

# A SCALABLE CODING FRAMEWORK FOR EFFICIENT VIDEO ADAPTATION

*N. Šprljan, M. Mrak, G. C. K. Abhayaratne, E. Izquierdo*

Multimedia and Vision Lab, Dept. of Electronic Engineering  
Queen Mary, University of London, London E1 4NS, United Kingdom

Email: {nikola.sprljan, marta.mrak, charith.abhayaratne, ebroul.izquierdo}@elec.qmul.ac.uk

## ABSTRACT

Current digital video applications require video coding techniques that cater a wide range of quality levels, spatial resolutions and frame rates supporting different user preferences, varying transmission bandwidths and terminal capabilities. Efficient adaptation of video content is vital in such application environments. Encoding video in scalable formats support fast and efficient adaptation. This paper presents a flexible scalable video coding framework that supports mandatory scalability functionalities required for video adaptation. The encoder, decoder and extractor modules of the framework, the scalable bits stream descriptions and video adaptation are presented.

## 1. INTRODUCTION

The emergence of various types of communication and distribution networks has enabled various new applications. The continuous increase of available bandwidth have put the multimedia content at the forefront of consumer interest, but to deliver this content to anywhere at anytime a high degree of adaptability is sought. Target video distribution systems range from low-bandwidth dial-up connections to high-bandwidth dedicated video links supporting high-definition digital television while the displaying hardware platforms vary from hand-held communication devices to cinema-quality projectors. This problem of adaptability is traditionally addressed by multimedia content transcoding, which in the case of video content is too complex to be widely used. Here scalable multimedia comes into the picture as it allows low complexity adaptation. Scalability of a particular piece of digital information refers to its ability to be instantiated at various levels of fidelity. Smaller subsets of the whole bitstream give representation at lower quality, lower resolution, less information, *etc.*

In this paper we present a flexible scalable video coding (SVC) framework for efficient video adaptation, developed within the EC FP6 project - aceMedia [1], which targets knowledge discovery and embedded self-adaptability to enable content to be self organising, self annotating, self associating and more importantly self reformatting, by support-

ing adaptability to its context, such as network, terminal and user preference. The aceMedia project is built around the concept of an ACE, an *autonomous content entity*, which consists of content, metadata and associated intelligence. An ACE is capable of adapting itself to the varying context conditions. Such context-aware adaptability is at the heart of the aceMedia project and is expected to achieve this by means of fully scalable representation of content. Thus, this paper addresses scalable video representation and associated adaptability functionalities.

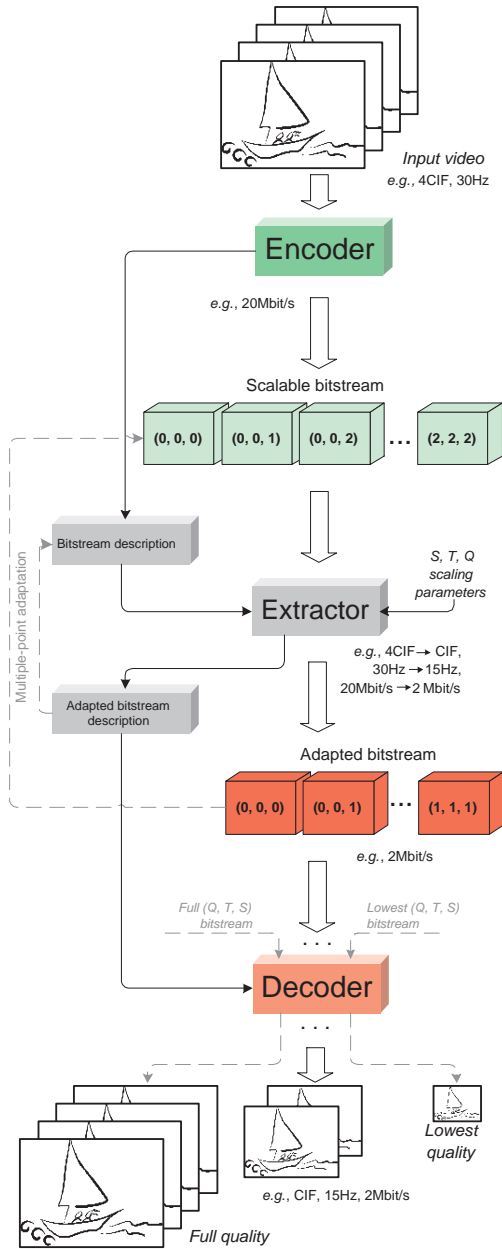
## 2. BACKGROUND

Over the years, scalable coding of multimedia - image, audio and video, has received a considerable attention in terms of both methodology and application. The main parts of such scalable coding algorithms include a hierarchical sub-band decomposition and embedded coding, yielding an embedded bitstream. Scalable bitstreams are usually represented by two parameters: *tiers* and *layers*. Embeddedness of the scalable bitstream indicates that each level of fidelity, corresponding to a particular layer of the bitstream, includes all of the lower layers corresponding to the lower levels of fidelity. The complete bitstream, consisting of all layers, represents the content in its full fidelity, which can be losslessly encoded media. The number of tiers nested in a scalable bitstream refers to the number of different types of scalability. Video usually requires scalability in quality, resolution and frame rate, therefore it is three-tier scalable.

Scalable image coding [2] and scalable audio coding [3] use 2D and 1D wavelet transforms, respectively, followed by embedded coding of the resulting wavelet coefficients. Similarly, 3D separable wavelet transforms have been used as the basis for scalable video coding. In early work on 3D wavelets based scalable video coding, the wavelet transform in the temporal direction has been performed either without [4] or with [5] motion compensated prediction. Although these 3D wavelets based methods did not yield high coding gains compared to the algorithms that use interframe motion compensated prediction techniques, they provided a framework for highly scalable video coding. Moreover, the

---

This research was supported by the European Commission under contract FP6-001765 aceMedia.



**Fig. 1.** Block diagram of the scalable coding

*lifting scheme* for wavelets has recently inspired a new temporal decomposition framework called *motion compensated temporal filtering* (MCTF) [5], in which the motion compensation has been incorporated into the temporal wavelet transform, thus improving both coding gain and scalability functionalities. Recent research on scalable video coding [6, 7] has resulted in efficient scalable video frameworks using combinations of spatio-temporal transform techniques and 3D embedded entropy coding. All these frameworks consist of two-step spatio-temporal decompositions: MCTF and 2D spatial transform. The spatio-temporal decomposi-

tions can be ordered in two ways: the MCTF followed by the spatial transform, *i.e.*,  $t+2D$  framework [6], and the spatial transform followed by the MCTF in transform domain, *i.e.*,  $2D+t$  framework [7]. The multiresolution (MR) structure resulting from the MCTF wavelet transform and the 2D subband decomposition enables temporal and spatial resolution scalability, respectively. These transforms followed by 2D or 3D embedded coding schemes enable achieving fine granular quality scalability.

### 3. SCALABLE VIDEO CODING FRAMEWORK

A scalable content coding framework can be represented by three modules: encoder, extractor and decoder. Fig. 1 shows a block diagram of a generalised scalable content coding framework.

The content is encoded only once by the encoder module. It outputs a scalable bitstream and its associated bitstream description, which can be interleaved within the video bitstream or given in a separate stream. The resulting bitstream is of the maximum required quality which can be quasi-lossless or even lossless. The main aim of the extractor is to truncate the scalable bitstream according to the input scaling parameters and to generate the adapted bitstream and its description. As shown on Fig. 1, the adapted bitstream is also scalable and can be fed back into the extractor together with its associated bitstream description. This scenario corresponds to the situation of multiple-point adaptation where the adapted bitstream is sent to the consequent network node and is adapted by another extractor. Finally, decoder is capable of decoding any adapted bitstream. The main technologies behind these modules are described in detail in following subsections.

#### 3.1. Encoder / Decoder Modules

The SVC framework presented in this paper follows the  $t+2D$  architecture. The encoder, shown in Fig. 2, is represented by two separate functional blocks: the spatio-temporal transform and entropy coding.

The spatio-temporal block includes scalable hierarchical motion estimation (HME), MCTF and the spatial wavelet transform. The MCTF is performed on a group of frames (GOF). A single level of the MCTF consists of splitting the frames in a GOF into odd and even indexed frames followed by two lifting steps: prediction and update. The prediction step uses either uni-directional or bi-directional motion compensated prediction of odd indexed frames using even indexed frames. This MCTF step yields high pass frames and corresponding motion vector fields. The update step uses the same motion vector fields to update the even indexed frames with the prediction step residuals. Since the motion vectors used in the update step are inverted, the pix-

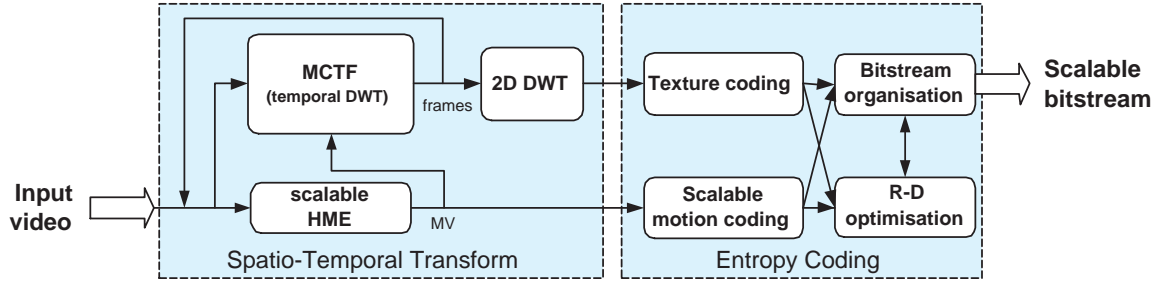


Fig. 2. Modules of the encoder

els in the even indexed frames are connected to those of the odd indexed frames in two ways: connected or unconnected. The update step is applied only to the connected pixels. In the case where a pixel is connected to more than one the pixel in the other frame, the average is used. Although it is widely considered that the MCTF/2D-DWT produces a hierarchical spatio-temporal decomposition, the motion vector fields obtained from ME are hierarchical only in the temporal direction. Since the motion vector fields are generated for a whole frame at the highest resolution and losslessly coded in a non-scalable manner as in conventional non-scalable video codecs, such motion fields would remain a fixed overhead cost at all other lower spatial resolution and bit-rate levels. Therefore, scalable motion vectors are vital for enhanced performance at lower spatial resolutions and bit-rates. In previous publications we showed how to achieve this by means of scalable generation [8] and scalable coding [9] of motion vectors. The result of this lifting process is a set of two types of *temporal* frames, one corresponding to the low-pass frames containing the motion compensated average, and the other to the high-pass frames containing the motion compensated difference. The process of motion estimation and MCTF can then further be applied to the obtained set of low-pass frames, and with the continuation of such iteration a desired level of dyadic temporal decomposition depth can be reached. If the motion estimation method is efficient in a way that it captured the motion accurately, the MCTF achieves concentration of most of the GOF energy in the final set of low-pass temporal frames.

The next step is the spatial decorrelation and in our framework it is achieved by means of 2D separable discrete wavelet transform (DWT), specifically by lifting implementation of the 9/7 wavelet filter. The outputs of the spatio-temporal transform are sets of motion vector fields and spatio-temporal subbands. The spatial and temporal scalability is thus given by performing the inverse spatio-temporal transform on particular subbands corresponding to the required level of spatio-temporal resolution.

The second block of the encoder consists of scalable motion vector coding, embedded texture coding, joint rate-distortion (R-D) optimisation and the bitstream organisation module. The spatio-temporal subbands, that are produced

in the spatio-temporal transform block, are sent to the embedded texture coding block. To introduce a support for the quality scalability, the texture coder produces an embedded bitstream for each of the spatio-temporal subbands. We use the embedded zerotree block coder (EZBC) [6]. The final two modules, R-D optimisation and bitstream organisation are responsible for optimum allocation of the bitstream across the quality layers and for generation of the bitstream description compliant to the targeted type of extractor.

### 3.2. Bitstream description and extractor

The bitstream is organised according to a three-tier spatio-temporal-quality scalability structure. For easier interpretation of the extraction process, the bitstream with such structure can be represented in a 3D *QTS* (Quality, Temporal resolution, Spatial resolution) space, with different parts of the bitstream at different coordinates, as shown in Fig. 3. The coordinates are determined by the quality layer and spatio-temporal subband that a particular part of the bitstream represents. We call this particular part an *atom* of the bitstream, since it represents the smallest entity that can be added or removed from the bitstream. It should be noted here that the this analogy is not entirely consistent since the quality scalability is fine-granular due to the embeddedness of the texture coder, *i.e.*, each atom can be truncated up to the desired bit location and yet can be decoded correctly up to that point (with the assumption that all the dependency atoms are available). However, as the quality of the decoded video can vary significantly depending on the selection of the truncation points across the spatio-temporal subbands, the bit-allocation algorithm finds the optimal set and selects the quality layers accordingly. If we declare the lowest layer as the 0-*th*, then the highest quality, temporal and spatial layer in the initial bitstream are  $q_{max}$ ,  $t_{max}$  and  $s_{max}$ , respectively. With this notation, each atom can be specified with a  $(q, t, s)$  3-tuple, corresponding to its position in the *QTS* space, as in Fig. 3. If by  $(Q_i, T_j, S_k)$  we denote the desired quality, temporal and spatial resolution, where  $Q_i \in \{Q_0, \dots, Q_{q_{max}}\}$ ,  $T_j \in \{T_0, \dots, T_{t_{max}}\}$  and  $S_k \in \{S_0, \dots, S_{s_{max}}\}$ , the extractor produces the bitstream

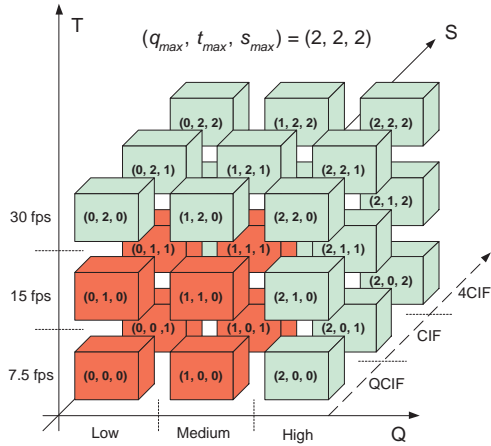


Fig. 3. 3D representation of a scalable video bitstream

which can be denoted as:

$$\bigcup_{q=0, t=0, s=0}^{i, j, k} (q, t, s).$$

In other words, it preserves all the atoms with the index lower or equal to the highest required and discards the rest.

Since quality layers correspond to the bit-planes of the encoded wavelet coefficients, the encoder and extractor produce constant-quality / variable-bit-rate streams. However, the extractor can also work in the average bit-rate mode, where the scalable bitstream is adapted to a required target bit-rate. To match the requested bit-rate, it firstly finds the maximum possible  $Q_i$  for a desired  $(T_j, S_k)$  such that average bit-rate across the GOFs for  $(Q_i, T_j, S_k)$  is lower than the desired bit-rate. Then it loops over all atoms ( $q = i, 0 \leq t \leq j, 0 \leq s \leq k$ ) and for each finds an optimal truncation point under constraint of the fixed total bit-rate.

In this way we can perform extraction on all possible bit-rates for all frame rates and resolutions that are supported in a scalable video bit-stream. As an example, we demonstrate the fine-granular quality scalability performance of our codec in Fig. 4. For comparison, we also show the performance of existing non-scalable standard codecs: MPEG-4 Part 10 (AVC) JM 9.3 and MPEG-2. For these two codecs the sequences are decoded from bitstreams encoded for a particular target bit-rate, while the full bit-rate/PSNR plot of our codec is produced by extracting from a single full-quality encoded bitstream.

#### 4. CONCLUSIONS

In this paper, we have presented the aceMedia scalable video framework that supports efficient video adaptation in order to achieve ACE adaptation objectives. The framework consists of the state-of-the-art spatio-temporal decomposition and encoding tools (MCTF, 9/7 2D DWT and EZBC),

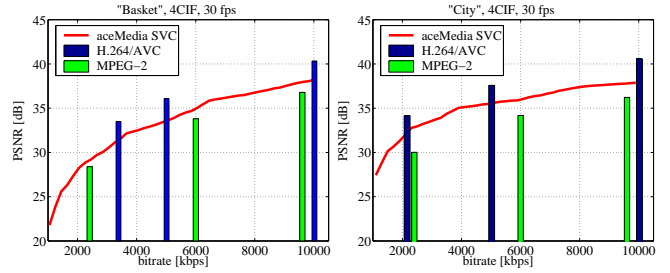


Fig. 4. PSNR results

and novel tools for scalable generation and coding of motion vectors for efficient low resolution adaptation. The bitstream organisation and description schemes support multiple-path adaptation enabling efficient content-aware adaptation of video in multimedia applications.

#### 5. REFERENCES

- [1] I. Kompatsiaris, Y. Avrithis, P. Hobson, and M. G. Strintzis, "Integrating knowledge, semantics and content for user-centred intelligent media services: The aceMedia project," in *Proc. Int. Work. on Image Analysis for Multimedia Interactive Services (WIAMIS)*, Apr. 2004.
- [2] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Trans. Image Processing*, vol. 9, no. 7, pp. 1158–1170, July 2000.
- [3] P. E. Kudumakis and M. B. Sandler, "Wavelet packets based scalable audio coding," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS)*, May 1996, vol. 2, pp. 41–44.
- [4] D. Taubman and A. Zakhor, "Multirate 3D subband coding of video," *IEEE Trans. Image Processing*, vol. 3, no. 5, pp. 572–589, Sept. 1994.
- [5] J.-R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Trans. Image Processing*, vol. 3, no. 5, pp. 559–571, Sept. 1994.
- [6] S.-T. Hsiang and J. W. Woods, "Embedded video coding using invertible motion compensated 3-d subband/wavelet filter bank," *Signal Processing : Image Communication*, vol. 16, no. 8, pp. 705–724, May 2001.
- [7] Y. Andreopoulos, M. van der Schaar, A. Munteanu, J. Barbarien, P. Schelkens, and J. Cornelis, "Complete-to-over-complete discrete wavelet transforms for scalable video coding with MCTF," in *Proc. SPIE Visual Communications and Image Processing*, July 2003, vol. Proc. SPIE 5150, pp. 719–731.
- [8] M. Mrak, N. Sprljan, G. C. K. Abhayaratne, and E. Izquierdo, "Scalable generation and coding of motion vectors for highly scalable video coding," in *Proc. Picture Coding Symp. (PCS)*, San Francisco, Dec. 2004.
- [9] M. Mrak, G. C. K. Abhayaratne, and E. Izquierdo, "On the influence of motion vector precision limiting in scalable video coding," in *Proc. Int. Conf. Signal Processing (ICSP)*, Aug. 2004, vol. 2, pp. 1143–1146.