Inter Mode Conversion of H.264 to MPEG-2 Video with Reduced Complexity

Sandro Moiron¹, Sergio Faria¹,², Pedro Assunçao¹,², Vitor Silva¹,³, Antonio Navarro¹,⁴
¹Instituto de Telecomunicações, Portugal; ²ESTG, Instituto Politécnico de Leiria, Portugal; ³DEEC, Universidade de Coimbra, Portugal; ⁴DET, Universidade de Aveiro, Portugal.

Abstract - In this paper we propose an efficient inter mode conversion method for low complexity H.264/AVC to MPEG-2 video transcoders. The novel coding tools introduced in H.264 are exploited along with the more classic ones of MPEG-2, such that computational complexity reductions up to 60% and quality gains are achieved in comparison with full recoding. The proposed method is suitable for software implementation of video transcoders in hardware platforms where the available resources are shared among multiple processes (e.g. home gateways).

I. INTRODUCTION

In the past few years, new multimedia services and applications have emerged driven by customer demand and advances in related technology, such as more efficient compression standards. However, the upcoming of new standards also gives rise to a growing need for providing backward compatibility during relatively long periods, in order to smooth technology transition and allow an easy upgrade of the consumer equipment. In this context, transcoding of multimedia content from new to older standards is necessary for services providers and multimedia consumers that want to increase their equipment's lifetime.

The actual service scenario is composed by a large amount of multimedia consumers using MPEG-2 based equipment, while H.264/AVC is yet on the marketplace offering 2-3 times better coding efficiency. This leads to conflicting requirements, where service providers and network operators want to benefit from the higher coding efficiency of H.264/AVC, but multimedia consumers may not see relevant advantages to upgrade their home equipment (e.g., DVD players, PVRs, etc). Video transcoding from H.264/AVC to MPEG-2 is a possible solution to ensure interoperability.

H.264/AVC to MPEG-2 transcoding has already been addressed in [1], [2] and [3], but they only deal with P slices sequences, whilst I slices have been recently addressed in [4]. The main novelty of the transcoder proposed in this paper is the B slice processing, which are the most complex ones. The most relevant coding parameters, such as motion vectors and macroblock type, are extracted from the H.264/AVC bitstream. These extracted parameters are reused such that relevant information previously processed by the H.264/AVC encoder is reused to encode the MPEG-2 bitstream, after a low complexity process for format adaptation.

The following section describes the proposed method to efficiently convert Inter coded macroblocks from H.264/AVC to MPEG-2. Sections III present and discuss the experimental results. Finally, section IV draws some conclusions about this method and presents further work for this application.

II. MODE CONVERSION

Although H.264/AVC and MPEG-2 use the same block-based coding paradigm, there are significant differences between them. The H.264/AVC standard supports the partitioning of 16×16 macroblock into smaller partitions. In the MPEG-2 side, the frame encoding of macroblock partitions is an unsupported feature. Therefore, the macroblock conversion is restricted at the output to a unique 16×16 type, which is not a direct conversion as described in Figure 1.

The conversion of a multi-partitioned macroblock to a unique macroblock implies to convert the global motion, described by several motion vectors, into a unique motion vector, that describes the global motion of the respective macroblock. Since each motion vector can point to a different direction, merging these motion vectors to a unique is not straightforward. Therefore, the candidate motion vector is determined by minimising the residue of the macroblock built with the multiple partitions, using the respective motion vectors. The unique motion vector is obtained based on the reference block displacement that achieve the smallest residue.

The high performance achieved by H.264/AVC is mainly related to the contribution of several newly introduced tools. While MPEG-2 use for reference prediction only adjacent pictures, H.264/AVC can use a list of up to 16 reference pictures. The use of reference pictures beyond MPEG-2 limits makes the prediction incompatible. Therefore, the proposed motion vector scaling allows the conversion of the source reference predictions into MPEG-2 compatible ones. The motion vector is scaled according to the temporal distance (Δt) between the reference picture and the current one (Cref) as given by

\[ MV_{MPEG2} = \text{round}\left(\frac{MV_{H.264}}{\Delta_t}\right), \text{where } \Delta_t = \frac{C_{ref} - H.264ref_{ref}}{C_{ref} - MPEG2ref_{ref}}. \]
Scaling the motion vector with this factor will make the reference picture compatible with MPEG-2. Out of boundary motion vectors are also converted into MPEG-2 type motion vectors when converting B slices.

The motion vector conversion for this case assumes a constant movement of the macroblock area. Therefore, it is possible to estimate the corresponding position in the opposite direction of the current reference picture. The reverse motion vector is obtained by flipping the source motion vector in both axes. This technique can only be applied in B slices, where the prediction direction can be either forward or backward.

Since the previous method introduces significant quality gains, its use has been extended to the remaining cases, and further improved adding bidirectional prediction. The conversion from uni to bi-directional prediction can be achieved by combining the motion vector from the previous method with the source motion vector. Additionally to these methods, it was used a refinement with ½ pixel search window to fine tune the selected motion vectors.

This process is highly efficient since it is intensively used by H.264/AVC, and its implementation avoids the full recalculation of the MPEG-2 motion vector.

### III. EXPERIMENTAL RESULTS

The performed simulations intend to compare the modified transcoder "Fast Transcoding" with the classical cascade transcoder "Full Recoding". It was used three distinct test sequences (Mobile, Stockholm and Shields) each one with 250 frames and 720×576 pixel size. These sequences were encoded using H.264/AVC JM10.2 at 5Mbps, following an 'IBPB' encoding pattern with GOP (Group Of Pictures) size 12 and allowing 5 reference pictures. The transcoder is composed by an H.264/AVC JM10.2 decoder and a MPEG-2 encoder, v1.2, from the MPEG Software Simulation Group. As shown in Figure 2 and Figure 3, the objective image quality achieved by the modified and the reference transcoder is very similar for both slice types. The computational time savings for Stockholm sequence is 60% for P slices and 51% for B slices, achieving a global saving of 55%, as depicted in Figure 4. All simulation results are consistent for the remaining two sequences, Mobile (52%) and Shields (57%).

### IV. CONCLUSION

The proposed video transcoder exploits the H.264/AVC bitstream information, to perform a fast mode decision. This task strongly relies on the reuse of the H.264/AVC motion vectors information, namely these affected by multiple block partitions, multiple references and out of slice boundaries motion vectors. Thus, it is able to achieve a significant 60% computational time reduction when compared with the reference transcoder and without image quality penalty. The proposed transcoder can be used as an additional module for a home gateway using wireless content distribution scenarios. The implementation of an R-D optimisation is proposed for further improvement.

### REFERENCES


