

H.263 TO H.264 TRANSCODING USING DATA MINING.

*Gerardo Fernández-Escribano¹, Jens Bialkowski², Hari Kalva³,
Pedro Cuenca¹, Luis Orozco-Barbosa¹ and André Kaup²*

¹**Instituto de Investigación en Informática de Albacete. Universidad de Castilla-La Mancha.
Campus Universitario, 02071 Albacete, SPAIN**

²**Chair of Multimedia Communications and Signal Processing, University of Erlangen-Nuremberg,
Cauerstr. 7, 91058 Erlangen, GERMANY**

³**Department of Computer Science and Engineering, Florida Atlantic University, Boca Raton, FL, USA**

ABSTRACT

In this paper, we propose the use of data mining algorithms to create a macroblock partition mode decision algorithm for inter-frame prediction, to be used as part of a high-efficient H.263 to H.264 transcoder. We use machine learning tools to exploit the correlation and derive decision trees to classify the incoming H.263 MC residual into one of the several coding modes in H.264. The proposed approach reduces the H.264 MB mode computation process into a decision tree lookup with very low complexity. Experimental results show that the proposed approach reduces the inter-prediction complexity by as much as 60% while maintaining the coding efficiency.

Index Terms— H.263, H.264, Data Mining

1. INTRODUCTION

The H.264/AVC standard [1] achieves much higher coding efficiency than the MPEG-2, H.263 and MPEG-4 standards, due to its improved inter and intra prediction modes at the expense of higher computation complexity. Therefore, H.264/AVC is a strong candidate for a wide range of applications in the future.

Transcoding MPEG-2, H.263 and MPEG-4 video to H.264 is important to enable gradual migration to H.264. For example the heterogeneous transcoding of H.263 [1] into H.264/AVC[2] bitstreams for UMTS or DVB-H networks will become necessary in the near future. However, given the significant differences between the H.263 standard and the H.264 coding algorithms, transcoding is a highly complex task and new approaches to transcoding are necessary [3].

The H.263 to H.264 complexity reduction techniques reported in the literature fall into two categories: 1) the mapping of the MB modes of the incoming H.263 video into the H.264 MB mode and 2) the selective evaluation of H.264 MB modes by means of heuristics. Because of the large number of inter and intra MB coding modes supported by H.264, there is no one-to-one mapping between the H.263 and H.264 MB modes. In this work, we present a novel approach that is not limited by the inefficiencies of mode mapping or selective evaluation approaches. The proposed approach is based on the hypothesis that the H.264 MB coding mode decisions have a high correlation with the distribution of the motion compensated residual in H.263 video. This hypothesis has been demonstrated to be very efficient in heterogeneous MPEG-2/H.264 video transcoders [4]. In this work, we use machine learning tools again to exploit the correlation and derive decision trees to classify the incoming H.263 MC residual into one of the several coding modes in H.264. The proposed approach reduces the H.264 MB mode computation process into a decision tree lookup with very low complexity.

The rest of the paper is organized as follows. Section 2 reviews the principles of operation of the prediction of inter-coded macroblocks in P-slices used by the H.264 encoding standard. Section 3 introduces our macroblock partition mode decision algorithm specifically designed for H.263 to H.264 transcoders. In Section 4, we carry out a performance evaluation of the proposed algorithm in terms of its computational complexity and rate-distortion results. Finally, Section 5 draws our conclusions.

2. MACROBLOCK MODE DECISION OF H.264 AND THE RELATED SIDE INFORMATION OF H.263

One of the main tasks of the transcoder to be developed herein is to find the H.264 macroblock mode given the macroblock mode of H.263. The parameters used in this approach are the macroblock mode, the mean and the variance of the related decoded residual image and the

This work was supported by the Ministry of Science and Technology of Spain under CONSOLIDER Project CSD2006-46, CICYT Project TIN2006-15516-C04-02, the Council of Science and Technology of Castilla-La Mancha under Project PAI06-0106 and FEDER

coded block pattern. We will now discuss the mode selection process of H.264 with respect to the input data of H.263. Only the baseline profile of H.263 will be taken into account.

Since there is no intra prediction in the baseline profile of H.263, the residual image of H.263 is simply the quantized image itself. As a result the mean value of the pixels of the intra MBs will be higher than the mean of inter MBs. This feature can be used to build the decision tree. Typically the coded block pattern assigns the coded coefficients to each transform block here and is not useful to obtain a macroblock mode of H.264.

In contrast, the amplitude of the mean and the variance of inter macroblocks of H.263 can greatly vary. This is due to the fact that only one motion vector is given for the whole H.263 MB. In some cases, e.g. fixed background, this prediction is sufficient and leads to small mean and variance of the macroblocks. However, if the prediction is performed at the edge of moving objects, the macroblock size of 16x16 pixels is too large which results on parts of the MB with great difference between the image and predictor. In those cases, the mean and variance values will be higher than in the background areas.

The output bitstream of our transcoder is H.264 baseline profile. Given the side information discussed above, the transcoder has to choose among various intra and inter macroblock modes and partitions. For the sake of completeness, the different macroblock modes will be discussed now. H.264 uses intra prediction in the pixel domain using the neighboring pixels of the partition to be coded. To different sizes of prediction patterns are possible - 16x16 or 4x4. If Intra16x16 is chosen, the prediction is performed given one direction for the full macroblock using pixels of the macroblocks above, to the left and left-above the current macroblock. For Intra4x4 the encoder has to assign a direction for each of the sixteen 4x4 partitions of a macroblock independently.

The mode decision of inter macroblocks involves an even greater range of parameters. In the case that the transcoding output is an inter macroblock, the most important step is to find the correct partition size. The macroblock of H.264 can be freely subdivided into a large number of different partitions resulting in a reduced residual energy.

There exist three coarse subdivisions, 16x16 pixels with one motion vector and 16x8 or 8x16 pixels with two different motion vectors. The macroblock can further be subdivided into sub-blocks of 8x8. It is possible to select partitions of 8x8, 8x4, 4x8 and 4x4, each with a different motion vector. The selection of these modes is based on the partition size 8x8. Although it is not possible to have different size partitions, such as 8x4 and 4x4 in the same sub-block of

8x8, the number of possible subdivisions is very high. This method of partitioning a macroblock into different subblocks is also known as tree-structured motion compensation. Figure 1 summarizes the different partition block sizes defined in H.264. The goal is to avoid areas of high residual energy at the cost of side information.

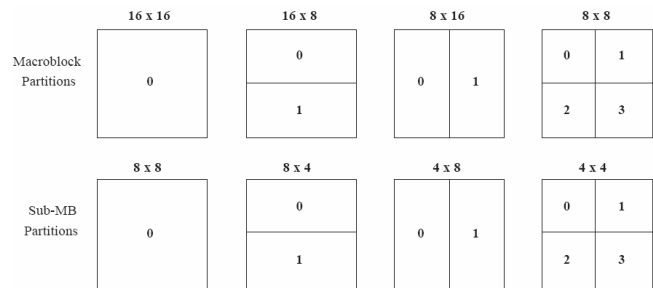


Figure 1. Macroblock partitions, sub-macroblock partitions and partition scans.

3. THE DECISION TREE BASE ON DATA MINING

In this section, we introduce the proposed macroblock partition mode decision algorithm aiming to reduce the computational time used for making the inter-frame prediction. This goal is achieved by making use of the H.263 MB coding mode, the coded block pattern (CBPC) in H.263 and the mean and variance of the residual information for this MB calculated for its 4x4 sub-blocks (see Figure 3.a, 3.b and 3.c). H.263 uses 16x16 motion compensation (MC). The MC residual can thus be exploited to understand the temporal correlation of variable block sizes in H.264. The open source WEKA [5] data mining tool is used to discover the relationship between this variables. We use the J.48 algorithm [6] and the training set was made with the flower garden H.263 CIF sequence, encoded with QP = 12 (around 1.5 Mbps), with only I and P frames. We used the P frames to create the training dataset. Each P frame in the sequence, in the CIF format, has up to 396 P MBs. A large training set can thus be easily created using a large video sequence. We found that this one sequence is sufficient to capture the mean and the variance distribution of the residual in H.263 coded video sequences and the decision tree made with this training set was able to make accurate mode decisions for all tested sequences.

We use three WEKA decision trees, shown in the picture with grey balls. The Decision tree works as follows:

- Node 1. The first level of the decision will be select only skip, intra, 8x8 modes or 16x16 modes, depending of the rules given by the first decision tree created with the WEKA data mining tool.
- Node 2. The inputs for this node are skip-mode MBs candidates to be skipped, as proposed by the

node 1. This node evaluates only the H.264 16x16 mode (without the modes 16x8 or 8x16). Then, the node selects the best option, skip or inter 16x16.

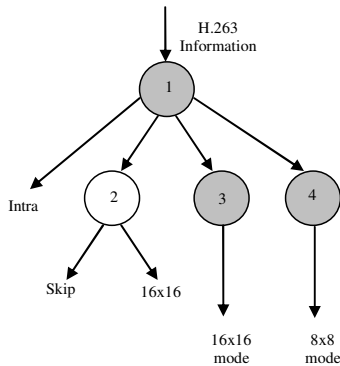


Figure 2. The Decision Tree

- Node 3. The inputs for this node are the 16x16 MBs coming from the node 1. The output of this node is the mode that will be use for coding the MB: 16x8 or 8x16.
- Node 4. In this node we use a tree decision generated with WEKA to decide which 8x8 mode should be used to encode the MB in H.264.

The WEKA tool was used to determine different sets of mean and variance thresholds for the H.263 residual. Due to space constraints we cannot show all the rules that are evaluated in the WEKA decision nodes. Each grey decision node in the Figure 2, implements a decision tree. A decision tree can be made by first creating a file with the 16 mean and variance values for each MB, the H.263 MB coding mode, the coded block pattern (CBPC) and the corresponding H.264 MB coding mode decision for that MB as determined by the H.264 reference software. Since the MB mode decision, and hence the thresholds, depend on the quantization parameter (QP), the decision tree thresholds are computed for a mid-QP of 25 and then adjusted for other QPs. Since the quantization step size in H.264 doubles when QP increases by 6, the thresholds are adjusted by 12.5% for a change in QP of 1.

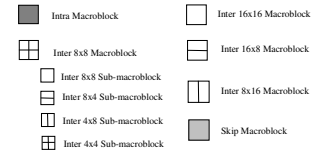
4. PERFORMANCE EVALUATION

In order to evaluate the proposed macroblock partition mode decision algorithm, we have implemented the proposed approach based on the H.264 reference software [7] (version 10.2). The metrics we have been interested on are Δ BitRate and Δ PSNR (rate distortion function), and Time Reduction. Throughout our experiments, we have used various video sequences exhibiting different spatial characteristics and different size formats (CIF and QCIF). We use Q factors from QP = 28 to QP = 40. The size of the GOP is 12 frames; where the first frame of every GOP was encoded as I-frame,

and the rest of the frames of the GOP were encoded as a P-frames. The rate control was disabled for all the simulations. Figures 3.e and 3.f show the differences between the inter mode selection made by the H.264 standard, and the proposed algorithm, with a value of 28 for QP in the Paris sequence, CIF format.



(a) Second Y frame (first P frame), sequence Paris (CIF).



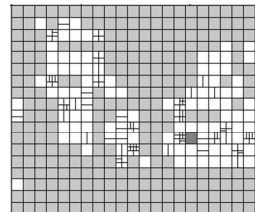
(b) Different kinds of Inter Macroblocks in the grid pictures.



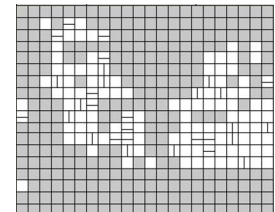
(c) Mean of the first P frame, sequence Paris (CIF).



(d) Variance of the first P frame, sequence Paris (CIF).



(e) H.264 QP = 28 First P frame, sequence Paris (CIF).



(f) Proposed QP = 28 First P frame, sequence Paris (CIF).

Figure 3. Macroblock partitions generated by the proposed algorithm for the first P-frame in the football sequence, based on the mean and the variance.

Figure 4 shows the RD results of applying the H.264 full estimation algorithm and the proposed macroblock partition mode decision algorithm.

Table 1 shows the time reduction for the proposed approach, compared with the RD optimized encoding stage, for encoding 200 frames of each sequence, (using a QP factor between 28 and 40) for different video sequences and size formats. The comparison metrics were produced and tabulated based on the difference of coding time (Δ Time), the PSNR difference (Δ PSNR) and the bit-rate difference (Δ BitRate). PSNR and bit-rate differences are calculated according to the numerical averages between the RD-curves derived from JM encoder (RD-Optimized option enabled) and the algorithm under study. The detail procedures in calculating these differences can be found from JVT documents authored by Bjontegaard [8], which is recommended by JVT Test Model Ad Hoc Group [9]. Note that PSNR and bit-rate differences should be regarded as equivalent, i.e., there is either the decrease in PSNR or the increase in bit-rate, but not both at the same time.

Table 1. Δ BitRate, Δ PSNR and Time Reduction results.

| Sequence, Format, QP's | | | Δ BitRate, Δ PSNR and Time Reduction (mean) from H.264 (RD-Optimized) | | |
|------------------------|------|---------------|--|--------------------|----------------------|
| | | | Proposed | | |
| | | | Time (%) | Δ PSNR (dB) | Δ BitRate (%) |
| Akiyo | CIF | (28,32,36,40) | - 62,12 | - 0,003 | 0,18 |
| Paris | CIF | (28,32,36,40) | - 61,00 | - 0,021 | 0,95 |
| Tempete | CIF | (28,32,36,40) | - 61,18 | - 0,040 | 2,75 |
| Akiyo | QCIF | (28,32,36,40) | - 66,67 | - 0,005 | 0,22 |
| Paris | QCIF | (28,32,36,40) | - 61,39 | - 0,050 | 2,70 |
| Tempete | QCIF | (28,32,36,40) | - 61,19 | - 0,051 | 2,71 |

As seen from Table 1, the algorithm presents negligible loss of video quality, with a slight increment in the bit rate. However, in terms of time saving, which is a critical issue in video transcoding applications. Our results show that the proposed algorithm is able to maintain a good picture quality while considerably reducing the number of operations to be performed.

5. CONCLUSIONS

In this paper, we showed data mining algorithms can be used to create a macroblock partition mode decision algorithm for inter-frame prediction, to be used as part of a high-efficient H.263 to H.264 transcoder. The proposed algorithms used data mining techniques to exploit the temporal correlation in the H.263 MC residual. The WEKA tool was used to develop decision trees for H.264 coding mode decision. Our results show that the proposed algorithm is able to maintain a good picture quality while considerably reducing the computational complexity by 60% on average.

6. REFERENCES

[1] G. Sullivan, Ed.: ITU-T Recommendation H.263: Transmission of Non-Telephone Signals; Video Coding for Low Bit Rate Communication, ITU-T SG 11, Jan. 1998

[2] ITU-T RECOMMENDATION H.264 “Advanced Video Coding for Generic Audiovisual Services”. May 2003.

[3] J. Bialkowski, M. Barkowsky, A. Kaup, “Overview of Low-Complexity Video Transcoding From H.263 to H.264,” Proceeding of the International Conference on Multimedia & Expo ICME 2006, July 2006.

[4] G. Fernandez-Escribano, H. Kalva, P. Cuenca and L. Orozco-Barbosa “A Very Low Complexity MPEG-2 to H.264 Transcoding Using Machine Learning”. Proceeding of the ACM Multimedia 2006, Santa Barbara (California), USA, October 2006, pp. 931-940.

[5] Ian H. Witten and Eibe Frank, “Data Mining: Practical machine learning tools and techniques”, 2nd Edition, Morgan Kaufmann, San Francisco, 2005.

[6] J.R. Quinlan, “C4.5: programs for machine learning”, Morgan Kaufmann, 1993.

[7] Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, Reference Software to Committee Draft. JVT-F100 JM10.2. 2004.

[8] G. Bjontegaard, “Calculation of Average PSNR Differences between RD-Curves”, presented at the 13th VCEG-M33 Meeting, Austin, TX, April 2001.

[9] JVT Test Model Ad Hoc Group, “Evaluation Sheet fro Motion Estimation”, Draft version 4, February 2003.

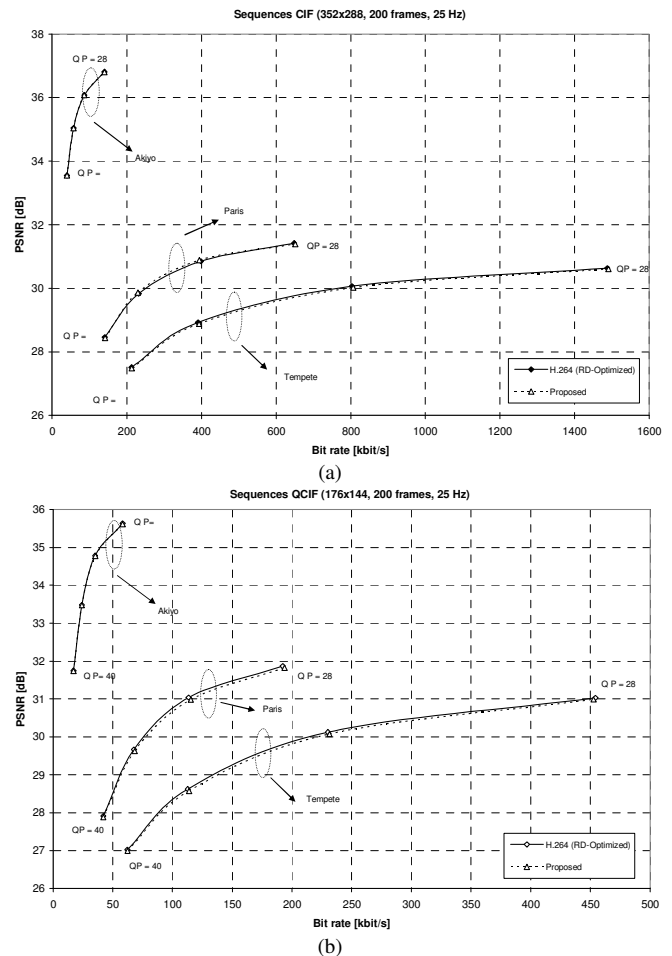


Figure 4. Rate Distortion Results. (a) CIF. (b) QCIF.