone to a distributed architecture, the system should evolve technically and it demands a lot of programming effort to avoid data conflicts, deadlocks, inconsistencies, as well as to maximize parallel processing, security, scalability and other challenges. As a plus, segmentation, annotation and feature recognition can be supported to describe the content of media resources. Segmentation and annotation can be done manually and feature recognition is a way to automate these requirements, increasing productivity when dozens of resources are waiting to be described [42]. Lastly, multi-domain can profit from distribution and annotation to allow the access of several specialists working in collaboration on multimedia description.

Considering these requirements, a number of relevant works have been developed by the industry and academy. Among these works, we start mentioning MMSRS (Multimedia Storage and Retrieval System) [33], that aims to provide enough storage capacity, efficient extraction of video fragments and improved quality of stream, but it is domain specific, focusing only on medical image and video data. The work of Doller et al. [7] extends a multimedia database system (MMDB) to use query, index, storage and content description according to MPEG-7, the Moving Picture Experts Group (MPEG) standard for multimedia content description [21], but when making a distinction between indexation and content description, it might lose relevant keys for media identification and being limited to the MPEG-7 specification diminishes the representability of annotated segments.

In fact, there are many MPEG-7 based software, but most of them are designed to a specific media type [20, 36]. This standard is also criticized due to its lack of expressiveness regarding semantic annotations, since

	Multiple Media	Distributed	Segmentation	Annotation	Features Recognition	Multi-domain
MPEG-7 MMDB	+	+−	+	+	+−	+
SemaPlorer	+−	+−	−	+	+−	+−
UCM	+	+	+	+	−	+

Fig. 1. Comparison of most relevant systems

they are decoupled from low-level features and may lead to ambiguity and limited interoperability [14].

SemaPlorer [31] is an application that enables users to explore and view semantic data. The storage infrastructure consists of different semantic data sources, even though it is mostly focused on images and its scalability approach is based on cloud infrastructure, the Amazon's Elastic Computing Cloud (EC2), not in cloud implementation, making it dependent of a cloud service provider.

The User-Centered Multimedia (UCM) environment intends to put users as an active and controlling element in the process of presentation creation, selection and rendering [4]. It segments the media content in diverse components in order to enable users to navigate through the content other than through a timeline-based navigation. It also enables users to annotate segments. However, these features are not so advanced because they are more focused on the user interaction.

Figure 1 compiles a matrix comparing qualitatively the most relevant works investigated. The level in which a system implements a requirement is represented by three signs:

1. '+' means that the work fully addresses the requirements;
2. '−' means that the work does not address the requirement; and
3. '+−' means that the work addresses the requirements to a certain extent, not completely.

The matrix highlights UCM, MPEG-7 MMDB and SemaPlorer as the ones that better fulfill the requirements.

Other applications are compliant with MPEG-21, the MPEG standard for multimedia framework specifications [13]. This standard specifies that a multimedia framework should implement digital items, or media resources, as the object that allows interaction between two or more users. Interaction can be seen as creating, providing, modifying, archiving, and others. User can be seen as individuals, consumers, communities, organizations, governments and others that work with media. In these applications, authors have full control over the process of their digital items [18,27].

In spite of its adoption, MPEG-21 has a series of shortcomings. Specifically about archiving, MPEG-21 states that it is a recommendation for further work [3]. The same for content description, in which the encoding of XML defined by the MPEG-7 standard is planned to be extended to fulfill MPEG-21's requirements. The work of Schrijver et al. [32] has identified scalability problems on the specification that may lead to increasing memory consumption that cannot simply be solved by software optimizations. Shapiro et al. [34] have identified that MPEG-21 do not cover how to manage multiple requests and shared resources, which is unacceptable in distributed systems.

Therefore, we do not fully support the adoption of standards because outdated standards, such as MPEG-7 and MPEG-21, might prevent us from taking benefit of recent technology advances. Furthermore, it is difficult to find a standard that would cover multiple media – image, video, audio and 3D models. Such existing standards have not yet been widely adopted due to their limitations, complexity and slow evolution.

Considering segmentation and annotation, we observed that most of them are specific to a type of media or domain, lacking a better support for several media and semantics representations. Santini [30] calls attention to the fact that representing the nature of a multimedia signal is a very complex task, being influenced by cultural aspects and signals having an iconic or a symbolic representation. A picture of a red Ferrari denotes a red Ferrari, which is an icon, but also denotes richness and success, which is a symbol. This differentiation is barely considered by standards. Kompatsiaris and Hobson [15] provide comprehensive critical overview of the existing and emerging methods of multimedia semantic analysis and multimedia ontologies. They found that at the current stage of progress in this field, it is important to understand the advantages and limitations of automated semantic analysis in terms of several factors such as reliability, accuracy, tractability at the back-end system level, simplicity and comprehension at user presentation level to provide benefits for personal, professional, enterprise and scientific users.

3. Multimedia archiving framework

As previously described, it is primordial that the evolution of multimedia archiving systems goes in the direction of rich representation of content and extensible user interaction.

Taking this evolution as a strategic vision, we decided to consolidate our research works in signal processing and multimodal interaction, identifying what could be a technological basis to make more robust solutions for these fields. Signal processing research works with large datasets of media and these media should be organized somehow to fulfill machine learning algorithms, to preserve obtained results, to allow parallel processing of the same dataset, and so on. Multimodal applications count on signal processing to give more freedom to user interaction and rely on long term memory to improve the precision of recognition algorithms.

The consolidation resulted in the development of a multimedia archiving framework, aiming to help researchers on the investigation of problems related to the processing of multimedia datasets. It works as a back-end for applications that perform recognition on media files and need to store the recognized data. At the same time, the support for manual segmentation and annotation is needed to fix existing data or when recognition is not present, or it is not robust enough for the case.

This multimedia archiving framework is denominated Yasmim. It is based on the principle that framework is a reusable design together with an implementation [9]. The design represents the model of the problem being addressed in a particular domain and the implementation defines how this model can be performed, completely or partially [28]. Hereafter, we explain how the framework was designed to fulfill the needs of multimedia archival.

3.1. Modules

Yasmim was developed in a modular way to allow its extensibility according to the needs of its several applications. Figure 2 depicts the interdependency between the modules of the framework. The *resource* module manages media files by organizing them in the file system, avoiding duplicity, updating, removing, versioning, etc. The *segmentation* module accesses resources to delimit their relevant content. The *annotation* module describes the segments. Finally, the *retrieval* module uses information from resources, segments and annotations to retrieve media to users.

Resources are media supported by Yasmim. The framework is able to segment and annotate images, audios, videos, animations and 3D models. Going into details in which one of these forms, we have a) image as a static, bidimensional and can be bitmap or vectorial. A bitmap image is composed of pixels and each pixel is endowed of color and dimension, influencing the image size and resolution. While images propagate electromagnetic waves, from a shorter length (violet) to a longer length (red), audio propagates sound waves, an energy that is only able to go from a point to another through a material medium capable of vibrating. In a range of 20Hz to 20 Khz, sound waves can be arranged to transmit information that human beings are able to perceive, such as musical notes, speech, noise, etc. Video is a sequence of images that reproduces situations or expressiveness, sometimes synchronized with audio to enhance perceptions and interpretations. Video files might become very large, making them resources that are difficult to store, manage and distribute. Considering the flexibility of vectorial images, coordinated changes on mathematical variables can imply on effective animations. These animations can be predefined or responsive to user interactions. A 3D graphic is composed of elements with width, height and depth. A point in a 3D model is called a vertex, and at least four vertices are necessary to create a basic 3D object. For more complex objects, thousands of vertices are mathematically positioned to represent forms in a virtual space. 3D models are vectorial, thus they have the same advantages of the graphics discussed previously.

3.2. Segmentation

Segmentation, or fragmentation, consists of delimiting meaningful regions of media content [34]. What determines whether the region is useful or not is the purpose of the system application. This purpose guides the direct intervention of the end-user when manually creating segments and also defines the recognition techniques to automatically recognize media content. The segmentation model takes into consideration spatial and temporal dimensions of the content in order to delimit identified meanings. These dimensions define the types of segment, which are: spatial, temporal and spatio-temporal. Each one of these types suites one or more types of media.

A spatial segment delimits a static region in visual media content. It can be bidimensional or tridimensional. Bidimensional segments contain a set of points involving a region of interest:

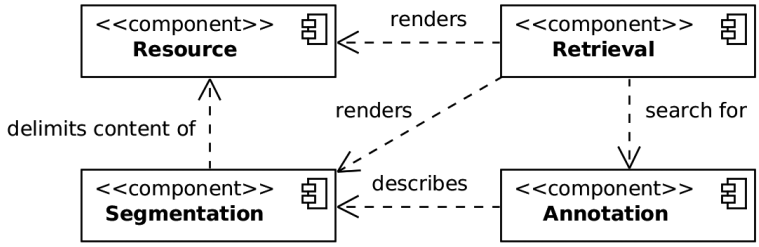

Fig. 2. Modules and dependencies.

$$S_s = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$$

where x and y are coordinates of the points in the Cartesian plane of the content. The value of x and y correspond to the pixels' position of a bitmap media. In vectorial media, x and y correspond to the current canvas size, where the number of pixels is dynamically defined by the graphical controller. A tridimensional segment contains a set of vertices:

$$S_s = \{(x_1, y_1, z_1), (x_2, y_2, z_2), ..., (x_n, y_n, z_n)\}.$$

A temporal segment delimits a sequential region in an audio or video. It defines when the relevant information starts and when it ends, but without any spatial delimitation. The definition of the temporal segment is:

$$S_t = [T_s, T_e]$$

where T_s is the starting timestamp and T_e is the ending timestamp, both included.

The spatio-temporal segment is a merge of the Spatial and Temporal segments concepts. It associates a time tag for each one of an uninterrupted sequence of frames. The formal definition is:

$$S_{st} = [T_s(S_{s1}, S_{s2}, ..., S_{sn}), T_e(S_{s1}, S_{s2}, ..., S_{sn})]$$

where T is a timestamp of the temporal segment and Ss is a spatial segment in a certain instant of time. Each timestamp can be correspondent to one or more spatial segments.

In general, none of the above segment types has constraints on overlapping of two or more segments. In theory, overlapping segments of the same type or different types does not have any advantage or disadvantage, but it could become a concern when those segments are annotated, since it could generate duplicity or require disambiguation. In addition, a segment is continuous and indivisible (immutable), simplifying its handling.

Table 1 shows the compatibility between segments and media types.

Table 1
Comparison of most relevant systems

	Spatial	Temporal	Spatio-temporal
Image	x		
Audio		x	
Video	x	x	x
Animation	x	x	x
3D models	x	x	x

3.3. Linking segments

Segments by themselves are capable of meeting the needs to delimit several content samples. It allows a precise attribution of meaning to the right location, using annotations. However, there still exist gaps to fill in terms of representativeness. These gaps are in the limbo between segments; and our approach to fill them is to use links.

Links are relationships between segments. Figure 3 shows links annotated in the same way that segments are annotated, complementing the content representation. Take as an example a bitmap image which shows overlapped elements, as in the case of the coffee maker in this figure that is partially hidden by the arm. When a part of an object is hidden by another object, its visibility is thus restricted to distinct areas that actually represent what can be seen of that object. Eventually only one segment is not able to represent the entire object since parts of it are hidden, but if we use more segments and link them, all visible details of the object will have the same meaning. In this case, the bottom and the top left of the coffee maker are distinct segments that are linked to represent only one object: the coffee maker. A link is also used to connect two parts of an object (e.g. half lemons) and indicate that together they represent one object: a lemon. Another example: two spatio-temporal segments trace a football player and a ball, respectively. The link between these segments can describe the intention of the player, which is to chase and handle the ball to achieve the goal.

The proliferation of links results in graphs where segments are seen as nodes. The arrangement of a graph

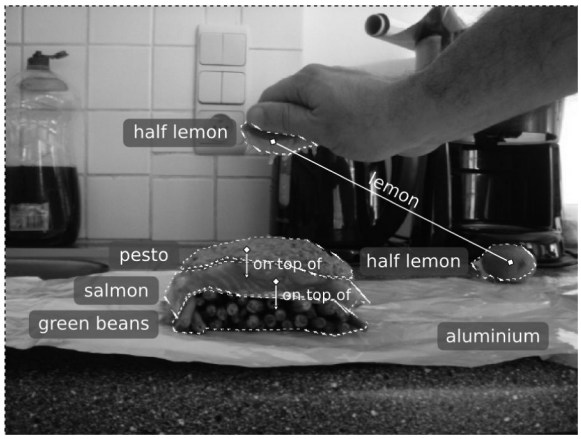

Fig. 3. Links relating segments.

has several configurations, such as: *sequence*, *hierarchy*, *composition*, *cause and effect*, and others. Seen as a sequence, the graph indicates that there is a logical order between the segments, a hierarchy indicates the refinement of a large segment in several smaller ones, a composition connects the parts of a bigger element, and a case and effect indicates the impact that a segment may cause on other segments.

A link connects two and only two segments. It can be unidirectional, when a segment has influence over another, or bidirectional, when both sides are complementary. As we can observe, the contribution that links brings to the content detailing is as important as the segmentation in itself.

3.4. Annotation of segments

Annotation is the assignment of meaning to segments in order to describe media content and, consequently, index them for subsequent media retrieval. The annotation should be made during the creation of segments to avoid saving them without the appropriate indexation. Applying automatic recognition techniques, annotation is which element was recognized and segmentation is where it was recognized. When done manually by the user, the region of interest is selected and the meaning is written or dragged to the segment.

It is possible to extract annotation from a lower level. Some media formats already contain metadata describing the content of the file. The MP3 format, for instance, has a metadata container entitled ID3 [23], used by authors, distributors and others to describe music properties, such as artist, song title, genre, album, etc. These data are of the high semantic level, which can be seen as annotation data and are relevant for media indexation. However, metadata containers make annotation prior to the creation of any segment, which is necessary to associate annotations to the media. To address this specific situation, for every added media, a segment is automatically created to represent the content as a whole, and then, all intrinsic metadata are transformed in annotations and associated with this global segment.

Table 2
Classification of supported annotations

Category	Supported annotation types
Structural	Property
Linguistic	Tagging, transcription, description and adhoc
Semantic	Domain concepts

Yasmim supports several annotation types. They go from a simplistic to a robust form of knowledge representation, giving more flexibility to different user profiles and being domain-independent. They can be assigned to segments and links, covering from simple to complex media content. Due to the number of annotations supported, we are adopting the classification of annotations introduced by [12] as used in [29]. There are three categories:

- *Structural Annotation*: describes physical and logical properties of the content (e.g. for a video: its definition, frame rate, dimensions, format, etc.), usually extracted from low level media file processing and from media descriptors.
- *Linguistic Annotation*: has the pure format of text (words, sentences and narratives) and oriented to human readability, structured according to the grammar of the language. They could be referred to also as Textual Annotations or Lexical Annotation.
- *Semantic Annotation*: corresponds to the addition of semantic data or meta data to the content given one or more agreed ontology. Oriented to machine readability, adopts the structure of a graph composed of triples (subject + predicate + object).

Table 2 classifies the supported annotation types within these categories.

The types are described as follows and they are based on a layering of annotation that reflects different aspects of representation.

3.4.1. Property

A property is an annotation more related to media files characteristics than to its content. It is collected during the initial processing, where embedded algorithms are executed automatically, or can be informed manually by the user, if necessary. Examples of prop-

erties are dimensions, resolution, size, format, volume, duration, etc. A property is composed of a *key* and a *value*, where the key qualifies its respective value. From the examples, we have $<size> = 25\ GB$, where *size* is the key and *25 GB* is the value of the key.

3.4.2. Tagging

Tagging is the assignment of keywords to the media content. Each keyword represents a simple word that identifies the content in the segment or in the links between segments. Keywords are simple, efficient and widely used nowadays to create indexes of information on the web. However, it has limited representativeness when compared with other forms of annotation, although it is more practical for most people and more efficient in terms of searching because most database systems nowadays have good support for text searching.

3.4.3. Transcription

Transcription is a textual and complete description of a monologue, dialog or music lyrics. Practical applications are the automatic recognition of speech in audio sequences, sub-titles, Optical Character Recognition (OCR) in images containing text, etc. It is precise in terms of content representation, but very complex in terms of computation because the extraction of meaning depends on the syntactic and semantic analysis of the transcription The search is not so efficient either. Large textual content demands long processing time for each annotation registry. Syntactic and semantic analyses are needed to allow computers to identify meaning in the text. However, none of these disadvantages decreases the need for this kind of annotation technique.

3.4.4. Description

Description is a detailed text explaining the content of media. It has the same advantages and disadvantages of transcription, but with a different purpose. Practical applications are story telling material, textual summarization, situation description, scenario-based prototypes, etc.

3.4.5. AdHoc

AdHoc annotations do not have commitment to be accurate in terms of content meaning. They could represent opinions, comments, external links, references, etc. AdHoc also does not have any priority in the searching mechanism and it is retrieved when the related media is already available for the user, appearing as additional or complementary information. This is due to the fact that AdHoc annotations are informal, free-text, and can lead to erroneous decisions [15].

3.4.6. Domain concepts

A good balance of representativeness and performance is the use of ontologies to annotate segments. Ontology is used to describe a domain of knowledge [1], which is composed of taxonomy of concepts from a certain domain of knowledge, semantic relationships between these concepts, and instances of these concepts that are representations of scenarios under the modeled domain. Concepts are more representative than tagging because they are well positioned in the domain, but they are less efficient than tagging because there is a cost of exploring the graph of concepts related to them. Comparing with transcription and description, ontologies are less representative than full transcription, but more explicit, computational friendly, and enable derivation of implicit knowledge through automated inference.

When an ontology is designed, it describes only one domain of knowledge in order to be cohesive and contribute to its coherence, reuse, compatibility with other ontologies and less risk of duplicity and ambiguity. On the other hand, when annotating a media, the user may find more than one domain represented in its content, showing that the support for multiple domains is needed to address real world situations.

To illustrate this we have considered the scenario in Fig. 4. It shows the preparation of a recipe called "*Salmon Au Pesto and Green Beans*". It is possible to identify two domains: a) *cooking* brings together all ingredients to prepare the dish: lemon, pesto, salmon and green beans; and b) *kitchen* describes objects in the location where the dish is prepared: balcony, sink, water boiler, coffee machine, soap bottle, and aluminum. These domains are not necessarily complementary, for instance, the soap bottle is not one of the ingredients, and the ontologies are applied on distinct objects.

Now we present an example where multiple annotations are assigned to the same segment: a picture shows a lot of vegetables and three specialists are instructed to annotate this picture. A nutritionist would describe the nutritional properties of each vegetable. A farmer would describe what is necessary to cultivate those vegetables and the behavior of the tilth through the seasons, while a chef would see possible combinations of ingredients for an exotic dish. It shows that each specialist needs his/her accessible domain of knowledge to be able to create comprehensive annotations, and the domain of one specialist might not be useful to other specialists.

Yasmim allows the use of multiple ontologies to annotate segments and links. While observing the media,

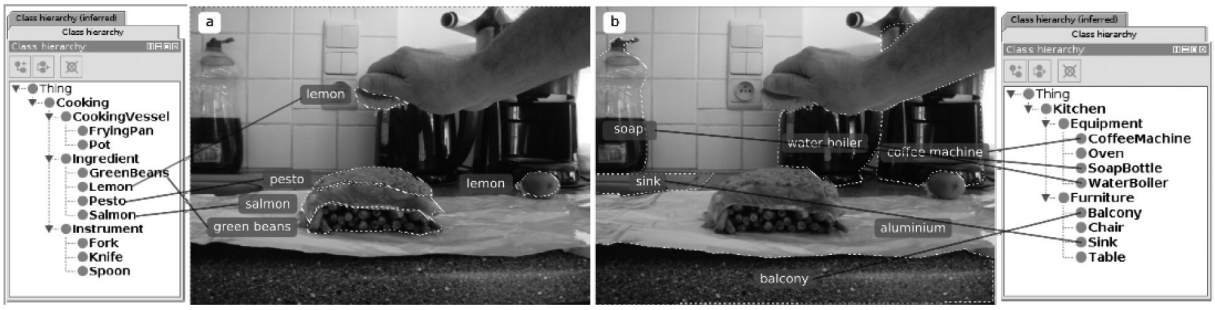

Fig. 4. Annotation using (a) cooking and (b) kitchen ontologies.

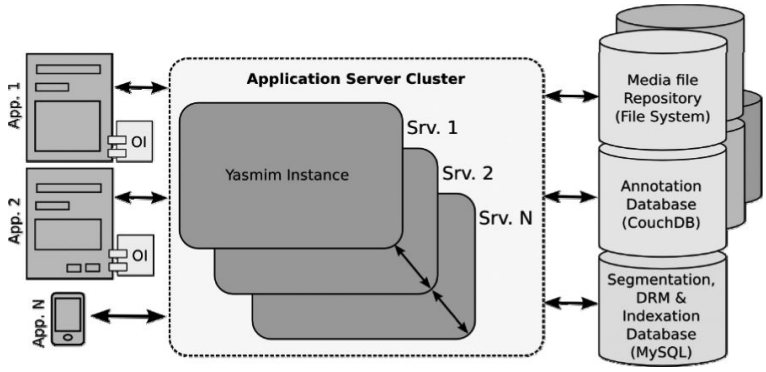

Fig. 5. Yasmim architecture.

the user decides which ontology is more appropriate for the case, considering his/her domain of expertise. Each annotation type is presented in a separate layer over the media content, except for the domain concept, which is not a unique layer, but one layer for each one of the applied ontology.

4. Architecture and implementation

Yasmim's architecture was designed to provide scalability, extensibility, and robustness. It is scalable because it is stateless and uses a peer-based distributed database with bi-directional replication, allowing dynamic addition of new server nodes as the demand increases. It is extensible because several existing solutions can be used with a minimal integration effort. It is robust because the chosen technologies are extensively applied on many other solutions.

Yasmim runs entirely on the server. In order to access and maintain media, client applications should be developed to access the server. The communication is made through web services, using Internet protocols. This approach is more widely accessible in comparison to a recent one proposed by Y. Chen et al. [5], despite being less efficient. To minimize this drawback, these web services are stateless, allowing the system to dedicate all its memory to process media. It also simplifies the use of several machines because it is not necessary to synchronize temporary data between them.

Figure 5 depicts the general architecture of a system using Yasmim for multimedia archiving. Yasmim is in the middle, intermediating data between several data sources and clients. The middleware behind it is an application server called Glassfish [24]. It can manage several instances of the same application spread on several machines, expanding the processing capability according to users' demands. The application manages the information that comes from clients and organizes them in several databases.

There are three data sources:

1. *Media File Repository*: media files are stored in the file system. The optimal efficiency on file access depends on the operating system and the storage system in use.
2. *Annotation Database* is a document database server called CouchDB [2], which processes text more efficiently than relational databases.
3. *Segmentation and Indexation Database* is relational (MySQL [25]), used to store references to

Table 3
Catalog of Yasmim web services

Service	Method	URI	Parameters
Save media	POST	. . . /[type]	
Get media	GET	. . . /[type]/[id]	version=#
			width=#
			height=#
			rotate=#
			format=*format-name*
			filter=*filter-name*
			sample=*true/false*
		. . . /[type]s	search=*keywords*
		. . . /	
Remove media	DELETE	. . . /[id]	version=#
Save segment	POST	. . . /[media-id]/segment	
Get segments	GET	. . . /[media-id]/segments	shape=*shape-name*
			type=*type-name*
			duration=#
			search=*keywords*
Get segment	GET	. . . /[media-id]/segment/[id]	binary=*true/false*
Update segment	PUT	. . . /[media-id]/segment/[id]	
Remove segment	DELETE	. . . /[media-id]/segment/[id]	
Save annotation	POST	. . . /segment/[seg-id]/annotation	
Get annotations	GET	. . . /segment/[seg-id]/annotations	search=*keywords*
			type=*type-name*
		. . . /[media-id]/annotations	
Get annotation	GET	. . . /segment/[seg-id]/annotation/[id]	
Update annotation	PUT	. . . /segment/[seg-id]/annotation/[id]	
Remove annotation	DELETE	. . . /segment/[seg-id]/annotation/[id]	

files in the repository because of its robust indexation mechanisms. It is also used to save segmentation data because database tables have better support to store and retrieve numbers.

The role of the client side is to process heavy operations, such as the support for several modalities, automatic segmentation, automatic extraction of meaning, and also to provide rich user interaction for intuitive manual segmentation and annotation. The data is synchronized with the server, making the media and all related data available for searching and sharing.

The web service architecture is based on REST architectural style [10], where the most important aspect is the addressability of resources. Every media and related information is reachable through a unique identifier. This identifier follows the URI (Uniform Resource Identifier) standard, which is used by HTTP (HiperText Transfer Protocol) to locate resources on the web. With a REST-based framework, we could attach segments and annotations to media, making slight modifications on the URI. This way, not only search mechanisms can benefit from the media description, but many other practical applications as well, since REST web services are easily accessible by any socket library. The key abstraction of information and data in REST is a resource, which is aligned with how we define media, thus every media has its unique URI, and related information is obtained extending or parameterizing media's URI.

The relevant services for general understanding are listed on Table 3. The first column indicates the name of the service, helping the developer to identify which service is more appropriate for his/her needs. Second column shows HTTP methods that could be GET, POST, PUT, and DELETE. The third column shows the relative URI, starting with "http://[server-name/domain]/resources". The brackets indicate that there is a value to fulfill. This value could be predefined, which is the case of [type], or generated, which is the case of [id]. The last column lists the parameters to be appended to the URI. None of the parameters is mandatory, except for the parameter "search" in the "Get Media" service to avoid a high amount of records retrieved.

The value [type] can assume the following values: "image", "video", "audio", "animation", and "3d", which are the types of media supported by Yasmim. These values are mainly useful to summarize the available services. [id] is a UUID (Universally Unique Identifier), an alphanumeric string of 32 characters with so many combinations that it theoretically never repeats for two different records. Because each id is unique, data synchronization, replication and merges are very simplified. UUID is used to define all ids, thus [id],

[media-id], and [seg-id] follow the same rules. Each parameter starts after a semicolon and can be written in any order. Some parameters are not appropriate for all types of media. "rotate", for example, cannot be applied to an audio file (Get Media) and "duration" cannot be applied to a spatial segment (Get Segments). Only media cannot be updated because media files are immutable. Segments and annotations can be inserted, updated, caught and deleted normally. In case a media file needs to be updated, a new version is created with the new file and the previous version is kept historically.

5. Multimodal support

A relevant part of the system is its support for multimodal interactions. We argue that the amount of information present in multimedia files demands enhanced user experience. Indeed, the richness of the user activity involved while interacting with such systems calls for multimodal interactions in order to provide intuitive ways of controlling the system, allowing expressiveness and intuitiveness beyond typing and pointing.

In terms of modality support, only a few multimedia systems were identified. Parageorgiou et al. [26] proposed a multimedia, multilingual and multimodal research system, the Combined Image & Word Spotting (CIMWOS), which is robust in terms of archiving, indexation and retrieval, but limited in terms of the number of supported modalities and types of content. The work of Peng Dai et al. [6] also proposes a multimodal archiving system. However, it is used for a particular case, which is meeting scenarios, and considerable changes are needed to support other domains. Lastly, Jyi-Shane Liu et al. [19] propose a very concise workflow with well delimited segmentation and annotation phases, but the solution only supports the annotation of pictures. Therefore, there is a lack of multipurpose multimedia archiving system that supports several media and multiple multimodal interactions.

Multimodal interaction can be seen as a multimedia form of input for multimedia systems because the interaction may occur due to the analysis of patterns present in streaming of images, sounds, and other signal-based sources. It leads to a match of technologies that are complementary. In addition, we are working with association of meaning and complex content, thus we believe that this kind of application requires improved representativeness of user intentions, which is sometimes difficult to represent using keyboard and mouse only.

The multimodal support is implemented by a desktop application that uses Yasmim framework for multimedia requirements. It runs in the user machine because the application needs to access local resources, such as processors, memory and connected devices, due to its high demanding processes. Devices can be camera, microphone, touch screens, remote controls, and even integrated devices like mobile phones and PDAs (Personal Digital Assistant).

The interaction modalities are related to human sensors defined as:

- *vision*: gesture recognition and movement analysis;
- *acoustic*: speech recognition;
- *haptic*: device vibration; and
- *sensor*: accelerometers, multi-touch, RFID (Radio Frequence IDentification), etc.

The use of modalities is technically complex and an attentive study is necessary to define how they should be adopted by the system. The result of this study is a list of user tasks and their respective compatible modalities. For the case of a multimedia system, the following common tasks were identified:

- create segments on media content;
- annotate segments;
- navigate in large media content;
- search for media;
- select resource;
- zoom, rotate, and translate media;
- increase/decrease volume; and
- play, stop, pause, fast forward, and fast rewind media.

Table 2 puts user tasks, modalities and techniques side by side, judging which kind of interaction is more appropriate for each task. For instance, it shows gesture recognition as a vision modality to segment, navigate, select, zoom, rotate, translate, and change volume of media resources. Speech recognition is an acoustic modality and multi-touch, sketching, and hand-held sensing are sensor-based.

When creating segments, users can make gestures to control the cursor and draw geometric shapes that are converted in spatial and spatio-temporal segments. The same for sketching and multi-touch, but the last one is also used to define starting and ending points of temporal segments. As seen in Section 7, annotations still have a textual representation and speech recognition is indicated to collect text from speech and use it as segment's annotations. When navigating through

Table 4
Classification of supported annotations

Task	Modality	Technique
Create segment	Vision, sensor	Gesture recognition, multi-touch, sketching
Annotate segment	Acoustic	Speech recognition
Navigate	Vision, acoustic, sensor	Gesture recognition, speech recognition, multi-touch
Select resource	Vision, sensor	Gesture recognition, multi-touch
Zoom, rotate, and translate	Vision, acoustic, sensor	Gesture recognition, speech recognition, multi-touch
Increase/decrease volume	Vision, acoustic, sensor	Gesture recognition, speech recognition, hand-held sensing
Play, stop, pause, fast forward, and fast rewind	Acoustic, Sensor	Speech recognition, hand-held sensing

retrieved media, gestures, voice commands and multi-touch move the content to the left, to the right, up and down, overwriting commands such as "next", "previous", and keyboard's arrows. Once the expected content is found, users can use gestures or multi-touch to select it and perform other operations, such as zooming, rotation, translation, and volume control. When using gestures we mean very distinct arms and hands movements, single finger position, but ignoring signs made with fingers. It may avoid users to memorize predefined hand gesture commands, which are compensated by other techniques such as speech, and hand-held sensing, also used to play, pause, stop, fast forward and fast rewind audio and video contents.

To handle the technical complexity of multimodal applications, we are using a component-based solution named OpenInterface (OI) platform [16]. OI has some advantages:

– *Execute components developed in different platforms*: OI was developed in C++ and it has bindings for several compiled languages. This is necessary because the chosen technology influences the quality of modality implementations.
– *Compose components in a pipeline*: components can be connected to each other when the output of a component is compatible with the input of other components. It creates a pipeline to implement the modality, from simple to complex techniques.
– *Has a large repository of compatible components available*: the OI Repository holds a collection of interaction elements for use in the development of interactive systems, especially those constructed using the OI Platform. It is available at: http://forge.openinterface.org.

Fig. 6. Finger tracking for selection and navigation.

As developers benefit from Yasmim to implement multimedia applications, they also benefit from the variety of existing components to make these applications multimodal. From the repository, we implement hand gesture recognition by combining OpenCV (Open Source Computer Vision) components (background, foreground extraction, conditional dilatation and connected components), allowing users to select a region of interest using at most two hands [38]. We have experienced other components to allow hand gestures and single stroke gestures recognition, as we can see in Fig. 6, with finger tracking. For this experiment, we have applied the "1$ recognizer" [40], a simple gesture recognition algorithm similar to Dynamic Time Warping, and a Java version of the Hidden Markov Model Toolkit (HTK) [41], depending on the type of gesture to learn. A waving gesture (based on optical flow analysis) was prototyped in order to horizontally navigate through the library.

To annotate segments, users pronounce textual cues, which are especially useful when the task is performed in a group, where most people do not have access to the keyboard. In this case, the computer's microphone is enough to collect speech and process it with Sphinx-4 [39] component, using 8 Gaussian triphone models trained on the Wall Street Journal Corpus, with a lexicon composed of 5000 words.

6. Discussions

We have considered the three most relevant systems presented in Section 2 and compared them with Yasmim using the same methodology of Fig. 1. They are not fully considered as framework because they were not fundamentally designed to be part of a multimedia system, but to be the extensible multimedia system.

	Multiple Media	Distributed	Segmentation	Annotation	Features Recognition	Multi-domain
MPEG-7 MMDB	+	+−	+	+	+−	+
SemaPlorer	+−	+−	−	+	+−	+−
UCM	+	+	+	+	−	+
Yasmim	+	+	+	+	−	+

Fig. 7. Comparison of systems considering Yasmim.

However, they are comparable with a pure framework like Yasmim because they comply with the set of requirements listed previously. Therefore, this comparison is more positioned in a level of requirements than in the system as a whole.

Figure 7 depicts a version of the matrix presented in Fig. 1, including Yasmim in the comparison. Some columns are highlighted when Yasmim implements the respective requirement. From those requirements, Yasmim implements five, which are: support for multiple media, distributed, segmentation, annotation, and multi-domain.

Going into details in each requirement:

- *Multiple media*: it is able to segment and annotate images, audios, animations, videos and 3D models. Within these media forms, there are also dozens of formats and the framework can support most of them, inheriting this capability from the Java^{TM} platform.
- *Distributed*: it provides REST web services widely available on the network that are accessible for clients developed in different languages, platforms and devices. In essence, every technology that has support for communication using sockets can access Yasmim's data and perform operations on the server, characterizing a modern approach of distribution, aligned with existing web architectures.
- *Segmentation*: it implements spatial, temporal and spatio-temporal segments as vectorial structures, making it applicable not only for bitmap content, but also for vectorial content. In addition, segments can be mutually linked, creating a graph of connections and filling the gap of meanings floating in the limbo not segmented.
- *Annotation*: it provides a greater coverage of the descriptive formats that can be given to segments, which are: property, tagging, transcription, description, adhoc and domain concepts.
- *Multi-domain:* it allows the use of multiple ontologies to annotate segments and links. While observing media, users decide which ontology is more appropriate for the case, considering his/her domain of expertise.

This framework transfers the responsibility of feature recognition and modality implementation for the clients because these processing can be costly, thus impacting the performance of the system as a whole if done in the server (centralized). On the other hand, it would be very necessary for mobile devices, which do not have enough processing power. Or some recognition made on the server can be saved and retrieved faster the next time it is requested. None of the systems is comfortable with feature recognition.MPEG-7 MMDB can run it only if third-party developers implement it and attach it as a plugin. MPEG-7 MMDB does not distribute these plugins by default. SemaPlorer is able to recognize some information in the context, but aided by additional metadata retrieved from the server.

In the requirements that all are competing for, such as multiple-media support, distribution, annotation, and multi-domain, each technology has its own goals, thus the implementation of these requirements are naturally different. What it is important to emphasize are the trends that these kinds of technologies are following and how the newest platforms and systems are allowing the evolution of multimedia solutions.

During the development of this research, we faced some challenges that may demand special attention in future investigations. Since media files are immutable, a version control is implemented to preserve their integrity in a long term archiving and keep all modifications historically organized in different copies. However, it presented itself as less cost effective because of the excessive use of disc space. Three modifications mean an average of 3 times the initial memory allocation. In fact, we believe that segments can also be applied on version control optimizations, keeping the historic of modified parts only. This is relatively challenging because in case of modifications, the content should be re-analyzed in order to update existing segments due to content shifting. Segments may also be useful to make DRM (Digital Right Management) more flexible, increasing the granularity of rights by setting them to segments and annotations

The use of ontology to annotate media makes this work domain-independent, but it leads to the need of additional attention when designing different domains. For example, specialists from the same field might produce distinct versions of the same domain; from dif-

ferent fields might produce ambiguous and duplicated concepts. Therefore, when considering multiple domains, a methodology to model those domains is needed in order to guide specialists to avoid such problems.

We are reusing available multimodal components from the OI repository. These components were developed by other researchers, specialized on their specific modalities. We consider that usability evaluations are under their responsibility, whose results can be used by developers as criteria of choice. At the same time, some level of evaluation should be done in the future because the context of use (user, environment, and platform) has changed and the combination of modalities may be also verified.

7. Conclusion

We have presented in this paper a multi-purpose, extensible and scalable framework for multimedia archiving to enhance the capacity of content description, using enhanced user interactions. This framework is extensible and with multi-purposes, allowing its application on conceptually distinct problems.

We have contributed to the field of multimedia systems by simplifying the way multimedia is added to existing applications and making distributed multimedia management accessible for non-specialized developers. With vector-based segments, Yasmim supports multiple media and formats. These segments are described with a consolidated offering of annotations, providing a good coverage of descriptive formats, emphasizing the multi-domain support, by the use of ontologies to describe several domains present in segments. The client-server communication is implemented using REST web services, which has been, as far as our knowledge concerns, a pioneer initiative in terms of multimedia systems. It even allows the access to Yasmim from simple HTML and Javascript implementation.

Our experience with multimodal have shown that a significant effort should be continuously made on the integration of multimedia and multimodal technologies because it extends user's expressiveness, allowing inputs that it is not easily represented by textual means. Therefore, having modalities as a multimedia input, a future work is to reshape the output as multimedia too. To achieve that, the retrieval should be rethought from the traditional list of relevant results to a composition of segments organized in a way that tells a logical sequence of segments, as a documentary, in a single piece of generated audio-visual streaming. Multimodal interaction comes back to the scene by improving the interaction through the resulting media.

References

[1] G. Antoniou and F. van Harmelen, A Semantic Web Primer (Cooperative Information Systems), The MIT Press, 2004.

[2] Apache Foundation, CouchDB Project, [Online] Available: http://couchdb.apache.org. [Accessed: Jan 25, 2010].

[3] J. Bormans and K. Hill, MPEG-21 overview – version 5. Tech. Rep. 21000, ISO/IEC, Shamghai, 2002.

[4] D. Bulterman, User-centered control within multimedia presentations, *Multimedia Systems* **12**(4–5) (2007), 423–438.

[5] Y. Chen, Y. Wu, B. Wang and K.J. Ray Liu, An auction-based framework for multimedia streaming over cognitive radio networks, IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP), Dallas, Texas, Mar. 2010.

[6] P. Dai, L. Tao and G. Xu, Dynamic context driven human detection and tracking in meeting scenarios. VISAPP (Special Sessions) 2007, pp. 31–40.

[7] M. Doller and H. Kosch, The MPEG-7 Multimedia Database System (MPEG-7 MMDB), *Journal of Systems and Software* **81**(9) (2008), 1559–1580.

[8] Communications and Remote Sensing Laboratory, Multimodal search interface in audiovisual content – IRMA, TELE Lab. [Online] Available: http://www.tele.ucl.ac.be/view-project.php?name=IRMA [Accessed: Jan. 25, 2010].

[9] M. Fayad, D. Schmidt and R. Johnson, Building application frameworks: object-oriented foundations of framework design. Wiley & Sons, 1999.

[10] R. T. Fielding, *Architectural Styles and the Design of Network-Based Software Architectures*, Ph.D. dissertation, University of California, Irvine, CA, USA, 2000.

[11] J. Gantz and D. Reinsel, *As the Economy Contracts*, the Digital Universe Expands, IDC, May, 2009.

[12] S. Handschuh, *Semantic Annotation of Resources in the Semantic Web*, In: Semantic Web Services: Concepts, Technologies, and Applications, 2007, pp. 135–155.

[13] ISO, Multimedia framework {MPEG-21}- Part 1: Vision, Technologies and Strategy, 2004.

[14] Y. Kompatsiaris and P. Hobson, Semantic Web Services: Concepts, Technologies, and Applications, Chapter 1, Introduction to Semantic Multimedia, 2008, pp. 3–13.

[15] Y. Kompatsiaris and P. Hobson, Semantic Multimedia and Ontologies, Springer Science+Business Media, LLC 2008.

[16] L. Lawson, A. Al-Akkad, J. Vanderdonckt and B. Macq, An open source workbench for prototyping multimodal interactions based on off-the-shelf heterogeneous components. EICS '09. ACM, New York, NY, 2009. pp. 245–254.

[17] L. Lawson, *Engineering Multimodal Interactions through Rapid Integration of Heterogeneous Components*, PhD Thesis. Université catholique de Louvain, May, 2010.

[18] A. Lie, K. Grythe and I. Balasingham, *On the use of the MPEG-21 Framework in Medical Wireless Sensor Networks*, In: Applied Sciences on Biomedical and Communication Technologies, 2008. ISABEL '08, 2008, pp. 1–5.

[19] J.-S. Liu, M.-H. Tseng and T.-K. Huang, Mediating team work for digital heritage archiving. JCDL'04, ACM, 2004, pp. 259–268.

[20] M. Lux, Caliph & Emir: MPEG-7 photo annotation and retrieval, ACM Multimedia, 2009, pp. 925–926.

[21] J.M. Martinez Sanchez, R. Koenen and F. Pereira, MPEG-7: The Generic Multimedia Content Description Standard, *Part 1, IEEE MultiMedia* **9**(2) (2002), 78–87.

[22] Multitel, "3DMedia", MediaTic. [Online] Available: http://mediatic.multitel.be/platforms/3dmedia.html. [Accessed: Jan. 25, 2010].

[23] M. Nilsson, ID3 Tag Version 2.3.0, [Online] Available: http://www.id3.org/. [Accessed: Jan 25, 2010].
[24] Oracle Corp. Glassfish Application Server, [Online] Available: https://glassfish.dev.java.net, [Jan. 25, 2010].
[25] Oracle Corp., MySQL, [Online] Available: http://mysql.com. [Accessed: Jan 25, 2010].
[26] H. Papageorgiou, P. Prokopidis, A. Protopapas and G. Carayannis, Multimedia indexing and retrieval using natural language, speech and image processing methods. Multimedia Content and the Semantic Web. John Wiley & Sons, 2005, pp. 279–297.
[27] C. Poppe, F. De Keukelaere, S. De Zutter, S. De Bruyne, W. De Neve and R. Van de Walle, Predictable Processing of Multimedia Content, Using MPEG-21, Digital Item Processing, PCM, 2007, pp. 549–558.
[28] D. Riehle, Framework design: a role modeling approach, Ph.D. thesis, ETH Zurich, 2000.
[29] F. Rinaldi, J. Dowdall, M. Hess, J. Ellman, G.P. Zarri, A. Persidis, L. Bernard and H. Karanikas, Multilayer annotations in Parmenides, In: K-CAP2003 Workshop on Knowledge Markup and Semantic Annotation, 2003.
[30] S. Santini, Multimedia Semiotics, Encyclopedia of Multimedia, Springer, 2006.
[31] S. Schenk, C. Saathoff, S. Staab and A. Scherp, SemaPlorer – Interactive semantic exploration of data and media based on a federated cloud infrastructure, *J Web Sem* **7**(4) (2009), 298–304.
[32] D. De Schrijver, W. De Neve, K. De Wolf, R. De Sutter and R. Van de Walle, An optimized MPEG-21 BSDL framework for the adaptation of scalable bitstreams, *J Visual Communication and Image Representation* **18**(3) (2007), 217–239.
[33] R. Slota, H. Kosch, D. Nikolow, M. Pogoda, K. Breidler and S. Podlipnig, MMSRS – Multimedia Storage and Retrieval System for a Distributed Medical Information System, HPCN Europe, 2000, pp. 517–524.
[34] L. Shapiro and G.C. Stockman, *Computer Vision*, Prentice Hall, 2001.
[35] A.A. Sofokleous and M.C. Angelides, DCAF: An MPEG-21 Dynamic Content Adaptation Framework, *Multimedia Tools Appl* **40**(2) (2008), 151–182.
[36] M. Spaniol, R. Klamma and T. Waitz, MECCA-Learn: A Community Based Collaborative Course Management System for Media-Rich Curricula in the Film Studies, ICWL, 2005, pp. 131–143.
[37] S. Tsekeridou, A. Kokonozi, K. Stavroglou and C. Chamzas, MPEG-7 Based Music Metadata Extensions for Traditional Greek, Music Retrieval, MRCS, 2006, pp. 426–433.
[38] A.D. Wilson, Robust computer vision-based detection of pinching for one and two-handed gesture input. UIST'06. ACM, New York, NY, 2006. pp. 255–258.
[39] W. Walker, Sphinx-4: A fexible open source framework for speech recognition. Technical Report SMLI TR2004-0811, Sun Mycrosystems Inc., 2004.
[40] J.O. Wobbrock, A.D. Wilson and Y. Li, Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes. UIST'07. ACM, New York, NY, 2007. pp. 159–168.
[41] S.J. Young and S.J. Young, The HTK hidden markov model toolkit: design and philosophy, *Entropic Cambridge Research Laboratory* **2** (1994), 2–44.
[42] Y. Wang, Z. Liu and J. Huang, Multimedia content analysis using both audio and visual clues, *IEEE Signal Processing Magazine* **17** (2000), 12–36.