Overview of the MoMuSys Extension Activities: New Algorithms, Tools and Applications for Mobile Multimedia Systems

Hendrik Fuchs [(1)], Michael Wollborn [(2)]

(1) Robert Bosch GmbH, Corporate Research and Development, Multimedia Systems, FV/SLM
Robert-Bosch-Straße 200, D-31139 Hildesheim, Germany, E-Mail: Hendrik.Fuchs@fr.bosch.de

(2) Universität Hannover, Institut für Theoretische Nachrichtentechnik und Informationsverarbeitung, Appelstraße 9A, D-30167 Hannover, Germany, E-Mail: wollborn@tnt.uni-hannover.de

Abstract

This paper gives an outline of the targets, achievements and future plans of the recently started MoMuSys extension work called "New Tasks for MPEG-4". It consists of three different parts: (1) development of new algorithms for MPEG-4 Version 2; (2) development of a semi-automatic video object generation tool for content creation purposes; (3) development of an MPEG-4 based mobile virtual conferencing system.

Introduction

The main objective of the project Mobile Multimedia Systems (*MoMuSys*) [1] is to develop and validate the technical elements necessary to provide new audio-visual functionalities for mobile multimedia systems. Such functionalities, which include among others content manipulation, content scalability and content based access, are identified in the context of ISO MPEG-4 [2] which is going to become the standard for coding of audio-visual information in multimedia systems. They must be achieved with algorithms that provide very efficient compression on the one hand and robustness against transmission errors on the other.

MoMuSys is a European initiative for pushing technical developments and standardisation for multimedia. The Advanced Communications Technologies and Services (*ACTS*) programme provided the basis for forming a European collaboration platform to influence international standardisation and to push the technical development in Europe. Several standardisation bodies like ISO-MPEG, ITU-T SG 16 or ETSI SMG are targets for active contributions from *MoMuSys*. In particular, close collaboration with the MPEG-4 standardisation effort ensures the international relevance of the work. In MPEG-4 all the key players on the forefront of multimedia technology are united working towards a common goal, and *MoMuSys* as a European project bundles the European efforts into a powerful force within this world-wide standardisation platform.

Besides the development and standardisation efforts, the heart of the *MoMuSys* project is the evaluation of the technology developed by performing usage trials. This also allows to gather experience with new functionalities in advanced multimedia applications. For that purpose real time terminals using MPEG-4 technology have been developed and built. These terminals were part of several field trials with expert and non-expert users from many different European countries. The results of these field trials have been co-operatively evaluated together with the European TAPESTRIES project and were reported to MPEG.

In 1998, the project has been extended by the following three activities:

1. Continue the successful *MoMuSys* work in the development of MPEG-4 Version 2, i.e. contribution of technical proposals, participation in Core Experiments and maintenance of the reference software.
2. Develop a semi-automatic video object generation tool for content creation purposes, in particular to allow the user-friendly creation of generic MPEG-4 content with good visual quality.
3. Develop an MPEG-4 based mobile conferencing system, combining 2D videos of the participants, the possibility for presentations and sharing of applications in a virtual 3D conference room.

These three main areas, which shall take benefit of the so far existing project efforts to date, as well as of the experience and software implementations of the past, are described in the following.

New algorithms for MPEG-4 version 2

During the development of MPEG-4 Version 1 [2][5], *MoMuSys* has played a key role in the definition and the development of this new multimedia standard. Now, as the Version 1 is finished and the further work in MPEG concentrates nearly completely on Version 2 issues, the successful work of *MoMuSys* with respect to defining requirements and developing algorithms shall be continued also for this phase. Since the activities of the project in this area are manifold, the developed techniques can not be described in detail in this paper. Therefore, only a short summary of each activity is presented.

In ***Audio***, a perceptual audio coding scheme with low delay for MPEG-4 Version 2 is developed. Traditionally perceptual audio coding, which is known to deliver high sound quality even at low bit rates for a broad range of audio signals, and low-delay coding for communication purposes have been separate worlds. On one hand, perceptual audio codecs provide excellent subjective audio quality for a broad range of signals including speech at bit rates down to 16 kbps. The delay of such a coder/decoder chain usually exceeds 200 ms at very low data rates and in this way is not acceptable for interactive two-way communication, such as telephony or teleconferencing. On the other hand, speech coders, e.g. such as those recommended by ITU-T, meet the delay requirements for these applications, but do not perform very well for non-speech signals. The new developed codec is designed to combine the advantages of perceptual audio coding with the low delay required for two-way communication. It is closely derived from the ISO/MPEG-2/4 Advanced Audio Coding (AAC) scheme, and currently forms the baseline of development within Version 2 of the MPEG-4 audio standard for the work item “Low Delay General Audio Coding” meeting the requirement for a maximum algorithmic delay of 20 ms.

In ***Video***, several topics have been worked upon. For encoder optimisation, two rate control techniques for multi video objects (VO) have been developed, based on a generic object-based rate control architecture. For texture coding, a fast implementation of the "Shape Adaptive DCT" (SADCT) algorithm has been developed. The SADCT is an object-based texture coding technique which is part of a new Version 2 profile that is currently under consideration. Further, an extension of the Version 1 syntax referred to as “Multiple Auxiliary Components” (MAC) was proposed and is considered for adoption into Version 2. This technique enables the definition of auxiliary components, which can contain additional data related to the video object, like transparency, disparity, depth, additional texture, and which are encoded like the gray-level shape in Version 1. Up to 3 auxiliary components are possible, including the grayscale or alpha shape. Possible application areas are multiview compression and reconstruction, where a dense disparity map is needed, multi-channel image acquisition (e.g. infrared channel plus normal color channels), and studio production environment, where auxiliary components are often used to drive specific geometric, depth or blending effects. Finally, first proposals for fine grain scalability coding were made, a topic which is of high interest regarding features like server resource sharing, statistical multiplexing, channel capacity adaptation and error protection, CPU processing capability adaptation, congestion prevention to name only a few.

In ***SNHC*** (***S****ynthetic* ***N****atural* ***H****ybrid* ***C****oding*), the work concentrated on 3D model coding, where an algorithm for progressive subdivision of 3D meshes was developed and implemented. Subdivision methods [7] are used to represent a 3D object hierarchically by approximating its bounding surface with a set of polygonal control meshes that are inherently nested. Those nested meshes define a pyramid of LODs (Levels Of Detail) highly suitable for the multi-resolution coding of the given object [6]. With interpolative subdivision schemes, it is not necessary to explicitly code the positions of all vertices of the finest LOD, which can be extremely expensive. Instead, only the coordinates of the coarsest LOD vertices and a set of details, to be added to the new vertices that appear as the subdivision process goes, need to be described. Those details can then be coded efficiently using wavelet-based techniques similar to those that have already been used to code images or functions defined on different kinds of surfaces.

In ***Systems*** and ***DMIF*** *(**D**elivery **M**ultimedia **I**ntegration **F**ramework)* the connection between an H.324 terminal and an MPEG-4 terminal has been investigated in order to find out which actions have to be taken to enable such a connection. Further, a contribution on Media and Quality of Service (QoS) management has been made, and the work on a software implementation for QoS management has been started.

Finally, the maintenance and integration of the MoMuSys MPEG-4 Visual reference software has been continued. Here, after finalisation of the Version 1 FDIS software, the integration and maintenance of the Version 2 software was initiated together with MPEG, including the development of a respective time and task schedule for the integration of the Version 2 items.

VOGUE – a tool for *V*ideo *O*bject *G*eneration with *U*ser-friendly *E*nvironment

The experiences made in the past years of the project lead to the conclusion that the creation of generic MPEG-4 content with high quality is very complicated, partially impossible and in any case requires a high amount of hand work consuming much time and cost. However, it seemed very important for the success of MPEG-4 in the market place to provide the potential users with a tool for video object generation, that allows an economical production of high quality generic MPEG-4 content. Therefore, in this task of the extension work such a tool [3] is currently developed, being an integrated solution combining efficient analysis tools and user-friendly interaction via a graphical user interface. Up to date, three different segmentation algorithms for spatial segmentation, object tracking and temporal segmentation have been integrated and a respective graphical user interface has been developed.

The Graphical User Interface (GUI) encapsulates the three implemented segmentation algorithms, and provides the means for the user to interact with the overall segmentation process. It has been implemented on a MS Windows environment, using JAVA in order to allow a high flexibility in the development phase and an easy portability to other OS platforms. Also, a minimal effort is needed to implement an applet version for running in an intranet/internet browser.

The GUI supports the general interfacing requests of an application dealing with video, such as sequence selection, viewing with resize possibility, or saving, open, save or visualise partitions, and, last not least, launching the segmentation algorithms. According to the selected operation mode a context sensitive toolbar is displayed as an alternative to the respective menu options. In Figure 1, an exemplary snapshot of the GUI is shown while performing a spatial segmentation task. Therefore, on the left hand side the image to be segmented can be selected from a video sequence. On the right hand side, a partition is displayed with a different colour for each region. At the same time, this colour information is superimposed to the luminance of the original image for better evaluation of contour precision.

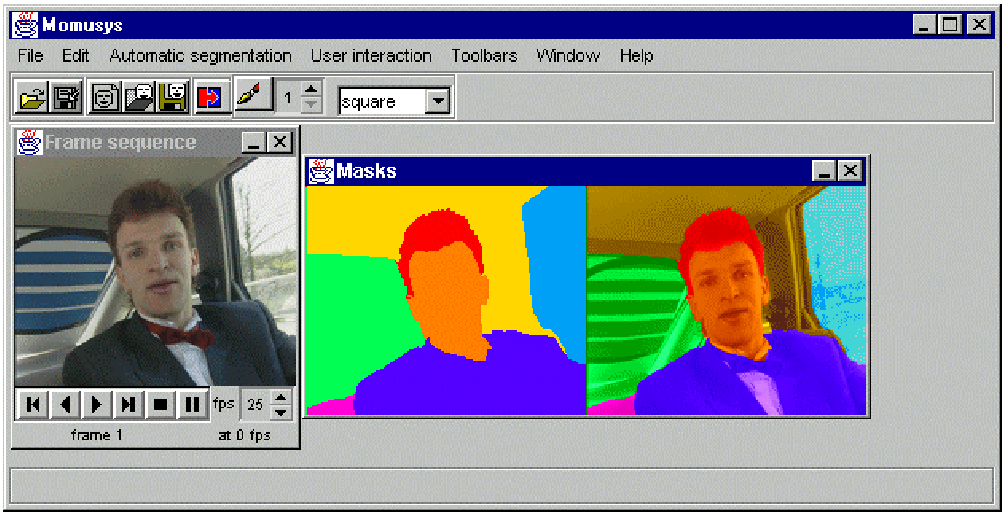

Figure 1 User interface of the MoMuSys' Video Object generation tool

Up to now, the user has the possibility to automatically segment highlight regions in different colours which may constitute an object according to the spatial segmentation algorithm. He can then combine these automatic suggestions of that tool to construct a meaningful object. Similar examples could be shown for usage of the temporal segmentation and of the object tracking functionality.

The spatial segmentation algorithm permits to define the segmentation mask of the object of interest. It is based on a multi-scale segmentation, consisting of a family of partitions that represents an image at different resolution levels. The coarsest level considers the image as a whole (as only one region) and finer partitions are always included in coarser ones. This means that a finer level is obtained by re-segmenting the regions of the previous level.

For the implemented morphological multi-scale segmentation, the finest resolution level contains all those regions that are determined by applying a watershed algorithm to the gradient image. Coarser partitions are then obtained by merging neighbouring regions. These fusions are ranked with a psycho-visual criterion, which comprises a combination of size and contrast and gives a good estimation of the visual significance of a region. The information of the whole multi-scale segmentation is stored in a tree which is obtained during the flooding process of the watershed algorithm. Thus, the complexity of the algorithm is equivalent to a single watershed, but instead of having a simple partition we obtain a whole family of partitions.

From the multi-scale segmentation presented in the previous paragraph, the user can obtain the desired partition. In order to make this process flexible and user-friendly, several tools are offered for interaction like selection of a number of regions, re-segmenting a selected region, coarsening a selected region (merging it with its most similar neighbours) and drawing a marker for each region. With these tools the user can easily define the object of interest and save the respective object mask to disk for further use.

The object tracking approach that is used relies on the concept of partition projection, that consists in accommodating the previous partition to the information in the current frame. Thus, once the object partition is available for an initial image, it can be automatically tracked through the following images of the sequence. The tracking algorithm requires spatial homogeneity of the regions to be projected. Since this requirement may not be fulfilled by the object partition, the latter one is re-segmented using the spatial segmentation algorithm described previously. The partition obtained is called *texture partition* of the previous frame, and it guarantees the spatial homogeneity of each region.

The texture partition of the previous image is projected into the current frame to obtain the texture partition at the current image. The projection of the texture partition is carried out in two steps: motion compensation and a fitting process that accommodates the motion compensated contours to the colour boundaries. New regions in the image can be detected during the fitting process or by an additional step of new region extraction. Detected new regions are analysed and an automatic decision is made on whether or not they belong to the object being tracked.

The result of the automatic tracking is displayed to the user, who has the possibility to stop the execution and ask for refinements of the object mask by adding or removing regions using the integrated user interaction tools of the GUI. By this, a sequence of object masks is generated which can be stored to disk for further use.

The temporal segmentation is useful when the user is interested in moving objects, i.e. it provides a functionality similar to the object tracking. However, in this case no initial object mask is required but the algorithm automatically detects moving objects in the image sequence which are then segmented. Thus, the temporal segmentation can be operated in two modes: in the completely automatic mode each moving object in the scene is segmented once the user has started the process, while in the semi-automatic mode the user can interact with the segmentation at any time he likes.

The implemented temporal segmentation algorithm is based on a change detection followed by a motion analysis. It consists of the following steps: (1) camera motion estimation and compensation, assuming that the background of the scene is a rigid plane, (2) scene cut detection based on the evaluation of the mean square difference between the camera motion compensated previous frame und the current frame, (3) estimation of a change detection mask by thresholding the difference image followed by a relaxation, (4) elimination of uncovered background by evaluating an estimated displacement vector field, and finally, (5) adaptation of the mask to luminance edges of the current frame. The algorithm has been adopted as an informative annex of the ISO/MPEG-4 standard [5] and is part of the COST 211 Analysis Model [4].

The temporal segmentation results are displayed to the user, who has the possibility to stop the execution and to refine the process parameters or the segmentation results. Further, it is possible to start the whole process with an initial mask from the spatial segmentation, to specify the area where objects should be detected or to calibrate the segmentation with an a-priori known background image.

MPEG-4 based mobile conferencing system

Besides the availability of MPEG-4 content the definition and development of meaningful MPEG-4 application scenarios is crucial for a wide acceptance and dissemination of this new technology. Therefore, based on the experiences and developments from the former usage field trials, a 'Next generation conference system' completely based on MPEG-4 technology will be developed and demonstrated. This system intends to bring remotely located participants together in a virtual conference room, which not only shows 2D video and audio of the participating people, but also allows the presentation of information (slides, video, 3D, etc.)

and supports co-operative work e.g. by application sharing. A conceptual draft of how such a virtual conference system could look like is depicted in Figure 2.

Figure 2 Conceptual draft of a virtual conference

The room is described by a BIFS scene [2]. The scene description contains references to Audio Visual Objects (AVOs) representing the voice and the picture of the participants.

The scene description itself and also the AVOs are transmitted by elementary streams that are essential components of the architecture of the MPEG-4 system part [2]. The overall system design of the conference system is based on the separate transmission of the audio/video elementary streams on the one hand and all the other information on the other hand. This allows the transmission of the audio/video data by an unreliable multicast connection, provided by UDP/IP while the sensitive scene description information can be transmitted by a reliable connection, provided by TCP/IP.

The overall system design is based on the separate transmission of the audio/video data via a multicast protocol in a client-server structure. Therefore, Inter- and Intranet technology will be used as the underlying transport technology, while normal PCs or workstations are used for the server and for the client terminals. The assumption of different data streams leads to the overall system architecture depicted in Figure 3

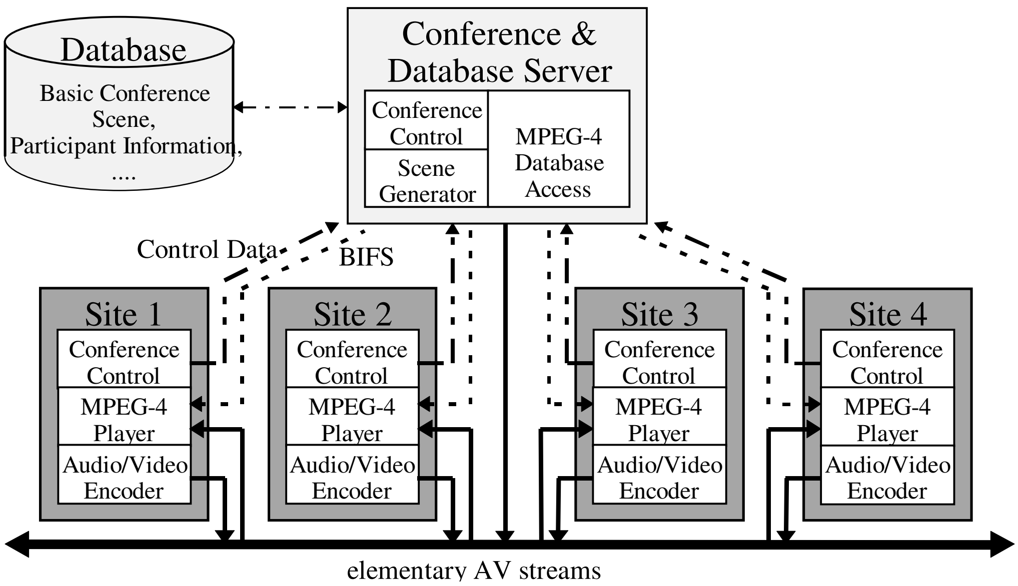

Figure 3 Client-Server Architecture

The conference server manages the virtual meeting room, the participants and their actions. It composes the scene description of the conference room tailored for a dedicated client and sends it to this client. Changes of the virtual scene are transmitted as BIFS updates.

A set of conference room description templates is stored in a multimedia database, such that a participant can easily select one. The data base also provides MPEG-4 pre-encoded material used in the conference.

The task of the conference client is two folded:

- It has to encode the audio and video information of the participant in an efficient way taking into account the limited bandwidth when DECT or ISDN networks are used. For the transmission of real-time data streams two modes can be used. In the *multicast mode* the streams will be sent directly from one client to all other clients. In the *reflector mode*, the streams are sent to the server which redirects the streams to the clients.
- The client also receives the MPEG-4 data and control streams generated by the server that will be used to compose the 3-D scene. This also includes the mixing of the audio streams. The composed scene is presented to the participant.

Up to date, a first implementation of the conference client has been developed and demonstrated, whereas the complete system is still under construction. It is expected to demonstrate a full virtual conference by the end of the year.

Conclusion

In this paper, an overview of the targets, achievements and future plans of the recently started MoMuSys extension has been given. The algorithms developed and proposed for MPEG-4 Version 2 build a significant input to the standard as was already the case for Version 1. Some of the algorithms are already accepted as baseline for the further standardisation work, and some are under consideration for inclusion into a new Version 2 profile. The video object generation tool provides a flexible and user-friendly environment for creating high quality MPEG-4 content, and it will be one important step forward in the field of MPEG-4 content creation. Nevertheless, much more work is needed in this area. The virtual conference system will be an attractive platform for demonstrating the MoMuSys achievements to the public and might be the basis for related applications. Together with the results already achieved in the first years of the project, the extension work will complete this very successful ACTS project.

References

[1] ACTS MOMUSYS Project Information. (http://www.tnt.uni-hannover.de/project/eu/momusys)

[2] ISO/IEC JTC1/SC29/WG11 Doc. N2725, “MPEG-4 Overview”, March 1999. (http://www.cselt.it/mpeg/standards/mpeg-4/mpeg-4.htm)

[3] B. Marcotegui, P. Correia, F. Marques, R. Mech, R. Rosa, M. Wollborn, F. Zanoguera, "A Video Object Generation Tool Allowing User-friendly Interaction", submitted for International Conference on Image Processing 1999 (ICIP '99), Kobe, Japan, October 1999

[4] A. Alatan, L. Onural, M. Wollborn, R. Mech, E. Tuncel, T. Sikora, "Image Sequence Analysis for Emerging Interactive Multimedia Services - The European COST 211 Framework", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 8, No. 7, November 1998, pp. 802-813.

[5] ISO/IEC JTC1/SC29/WG11 Doc. N2502, “Information Technology – Generic Coding of Audio-visual Objects, Part 2: Visual, ISO/IEC 14496-2”, Final Draft of International Standard, October 1998.

[6] M. Eckert et al., “Multiresolution Analysis of Arbitrary Meshes”, Computer Graphics Proceedings 1995 (ACM SIGGRAPH'95).

[7] J. Schweitzer: “Analysis and Application of Subdivision Surfaces”, Ph.D. thesis, University of Washington, 1996.