COMPLEXITY SCALABLE MODE-BASED H.263 VIDEO TRANSCODING

Joseph Yeh   *Gene Cheung*

Dept of EECS
University of California at Berkeley

HP Labs, Japan
Hewlett-Packard Japan, Ltd

ABSTRACT

While bitrate-reduction video transcoding is a mature research topic, we introduce a new paradigm for the old problem: designing a complexity scalable transcoder where computation can be gracefully traded off with transcoded video quality. Complexity scalability is important when a time-varying number of real-time transcoding sessions must be maintained by the transcoder simultaneously. We present a complexity scalable transcoding algorithm for H.263 based on coding mode transition probabilities. Preliminary results are presented to show potential tradeoffs between computation and transcoded video quality in PSNR.

1. INTRODUCTION

While part of the networking community is investing heavily on research and development of content delivery networks (CDNs) — an overlay network infrastructure that reliably and efficiently caches popular contents across the Internet for large number of users, another part of the community is investigating the next evolution beyond simple content distribution. The IETF working group that concerns itself with this effort is called Open Pluggable Edge Services (OPES) [1], and its vision is a future of application component networks that provide services to content at network edge before delivery to content consumers.

An example of this type of network application is real-time transcoding of media content suitable for consumption by the requested client. Because the number of transcoding requests is time-variant, a good network component responsible for transcoding must be able to scale to a large number of service requests, even at the expense of resulting transcoding quality. In other words, a *complexity scalable* transcoder, with the ability to trade off transcoding performance with the number of concurrent transcoding sessions, would be desirable as a network component in the OPES framework. This is the topic we are investigating in this paper. In particular, among all possible transcoding operations, we are focusing on the rate-reduction of H.263 [2] video stream. Our methodology, however, is fairly general and can conceivably be applied to rate-reduction of other video coding standards such as MPEG4 and H.26L. In the case of the latter, there are many more modes available than in H.263, making our presented approach even more applicable.

While transcoding itself is a fairly mature research topic [3] [4] [5] [6], casting transcoding in the framework of complexity scalability is new. Like earlier work in complexity scalability under different contexts [7] [8], the challenge is to find the optimal tradeoff between compression quality and computational complexity given the desired resulting transcoded bit rate. Towards that goal, we have developed a complexity scalable transcoding algorithm called *probabilistic mode-based transcoding algorithm* that uses a *variable* amount of side information made available during the first stage of the transcoding process (decoding stage) for the second stage (re-encoding stage). The more side information used, the more computationally efficient the transcoding algorithm is, at the expense of lower transcoded video quality.

The next section discusses related work. Section 3 then presents our proposed algorithm. Section 4 presents results and Section 5 concludes the paper.

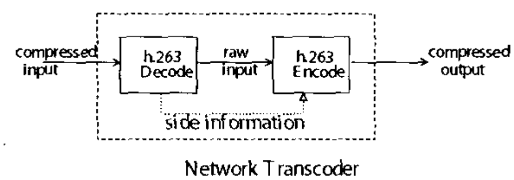

Fig. 1. Transcoder Architecture

2. RELATED WORK

2.1. Transcoding

A typical transcoding architecture, as shown in Figure 1, consists of two stages: a decoding stage that converts the compressed input bitstream to an uncompressed frame representation of the input, followed by a re-encoding stage that encodes the input to a lower bit rate by employing stronger quantization. *Side information*, made available during the first stage, is often used as an aid for the second stage.

Effectively using side information is important, as it is well known that while decoding a video stream requires little computation, encoding requires much more computation due to motion estimation. To save computation using side information, earlier work [3] [4] reuses motion vectors and coding modes of the original compressed input, available as side information during decoding, in the compressed output. As shown in [5] [6], however, the new quantization often means the original motion vectors (MVs) and modes are no longer optimal in a rate-distortion sense, due to changes in the reference pictures available to the decoder. To alleviate this problem, [5] and [6] propose to use a smaller search space to refine the original MVs. While we acknowledge the importance of motion re-estimation during re-encoding, we are doing so in a complexity scalable fashion, trading off computation with transcoding quality.

2.2. Complexity Scalability

The study of tradeoff among rate, distortion and complexity for vector search in multi-dimensional space has been examined in [7] and [8]. We leverage on the variable complexity algorithms presented in [8], where distortion is traded off with complexity in the case of motion estimation. For a macroblock (MB) in a current frame, indicated by vector $\vec{l}$, finding the best matched MB in a previous frame, indicated by $\vec{l}+\vec{v}$, means identifying the most suitable MV $\vec{v}^*$ in a search space S that minimizes some metric, like the commonly used sum of absolute difference (SAD):

$$\vec{v}^* = \arg\min_{\vec{v}\in S} SAD(\vec{l}+\vec{v},\, \vec{l}) \tag{1}$$

SAD is the sum of l^1 norm of the difference in pixel intensity between all corresponding pixels of two macroblocks (MBs):

$$SAD(\vec{l}+\vec{v},\vec{l}) = \sum_{\vec{\delta}\in B} |I_{t-1}(\vec{l}+\vec{v}+\vec{\delta}) - I_t(\vec{l}+\vec{\delta})| \tag{2}$$

where $I_t(\vec{v})$ is the pixel intensity value of pixel of frame t indicated by vector $\vec{v}$, and B contains the set of offset vectors δ's that points to all pixels for a MB with top left corner at $(0,0)$.

As discussed in [8], there are essentially two classes of methods to speed up the search in (1). *Fast Matching* speeds up the evaluation of SAD by reducing the number of pixels in B to a smaller B'. *Fast search* reduces the number of MV candidates by varying the size of S. We focus on fast search in this paper, but we differ from [8] in that in the case of transcoding, previously selected motion vector for each MB in the encoded bitstream can be used as an good initial motion vector estimate during motion re-estimation.

3. PROBABILISTIC MODE BASED ALGORITHM

The key to designing a complexity scalable transcoding algorithm is how to effectively utilize the side information made available during the decoding stage of transcoding to scale back computation during the re-encoding stage. The side information are the previously used MVs and coding modes for macroblocks in a P-frame.

3.1. Correlation in Motion Vectors

Before we discuss our complexity scalable algorithm, we first examine the correlation between the MV used in the previous encoding and the new optimal MV using exhaustive search in the entire search space after re-quantization for every macroblock (MB) in a P-frame, as done by Youn *et. al.*[6]. We accomplish that by recording the vector difference between the two vectors for every MB for 30 frames of QCIF-sized test sequences `MotherDaughter`, `Foreman`, `Coast`, and `Trevor`. The resulting differences are counted for all MBs, normalized to 1, and plotted in Figure 2 for difference pairs of quantizers (5,6) and (5,25). We first notice that the plots are fairly symmetric, and the peaks are centered around the origin as expected. We further notice that as the difference between the quantizers increases, the plots become more evenly distributed away from the center (the probability that a MV remains the same is truncated). Figure 3 shows the required search area to reach a given percentage of ideal MVs as a function of the difference between quantization factors, demonstrating a somewhat linear relationship between the two.

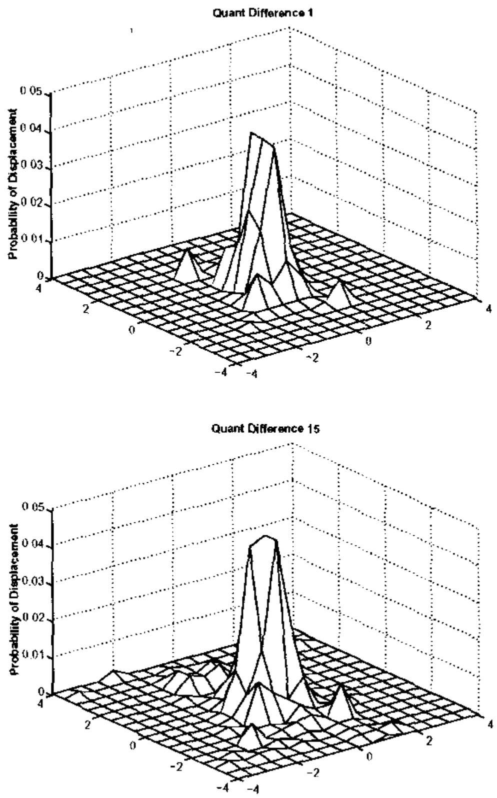

Fig. 2. Correlation between Previous and New MVs

What we can conclude is that a small refine search space S' centered around the previous MV is sufficient to capture the optimal MV, where S' depends on the difference between the quantization parameters.

3.2. Correlation in Coding Modes

While complex coding mode selection algorithms have been proposed [9] [10] so that modes of a group of MBs can be jointly and optimally selected, the complexity required for [9] and [10] remains high and is likely not practical for real-time transcoding. Instead, we focus on selecting modes on a MB-by-MB basis in a rate-distortion optimal manner using Lagrange multiplier:

$$m^* = \arg\min_{m\in\mathcal{M}} d_i(m) + \lambda r_i(m) \tag{3}$$

where $d_i(m)$ and $r_i(m)$ are the distortion and rate contribution for MB i given mode m respectively, and $\mathcal{M}$ is the set of modes available for each MB. The Lagrange multiplier λ can selected using one of many well-known methods such as [11].

For H.263 [2], $\mathcal{M}$ essentially has four modes: INTRA (intra-block coding), INTER (inter-block coding with 1 MV), INTER4V (inter-block coding with 4 MVs, one for each 8x8 sub-block) and SKIP (repeat block from previous frame). As with MV, the optimal

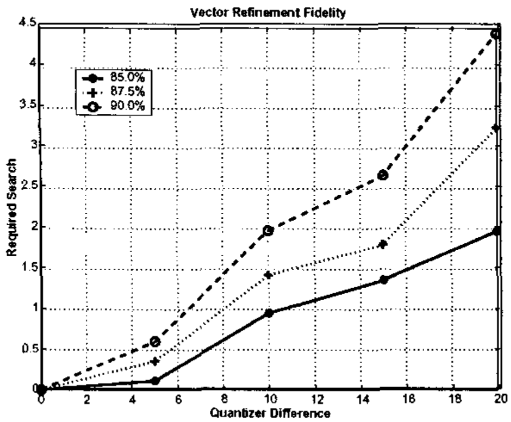

Fig. 3. Search Range Needed for Given Percentage of MVs Graph points are interpolated from observed data points.

coding mode for a MB will likely change after re-quantization. To empirically see how likely a MB changes from mode X to Y, we repeat the experiment in section 3.1 but collect statistics for coding modes instead, analyzing mode transitions incurred when the input bitstream to the transcoder has a certain quantization factor and the transcoder encodes the output bitstream with another quantization factor. The results, shown in Figure 4, seem to indicate little variation between transition probabilities for different combinations of original and output quantization factors.

3.3. Algorithm Development

Having established the correlation between previous and new MVs and the correlation between previous and new coding modes, we now discuss our proposed probabilistic mode-based complexity scaling algorithm. For each complexity $N \in \{0, \ldots, |\mathcal{M}|^2\}$, we do the following:

1. Given the quantizer pair (q_1, q_2), where q_1 is the quantizer used in the initial bitstream and q_2 is the quantizer used in the transcoded bitstream for the reference frame, select the appropriate MV search space $S_{q_2 - q_1}$ for motion re-estimation (1).
2. Mark the N entries with the N highest probabilities in the mode-mode table of Figure 4, where the probability of each entry of modes (M_x, M_y) is probability of the input bitstream mode M_x times the probability of the output bitstream mode M_y.
3. For each previous mode X, the mode set for the new mode is $\mathcal{M}_x$ containing all marked modes.
4. During encoding stage, for a given MB i with previous mode $\tilde{m}_i$, we solve (3) where mode set $\mathcal{M}$ is $\mathcal{M}_{\tilde{m}_i}$.

In step 4, because the mode set is constrained depending on the previous mode, we only need to perform motion estimation (1) when INTER or INTER4V are in the reduced mode set. Notice that when $N = 0$ and when $N = |\mathcal{M}|^2$, we get the two base cases on the distortion complexity curve: no mode re-selection and no motion re-estimation versus full mode re-selection and motion

previous mode w/ probability (quant = 5)	new mode	transition probability (quant = 15)	transition probability (quant = 25)
intra	intra	0.506	0.420
0.007	inter	0.284	0.222
	inter4v	0.185	0.259
	skip	0.025	0.099
inter	intra	0.071	0.055
0.427	inter	0.319	0.287
	inter4v	0.401	0.390
	skip	0.209	0.269
inter4v	intra	0.096	0.082
0.363	inter	0.250	0.229
	inter4v	0.544	0.539
	skip	0.110	0.151
skip	intra	0.002	0.002
0.203	inter	0.063	0.068
	inter4v	0.088	0.126
	skip	0.847	0.804

previous mode w/ probability (quant = 10)	new mode	transition probability (quant = 20)	transition probability (quant = 25)
intra	intra	0.267	0.229
0.062	inter	0.225	0.215
	inter4v	0.453	0.472
	skip	0.054	0.084
inter	intra	0.075	0.058
0.297	inter	0.365	0.350
	inter4v	0.393	0.402
	skip	0.167	0.190
inter4v	intra	0.076	0.066
0.364	inter	0.219	0.227
	inter4v	0.574	0.559
	skip	0.132	0.148
skip	intra	0.006	0.003
0.277	inter	0.070	0.075
	inter4v	0.118	0.134
	skip	0.806	0.788

Fig. 4. Correlation between Previous and New Coding Modes

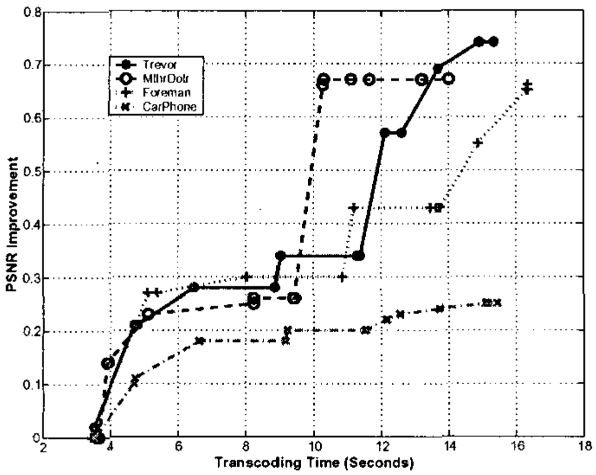

Fig. 5. Signal Fidelity versus Computation Time

re-estimation. When we need to consider mode transitions from INTRA to INTER or INTER4V, we execute a full motion search as opposed to a limited one based upon the quantizer pair. Since the amount of INTRA blocks are relatively few for the bitrates under consideration, this does drastically affect the overall complexity.

4. RESULTS

Figure 5 shows results of transcoding experiments performed on four sequences; `Trevor`, `MotherDaughter`, the panning part of `Foreman`, and `Carphone`. For each sequence, we transcode 61 frames of a 30 f/s 200-300 kpbs stream down to 80 kbps (30 kbps in the case of `MotherDaughter`). These different bitrates correspond to quantization factors of 3 to 5 in the original sequences to factors of 22 to 25 in the transcoded sequences. We run the transcoder at each complexity level, noting the total time taken to transcode the bitstream and the improvement in PSNR achieved over reusing all modes and motion vectors in the input bitstream. The simulations were done on a 550 Mhz Pentium III with 256 MB of main memory, and C-code was compiled with `gcc` with the `-O3` option. The reader should note that currently available processors run at up to six times the speed of our test platform, so there are many possibilities for real-time performance.

As expected, the time of computation increases as the complexity level increases. The PSNR also improves as the complexity level increases, although the total amount of increase over all the complexity levels varies with the type of sequence. `Carphone` shows the least amount of increase, since the sequence is relatively stationary compared to the others. In `Mother Daughter`, we observe a sudden jump in PSNR at complexity level 10, where the PSNR is about the same as it is at full complexity. This is because at that complexity level, macroblocks that are skipped in the original bitstream are allowed to be re-evaluated for motion vectors and residual data as opposed to just being skipped in the transcoded bitstream. We believe that this makes a great difference because reference frames in the transcoded sequence are so degraded in comparison to reference frames in the original sequence that residual data is much more necessary. In general, the probabilities presented in Figure 2 may not necessarily translate into equivalent opportunities for distortion reduction.

5. CONCLUSION AND FUTURE DIRECTIONS

In this paper, we show that video transcoding can be done in a *complexity scalable* fashion by using a probabilistic mode-based complexity scaling algorithm, which goes beyond motion refinement techniques by appropriately selecting a set of mode transitions to consider based upon prior estimations of the transition probabilities. We demonstrated its use in reaching complexity-performance points in between the extremes of redoing all mode decisions and motion vector searches and using all mode decisions and motion vectors from the incoming bitstream.

Future directions for improving our algorithm may include finding ways to combine the presented algorithm with probabilistic methods of varying the motion search as presented by Lengwehasatit and Ortega[8], and adaptive motion refinement techniques as presented by Youn *et. al.*[6], along with working on the overall computational efficiency of all complexity levels.

6. REFERENCES

[1] G. Tomlinson et. al., "A model for open pluggable edge services," July 2001, IETF Internet-draft: draft-tomlinson-opes-model-00.txt.

[2] ITU-T Recommendation H.263, *Video Coding for Low Bitrate Communication*, February 1998.

[3] P. Assuncao and M. Ghanbari, "A frequency-domain video transcoder for dynamic bit-rate reduction of mpeg-2 bit streams," in *IEEE Trans. on CSVT*, December 1998, vol. 8, no.8.

[4] S. Wee, J. Apostolopoulos, and N. Feamster, "Field-to-frame transcoding with temporal and spatial downsampling," in *ICIP*, October 1999.

[5] N. Bjork and C. Christopoulos, "Video transcoding for universal multimedia access," in *ACM Multimedia*, 2000.

[6] J. Youn, M. Sun, and C. Lin, "Motion vector refinement for high-performance transcoding," in *IEEE Trans. on Multimedia*, March 1999, vol. 1, no.1.

[7] K. Lengwehasatit and A. Ortega, "Complexity-distortion tradeoffs in vector matching based on probabilistic partial distance techniques," in *DCC*, 1999.

[8] K. Lengwehasatit and A. Ortega, "Computationally scalable partial distance based fast search motion estimation," in *ICIP*, 2000.

[9] T. Wiegand et. al., "Rate-distortion optimized mode selection for very low bit rate video coding and the emerging h.263 standard," in *IEEE Trans. on CSVT*, April 1996, vol. 6, no.2.

[10] G. Cheung, "Directed acyclic graph based mode optimization for h.263 video encoding," in *ICIP*, 2001.

[11] T. Wiegand and Bernd Girod, "Lagrange multiplier selection in hybrid video coder control," in *ICIP*, September 2001.