A FAST DOWNSIZING VIDEO TRANSCODER FOR H.264/AVC WITH RATE-DISTORTION OPTIMAL MODE DECISION

Huifeng Shen[1†], Xiaoyan Sun[2], Feng Wu[2], Houqiang Li[3], Shipeng Li[2]
[1,3]University of Science & Technology of China, [2]Microsoft Research Asia
[1]shenhf@mail.ustc.edu.cn, [2]{xysun, fengwu, spli}@microsoft.com, [3]lihq@ustc.edu.cn

ABSTRACT

This paper focuses on the mode decision and motion selection problem when H.264/AVC video streams are transcoded in spatial resolution. A fast downsizing transcoding scheme is developed in which a new rate-distortion (R-D) optimal mode decision mechanism is presented for high speed transcoding as well as high coding efficiency. A model for estimating relative prediction errors is applied in this paper, which is free from computation of interpolation and SAD/SSD computation. Based on the selected model, a motion refinement within a distance of 1 pixel is performed after mode decision. Experimental results demonstrate that our method can significantly speed up the spatial resolution reduction process, while achieving high coding efficiency.

1. INTRODUCTION

Nowadays, more and more devices that can playback video content, such as cell phones, pocket PCs and portable media centers, etc., are going to be involved in multimedia applications. However, these devices are quite different in display resolutions and access bandwidths. How to effectively meet the different demands of various devices becomes very challenging. Video adaptation through downsizing is one of the most promising methods, which can provide dynamic adjustment of bit-rate and display resolution to meet various requirements of devices as well as heterogeneous networks.

Some studies [1]~[4] have addressed the problem on spatial resolution reduction for existing video coding standards including MPEG1/2/4 and H.261/H.263. These spatial transcoders can be generally classified into two categories: pixel-domain transcoders and DCT-domain transcoders. As pixel-domain motion compensations are always employed, pixel-domain transcoders are drift-free but relatively complicated; whereas DCT-domain transcodings perform directly in DCT domain to reduce complexity, which would in return lead to quality degradation due to drift errors.

The up-to-date H.264/AVC [5][6] video coding standard provides a significant improvement in terms of coding efficiency by involving complicated mode decision as well as motion estimation. It motivates the development of corresponding transcoding technology for multimedia adaptation and universal multimedia access. It can be observed that the modules including sub/quarter-pixel interpolation, in-loop de-blocking filter and intra prediction in H.264/AVC standard will cause severe drift error and thus prevent the spatial resolution reduction from performing in DCT domain. However, in the case of pixel-domain transcoders, rate-distortion (R-D) optimal motion search and mode decision [6] of H.264/AVC are the main bottlenecks of speeding up the transcoding process. Therefore, several methods have been presented in literatures focusing on the complexity reduction in obtaining the R-D optimal (RDO) motion and mode information [7]~[9]. Zhang et al. [7] propose a mode mapping method, which is time-saving but has about 3 dB losses. An area-weighted vector median motion estimation method is proposed in [8] and another method consisting of bottom-up motion re-estimation, rapid mode decision and adaptive motion refinement is proposed in [9]. These two methods can achieve a comparable coding efficiency with fully re-encoding methods. However, since multiple modes have to be evaluated for each macroblock in the motion search stage, they are still computationally complex. There are also potentials to further accelerate the spatial resolution reduction process.

In this paper, we propose an approach to enable fast R-D optimal spatial resolution reduction of the H.264/AVC-coded videos, where the resizing ratio is 2:1. In this method, the input motion and mode information are firstly down-scaled. Based on the downscaled motion and mode information, a fast R-D optimal mode decision mechanism is presented to determine the optimal mode and the corresponding motion vectors. Then, motion refinement is performed based on the selected mode and motion, where the search range is only 1 pixel. The main contribution of this paper lies in the development of the fast prediction error estimation method for the RDO framework and the corresponding transcoding scheme. Experimental results have demonstrated that our method can offer similar performance and nearly 3 times of speed up compared with the method described in [9].

The paper is organized as follows: Section 2 presents the down-conversion process of mode and motion information; Section 3 describes the R-D optimal mode

† The work is done while the author is with Microsoft Research Asia

decision method in detail; Section 4 shows the motion refinement stage. Experimental results are given in Section 5. Section 6 concludes this paper.

2. DOWNCONVERSION OF MODE AND MOTION

H.264/AVC exploits multiple macroblock partition modes for motion-compensated prediction. The luminance component of each macroblock can be partitioned into 16x16, 16x8, 8x16 or 8x8. Further, the 8x8 sub-macroblock can be partitioned into 8x8, 8x4, 4x8, or 4x4.

During the downsizing transcoding, the partition mode of each 8x8 block in the low resolution stream is achieved by zooming out the partition mode of the corresponding macroblock in the high resolution stream, as illustrated in Figure 1. Meanwhile, the motion vectors (MVs) of the low resolution stream are obtained by scaling down to the half value of the motion vectors of the corresponding block in the high resolution. If the macroblock in the high resolution stream is of INTRA mode, the MV of the corresponding block in the low resolution is generated from the median MV value of the adjacent blocks.

Notice that as a result of the down-conversion process, only 8x8 macroblock partitions are employed in the low resolution stream. Thus, the performance of the low resolution stream would be degraded due to the lack of large partition modes. To cope with this problem, R-D optimal mode decision should be performed to improve the coding efficiency of the low resolution stream.

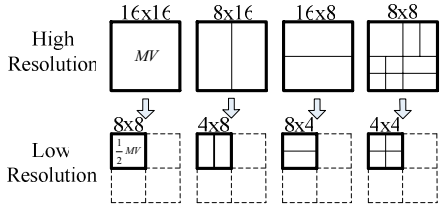

Figure 1. The down-converting process of modes and motions. One macroblock in the high resolution is corresponding to an 8x8 sub-macroblock in the low resolution, as denoted by bold lines. Partition modes in the low resolution are derived by zooming out the corresponding macroblock modes in the high resolution, as indicated by real lines.

3. PROPOSED MODE DECISION MECHANISM

In this section, we propose a fast R-D optimal mode decision method based on the down-scaled mode and motion information.

3.1. Conventional Mode Decision

In the conventional R-D optimal macroblock mode decision [6], the optimal mode is selected by minimizing

$$J = D_{REC} + \lambda_{MODE} R_{TOTAL}, \quad (1)$$

where D_{REC} denotes the distortion resulted from the reconstructed signals, R_{TOTAL} denotes the total bits, including motion bits and texture bits, which are spent in coding the macroblock. In general, it has to code the macroblock in every mode and thus becomes very time-consuming.

To reduce the complexity, the mode decision method could be modified to

$$\min J = \min(D_{DFD} + \lambda_{MOTION} R_{MOTION}), \quad (2)$$

where D_{DFD} denotes the difference (SSD/SAD) between the prediction signals and the original signals, that is, prediction errors, and R_{MOTION} denotes the bits which are spent on the motion vectors. However, the interpolation in the MC loop and SAD/SSD computation still lead to considerable cost of processing time.

3.2. Proposed Mode Decision Scheme

To reduce the complexity of mode decision, a new mode decision method is proposed to be free from the interpolation and SSD/SAD computation indicated in (2). Given the downscaled mode and motion information, the mode decision is formulated as

$$\min J = \min(\Psi + \lambda_{MOTION} R_{MOTION}), \quad (3)$$

Here R_{MOTION} is as the same as that in equation (2) and Ψ represents the relative prediction error described in the following subsection.

3.2.1. Prediction Error Estimation

In the case of downsizing transcoding, it is reasonable to assume that the bit-rate of the high resolution stream is high enough and the extracted motion vectors reflect the real motion. Moreover, it could be observed that the down-converted motion information also roughly reflects the motion in the low resolution stream. Here we use mv_H to denote the motion vectors derived from those of the high resolution stream by down-conversion. Let MV_L represent the set of the candidate motion vectors used in the low resolution stream. f_H is used to represent the prediction signal obtained by mv_H; and f_i is used to represent the prediction signal corresponding to mv_i ($mv_i \in MV_L$). Let f_{org} denote the original signal. Thus, the prediction error is estimated by equation (4), that is,

$$\Psi = (f_{org} - f_i)^2 - (f_{org} - f_H)^2 \approx (f_i - f_H)^2. \quad (4)$$

Here $(f_{org} - f_i)^2$ and $(f_{org} - f_H)^2$ denote the prediction errors when using mv_i, and mv_H, respectively. The above approximation is made based on that $(f_{org} - f_i)^2$ would be larger than $(f_{org} - f_H)^2$ in most cases and $|f_i - f_H|$ is commonly much larger than $|f_{org} - f_H|$.

As shown in (4), Ψ indicates the difference between two predictions resulting from two motion vectors. According to [10], it is highly related to the motion vector mean-squared error (MSE) of the two motion vectors and the power spectral density of the prediction signals. Thus, it has been formulated as

$$\Psi \approx (f_i - f_H)^2 \approx \varphi_x |\Delta mv_x|^2 + \varphi_y |\Delta mv_y|^2 + \varphi_{xy} |\Delta mv_x \Delta mv_y| \quad (5)$$

Here, $(\Delta mv_x, \Delta mv_y)^t$ denotes the difference between mv_i and mv_H, and

$$\varphi_x = \frac{1}{(2\pi)^2} \iint_{(-\pi,\pi]} S(\vec{\omega})\omega_1^2 d\omega_1 d\omega_2,$$
$$\varphi_y = \frac{1}{(2\pi)^2} \iint_{(-\pi,\pi]} S(\vec{\omega})\omega_2^2 d\omega_1 d\omega_2, \quad (6)$$
$$\varphi_{xy} = \frac{2}{(2\pi)^2} \iint_{(-\pi,\pi]} S(\vec{\omega})\omega_1\omega_2 d\omega_1 d\omega_2.$$

Here, $\vec{\omega} = (\omega_1, \omega_2)^t$ denotes the two-dimensional frequency, and $S(\vec{\omega})$ represents the power spectral density (PSD) of the prediction signal using mv_H. $S(\vec{\omega})$ can be approximated by the PSD of the reconstructed signal which is evaluated by the square of two-dimensional Fast Fourier Transform (2-D FFT) coefficients of each 4x4 reconstructed block. Furthermore, Discrete Cosine Transform (DCT) can also get the intensity of each discrete frequency point in the frequency domain, and get PSD through squaring the intensity. To further reduce the complexity, the DCT-like 4x4 integer transform [11] is introduced in our method to approximate the PSD for each 4x4 block. Obviously, the generation of Ψ is indeed free from the interpolation and SSD/SAD computation.

3.2.2. Bottom-Up Mode Limitation

In this subsection, the mode decision process is further accelerated by bottom-up mode limitation with regard to the downscaled mode and motion information. Firstly, the decision method described in [9] is adopted in our scheme to decide whether the current macroblock is coded as INTRA or not.

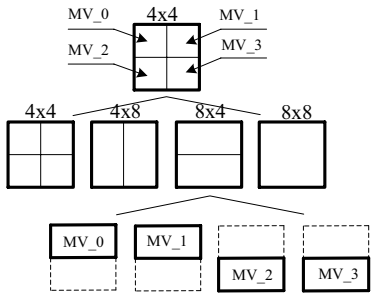

Figure 2. The bottom-up sub-macroblock mode decision.

If the macroblock is decided to be coded as INTER modes, sub-macroblock mode decision is first performed since the initial downscaled macroblock mode in the low resolution stream is the 8x8 mode. As illustrated in Figure 2, the sub-macroblock mode decision routes from the initial mode to larger-block-size modes, and the mode that results in minimal cost according to (3) is selected as the final sub-macroblock mode. For example, the sub-macroblock modes, 4x4, 4x8, 8x4 and 8x8, will be checked if the initial down-converted mode of current sub-macroblock is the 4x4 mode. Along with the mode decision, the motion selection is also performed following the criteria (3). As illustrated in Figure 2, when the cost of the 8x4 sub-macroblock mode is computing, the MVs of the two corresponding 4x4 blocks will be used in the cost calculation, respectively. The MV bringing on less cost is selected as the *MV* for this partition.

After sub-macroblock mode decision, the rest of modes, 8x16, 16x8 and 16x16, will be checked by the mode decision on the same principle to finally determine the partition mode of this macroblock.

4. MOTION REFINEMENT

However, using the initial MVs directly in downsizing transcoding will lead to significant efficiency loss. Therefore, motion refinement is introduced after the mode decision to retain good coding efficiency. In our scheme, initial MV obtained by mode decision is selected as the search center of the motion refinement. The search range of the motion refinement can be variable to pursuit different coding performance. In our scheme, as the initial MVs are obtained by the proposed R-D mode decision, the initial MVs are trustworthy so that the search range is only 1 integer pixel. Finally, the MVs of the macroblock are determined according to (2).

5. EXPERIMENTAL RESULTS AND ANALYSIS

Experiments have been done to illustrate the effectiveness of the proposed downsizing transcoding method. CIF test sequences, Foreman, News, Mobile and Football, are utilized in the experiments. The streams at CIF resolution are generated by H.264/AVC encoder with QP 22. The first frame is coded as I frame and the others P frames. Only one reference frame is used. The simulation platform is WIN.XP running on P4 3.0GHz with 1GB RAM.

Two other downsizing transcoding schemes are evaluated in the experiments. In *Full Search* method, the high resolution stream is decoded firstly; then the resulting CIF video is down-sampled to QCIF video; finally the QCIF sequence is re-encoded by H.264/AVC encoder with full-scale motion search whose range is 16. Furthermore, the method proposed by Li et al. in [9] has been implemented for comparison. To the authors' best knowledge, it could be regarded as one of the best schemes in terms of good coding efficiency and high speed transcoding.

The performance comparisons of the test sequences are shown in Figure 3 and Figure 4. *Proposed_Real* means that the prediction errors used in the mode decision is obtained by computing SSD; *Proposed_Estimated* means that the prediction errors used in the mode decision are estimated using the formula (5). It can be seen that, for News sequence, our method achieves the same coding efficiency compared with *Li's method*; for the other sequences, there is a coding efficiency loss up to 0.5 dB in comparison with *Li's method*. The performance gap between our method and the *Full Search* is limited to 1dB. However, as given in Figure 4, our scheme has the fastest speed, which can speed up the

transcoding process to 3 and 15 times compared with *Li's Method* and *Full Search*, respectively.

It can be seen that our scheme provides best performance in terms of high coding efficiency as well as high speed transcoding. Compared with other schemes, the proposed method succeeds in the R-D optimal mode decision so that only one mode for each macroblock needs to do motion refinement. Moreover, the search range in our method is only 1, while the search range is averagely 3 in *Li's method*. Though our R-D optimal mode decision would be cost-plus, the complexity of the mode decision is reduced significantly by the proposed prediction error estimation method.

6. CONCLUSIONS

In this paper, a fast downsizing transcoding scheme is proposed in which a new R-D optimal mode decision method is presented to speed up the process and maintain high coding efficiency. In the proposed transcoding scheme, the initial motions and modes are achieved by down-conversion of those of the high resolution stream. Based on these mode and motion information, our proposed R-D optimal mode decision decides the final mode and selects a coarse motion as well. Then, the motion refinement is performed with regard to the final mode and coarse motion. Experimental results demonstrate that our method can greatly speed up the spatial transcoding process and maintain the good coding efficiency.

7. REFERENCES

[1] A. Vetro, C. Christopoulos, and H. Sun, "Video Transcoding Architectures and Techniques: An Overview", IEEE Signal processing magazine, March 2003.

[2] J. Xin, C. –W. Lin, and M. –T. Sun, "Digital Video Transcoding", Proceedings of the IEEE, Vol. 93, No. 1, Jan 2005

[3] P. Yin, A. Vetro, B. Liu and H. Sun, "Drift Compensation for Reduced Spatial Resolution Transcoding", IEEE Trans. on Circuits and Systems for Video Technology, Vol. 12, Issue 11, pp. 1009-1020, Nov. 2002

[4] H. Sun, L. –P. Chau, "An Efficient Arbitrary Downsizing Algorithm for Video Transcoding", IEEE Trans. on Circuits and Systems for Video Technology, Vol. 14, No. 6, Jun. 2004

[5] ITU-T and ISO/IEC JTC1, "Advanced Video Coding for Generic Audiovisual Services", ITU-T Recommendation H.264 – ISO/IEC 14496 AVC, 2003

[6] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-Constrained Coder Control and Comparison of Video Coding Standards", IEEE Trans. on Circuits and Systems for Video Technology, Vol. 13, No. 17, July 2003.

[7] P. Zhang, Y. Lu, Q. Huang, W. Gao, "Mode Mapping Method for H.264/AVC Spatial Downscaling Transcoding", International Conference on Image Processing, 2004

[8] Y. –P. Tan, H. Sun, "Fast Motion Re-Estimation for Arbitrary Downsizing Video Transcoding Using H.264/AVC Standard", IEEE Trans. on Consumer Electronics, Vol. 50, No. 3, Aug. 2004

[9] C. -H. Li, C. -N. Wang, and T. Chiang, "A Fast Downsizing Video Transcoder Based on H.264/AVC Standard", PCM 2004, LNCS 3333, pp. 215-223, 2004

[10] A. Secker and D. Taubman, "Highly Scalable Video Compression With Scalable Motion Coding", IEEE Trans. on Image Processing, Vol. 13, No. 8, August 2004.

[11] H. S. Malvar, A. Hallapuro, M. Karczewicz, L. Kerofsky, "Low-Complexity Transform and Quantization in H.264/AVC", IEEE Trans. on Circuits and Systems for Video Technology, Vol. 13, No. 7, Jul. 2003.

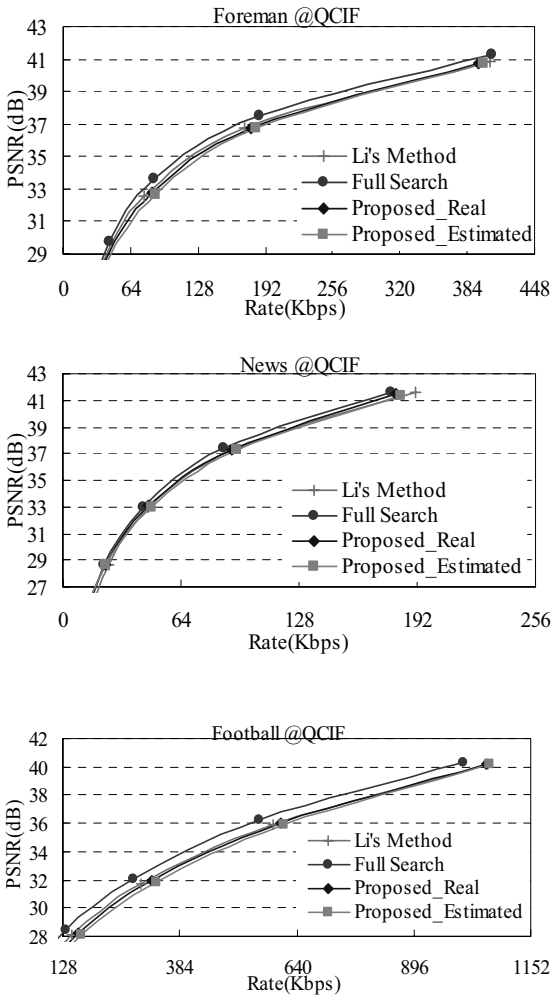

Figure 3. The coding performance comparison between different transcoders.

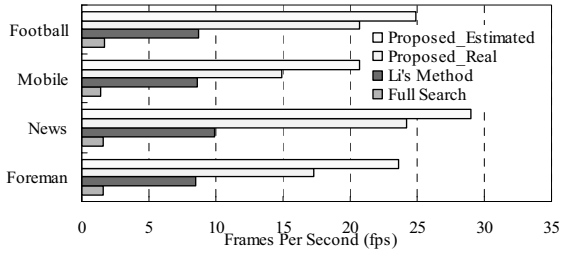

Figure 4. Comparison on transcoding speed between different transcoders.