IMPROVING SCALABLE VIDEO ADAPTATION IN A KNOWLEDGE-BASED FRAMEWORK

F. López[1], G. Nur[2], S. Dogan[2], H.K. Arachchi[2], M. Mrak[2], J.M. Martínez[1], N. García[3], A. Kondoz[2]

[1]VPULab, EPS – Universidad Autónoma de Madrid, Spain {f.lopez, josem.martinez}@uam.es
[2]I-Lab, FEPS – University of Surrey, Guildford, UK
{g.nur, s.dogan, h.kodikaraarachchi, m.mrak, a.kondoz}@surrey.ac.uk
[3]GTI, ETSIT – Universidad Politécnica de Madrid, Spain narciso@gti.ssr.upm.es

ABSTRACT

In a knowledge-based content adaptation framework, video adaptation can be performed in a series of steps, named conversions. The high-level decision phase in such a framework occasionally encounters several feasible parameter values of a specific conversion. This paper proposes to transfer further decisions to a low-level phase that decides which parameters maximise the quality of the adaptation. Particularly when more than one solution are available, an innovative quality measure is used for selecting the best values for the parameters among the set of values that fulfil the adaptation constraints in the case of scalable video.

1. INTRODUCTION

Future social networks will rely on advanced multimedia content sharing services. In order to fully exploit visual content access capabilities, new technologies have to be developed, which will maximise the multimedia quality provided to end users. Those technologies can be based on emerging knowledge-based content adaptation frameworks [1][2][3]. In such systems, rich multimedia content can be adapted in a sequence of conversions that optimise the content for consumption by end users. Knowledge-based frameworks provide methods to automatically identify a sequence of conversions and the associated set of parameters for adapting media content to the constraints of user terminals and access networks as well as user preferences. Sometimes, knowledge-based methods determine that different combinations of parameters produce different adaptation results, all of which fulfil the same set of constraints. This can be observed, among others, while adapting scalable video streams. In such cases, it is possible to introduce additional decision mechanisms in order to improve the resulting quality and efficiency of the adaptation. This work investigates a new method for taking into account network bandwidth constraints, in terms of bit rate restrictions and video quality evaluations [4]. The paper shows how this information can be exploited in making additional decisions that improve the quality of the adaptation.

This paper is organised as follows. Section 2 presents the related state of the art. Section 3 explains the high-level decision phase and demonstrates why multiple solutions for a specific adaptation problem may exist. Subsequently, it discusses the selection criteria of the parameter values in the low-level decision phase. The low-level decision phase uses these scores to select the best possible values for the parameters. Section 4 presents the results of the experiments conducted to demonstrate the effectiveness of the method used in this paper. Section 5 concludes the paper.

2. STATE OF THE ART

Subjective evaluations conducted with real observers provide more accurate results for assessing video quality than objective quality assessment techniques [5]. However, subjective evaluations are time consuming and costly. *Video Quality Metric* (VQM) [5] is an objective quality assessment method, which has been proved to provide a good solution for such evaluations. VQM represents the subjective quality judgments of observers better compared to the other objective quality models (e.g., PSNR, *Structural Similarity Index* (SSIM), etc) [6], and therefore we use it as the principal quality assessment technique in this paper.

VQM comprises two phases for video quality assessments. The *Calibration phase* involves spatial alignment, valid region estimation, gain-offset calculation, and temporal alignment. Jerkiness, colour distortion, blurriness, global noise, block distortion of the calibrated video content are measured compared to the original one within the VQM's *Calculation phase* [5][6]. A VQM grade ranging from 0 (no impairment) to 1 (maximum impairment) is used in the paper, which will demonstrate how considering VQM improves the quality of adaptation subject to the constraints of the knowledge-based adaptation framework.

3. DECISION MAKING IN CAIN-21

CAIN-21 [3] is a knowledge-based adaptation engine that complies with the representation schema of the MPEG-21 standard [7]. The main objective of CAIN-21 is to provide a framework in which different multimedia adaptation tools can be integrated and tested. CAIN-21 provides an extensibility mechanism by which adaptation tools, named *Component Adaptation Tools* (CATs), can be added in a pluggable manner. Furthermore, this adaptation engine includes an automatic knowledge-based decision making process that enables the quick generation of *sequences of conversions* for multimedia adaptation. The source code and an online demo of its functionalities are publicly available at *http://cain21.sourceforge.net.*

3.1. The adaptation process

Fig. 1 illustrates the adaptation process in CAIN-21. The *adapt*() operation receives two MPEG-21 Part-2 [7] *Digital Items* (DIs). The *Content DI*[1] represents both a media resource and its metadata. In the context of this paper, the media resource is scalable video and the metadata is an MPEG-21 Part-7 [7] *AdaptationQoS* description. In this descriptor, the available layers of a scalable video are described, as well as the pre-computed bitstream components and utility of each adaptation. The *Context DI* stores the *usage environment* (terminal capabilities, network characteristics and user's preferences) and the *CAT Capabilities*. In CAIN-21, each CAT implements one or more conversions. The *CAT Capabilities* is a description of the conversions that a specific CAT implements, which is required by the CAIN-21 decision engine for making a decision. Each of the *CAT Capabilities* is divided into one or more *Conversion Capabilities* description elements, each one representing the inputs and outputs of a specific conversion that the CAT can perform.

During the invocation of the *decide*() function, CAIN-21 determines the *feasible set of sequences of conversions* (i.e., CAT executions along with necessary parameters) that is capable of adapting the video content (represented as the *Content DI*) to a particular usage environment (represented with the *Context DI*). In the simplest case, one of the shorter sequences of conversions from the above feasible set of sequences of conversions is chosen and transferred to the *execute*() function. This function executes the chosen sequence of conversions using the parameters that the *decide*() function has selected. At the end of the *execute*() function, the adapted *Content DI* is generated.

[1]MPEG-21 capitalises XML description elements. This paper adopts this rule.

3.2. The decision process

In [8], we reported the theoretical basis of the automatic decision mechanism that CAIN-21 implements. This section elaborates on an example to illustrate the operation of the decision mechanism. In the simplest case (i.e., where only one step is required), the decision mechanism performs two matching operations. The first matching is between the *Content DI* and the input properties of a *Conversion Capabilities* description element of each CAT. Clearly, not every *Conversion Capabilities* description element has to match the *Content DI*. If they match, this means that the corresponding conversion is able to process the *Content DI*. The second match takes place between the output properties of the *Conversion Capabilities* and the *Context DI*. If there is a match, then the corresponding conversion is able to produce content that can be consumed in the usage environment.

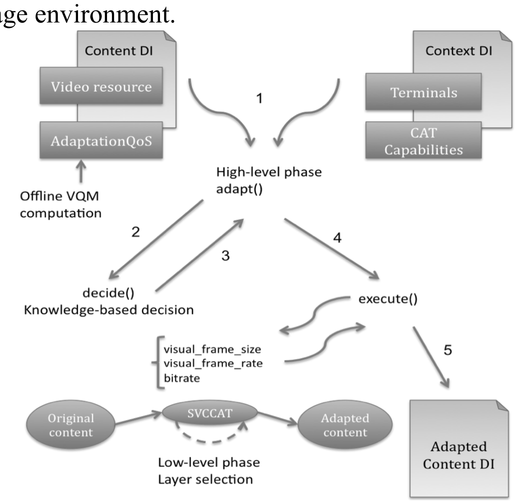

Fig. 1: Adaptation process in CAIN-21

3.3. Partial decisions

In this subsection, the partial decisions concept is demonstrated using an example scenario, which illustrates the delivery of *bus.xml Content DI* to the terminal labelled as *svc_no_audio_176x144_15fps*. To enable scalable video adaptation, a new CAT, named SVCCAT, has been implemented. Its *CAT Capabilities* are stored in the *svc_cat.xml*. Further details on this example can be found in the CAIN-21 demo.

For achieving this adaptation, the *decide*() function generates the following sequence of conversions: *initial* → *svc_without_audio_transcoder* → *goal*. In this sequence, the *initial* conversion state corresponds to the *Content DI* to be adapted, the *svc_without _audio_transcoder* conversion state corresponds to one of the conversions implemented in the SVCCAT and *goal* corresponds to the properties of the selected terminal. The *decide*() function produces the following source and target parameters for the *svc_without_audio_transcoder* conversion state:

Source parameters

visual_frame_size = {176x144, 352x288, 704x576}
visual_frame_rate = {3.75, 7.5, 15, 30}
visual_bitrate = [110.2 .. 4501]

Target parameters

visual_frame_size = {176x144}
visual_frame_rate = {3.75, 7.5, 15}
visual_bitrate = [110.2 .. 400]

The *visual_frame_size* source parameter indicates that the scalable input video can be adapted at three different frame resolutions (QCIF: 176x144, CIF: 352x288, and 4CIF: 704x576), and the target parameter indicates that the *decide*() function has selected QCIF as the frame resolution. This is due to the display size of the user terminal considered in this example. However, the *decide*() function has selected multiple frame rates and bit rates that correspond to the multiple values supported by the terminal. At this point, the *decide*() function in Fig. 1 terminates.

3.4. Layer selection

In the example described above, *decide*() has not provided a complete solution to the adaptation problem. Specifically, the *visual_frame_rate* and *visual_bitrate* parameters contain multiple values, and thus the layer has not been determined. In this case, the parameter selection is transferred from the *decide*() function to the *execute*() function. Moving decisions from the *decide*() function to the *execute*() function gives the CAT implementers the opportunity to make further decisions (such as quality-based). In our example, the SVCCAT (used inside the *execute*() function) is responsible for deciding the target frame rate, frame size and bit rate. For the layer selection, the SVCCAT looks up into the *AdaptationQoS* description (available in the CAIN-21 demo) and eliminates the layers that do not satisfy the bit rate, frame size, and frame rate constraints, respectively. Then, the layer that presents the best quality in terms of VQM among the remaining layers is selected as the target layer. Accordingly, the layer that has the frame and bit rates of 30 fps and 358.9 kbps, respectively, is selected as the target layer for the experiment discussed in Section 3.3.

It should be noted that VQM considers the visual artefacts due to frame rate differences, such as jerky and unnatural motion [5]. Different spatial resolutions could be matched by up or down sampling according to a common reference while performing fair VQM measurements. However, this leads to introducing up/down sampling artefacts, which would have a negative influence on the VQM measurements. Thus, the resolution re-sampling is not performed in the experiments reported in this paper.

4. EXPERIMENTS AND RESULTS

This section presents a set of video adaptation experiments to demonstrate the effectiveness of combining the high-level (knowledge-based) decision phase with the low-level (quality-based) layer selection phase.

Three different video sequences were used in the experiments and adapted to three different terminal profiles. These video sequences correspond to the following *Content DI*s: *bus.xml*, *football.xml* and *hall-monitor.xml*. In all of the experiments, the video sequences were encoded in a scalable format at three spatial resolutions (i.e., QCIF, CIF, 4CIF), with a base layer and one quality refinement layer, and with a GOP size of 8 using the JSVM 9.13.1 codec [9]. The frame rate of the original video content was 30 fps, and the frame rates of 30, 15, 7.5, and 3.75 fps were supported by the encoded video sequences. Data on the layers and the corresponding VQM grades were stored in the *AdaptationQoS* description of each *Content DI*.

The details of the first experiment have already been provided as an example in Section 3.3. The second experiment involved the adaptation of the *Content DI* named *football.xml* to the terminal labelled as *svc_with_audio_352x288_30fps*. The parameters that CAIN-21 transfers to the SVCCAT during this execution are:

Source parameters:

visual_frame_size = {176x144, 352x288, 704x576}
visual_frame_rate = {3.75, 7.5, 15, 30}
visual_bitrate = [118.5 .. 5739]

Target parameters

visual_frame_size = {176x144, 352x288}
visual_frame_rate = {3.75, 7.5, 15, 30}
visual_bitrate = [118.5 .. 2500]

In this experiment, *decide*() selects 16 layers (from Layer 0 to Layer 15) that fulfil the three constraints imposed by the parameters (i.e., up to CIF resolution, up to 30 fps frame rate and up to 2500 kbps). Table 1 shows the frame size, frame rate, and bit rate of the 16 layers. The PSNR and VQM results for these layers are also presented in the table. The layers with higher PSNR values have higher quality from the standpoint of PSNR. Conversely, the layers with lower VQM values have higher quality from the standpoint of VQM.

The SVCCAT selects Layer 15 due to its lowest VQM (i.e., the highest quality) as the target layer. Accordingly, the target frame size, frame rate, and bit rate are determined as CIF, 30 fps, and 2157 kbps, respectively. If PSNR had been used as the quality assessment technique to decide the target parameters, Layer 12 should have been selected as the target layer, which has the frame size and rate of CIF and 3.75 fps. The sequence used in this experiment, namely the "Football" video, has a high motion activity. Thus, the layer selected using VQM adequately reflects the need for a higher frame rate, so as to provide better solutions for enhancing user's perceptual quality [6].

The third experiment involved the adaptation of the *Content DI* named *hallmonitor.xml* to the terminal labelled as *svc_with_audio_352x288_30fps*. In addition, in this experiment, the network profile *umts_3g* has been enabled with *max_capacity* = 300 kbps. The parameters that CAIN-21 transfers to the SVCCAT during this execution are:

Source parameters:
visual_frame_size = {176x144, 352x288, 704x576}
visual_frame_rate = {3.75, 7.5, 15, 30}
visual_bitrate = [22.4 .. 1258.8]

Target parameters
visual_frame_size = {176x144, 352x288}
visual_frame_rate = {3.75, 7.5, 15, 30}
visual_bitrate = [22.4 .. 300]

Table 1: Layers in *football.xml*

Layer	Frame size (pixels)	Frame rate (fps)	Bit rate (kbps)	PSNR (dB)	VQM
0	176x144	3.75	118.5	34.601	0.562
1	176x144	7.5	181.9	33.421	0.533
2	176x144	15	263.8	32.429	0.484
3	176x144	30	355.7	31.959	0.432
4	176x144	3.75	189.3	38.012	0.447
5	176x144	7.5	289.5	36.644	0.402
6	176x144	15	414.7	35.595	0.387
7	176x144	30	557.6	34.906	0.373
8	352x288	3.75	525.8	35.629	0.520
9	352x288	7.5	807.6	34.539	0.479
10	352x288	15	1169.1	33.608	0.429
11	352x288	30	1579.3	33.113	0.367
12	352x288	3.75	724.1	38.947	0.392
13	352x288	7.5	1111.4	37.662	0.377
14	352x288	15	1598.3	36.664	0.342
15	352x288	30	2157	35.987	0.301

Table 2: Layers in *hallmonitor.xml*

Layer	Frame size (pixels)	Frame rate (fps)	Bit rate (kbps)	PSNR (dB)	VQM
0	176x144	3.75	22.4	37.886	0.353
1	176x144	7.5	29.4	37.594	0.348
2	176x144	15	37.3	37.373	0.338
3	176x144	30	44.3	37.287	0.329
4	176x144	3.75	35.0	39.018	0.318
5	176x144	7.5	45.9	38.724	0.310
6	176x144	15	58.3	38.513	0.297
7	176x144	30	70.0	38.365	0.294
8	352x288	3.75	128.7	37.573	0.329
9	352x288	7.5	168.3	37.422	0.325
10	352x288	15	216.4	37.258	0.315
11	352x288	30	268.5	37.141	0.312
12	352x288	3.75	165.2	39.192	0.292
13	352x288	7.5	218.1	38.760	0.287
14	352x288	15	282.0	38.437	0.273

In this experiment, the *decide*() function has limited the target bit rate to 300 kbps due to the network constraint. There are 15 layers in the *AdaptationQoS* that fulfil the constraints imposed by these parameters (from Layer 0 to layer 14), as shown in Table 2. The layer with the lowest VQM is Layer 14, and therefore the selected target frame size, frame rate and bit rate are CIF, 15 fps and 282.0 kbps, respectively. Finally, note that if the PSNR criterion had been used for quality assessment, Layer 12 would have been selected.

5. CONCLUSIONS

This paper has presented a new technique to improve the quality of scalable video adaptation by exploiting VQM information in a knowledge-based decision method. Specifically, the knowledge-based method selects the parameters that fulfil the constraints of the usage environment, which leads to a several feasible adaptation solutions. Subsequently, the use of the VQM information has been proposed to select the adaptation solutions (values of the parameters) that fulfil the user's perceptual quality in the best possible way. The innovation of this paper can be realised from the combination of knowledge-based decision techniques with objective quality assessments. The experimental results have demonstrated the effectiveness of this combination.

6. REFERENCES

[1] D. Jannach, K. Leopold, C. Timmerer, H. Hellwagner, "A knowledge-based framework for multimedia adaptation", Int. J. App. Intelligence, 24(2):109-125, Apr. 2006.

[2] P. Soetens, M. De Geyter, "Applying domain knowledge to multistep media adaptation based on semantic web services", Proc. WIAMIS'05, Apr. 2005.

[3] F. López, J.M. Martínez, N. García, "CAIN-21: An extensible and metadata-driven multimedia adaptation engine in the MPEG-21 framework", LNCS, 5887:114-125, Dec. 2009.

[4] C. Timmerer, V.H. Ortega, J.M. González, A. León, "Measuring quality of experience for MPEG-21-based cross-layer multimedia content adaptation", Proc. WISe'08, Apr. 2008.

[5] M.H. Pinson, S. Wolf, "A new standardised method for objectively measuring video quality", IEEE Trans. Broadcasting, 50:312-322, Sep. 2004.

[6] G. Nur, H.K. Arachchi, S. Dogan, A.M. Kondoz, "Evaluation of quality scalability layer selection bit rate adaptation of scalable video content", Proc. WIAMIS'09, 2009.

[7] I.S. Burnett, F. Pereira, R.V. de Walle, R. Koenen, "The MPEG-21 book", John Wiley and Sons, 2006.

[8] F. López, D. Jannach, J.M. Martínez, C. Timmerer, H. Hellwagner, N. García, "Multimedia adaptation decisions modelled as non-deterministic operations", Proc. WIAMIS'08, May 2008.

[9] JSVM 9.13.1 Software, downloaded from CVS server, garcon.ient.rwth-aachen.de/cvs/jv