An MPEG-7 Based Description Scheme For Video Analysis using Anthropocentric Video Content Descriptors

N. Vretos, V. Solachidis and I. Pitas

Department of Informatics University of Thessaloniki,54124, Thessaloniki, Greece
phone: +30-2310996304, fax: +30-2310996304
email:vretos,vasilis,pitas@aiia.csd.auth.gr web: aiia.csd.auth.gr

Abstract. MPEG-7 has emerged as the standard for multimedia data content description. As it is in its early age, it tries to evolve towards a direction in which semantic content description can be implemented. In this paper we provide a number of classes to extend the MPEG-7 standard so that it can handle the video media data, in a more uniform and anthropocentric way. Many descriptors (Ds) and description schemes (DSs) already provided by the MPEG-7 standard can help to implement semantics of a media. However, by grouping together several MPEG-7 classes and adding new Ds, better results in the video production and video analysis tasks can be produced. Several classes are proposed in this context and we show that the corresponding scheme produce a new profile which is more flexible in all types of applications as they are described in [1].

1 Introduction

Digital video is the most essential media nowadays. It is used in many multimedia applications such as communication, entertainment, education etc. It is very easy to conclude that video data increase exponentially with time and researchers are focusing on finding better ways in classification and retrieval applications for video databases. The way of constructing videos has also changed in the last years. The potential of digital videos gives producers better editing tools for a film production. Several applications have been proposed that help producers in doing modern film types and manipulate all the film data faster and more accurately. For a better manipulation of all the above, MPEG-7 standardizes the set of Ds, DSs, the description definition language (DDL) and the description encoding [2]-[6]. Despite its early age, many researchers have proposed several Ds and DSs to improve MPEG-7 performance in terms of semantic content description [7], [8].

In this paper we will try to develop several Ds and DSs that will describe digital video content in a better and more sophisticate way. Our efforts originate from the idea that a video entity is a part of several objects prior to and past to the editing process. We incorporate information provided by the preproduction process improve semantic media content description. The structure of the remainder of this paper is as follows. In Section 2 we describe the Ds and DSs. Section (3), we give some examples of use of real data. Finally in Section 4, a conclusion and future work directions are described.

2 MPEG-7 Video Descriptors

In the MPEG-7 standard several descriptors are defined which enable the implementation of video content descriptions.We believe that MPEG-7 must provide mechanisms that can propagate semantic entities from the preproduction to the postproduction phase.Table 1 illustrates a summary of all the classes that we propose in order to assure this connectivity between the pre and the post production phases.

Class Name	Characterization
Movie Class	Container Class
Version Class	Container Class
Scene Class	Container Class
Shot Class	Container Class
Take Class	Container Class
Frame Class	Object Class
Sound Class	Container Class
Actor Class	Object CLass
Object Appearance Class	Event Class
High Order Semantic Class	Container Class
Camera Class	Object CLass
Camera Use Class	Event Class
Lens Class	Object Class

Table 1. Classes introduced in the new framework in order to implement semantics in multimedia support

Before going any further in providing the classes' details, we explain the characterization of those classes and their relations (figure 1). Three types of classes are introduced: container, object and event classes. Container classes, as indicated by their name, contain other classes. For instance, a movie class contains scenes that in turn contain shots or takes, which in turn contain frames (optionally). This encapsulation can therefore be very informative in the relation between semantics characteristics of those classes, because parent classes can propagate semantics to child classes and vice versa. For example, a scene which is identified as a night scene, can propagate this semantic entity to its child classes (Take or Shot). This global approach of semantic entities not only facilitates the semantic extraction, but also gives a research framework in which low-level features can be statistically compared very fast in the semantic information extraction process. The object oriented interface (OOF) which is applied in this framework provides flexibility in the use and the implementation of algorithms

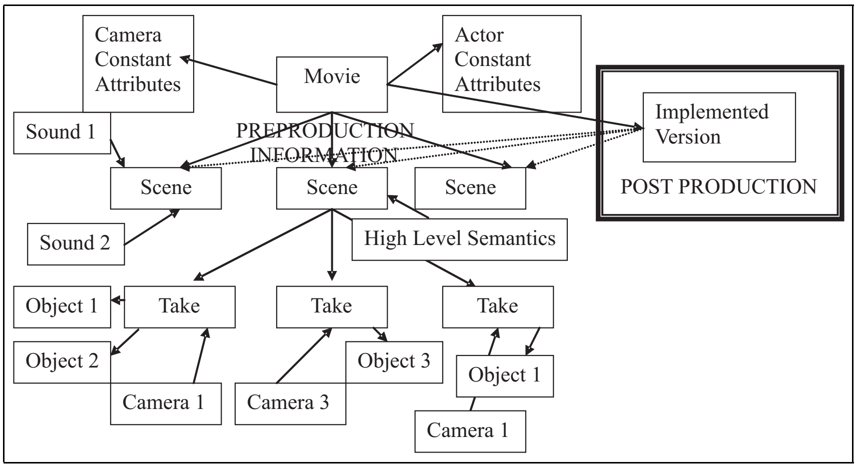

Fig. 1. The interoperability of all the classes enforce semantic entities propagation and exchange in the preproduction phase.

for high semantic extraction. On the other hand, object classes are constant physical objects or abstract objects that interact with the media. Any relations of the form "this object interacts with that object" are implemented within this interface. For example a "specific actor is playing in a specific movie", or "a specific camera is used in this take", or "a specific frame is contained by this take or shot". Finally, the events classes are used to implement the interaction of the objects with the movie. An example of this is "the specific camera object is "panning" on this take". From the examples we can clearly conclude that high semantics relation can be easily implemented with the use of those classes. More specifically:

The Movie Class is the root class of the hierarchy. It contains all other classes and describes semantically all the information for a piece of multimedia content. The movie class holds what can be considered as static movie information, such as crew, actors, cameras, director, title etc. It also holds the list of the movie's static scenes where different instances of the Scene class(described later) are stored. Finally, the movie class combines different segments of the static information in order to create different movie versions, within the Version class.

The Version Class encodes (contains) the playable versions of a movie. A movie can be built using this class. It makes references to the static information of the movie in order to collect different movie parts (take fragments) and construct a shot sequence (figure 2). It can also reference a part of an already defined version's scene. For instance, a movie resume (summary) can be made out of the director's scenes.

The Scene Class contains low-level information with respect to timing, like the start and end timecode of a scene and the duration. A scene theme tag

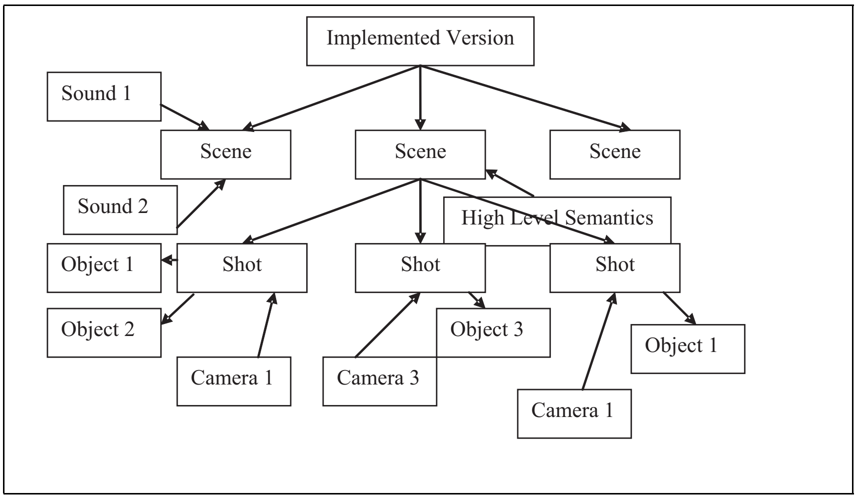

Fig. 2. Implemented versions collect static movie information in order to make a sequential playable movie. This can be achieved either with references from the static movie parts (Takes from pre production phase) or with new instantiated shot classes.

is made in order to make possible a semantic definition of each scene. The High Semantic tag is an instance of the High Order Semantic Class(described later), which gives the possibility to describe semantic actions and events by means of semantic data from the audiovisual content. It correlates higher-level information to provide a narrative description of the scene. Expert systems can be used to provide information for those tags. A sound array tag has also been introduced in a sense that sound and scene are highly correlated in a film. The sound array stores all instances of sounds that are present in a scene. The sound array, as we will see later in more detail, along with low-level information holds high level information, such as the persons that are currently talking or the song that is actually playing or the ambient sounds. Additionally the sound array also captures timecode information for every sound that is heard within the scene. Timecodes can of course overlap (for example flirting while dancing in a fine tune) and this information can be used for extraction of high level scene features. The take and shot array are restricted fields of the scene class, which means that one of them can be active for a particular instance. That class can be used for both pre and postproduction levels. The takes are used in the preproduction environment and the shots in the postproduction environment (within the Version Class).

The Shot Class holds semantic and low-level information about the shot. Two versions are proposed: one with frames and one without. In the second version, we consider the shot as the atom of the movie, which means that it is the essential element of a movie that cannot be divided any further. Several attributes of this class are common in both versions, like the serial number of the shot, the list of appearance and disappearance of an actor, the camera use, and others. The frameless version has a color information tag and a texture information tag, which give information about the color and the texture within the shot. The version with frames provides this feature within the frame class.

The Take Class implements the description of a take. Low-level and high level information is gathered within this class to provide semantic feature extraction, content description and also to facilitate the implementation of a production tool. A take is actually a continuous shot from a specific camera. In the production room the director and his/her colleagues segment takes in order to create a film version. This implementation tries to provide the users with editing ability, as well as assists the directors. For instance, if a director stores all the takes in a multimedia database and then creates an MPEG-7 file with this description scheme he/her will have the ability to easily operate on his film. The "Synchronized With" tag holds information about simultaneous takes. For example in a dialog scenario, where three takes are simultaneous, but taken from different cameras. Algorithms can be built to easily extract the simultaneous takes from a already produced film.

The Frame Class is the lowest level where feature extraction can take place. Within the frame class there are several pieces of information that we would like to store. In contrast with all other classes the frame is a purely spatial description of a video instance. No time information is stored in it. There is, of course, the absolute time of the frame and the local (in take / in shot) time of the frame but whatever information is associated with actor position, actor emotions, dominant color etc. does not have a time dimension. The non-temporality of the frame can be used for low-level feature extraction and also in a production tool for a frame-by-frame editing process (figure 3).

The Sound Class interfaces all kind of sounds that can appear in multimedia. Speech, music and noise are the basic parts of what can be heard in a movie context. This class holds also the time that a particular sound started and ended within a scene, and attributes that characterize this sound. The speech tag holds an array of speakers within the scene and also the general speech type (narration, monolog, dialog etc). The sound class provides useful information for high level feature extraction.

The Actor Class & The Object Appearance Class: The actor class implements all the information that is useful to describe an actor and also gives the possibility to seek one in a database, based on a visual description. Low-level information for the actor interactions with shots or takes is stored in the Object Appearance Class. This information include the time that the actor enters and leaves the shot. Also, if the actor re-enters the shot several times this class holds a list of time-in and time-out instances in order to handle that. A semantic list of what the actor/object is actually doing in the shot, like if he/she is running or just moving or killing someone, is stored for high-level feature extraction in the High Order Semantic Class. The Motion of the actor/object is held as low-level information. The Key Points List is used to describe any possible known Region Of Interest (ROI),

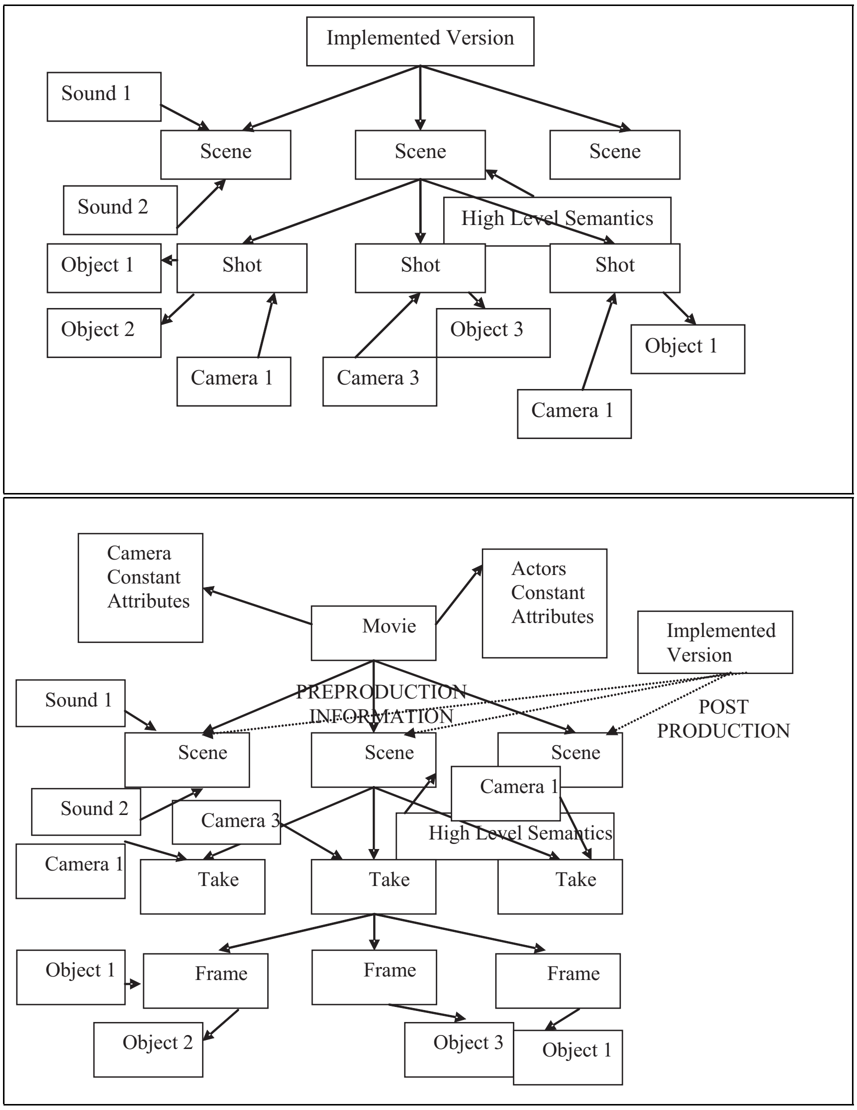

Fig. 3. Frames can be used with Take and Shot classes for a deeper analysis, giving rise to an exponential growth of the MPEG-7 file size.

like the bounding box, the convex hull, the features point etc. For instance, this implementation can provide the trajectory of a specific ROI like the eye or a lip's lower edge etc. The pose estimation of the face and the emotions of the actor (with an intensity value) within the shot can also be captured. The latter three tags can be used to create high-level information automatically. The sound class also holds higher-level information for the speakers and in combining the information we can generate high-level semantics based also in the sound content of the multimedia to provide instantiations of the High Order Semantic Class.

The High Order Semantic Class organizes and combines the semantic features, which are derived from the audiovisual content. For example, if a sequence of actors shaking hands is followed by low volume crowd noise and positives emotions this could probably be a countries leaders handshake. The Object's Narrative Identification hosts semantics of actors and objects in a scene context, for example (Tom's Car) and not only the object identification like (Tom, Car). The Action list refers to actions performed by actors or other objects like "car crashes before a wall" or "actor is eating". The events list holds information of events that might occur in a scene like "A plane crashes". Finally, the two semantic tags, Time and Location, define semantics for narrative time (night, day, exact narrative time etc) and narrative location (indoor, outdoor, cityscape, landscape etc).

The Camera & The Camera Use Class: The camera class holds all the information of a camera, such as manufacturer, type, lenses etc. The Camera Use class, which contains the camera's interaction with the film. The latter is very useful for low-level and high-level feature extraction from the film. The camera motion tag uses a string of characters that conform to the traditional camera motion styles. New styles have no need to re-implement the class. The current zoom factor and lens is used for feature extraction as well.

The Lens Class implements the characteristics of several lenses that will be used in the production of a movie or documentary. It is useful to know and store this information in order to better extract low-level features from the movie. Also for educational reasons one can search for movies that are recorded with a particular lens.

This OOF essentially constitutes a novel approach of digital video processing. We believe that in a video retrieval application one can post queries in a form that only simple low-level features cannot answer yet. Also, video annotation, has been proven [7]to be non productive, because of the objectivity of the annotators and the time consuming annotation process. The proposed classes have the ability to standardize the annotation context and they are defined in a way that low-level features can be integrated, in order to enable an automatic extraction of those high-level features. In figure 4 one can see the relation between semantic space and technical (low-level) features space. The proposed classes are an amalgam of low-level features, like histograms, FFTs etc, and high semantic entities, like scene theme, person recognition, emotions, sounds qualification, etc. Nowadays researchers are very interested in extracting high semantic features

[9],[10],[11]. Having this in mind, we believe that the proposed template can give the genre of a new digital video processing approach.

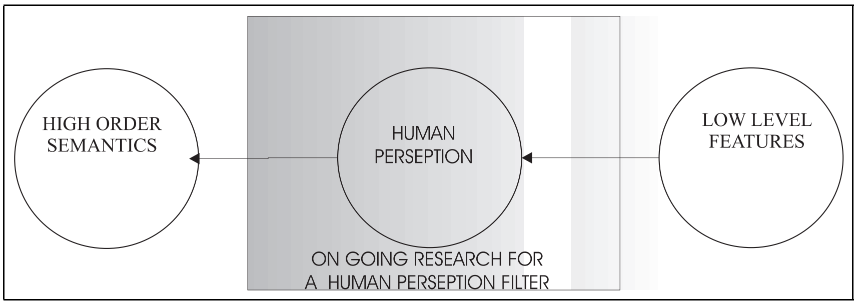

Fig. 4. Low-level features space is captured by human perception algorithms and this mechanism will produce the semantic content of the frame.

3 Examples of Use

3 types of examples, will be provide in this section . These include one for preproduction environment, one for postproduction and the combination of preproduction and postproduction.

- In preproduction environment the produced xml will have a movie class in witch takes,scenes,sound information and all the static information of the movie will be encapsulated. In Figure 5, one can visualize the encapsulation of those classes. Applications can handle the MPEG-7 file in order to produce helpful tips for the director while he/she is in the editing room or better assist him in editing with a smart agents application.
- In the second type of application all the types of the pull and push applications, as defined in the MPEG-7 standard[1] can be realized or improved. Retrieval application, will be able to handle in a more uniform way all kinds of request whether the latter are associated with low-level features or high semantic entities.
- Finally the 3d type of application, is the essential reason for implementing such a framework. In such an application a total interactivity with the user can be established, by providing even the ability to reediting a film and create different film version with the same amount of data. For example, by saving the takes of a film and altering the editing process, the user can reproduce the film from scratch and with no additional storage of heavy video files.Creating different permutation arrays of takes and scenes results in different versions of the same film. Nowadays media supports (DVD,VCD,etc) can hold a very large amount of data. Such an implementation can be realistic, if we consider that already produced DVDs provides users something more than just a

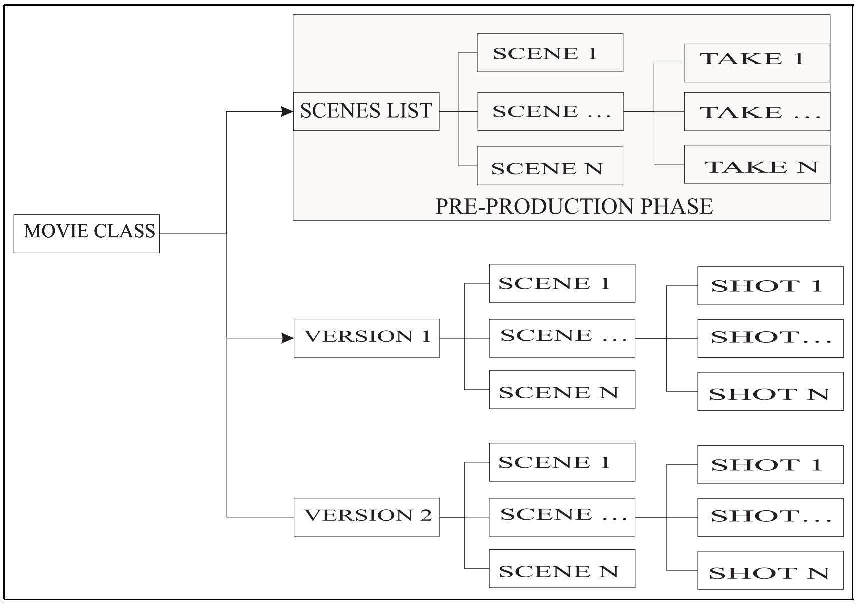

Fig. 5. Shots exist only in the post-production phase as fragments of takes from the pre-production phase, and inherite all semantic and low-level features extracted for the takes in the pre-production phase.This constitutes the semantic propagation mechanism. Several versions can be implemented without different video data files.

movie e.g. (behind the scene tracks, or director's cuts, actors' interviews etc).Media Interactivity in media seems to gain a big share in the market, the proposed OOF can provide a very useful product. It is obvious that such a tool can give rise to education programs, interactive television and all the new revolutionary approaches of multimedia world.

4 Conclusion and further work areas

As a conclusion we underline the fact that this framework can be very useful for the new era of image processing witch is focusing in semantic feature extraction and we believe that it can provide specific research goals. Having this in mind, in the future we will concentrate our research in filling automatically the classes' attributes. On going research has already underdevelopment tools for extraction of high semantic feature.

5 Acknowledgement

The work presented was developed within NM2, a European Integrated Project (http://www.ist-nm2.org), funded under the European Commission IST FP6 program.

References

[1] I.S.O: Information technology – multimedia content description interface - part 6: Reference software. (2001)

[2] I.S.O: Information technology – multimedia content description interface - part 1: Systems. (2001)

[3] I.S.O: Information technology – multimedia content description interface - part 2: Description definition language. (2001)

[4] I.S.O: Information technology – multimedia content description interface - part 3: Visual. (2001)

[5] I.S.O: Information technology – multimedia content description interface - part 4: Audio. (2001)

[6] I.S.O: Information technology – multimedia content description interface - part 5: Multimedia description schemes. (2001)

[7] Eakins, J.P.: Retrieval of still images by content. (2001) 111–138

[8] Vakali, A., Hacid, M., Elmagarmid, A.: Mpeg-7 based description schemes for multi-level video content classification. IVC **22** (2004) 367–378

[9] Krinidis, M., Stamou, G., Teutsch, H., Spors, S., Nikolaidis, N., Rabenstein, R., Pitas, I.: An audio-visual database for evaluating person tracking algorithms. In: Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing, Philadelphia, USA (2005) 452–455

[10] Kyperountas, M., Cernekova, Z., Kotropoulos, C., Gavrielides, M., Pitas, I.: Scene change detection using audiovisual clues. In: Proc. of Norwegian Conference on Image Processing and Pattern Recognition (NOBIM 2004), Stavanger, Norway (2004)

[11] Sikudova, E., Gavrielides, M.A., Pitas, I.: Extracting semantic information from art images. In: Proc. of International Conference on Computer Vision and Graphics 2004 (ICCVG 2004), Warsaw, Poland (2004)