Personalized adaptation and presentation of annotated videos for mobile applications

Sarah De Bruyne, Jan De Cock,
Rik Van de Walle
Department of Electronics and Information
Systems - Multimedia Lab
Ghent University - IBBT
Gaston Crommenlaan 8 bus 201, B-9050
Ledeberg-Ghent, Belgium
{sarah.debruyne, jan.decock,
rik.vandewalle}@ugent.be

Peter Hosten, Mark Asbach,
Mathias Wien
Institute of Communication Engineering
RWTH Aachen University
52056 Aachen, Germany
{hosten, asbach,
wien}@ient.rwth-aachen.de

Cyril Concolato
TELECOM ParisTech
46, rue Barrault
75013 Paris, France
cyril.concolato@telecom-paristech.fr

ABSTRACT

Personalized multimedia content which suites user preferences and the usage environment, and as a result improves the user experience, gains more importance. In this paper, we describe an architecture for personalized video adaptation and presentation for mobile applications which is guided by automatically generated annotations. By including this annotation information, more intelligent adaptation techniques can be realized which only reduce the quality of unimportant regions in case a bit rate reduction is necessary. Furthermore, presentation layers are added to enable advanced multimedia viewers to optimally present the interesting parts of a video in case the user wants to zoom in. This architecture is the result of collaborative research done in the EU FP6 IST INTERMEDIA project.

Categories and Subject Descriptors

I.4 [**Image Processing And Computer Vision**]: Miscellaneous

General Terms

Algorithms

Keywords

annotation, adaptation, rich media presentation, personalized multimedia

1. INTRODUCTION

Many situations exist where personalized multimedia is highly desirable in order to improve the user experience. Therefore, on the one hand, properties of multimedia need to match the current user situation such as the available network bandwidth, display device capabilities, etc. On the other hand, personal user preferences need to be taken into account to enable user-centric convergence of multimedia.

In this paper, we illustrate how multimedia annotations can guide adaptation and presentation techniques to create personalized multimedia for applications with limited bandwidth and display constrains, such as mobile devices. By combining these different research domains, higher user satisfaction can be achieved. The research described in this paper is the result of research performed in the scope of the EU FP6 IST project Interactive Media with Personal Networked Devices (INTERMEDIA) [2]. In particular, one of the objectives of this project is to generate a common vision on user-centric multimedia services in shared content environments to provide users with content personalized to their (semantic) user preferences and usage environments [1].

In the context of Universal Multimedia Access (UMA), efficient techniques are needed for the adaptation of video content. An important example is the reduction of the bitrate in order to satisfy the bandwidth constraints imposed by the network or the decoding capability of the terminal devices. Typically, these adaptation techniques will reduce the quality of the entire frame. However, by incorporating region of interest (ROI) information, more intelligent adaptations can be realized by assigning different priority levels to particular areas.

Unfortunately, content collections often lack any metadata which can be used to steer context-aware adaptations. Therefore, automatic content analysis and annotation techniques are of paramount importance. In INTERMEDIA, we focus on temporal segmentation and ROI detection as this information can guide the personalization of multimedia.

When consuming multimedia on devices with small displays, detailed information in video sequences can no longer be seen. Therefore, dynamic presentation layers are added which take into account the user preferences using interaction, the characteristics of the device, and the ROI information generated during the annotation process. As such, advanced multimedia viewers can be obtained which optimally present the ROIs.

The different aspects of personalized multimedia adaptation and presentation guided by annotations are further described in the remainder of this paper. In the next section, the automatic metadata generation is discussed. Section 3 and 4 elaborate on adaptation techniques and rich media presentations resp. which are guided by ROI information. Conclusions are drawn in Section 5.

2. AUTOMATIC CONTENT ANNOTATION

During the last years, the field of image understanding has made significant progress. Different tasks such as shot boundary detection, face detection, optical character recognition and even matching existing scripts to dialogs can now be handled by autonomous systems. Typically these techniques are used and evaluated in the context of information retrieval, i.e. searching digital libraries of stored media. An overview about this can be found in TrecVID [5].

In the context of UMA, a new application field for automatic image understanding has arisen. As described above, sensible and intelligent adaptation of media that originally has been authored for TV screens or cinema needs annotation.

The INTERMEDIA content annotation tool chain has been designed with personalized media adaptation and presentation in mind. It therefore extracts only those media characteristics that can be evaluated based on the current viewing situation to form an adaptation decision. At first, temporal segmentation is applied to find individual shots with mostly uniform media characteristics. This information is necessary for the following processing steps, but it can also be used for easily skimming content, skipping blocks or automatically creating a simple table of contents. Every shot is then analyzed for spatial partitioning. Without any further knowledge on the kind of media content, general criteria are necessary to differentiate between important and less important parts. For INTERMEDIA, we chose the concept of foreground versus background to identify ROIs. Based on such annotations, the adaptation process can be steered to assign higher priority to (hopefully) more important foreground objects than to the surrounding background parts. In parallel, specific objects are detected and tracked. Faces are important parts of typical visual media. Other kinds of objects could be interesting for certain domains like footballs or cars for sports, or certain animals for documentaries. If special objects are present, media presentations can be personalized even more. However, since there is no perfect and complete set of object categories in the general case, the general segmentation information is always present as a fallback. This general structure is depicted in Figure 1.

2.1 Temporal segmentation

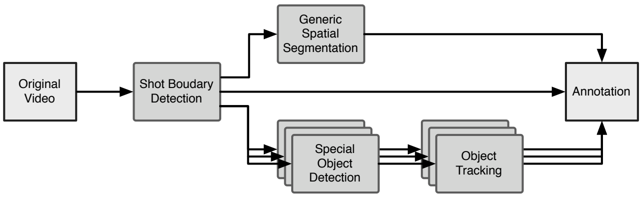

Figure 1: Annotation pipeline with generic and specialized object detection concepts.

The first step of the automatic annotation process is the detection of different scenes. Via shot boundary detection the temporal information in form of single shots is extracted. Shot boundaries (or scene cuts) are detected based on color histograms in HSV color space. By averaging over a couple of frames, small jumps in histogram entries are smoothed and only non-transient changes result in a jump in histogram differences that indicate a shot boundary. Also, a long-term comparison with the start of the current shot allows detecting gradual changes resulting from transition effects like wipes or dissolves.

2.2 Generic spatial segmentation

For every temporal segmentation, the spatial information in the different scenes is extracted. In INTERMEDIA, the generic object detection is based on motion compensated background subtraction. Background subtraction, being a standard approach in surveillance scenarios, can be applied to general video content when camera motion can be compensated for. The authors have presented an approach to estimate global motion and generate artificial backgrounds in [6]. Effectively background subtraction with such artificial backgrounds results in pixel-accurate masks or contours of all spatial regions that cannot be described by a background model.

2.3 Specialized object segmentation

In addition to generic spatial segmentation a specialized object segmentation algorithm extracts semantic information. An object detector based on a cascade of boosted classifiers [8] is trained for each kind of object that is going to be detected, e.g. frontal faces, as depicted in Figure 2. During the training process multiple weak classifiers based Haar-like features are generated and arranged in series resulting in a strong classifier. Once the cascade is trained, each subwindow in each frame is analyzed. This algorithm is very efficient since the most significant features are tested by the leading classifier in the cascade, so that a large number of negative examples is rejected in the early processing stages.

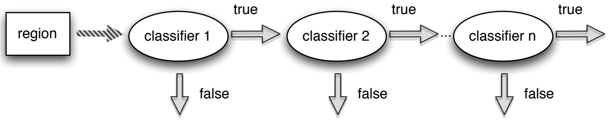

Figure 2: Cascade of weak classifiers.

However, only these objects, which have been trained, can be detected. A frontal face detector, for instance, will not be able to detect a rotated face in profile view. For that reason a Kalman filtering approach modeling the location, velocity and size of the object is used for tracking. That way the object can be detected and tracked over several frames, providing the information about size and location of specific objects (ROIs). Figure 3 illustrates the result of the face detection process.

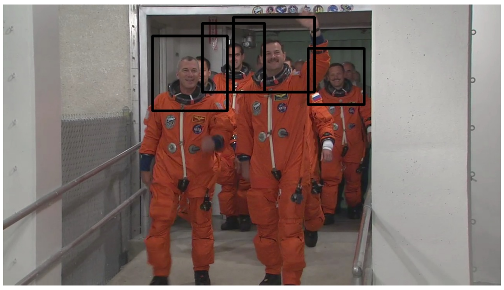

Figure 3: Bounding boxes indicating the result of the face detection process on the Crew sequence.

3. REGION-OF-INTEREST-BASED VIDEO ADAPTION

As described above, multimedia adaptation is required to for example match the bit rate of the video signal to the available network bandwidth. Although scalability provisions at the encoder side might allow easy adaptation of video streams, such as with the scalable extension of the H.264/AVC video coding standard (SVC), practical video encoders are likely to output single-layer video streams. Hence, adaptation of coded video content remains a challenging task. This is only reinforced by the high complexity of state of the art video coding algorithms.

As a straightforward solution of video adaptation, a coupled decoder and encoder might be used, where the output of the decoder is fed to the re-encoding process. Given the high computational complexity of both modules, and in particular the encoder, such a solution is not viable in typical use cases. In order to reduce the computational burden of the adaptation, it is pivotal that information from the incoming bitstream is reused during adaptation.

Transcoding solutions provide fast adaptation by reusing data of the input stream such as motion vectors and prediction modes. As a result, the search space is reduced during transcoding when compared to re-encoding, hereby allowing a significant increase in processing speed. The presented video transcoding module is able to reduce the bit rate of the incoming coded video signal to comply with the constraints imposed by the environment, such as the available network bandwidth. Typically, the bit rate of the video stream is determined by the coarseness of the quantization during encoding. When a reduction in bit rate is desired, this can be accomplished by requantizing the prediction error coefficients with a coarser quantization step size, which is indexed by the quantization parameter (QP, which can take values from 0 to 51). Typically, a small increase in QP will suffice for most desired adaptations.

Traditional transcoding techniques will reduce the quality of the entire frame [7]. However, by incorporating ROI information which is derived from the annotations, as described in Section 2, more intelligent adaptations can be realized. In this tool, the quality of the picture after transcoding is unaffected in the region(s) of interest, while the background quality will be reduced, resulting in a lowered bit rate for the overall video sequence. In this way, the data in the bitstream will be apportioned to the relevant regions in the video sequence, while overhead and quality of the less important background regions will be reduced. This is demonstrated in Figure 4, where the high quality is only maintained for the ROIs detected in Figure 3, while other regions are heavily quantized, leading to a significant bit rate reduction.

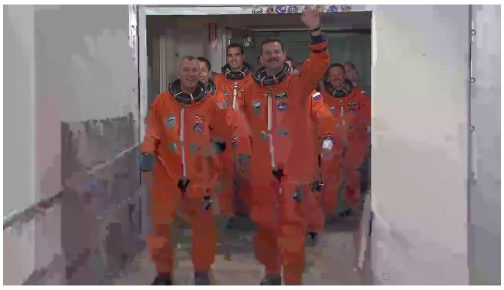

Figure 4: Crew sequence after ROI-based adaptation.

A high-level overview of the used transcoder architecture is given in Figure 5. The first component of the transcoder is a decoder loop, which reconstructs the pictures to the pixel domain, and stores these pictures in the buffer. For these decoded pictures, object detection can be applied, resulting in the ROIs. The macroblock indices associated with the ROIs are passed on to the encoder loop. For these macroblocks, no change in QP is incurred. Nonetheless, recalculation of the prediction error is necessary, since the prediction values may have changed. For the background macroblocks, requantization is executed with an increased QP. A second motion estimation step is avoided by passing the motion parameters from the incoming bitstream to the encoder loop. In this way, motion vectors, reference picture indices, macroblock partitioning, and prediction modes are reused and passed on to the output bitstream without additional computational complexity. This ‘shortcut’ results in significant computational complexity savings when compared to a coupled decoder-encoder.

Two strategies can be followed to resolve the issue of which QP to use for the background macroblocks during transcoding. On the one hand, a fixed increase in QP can be used, so that the output bit rate is a priori unknown. On the other hand, a rate control algorithm can steer the QP selection so that the appropriate reduction in bit rate is achieved after transcoding.

If desired, motion information can be changed to better reflect the updated information in the bitstream. Such a motion refinement step can help improve coding efficiency of the output bitstream, hereby helping to further improve video quality given the available bandwidth. In particular, in the case that ROI macroblocks are predicted based on non-ROI macroblocks, or vice versa, it is likely that prediction will benefit from an update in motion vectors or pre-

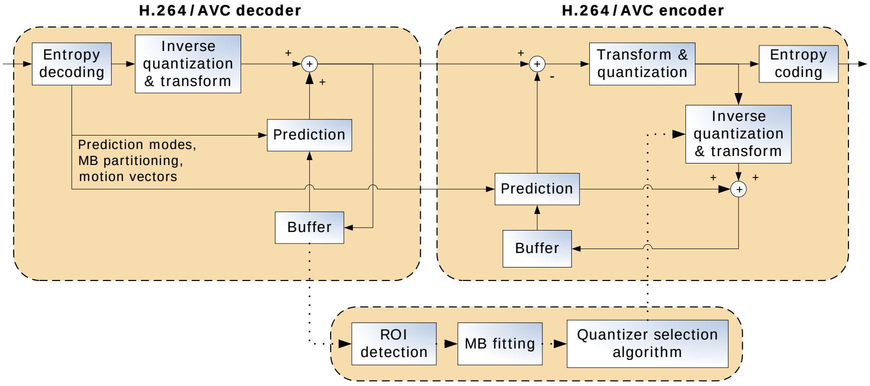

Figure 5: Overview of ROI-based adaptation (transcoder) tool.

diction modes. While this step can increase computational complexity, intelligent algorithms can be designed that benefit from the information in the input bitstream. This means that exhaustive motion estimation can still be avoided.

4. RICH MEDIA PRESENTATION

Section 2 presented how interesting objects can be located in a video, whereas Section 3 showed how this video can be adapted intelligently based on this information. This section describes how to interactively present the adapted video to a user when dealing with mobile devices with constrained displays based on this ROI information. In order to achieve a user-centric presentation, the following requirements should be met. The presentation system should:

1. be backward compatibility with simple audiovideo players, in order to display on every device;
2. be able to present multiple ROIs at the same time;
3. be able to present ROIs of rectangular and arbitrary shape;
4. be able to present dynamic ROIs, synchronized with the video;
5. allow a user to interact with ROIs;
6. preserve the pixel aspect ratio when viewing ROIs;
7. and enable adapted presentation according to the viewing device characteristics: screen size (in inches), screen resolution (in pixels), and screen aspect ratio.

These requirements lead us to the use of a scene description to describe presentation instructions. These presentation instructions indicate, to advanced multimedia players (also called rich-media players), where the ROIs are, how to display them on top of the video, how they change over time and how the user may interact and view them. When packaged properly, these instructions may be ignored by traditional audio-video players such as VLC, thereby fulfilling requirement 1. There are many candidate scene description technologies to fulfill our requirements. We can cite the Scalable Vector Graphics (SVG) language and its extension, Lightweight Application Scene Representation (LASeR); Flash, the de facto web standard for animated graphics and video presentation (e.g. as on YouTube); the Binary Format for Scenes (BIFS), or the Synchronized Multimedia Integration Language (SMIL).

In our scenario, the description of the presentation instructions is tightly coupled with the video. The video content is described as a stream. We therefore naturally decide to choose a stream-based description language. Additionally, since we require a packaging format capable of storing separately the scene description and the video (to fulfill requirement 1), we are therefore left with either MPEG-4 LASeR or MPEG-4 BIFS. Both languages are stream-based, can be created using XML or simply plain text, then compressed or not, and finally streamed over IP or stored along the video in an mp4 file, both allowing individual presentation of the video. In terms of expressiveness of the presentation, even though the detected ROIs are currently rectangular, we require a language capable of representing arbitrary shaped ROIs. Although both MPEG languages could allow it, we choose to create our presentation instructions using the MPEG-4 BIFS language [3] since this language supports texture mapping.

We present now the structure of these instructions, which consist of an initial scene (presented at T=0) and scene updates. Based on the ROI information extracted during the analysis, we first compute the maximum number n of ROIs per frame for the whole video duration.

With this information, we build an initial scene which consists of a video (*Shape* and *MovieTexture* nodes) on top of which n clickable rectangles (*Shape* and *Rectangle* nodes), initially invisible, are drawn. We also define *n+1* viewports (*Viewport* nodes) for each of the ROIs and for the non-zoomed version, used as the initial viewport. Upon a click (use of a *TouchSensor* node) on one of the ROI rectangles, the associated viewport is bound (using a *Route*, a *Conditional* node and the *set_bind* event of the *Viewport* node), and the video is therefore zoomed to show the appropriate ROI, as illustrated in Figure 6. The viewport also allows indicating if the pixel aspect ratio is to be preserved or not and if it is how to fill the rest of the viewport. An example is provided below, using the BIFS textual syntax. Note that the declaration of the prototype *RegionOfInterestProto* is omitted for brevity.

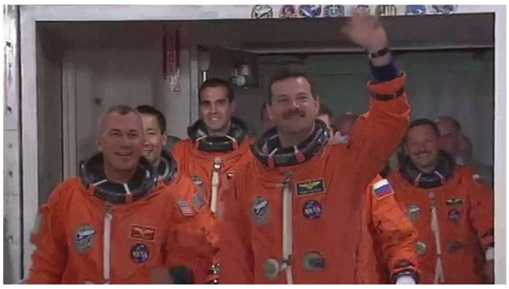

Figure 6: Illustration of user-driven presentation when zooming into a ROI.

```
OrderedGroup { children [
Shape { geometry Rectangle { size 1280 720 }
        appearance Appearance { texture
        MovieTexture { url "video.mp4" } } }
DEF VP_MAIN Viewport { fit 1 size 1280 720 }
DEF ROI_MAIN TouchSensor {}
DEF C_MAIN Conditional {
  buffer { REPLACE VP_MAIN.set_bind BY TRUE } }
DEF VP1 Viewport { fit 1 }
DEF C1 Conditional {
  buffer { REPLACE VP1.set_bind BY TRUE } }
...
Transform2D { translation -360 216 children [
   DEF ROI1 RegionOfInterestProto {}
   ...
] } ] }
ROUTE ROI_MAIN.isActive TO C_MAIN.activate
ROUTE ROI1.activate TO C1.activate ...
```

Finally, we build a new scene update for each frame where the ROI changes. Each update contains commands to hide/ show and set the position and size of the clickable rectangles, and to set the position and size of the corresponding viewports. Each update can contain a command to set the title of each ROI in order to include semantic information into the presentation. An example of update is provided below.

```
AT 40.04 { # time in milliseconds
REPLACE ROI1.hidden BY 0
REPLACE ROI1.keyword BY "Face1"
REPLACE ROI1.position BY 208 -160
REPLACE ROI1.size BY 48 48
REPLACE VP1.position BY -128 32
REPLACE VP1.size BY 48 48
REPLACE ROI2.hidden BY 0
REPLACE ROI2.keyword BY "Face2"
REPLACE ROI2.position BY 288 -160 ... }
```

The result of this generation process is then compressed into the BIFS binary format, packaged into an mp4 file together with the video and played with the GPAC Rich Media Player [4] on desktops or mobile devices.

5. CONCLUSIONS

In this paper, we describe an architecture for personalized adaptation and presentation of videos based on automatically extracted region-of-interest information. The goal of this approach is to deliver content to users with mobile devices with limited display and network capabilities in a user-centric way in order to improve the user experience. First, by using ROI information, more intelligent adaptations can be achieved by only degrading the quality of unimportant regions. Furthermore, rich media presentations are included to enable interactivity with these ROIs.

Future work includes object recognition to extend the annotion information, improved algorithms for adaptation, and annimations during the transitions between different ROIs in the presentation phase.

6. ACKNOWLEDGMENTS

The research activities that have been described in this paper were co-funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research-Flanders (FWOFlanders), the Belgian Federal Science Policy Office (BFSPO), and the European Union (within the framework of the NoE INTERMEDIA, IST-038419).

7. REFERENCES

[1] S. De Zutter, M. Asbach, S. De Bruyne, M. Unger, M. Wien, and R. Van de Walle. System architecture for semantic annotation and adaptation in content sharing environments. *The Visual Computer, International Journal of Computer Graphics*, 24(7-9):735–743, 7 2008.

[2] INTERMEDIA. Interactive Media with Personal Networked Devices (ist-1-38419), Feb. 2008. European Sixth Framework Programme (FP6) IST NoE co-funded project.

[3] ISO/IEC 14496-11:2005. *Information technology - Coding of audio-visual objects - Part 11: Scene description and application engine*. ISO, Geneva, Switzerland, December 2005.

[4] J. Le Feuvre, C. Concolato, and J.-C. Moissinac. GPAC: open source multimedia framework. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, pages 1009–1012, New York, NY, USA, 2007. ACM.

[5] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.

[6] M. Unger, M. Asbach, and P. Hosten. Enhanced background subtraction using global motion compensation and mosaicing. *15th IEEE International Conference on Image Processing*, pages 2708–2711, 2008.

[7] A. Vetro, C. Christopoulos, and H. Sun. Video transcoding architectures and techniques: An overview. *IEEE Signal Processing Mag.*, 20(2):18–29, 2003.

[8] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1:511–518, 2001.