An Efficient Video Transcoding Algorithm Between H.263 and MPEG-4 for Inter-Network Communications

Mehdi Jafari

Bahonar University of Kerman
Department of Mathematics & Computer Science
P.O.Box 14515-775, Kerman, Iran
mjafari@mail.uk.ac.ir

Shohreh Kasaei

Sharif University of Technology
Department of Computer Engineering
P.O.Box 11365-9517, Tehran, Iran
skasaei@sharif.edu

Abstract- **Video transcoding comprises the necessary operations for the conversion of a compressed video stream from one syntax to another one for inter-network communications, without the need of any further decoding and re-encoding process. The two most recent and popular video coders are H.263 and MPEG-4. As well as some similarities, there are a number of differences between MPEG-4 and the H.263 bitstream syntax. This paper presents an efficient algorithm for transcoding between MPEG-4 bitstream syntax and the H.263 bitstream syntax. This novel transcoding algorithm is proved to give highly improved service quality while reducing the complexity and time delay of conventional cascaded decoding/re-encoding process.**

Keywords- MPEG-4, H.263, Tanscoding, Inter-network communications.

I. INTRODUCTION

Due to the expansion and diversity of multimedia applications and the underlying networking platforms with their associated communication protocols, there has been a growing need for inter-network communications and media gateways. Video transcoding is a technique to convert one bitstream into another and there exist two different scenarios for this: homogeneous and inhomogeneous transcoding. Homogeneous transcoding reencodes bitstreams within the same standard but with different parameters, e.g. bit rate, frame rate and/or resolution of the pre-encoded video stream. In contrast, inhomogeneous transcoding extends this scenario to a conversion between different standards. This paper presents a novel full scale implementation of heterogeneous transcoding scheme that interconnects two different video standards operating at very low bit rates: MPEG-4 and H.263. Interoperability between these two very low bit-rate standards has increasingly been an important issue to operate them with the utmost compatibility with each other. Here, the main goals are to avoid cascaded decoding and re-encoding processes while maintaining the quality of service (QoS), reducing the processing power and most significantly the time delay associated with tandeming for delay sensitive applications such as two-way video communications. This paper is organized as follows. Section 2 gives brief idea about MPEG-4 and H.263 video coding algorithms. The general description of the mapping process is given in section 3. Algorithm details and simulation results for transcoding between H.263 and MPEG-4 are explained in section 4. Finally, the last section concludes the paper.

II. MPEG-4 AND H.263 PACKETISATION

This section describes the MPEG-4 and H.263 video coding standards according to [1] and [2] respectively. Also similarities and differences between the syntax of the two standards are discussed.

A. MPEG-4 Packetisation

MPEG-4 is an open standard of ISO [3]. The most important application of MPEG-4 will be in multimedia environment. The media that can use the coding tools of MPEG-4 includes computer networks, wireless communication networks, and the Internet. Perhaps the most fundamental shift in the MPEG-4 standard has been towards object-based or content-based coding, where a video scene can be handeled as a set of foreground and background objects rather than just as a series of rectangular frames. This type of coding opens up a wide range of possibilities (such as independent coding of different objects in a scene, reuse of scene components, compositing, and a high degree of interactivities). Unlike block-based video coders (such as MPEG-1,-2, H.261, and H.264), MPEG-4 detects entities in the video frame that the user can access and manipulate, hence providing the user with content-based functionalities for the processing and compression of any video scene. MPEG-4 defines a syntactic description language to describe the exact binary syntax of an audio-visual object bitstream, as well as that of the scene description information. A typical video syntax hierarchy is shown in figure 1.

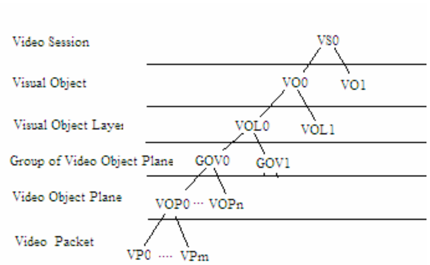

Figure 1: MPEG-4 video syntax hierarchy.

The *video session* (VS) is the highest syntactic structure of the coded video bitstream. AVS is a collection of one or more *visual objects* (VOs). The VO header information contains the start code followed by profile and level identification and a VO identification to indicate the type of object, which may be a still texture object, a mesh object, or a face object. A VO can consist of one or more *visual object layers* (VOLs). In the VOL, the VO can be coded with spatial or temporal scalability. *Group of Video* (GOV) is like the *group of picture* GOP [4] in MPEG-1 and -2, and *visual object plane* (VOP) is a video frame.

As shown in Figure 2, a VOP (or video frame) consists of a VOP header and several *video packets* (VPs). The VOP and VP headers contain the synchronization code and compression parameters. Each video frames starts with a start code. There are a number of start codes defined by MPEG-4 to make the decoding process clear and efficient [5]. Start codes are unique combination of bits that never occur in the video data. Each start code consists of a start code prefix followed by a start code value. The start code prefix is a string of twenty-three bits with the value zero followed by a single bit with the value one. The start code value is an eight bit integer, which identifies the type of start code. The VOP time parameter represents the number of the seconds elapsed since synchronization point marked by time stamp of the previously decoded *intra*- (I-) or *predictive*- (P-) VOP, in the decoding order. After the time parameter, the VOP quantization is added [6][8]. Apart from the start code, each video frame contains resynchronization markers at the boundaries of video packets. Also, data partitioning in MPEG-4 divides the VP into two parts. The header, motion, and shape data are coded in the first partition, while the less important texture information is placed in the second partition [8]. A VP header consists of *variable length coded* (VLC) macro block number, quantization scale parameter, and an optical *header extension code* (HEC) as shown in Figure 2(b). Each VP is partitioned into two portions separated by a DC-marker (in case of I-VOPs) or a motion marker (in case of P-VOPs). The MPEG-4 in addition to coding the texture and motion information traditionally encountered in block-based video coders, codes the shape of each VOP, as illustrated in Figure 2(b), so that the composition of objects can be done at the end decoder.

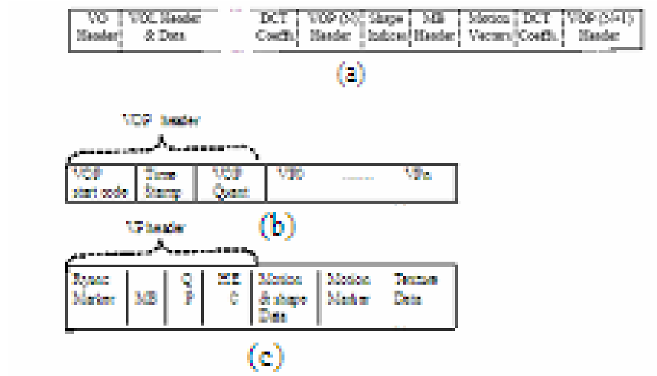

Figure 2: MPEG-4 Structure, (a) Bitstream syntax, (b) Video object plane (VOP) and (c) Video packet (VP).

B. Layering Syntax for ITU-T H.263 Video Coding Standard

The primary goal in the H.263 standard codec was coding of video at low or very low bit rates for applications such as mobile networks, public switched telephone network (PSTN) and the narrowband ISDN. This goal could only be achieved with small image size such as sub-QCIF and QCIF, at low frame rates. Today, this codec has been found so attractive that higher resolution pictures can also be coded at relatively low bit rates. The current standard recommends operation on five standard pictures of the CIF family, known as sub-QCIF, QCIF, CIF, 4CIF and 16CIF.

The syntax and layering structure of ITU-T H.263 video source coder are depicted in figure 3.

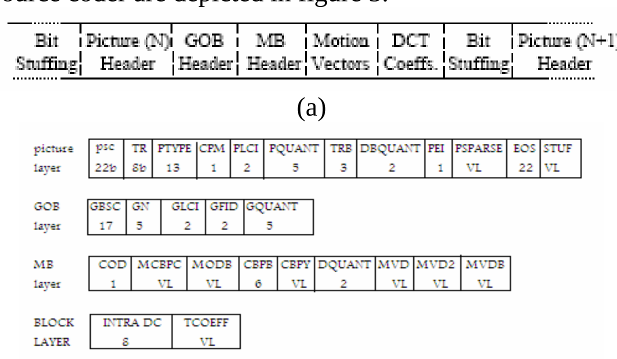

Figure 3: H.263 Structure (a) bitstream syntax, (b) Layering structure.

The next section presents the mapping of video parameters from H.263 to MPEG-4 or vice versa without the need of going through the conventional way of decoding and encoding.

III. METHODOLOGY OF TRANSCODING

The proposed transcoding algorithm maps the syntax of the two video standards from one to the other as depicted in Figure 4[7].

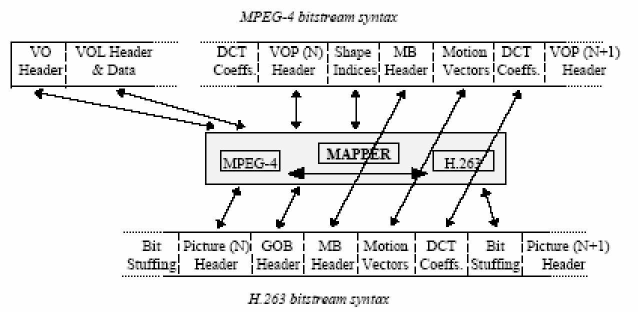

Figure 4: Bitsream syntax mapping between two standards.

MPEG-4 encoding algorithm has the ability to switch options for the use of different functionalities supported.
In case of switching off those many functionalities which are not provided by H.263, the encoder operates with a very close resemblance to H.263 baseline algorithm. In this case, the MPEG-4 video syntax is very similar to the current H.263 syntax, and many coding tools are common in two standards. Only very few changes are required to accommodate the tandeming of both standards without the need of any further decoding and encoding processes. There are several differences between the syntax of the two standards as:

A. VO and VOL headers

The video transcoder receives the incoming encoded bitstream and reads it without decoding. It then employs header extraction for the previous algorithm and new header insertion for the next algorithm. The extraction of MPEG-4's VO and VOL headers and data, and VOP headers from the MPEG-4 bitstream, insertion of H.263 picture headers to H.263 picture data. This is one way translation of the syntax from MPEG-4 to H.263.
After the picture headers, macroblock headers are mapped. and then MB data comprising motion vectors and discrete cosine transom coefficients along with the bit stuffing parts of the syntax are translated.

B. Intra-coded MBs

The other major distinction arises between MPEG-4 and H.263 VLC tables is the existence of a second Huffman table for the encoding of the most commonly occurring EVENTs [8][9] for blocks on MPEG-4 side. The EVENTs for intra luminance blocks, and intra chrominance and inter blocks of MPEG-4 are encoded by the use of two different VLC tables whilst all H.263 blocks use one of the same VLC tables for AC coefficients of MPEG-4. During the translaion of intra-coded MBs, the transcoder dequantises and requantises the received DC coefficients for correct decoding levels between different algorithms. Also, AC coefficients of intra-blocks belonging to MPEG-4 video make use of a new Huffman VLC table which is not defined in the H.263 algorithm. Hence, during transcoding from MPEG-4 to H.263, only these blocks need to be re-valuated. In the reverse direction, this problem does not exist.

C. Inter-coded MBs

For the transcoding of inter-MBs, the distinction arises in the MV mapping. Very careful attention should be paid to the mapping of MVs(Motion Vectors) in macroblock layer and TCOEFFs(Transform Cefficients) in block layer of the processed video frame. It is for the reason that MPEG-4 MV table has 65 indices leading to 65 different vector differences whilst H.263 table comprises of one less index. This extra vector index is mapped to the nearest vector difference. In the reverse path, this problem does not occur.

D. Motion Compensation

The last difference between the standards occurs in the motion compensation stage of the predicted frames. MPEG-4 standard evaluates a rounding parameter which does not exist in H.263. This parameters has arbitrary values during the motion compensation of predicted MBs, and it is signaled in the MPEG-4 header data [1]. For a pixel matching basis, this parameter should be forced to have a null value in order to comply with its absence in H.263. During the simulations, the effect of changing this rounding parameter to zero was experienced as a negligible loss in the *peak signal-to-noise ratio* (PSNR) levels of MPEG-4 video.

IV. SIMULATION RESULTS

Figure 5 shows the subjective quality obtained by using a bi-directional heterogeneous video transcoding algorithm between the H.263 and MPEG-4. It can be clearly seen that the transcoder performance is almost as good as the direct encoding/decoding scheme.

Figure 6 shows the component PSNR levels of H.263 to MPEG-4 direction and Figure 7 illustrates the reverse path PSNR values. Each diagram comprises three simulation results: (a) direct encoding/decoding as reference, (b) transcoding (c) cascaded decoding/re-encoding. As it is clear from the figures 6 and 7, the PSNR values for transcoded sequences in both directions are quite close to those of the direct encoding/decoding schemes. In both cases, the PSNRs of transcoded frames are superior over cascaded decoded/re-encoded frames, on average. This is due to the fact that transcoding uses the MVs of the incoming bitstream without decoding them wereas in cascaded decoding and re-encoding new MVs are calculated based on the lossy reconstructed pictures.

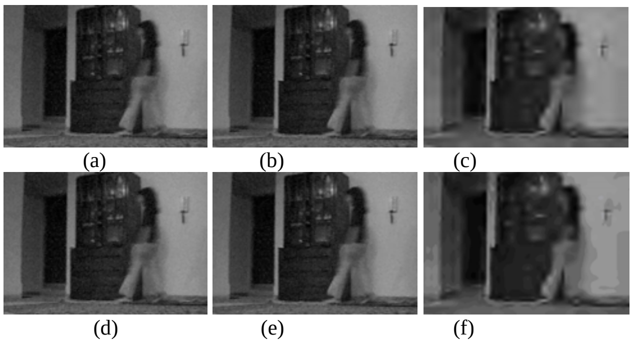

Figure 5: Subjective performance evaluation of heterogeneous video transcoding using 150 frames of the walking person sequence, (a) MPEG-4 direct encoded/decoded(direct), (b) MPEG-4 encoded and transcoded and H.263 decoded(transcoded), (c) MPEG-4 encoded/decoded and H.263 re-encoded/re-decoded(cascade), (d) H.263 direct encoded/ decoded (direct), (e) H.263 encoded and transcoded and MPEG-4 decoded (transcoded), (f) H.263 encoded/decoded and MPEG-4 re-encoded/re-decoded(cascade).

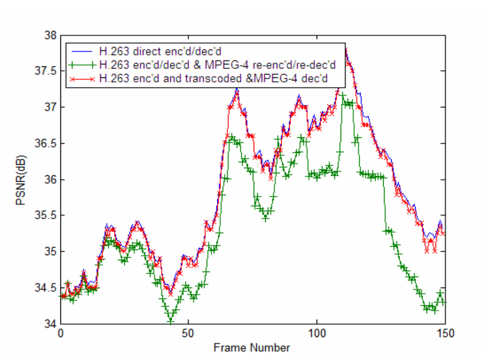

Figure 6: Performance evaluation of heterogeneous video transcoders using PSNR values, H.263 to MPEG-4 for walking person sequences.

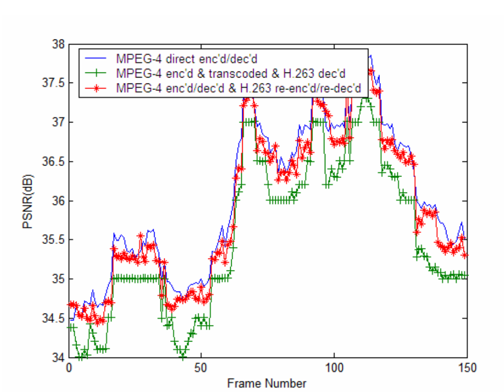

Figure 7.: Performance evaluation of heterogeneous video transcoders using PSNR values, MPEG-4 to H.263 for walking person sequences.

V. Conclusions

It is important and challenging to implement transcoders to interconnect standards and do the mapping without any significant delay caused by further decoding and encoding processes. In this paper a newly proposed algorithm between MPEG-4 and H.263 has been presented. It has been shown that the direct mapping for the bit patterns of the two standards can be achieved without degrading the video quality. The method also has the advantages of very low processing delay and complexity.

References

[1] T. Ebrahimi (ED): 'MPEG-4 video verification model version 8.0' , ISO/IEC JTC1/SC2/WG11 N1796, Stockholm meeting, July 1997

[2] Draft ITU-T Recommendation H.263: Video coding for low bitrate communication' May 1996.

[3] MPEG-4 Committee Draft,"Information technology Coding of Audio-Visual Objects: Visual ISO/IEC 144496-2," ISO/IEC JTC/SC29/WG11 N22o2, Tokyo, Mach 1998.

[4] T.Sikora, and Chiariglione,"The MPEG-4 Video Standardant its Potential for Future Multimedia Applications," Proceeding of IEEE, ISCAS conference, Hongkong, June 1997.

[5] A. Puri and A. Eleftherials. MPEG-4: An object-based multimedia coding standard supporting mobile applications. ACM Mobile Network and Applications, Vol. 3,June 1998,

pp. 5-32.

[6] W. Rabmer, M. Budagavi, and R. Talluri, proposed extensions to DMIF for supporting unequal error protection of MPEG-4 video over H.324 mobile networkes. DOC M4135. MPEG Atlantic, City meeting, October 1998.

[7] S., Dogan, A.H. , Sad, and Kondoz, A. M., 'Tandeming/transcoding issue between MPEG- and H.263' in RUGGIERI, M, (ED):' Moble and personl satellite communications 3 (Springer-Verlag, Great Britin, 1999), pp. 339-346.

[8] Z. Ahmed, S. Worrall, A. H. Sadka, A. Kondoz," A Novel Packetisation Scheme for MPEG-4 over 3-G Wireless System", VIE. 2005, Glasgow, UK, 4-6 April 2005, PP. 309-314.