Shanableh, T., Peixoto, E., & Izquierdo, E. (2013). MPEG-2 to HEVC video transcoding with content-based modeling. IEEE Transactions on Circuits and Systems for Video Technology, 23(7), 1191-1196.
DOI: 10.1109/TCSVT.2013.2241352

MPEG-2 to HEVC video transcoding with content-based modeling

Tamer Shanableh
American University of Sharjah
Sharjah, UAE
tshanableh@aus.edu

Eduardo Peixoto
Queen Mary, University of London, London, UK
eduardopeixoto@ieee.org

Ebroul Izquierdo
Queen Mary, University of London, London, UK
ebroul.izquierdo@elec.qmul.ac.uk

Abstract

This paper proposes an efficient MPEG-2 to HEVC video transcoder. The objective of the transcoder is to migrate the abundant MPEG-2 video content to the emerging HEVC video coding standard. The transcoder introduces a content-based machine learning solution to predict the depth of the final HEVC coding units. The proposed transcoder utilizes full re-encoding to find a mapping between the incoming MPEG-2 parameters and the outgoing HEVC depths of the coding units. Once the model is built, a switch to transcoding mode takes place. Hence the model is content-based and varies from one video sequence to another. The transcoder is compared against the full re-encoding using the default HEVC fast motion estimation. Using 5 HEVC test sequences, it is shown that a speed-up factor of up to 3 is achieved whilst reducing the bitrate of the incoming video by around 50%. In comparison to full re-encoding, an average of 3.9% excessive bitrate is encountered with an average PSNR drop of 0.1 dB. Since this is the first work to report on MPEG-2 to HEVC video transcoding then the reported results can be used as a benchmark for future transcoding research.

Keywords: Video transcoding; HEVC video coding; machine learning.

1. Introduction

Work on video transcoding started around two decades back. One of the first papers to report video transcoding proposed a H.261 single motion-compensation loop transcoder for bitrate reduction [1]. Soon after, MPEG-2 transcoding was proposed for bitrate adaptability [2]. Additionally, with the advent of heterogeneous communication networks and the variety of end-systems, MPEG-2 to H.261 and H.263 was proposed [3], likewise MPEG-2 single layer to multilayer transcoding was proposed for video broadcasting [4]. Video transcoding become more in demand as a codec interoperability tool when new codecs started to appear. Worth noting is the MPEG-2 to H.264/AVC transcoder [5], Wyner-Ziv Video into H.263+ transcoder [6] and H.264/AVC into wavelet-based scalable transcoder [7]. In an attempt to reduce the complexity if video transcoding, [8] proposed a rule-based machine learning approach for transcoding MPEG-2 into H.264/AVC. Noticeable computational time reductions are reported at the expense of moderate bitrate increase.

Apart from bitrate/resolution adaptability and codec interoperability, the use of video transcoding has been reported for adding error resiliency for pre-encoded videos [9]. Video transcoding is also used for data embedding as reported in [10].

Nowadays, with the advent of the new HEVC codec which is being developed by ISO MPEG and ITU-T VCEG [11], a need has emerged for a MPEG-2 to HEVC video transcoder. The MPEG-2 video content is abundant as it used in TV services and DVD videos. One drawback of MPEG-2 video is its compression efficiency in comparison to more recent video codecs like H.264/AVC and HEVC. One approach to making use of the existing MPEG-2 video is therefore to transcode it to an efficient video codec such as HEVC. By doing so, the file sizes of MPEG-2 video can be reduced by half whilst maintaining its quality as shown in this paper.

MPEG-2 to HEVC transcoding is a challenging task for a number of reasons. For instance, a typical size of a Coding Unit (CU) in HEVC is 64x64 whilst that in MPEG-2 is 16x16. Additionally, in HEVC P-frames can use more than one reference frame as a source of prediction, whereas in MPEG-2, P-frames are restricted to one reference frame. In HEVC, the smallest partition size can be 4x4 which is not supported by MPEG-2. All such differences make it difficult to reuse the incoming MPEG-2 motion parameters in the coding of the outgoing HEVC video stream. In this work a different approach is proposed in which the incoming MPEG-2 coding information is mapped into the splitting depth of the CUs as explained in the next section. The proposed work is unique in a number of aspects. First, it is the first to report on MPEG-2 to HEVC video transcoding. Second, it uses video-content dependent machine learning to speed up the transcoding process.

The rest of the paper is organized as follows. Section 2 presents the proposed transcoding architecture with the use of linear discriminant functions as a machine learning tool. In Section 3 the experimental setup and results are presented. Section 4 concludes the paper.

2. Proposed transcoding solution

As mentioned in the introduction, one advantage of transcoding MPEG-2 video into HEVC is to reduce the file size of compressed videos. Clearly, reducing the video size depends on a number of factors, the most obvious of which is the QP value of the HEVC encoder. An example of bitrate saving using a QP value of 30 is shown for a number of test sequences in Table 1. The results in the table are generated using a re-encoding approach as opposed to transcoding. That is, the incoming MPEG-2 video is fully decoded and the reconstructed images are then fully encoded using HEVC.

Sequence	MPEG-2 bitrate (Kbit/s)	HEVC bitrate (kbit/s)	Bitrate reduction (%)	Loss (dB)
Basketball 832x480(50Hz)	2462.2	1476	40.1	0.3
PartyScene 832x480(50Hz)	8756.1	4494	48.7	0.6
BQMall 832x480(60Hz)	3933.7	2003.7	49.1	0.6
RaceHorses 832x480(30Hz)	3371.6	1811.8	46.3	1.1
Vidyo1 1280x720(60Hz)	1312.6	663.9	49.4	0.0
Average			**46.7**	**0.5**

Table 1. MPEG-2 bitrate reduction when re-encoded to HEVC.

In the table, an average bitrate reduction of 46.7% is reported. It is also shown that the reduction is associated with an average drop in PSNR of 0.5 dB. Clearly, this quality loss can be decreased at the expense of bitrate reduction if the QP of the outgoing HEVC video is reduced. In video transcoding it is understood that the quantization error of the incoming MPEG-2 video cannot be recovered and it contributes to this moderate loss in reconstruction quality.

In the HEVC, the coding unit (CU) plays a similar role to the macroblock in the MPEG-2. However, in the HEVC, the largest CU size is an encoder parameter, but it is typically chosen as 64x64. A CU can be recursively split in four smaller CUs, until the minimum size of 8x8 is reached. Each CU is encoded independently and defines, among other encoding parameters, the type of prediction (inter, intra or skip), the prediction unit (PU), and if the CU will be split or not. If the CU is split, it is said that the region is encoded using a higher depth. Thus, the 64x64 CU is said to be at depth 0, the 32x32 is at depth 1, and so on. If the inter prediction is used, the CU can be further partitioned in different PUs. The codec offers 8 partition sizes for the PU: 2Nx2N, 2NxN, Nx2N, NxN, where 2N is the size of the CU, as well as the asymmetric partitions. For instance, for a 64x64 CU, the 2NxN partition would refer to two 64x32 partitions.

The HEVC encoder thus decides on how to encode a CU using rate-distortion optimization. The encoder recursively computes the cost for the CU at depth *n* and for the four sub-CUs at depth *n+1*, and then decides for the one that minimizes the cost. In addition to deciding on the depth level, for each CU, all PU sizes are tested. In this work, it is noticed that , typically, at least 50% of the computation time of the encoder is spent on determining the depth level of the CUs. Hence, in video transcoding to HEVC, if the transcoder can map the coding information of an incoming MPEG-2 video to the splitting depth of the outgoing CUs then a minimum speed up factor of 2 can be achieved.

To achieve this goal, we propose the transcoding architecture of Figure 1. The main idea is to formalize the mapping as a classification problem. Four classes are used according to the depth of the CUs. Thus, the transcoder would attempt to predict the depth of the current CU, and then encode that CU using the predicted depth, but testing all PUs for that depth. The incoming MPEG-2 macroblocks are organized into the largest CU sizes, that is, 4x4 macroblocks which corresponds to 64x64 pixels. The coding parameters and motion information of the reorganized macroblocks are used to generate feature vectors which can be mapped into one of the four depth classes.

It is shown in the proposed transcoder of Figure 1 that a variable number of incoming frames can be used for model generation. During which, the transcoder is running in re-encoding mode in which the MPEG-2 video is fully decoded and fully encoded using HEVC. Once the model weights are generated, a switch between re-encoding and transcoding takes place. The advantage of this arrangement is twofold. First, the model weights are video content-dependent, that is, the weights are generated based on the same video being transcoded, thus avoiding a ‘one size fits all’ model generation approach. The second advantage is that the model can be retrained at any point during the transcoding process. For instance, if an abrupt or gradual scene cut is detected then the model can be regenerated. One approach to detect such scenarios is to observe the percentage of intra-coded macroblocks or sudden changes in allocated bitrate in the incoming MPEG-2 video. Surely many other approaches can be used as reported in [12] and [13].

A relevant approach to the proposed training solution, although reported in a different context, is the computational complexity reduction of H.264/AVC mode selection [14]. It was proposed to apply full mode search to a subset of MBs in a given frame. The dominant modes are then stored for later use. The rest of the MBs are coded with a sub-optimal mode decision using dominant modes only.

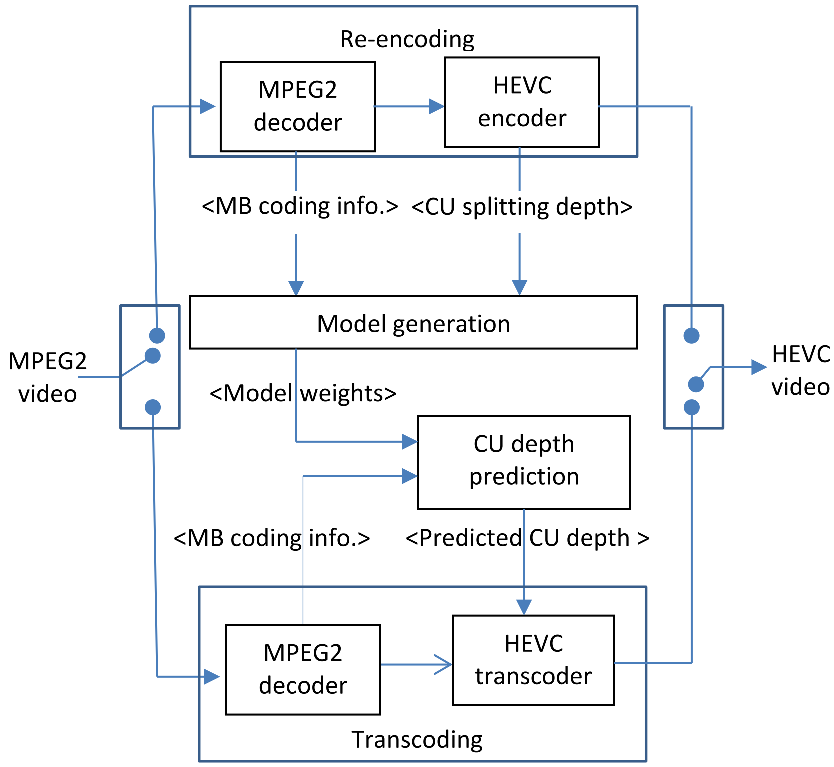

Figure 1. Proposed MPEG-2 to HEVC transcoding architecture.

The feature vectors that are extracted from the incoming MPEG-2 video are macroblock-based. The feature variables and their lengths are summarized in Table 2. Bear in mind that 4x4 MBs correspond to 64x64 pixels which is the size of the largest HEVC CU. The feature variables 1 to 3 are applicable to inter-coded macroblocks. If an incoming macroblock is intra-coded then a nil motion vector is used and the macroblock type is recorded in feature variable at index 4.

Feature index	Description	Feature length
1	Variance of the motion vectors of a square block of 16 MBs. This information corresponds to a CU at depth 0.	1
2	Variance of the motion vectors of each square block of 4 MBs in the block of 4x4 MBs. This information corresponds to a CU at depth 1.	4
3	The raw motion vector values of a square block of 16 MBs.	32
4	MB type in a 4x4 block of MBs	16
5	MB Coded Block Pattern (CBP) in a 4x4 block of MBs.	16
6	Number of coding bits per MB in a 4x4 block of MBs	16
7	Variance of pixel intensity per MB in a 4x4 block of MBs	16

Table 2. Summary of extracted feature variables from the incoming MPEG-2 video.

In this work, the mapping between the feature vectors and the splitting depth is performed using a linear classifier [15]. Once the feature variables are identified, the classification problem can be formalized as follows. Denote by $\mathbf{X}_i = [\mathbf{x}_{i,1}\ \mathbf{x}_{i,2}\ \cdots, \mathbf{x}_{i,N_i}]^T$ the sequence of feature vectors corresponding to the HEVC splitting depth of class i, where $\mathbf{X}_i$ is a $N_i \times M$ matrix, and M is the dimensionality of the feature vector. Concatenating all of the individual $\mathbf{X}_i$ matrices results in a matrix for all K classes such that $\mathbf{X} = [\mathbf{X}_1\quad \mathbf{X}_2\quad \cdots\quad \mathbf{X}_\mathbf{K}]^T$. For each HEVC splitting depth of class i, the training procedure computes an optimum weight vector by minimizing the distance between the HEVC splitting depth vector $\mathbf{y}_i$ and a linear combination of the training feature vectors $(\mathbf{X}\mathbf{w}_i)$ such that

$$\mathbf{w}_i^{opt} = \arg\min_{\mathbf{w}_i} \left\| \mathbf{X}\mathbf{w}_i - \mathbf{y}_i \right\|_2 \qquad (1)$$

The HEVC splitting depth vector for the i^{th} class, $\mathbf{y}_i$, is a column vector containing ones and zeros. That is, $\mathbf{y}_i = [\ \mathbf{0}_{N_1}, \mathbf{0}_{N_2}, \ldots, \mathbf{0}_{N_{i-1}}, \mathbf{1}_{N_i}, \mathbf{0}_{N_{i+1}}, \ldots, \mathbf{0}_{N_k}\ ]^{\mathrm{T}}$. The minimizing of the objective function of Equation (1) results in:

$$\boldsymbol{w}_i^{opt} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\boldsymbol{y}_i \qquad (2)$$

To classify a sequence of N_c MPEG-2 feature vectors $\mathbf{X}_c$, we are required to determine its class *c* as one of the enrolled classes in the set {0,…, K} where K = 3. This is done by evaluating the output sequences against all K models $\{\mathbf{w}_i^{opt}\}$ to obtain a set of score sequences $\{\mathbf{s}_i\}$ such as

$$\mathbf{s}_i = \mathbf{X}_c \mathbf{w}_i^{opt} \qquad (3)$$

The elements of the sequence $\mathbf{s}_i$ represent the scores of each MPEG-2 feature vector in the vector sequence $\mathbf{X}_c$. The class of the sequence $\mathbf{X}_c$ is determined by maximizing $\{g(\mathbf{s}_i)\}$ such as

$$c = \arg\max_i (g(\mathbf{s}_i)) \qquad (4)$$

where g is a function that outputs the mean of $\mathbf{s}_i$ such as

$$g(\mathbf{s}_i) = \frac{1}{N_c}\sum_{j=1}^{N_c} s_{i,\ j} \qquad (5)$$

Such a supervised learning technique is suitable for the proposed transcoder for a number of reasons. First, in the transcoding architecture of Figure 1, the re-encoding mode makes the supervised learning permissible because it facilitates access to both MPEG-2 feature vectors and corresponding CU depths of the HEVC encoder. Second, the computation of the model weights in Equation 2 is non-iterative and can be carried out at the transcoder. Lastly, in the transcoding mode of Figure 1, the classification can be performed mainly using a simple dot product operation as shown in Equation 3. Hence, the aforementioned linear classifier is a suitable choice for the proposed transcoder.

3. Expermental results

In the following experiments, five video test sequences are used. Namely; BasketballDrill (832x480, 50Hz), PartyScene (832x480, 50Hz), BQMall (832x480, 60Hz), RaceHorses (832x480, 30Hz) and Vidyo1 (1280x720, 60Hz). The sequences are MPEG-2 coded with IPPPPP… structure and a variable bitrate with a quantization step sizes of {12,15,20,23} out of 31. The HEVC coder uses the following corresponding set of QPs {25,27,29,30}. The QPs are chosen such that the reduction in bitrate is in the range of 45% to 50%. The coding structure is again IPPPPP… with 4 reference frames. The maximum CU size is set to 64x64. The asymmetric motion partitions tool and the adaptive loop filter tool are both enabled. HEVC reference software HM6.0 is used [16].In all cases, the HEVC uses the default fast motion estimation (a modified EPZS) and fast mode decision.

The first 25 frames of each sequence are used for model generation. A switch from re-encoding to transcoding is then applied.

The RD curves for the five test sequences are shown in Figure 2. The RD curves presented are for the MPEG-2 encoding results, the full HEVC re-encoding results and lastly, the proposed transcoding results. The comparison is valid because full re-encoding presents the best results that can be achieved in terms of both rate and distortion. It is shown in the figure that the proposed transcoding solution results in a moderate drop in PSNR and a slight increase in bitrate. Bear in mind that the maximum PSNR that can be reached is limited by the quality of the reconstructed MPEG-2 video. The figure also shows that transcoding MPEG-2 video into HEVC results in a considerable bitrate decrease at the expense of a slight drop in PSNR. In all cases, the average bitrate reduction is between 45% and 50% and the average drop in PSNR is 0.5 dB.

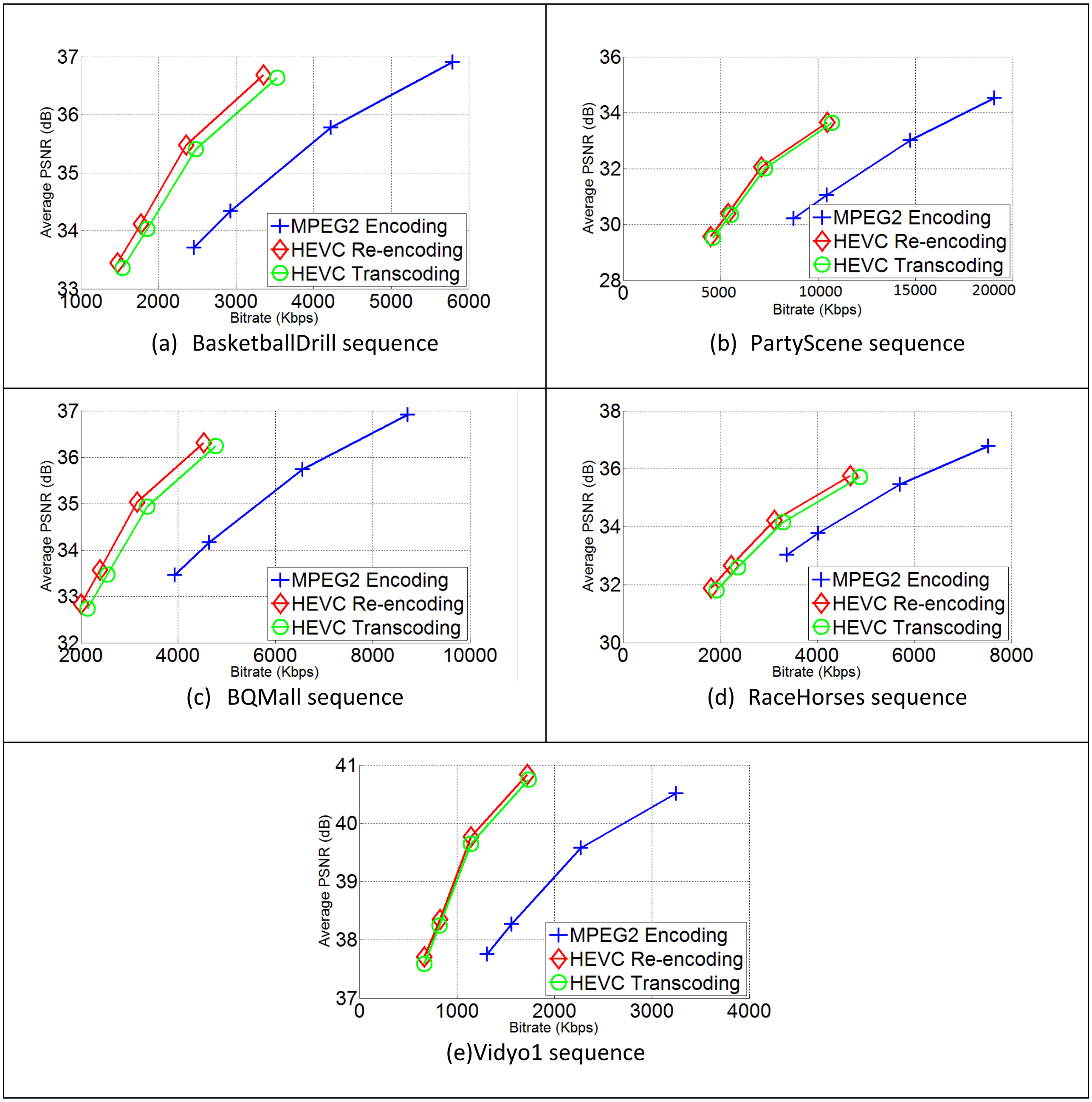

Figure 2. Rate-distortion curves for the proposed transcoding solution versus MPEG-2 encoding and full HEVC re-encoding.

We next examine the transcoding efficiency in terms of excessive bitrate and drop in PSNR. In part (a) of the figure, when compared to the full re-encoding case, the transcoding of the BasketballDrill sequence resulted in an average bitrate increase of 4.9%. The average drop in PSNR is 0.07 dB. In part (b) of the figure, the transcoding of the PartyScene sequence resulted in an average increase in bitrate of 2.5% and an average drop in PSNR of less than 0.04 dB. In part (c) of the figure, the transcoding of the BQMall sequence resulted in an average increase in bitrate of 6.4% and a drop in PSNR of 0.09 dB. Likewise, the transcoding of the RaceHorses sequence resulted in an average bitrate increase of 5.6% with a drop in PSNR of 0.1 dB. And lastly, an average bitrate increase of 0.2% incurred in the transcoding of the Vidyo1 sequence with a drop in PSNR of 0.1 dB as well.

In general the excessive bitrate caused by the transcoding solution is sequence dependent. However, the accuracy of the split depth prediction is also an important factor. Table 3 presents the accuracy of predicting the 4 split depths for both the training (the first 25 frames) and the testing sets (the rest of the 250 frames). It is shown in the table that the average training accuracy is 68% whilst the average testing accuracy is 62%. There is no direct relation between the classification accuracy and the excessive bitrate because the video content itself is a lurking variable in this case. For instance, although the excessive bitrate caused by transcoding the BQMall was the highest nonetheless, its classification accuracy is not the lowest.

	Classification rate	
Sequence	Training	Testing
Basketball	66%	62%
PartyScene	61%	58%
BQMall	69%	60%
RaceHorses	64%	59%
Vidyo1	78%	73%
Average	**68%**	**62%**

Table 3. Accuracy of split depth prediction

Lastly, the transcoding speed up factors are presented in Table 4. The speed up factors are computed in comparison to the HEVC full re-encoding. The computations are carried out on an Intel® Core ™ i5 CPU @ 2.4GHz. The OS is 64-bit Windows 7 Enterprise N. The results in the table show that the transcoding speed up factor is sequence dependent. However all speed up factors range roughly between 2 and 3.

	Transcoding Speedup				
QPs {MPEG2,HEVC}	Basket	Party	Mall	Horses	Vidyo1
{12,25}	2.33	1.83	2.24	2.13	2.91
{15,27}	2.32	1.92	2.26	2.23	3.12
{20,29}	2.55	2.01	2.41	2.27	3.18
{23,30}	2.59	2.09	2.46	2.77	3.26
Avg	**2.45**	**1.96**	**2.34**	**2.35**	**3.12**

Table 4. Transcoding speed-up factor.

4. Conclusion

A MPEG-2 to HEVC transcoder was proposed in this paper. The transcoder switches between full re-encoding and transcoding modes. The transcoder makes use of the re-encoding mode to map the incoming MPEG-2 macroblock coding parameters into the splitting depth of HEVC CUs. As such, the model weights are content-dependent and vary from one video sequence to the other. The model weights are then used to predict the splitting depth of the HEVC CUs which resulted in a computational speed up factor of up to 3. The transcoder was shown to reduce the bitrate of the incoming MPEG2 video by around 45%~50% with an average 0.5dB loss in PSNR. When compared to full re-encoding the transcoder resulted in an average 3.9% excessive bitrate with 0.1dB loss in PSNR.

Acknowledgement

This work was carried out during the academic visit of Tamer Shanableh to the Multimedia and Vision Lab in the school of Electronic Engineering and Computer Science at Queen Mary, University of London. The visit is conducted under a sabbatical award from the American University of Sharjah for the Spring 2012 semester.

References

[1]D. G. Morrison, M. E. Nilsson, and M. Ghanbari, “Reduction of the bit-rate of compressed video while in its coded form,” in Proc. Sixth Int. Workshop Packet Video, Portland, OR, September 1994.

[2] P. Assuncao, M. Ghanbari, “A frequency-domain video transcoder for dynamic bit rate reduction ofMPEG-2bitstreams,” IEEE Transactions on Circuits Syst. Video Technology, 8 (1998), pp. 953-967

[3] T. Shanableh, M. Ghanbari, "Heterogeneous video transcoding to lower spatio-temporal resolutions and different encoding formats," IEEE Transactions on Multimedia, 2(2), pp.101-110, Jun 2000

[4] T. Shanableh and M. Ghanbari, “Multilayer Transcoding with format portability for multicasting of single-layered video,” IEEE Transactions on Multimedia, 7(1), 1-15, February 2005

[5] T. Qiang and P. Nasiopoulos, "Efficient Motion Re-Estimation With Rate-Distortion Optimization for MPEG-2 to H.264/AVC Transcoding," IEEE Transactions on Circuits and Systems for Video Technology, 20(2), pp.262-274, February 2010

[6] E. Peixoto, R.L. de Queiroz and D. Mukherjee, "A Wyner-Ziv Video Transcoder," IEEE Transactions on Circuits and Systems for Video Technology, 20(2), pp.189-200, February 2010

[7] E. Peixoto, T. Zgaljic and E. Izquierdo, "Transcoding From Hybrid Nonscalable to Wavelet-Based Scalable Video Codecs," IEEE Transactions on Circuits and Systems for Video Technology, 22(4), pp.502-515, April 2012

[8] G. Fernandez-Escribano, H. Kalva, J.L. Martinez, P. Cuenca, L. Orozco-Barbosa and A. Garrido, "An MPEG-2 to H.264 Video Transcoder in the Baseline Profile," IEEE Transactions on Circuits and Systems for Video Technology, 20(5) , pp.763-768, May 2010

[9] T. Shanableh, T. May and F. Ishtiaq, "Error Resiliency Transcoding and Decoding Solutions Using Distributed Video Coding Techniques", Signal processing: Image communications, 23(8), September, 2008.

[10] T. Shanableh, "Matrix Encoding for Data Hiding Using Multilayer Video Coding and Transcoding Solutions," Signal processing: Image communications, accepted for publication.

[12] "Joint Call for Proposals on Video Compression Technology", ISO/ IEC JTC1/SC29 /WG11/ N11113, ITU-T Q6/16 document, VCEG-AM91, Jan. 2010.

[12] D. Lelescu, D. Schonfeld, "Statistical sequential analysis for real-time video scene change detection on compressed multimedia bitstream," IEEE Transactions on Multimedia, 5(1), pp. 106- 117, March 2003

[13] S. Kwang-deok, P. Seong, J. Soon-heung, "Wipe scene-change detector based on visual rhythm spectrum," IEEE Transactions on Consumer Electronics, 55(2), pp.831-838, May 2009

[14] T. da Fonseca, R. de Queiroz and D. Mukherjee, "Complexity-scalable H.264/AVC in an IPP-based video encoder," IEEE International Conference on Image Processing (ICIP), 17th, pp.2885-2888, 26-29 September 2010

[15] T. Shanableh and K. Assaleh, "Feature modeling using polynomial classifiers and stepwise regression," Neurocomputing, 73(10-12), June 2010

[16] I. K. Kim, K. Sugimoto, K. McCann, B.Bross, W.-J. Han, J.-R. Ohm and G. J. Sullivan, "HM6: High Efficiency Video Coding (HEVC) Test Model 6 (HM 6) Encoder Description", Doc. JCTVC-H1002, February, 2012.

© 2013 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other users, including reprinting/ republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted components of this work in other works.