

A NEURAL NETWORK BASED TRANSCODER FOR MPEG2 VIDEO COMPRESSION

H.C. Fu, Z.H. Chen, Y.Y. Xu

Department of Computer Engineering
National Chiao Tung University
Hsin-Chu, Taiwan, ROC

C.H. Wang

Mentor Data Systems
Hsin-Chu, Taiwan, ROC

ABSTRACT

In this paper, we proposed a neural network method for the high efficiency requantization in the design of a transcoder. In our design, there are two types of video bitrate control in the proposed transcoder. One is the global adjusting of quantizer scales in which the adjusting is based on the complication of the whole frame, the other is the adaptive adjusting of quantizer scales, that the adjusting is the complication of the current macroblock. From our experimental results, the prototype transcoder can achieve desirable bitrate (1.5 Mbps) with an acceptable image quality. In addition, we constructed a video multiplexer for PPV or NVOD applications on the proposed transcoder.

1. THE INTRODUCTION

The digital video has been gradually used by most of the broadcasting media. In the commercial TV system, if we want to put more programs into a channel, we need to decrease the bitrate of the original video stream. The device or software that converts the video bitstream with different bitrates is called transcoder.

In [1], a video system was proposed to compress and to multiplex several video sequences into a commercial TV channel. The compressor at the encoder side used the coding level to control the video bitrate and video image quality. Without consider the temporal and local quantization needs, their compression efficient is quite limited.

Recently, MPEG2 standard has been widely adapted in most of video system. The bitrate of a normally compressed video is about 6 Mbits/sec. In general, a 6MHz bandwidth is equivalent to 27Mbps of bitrates. Without any further compression, a commercial TV channel or satellite transponder can allow 4 digital video programs. For NVOD application, this kind of channel utilization is still not desirable. In [4] [5], several bit allocation methods were proposed to minimize the distortion of MPEG2 video under the constant bitrate. In [6], three kinds of transcoder were proposed, but none of them can keep up with the transcoded time and the video quality. In this paper, we proposed a neural network method to achieve high efficiency requantization for MPEG2 video compression.

The rest of this paper is organized as follows. In Section 2, we introduce the transcoder and its designed issues. In Section 3, we show the implementation of the proposed transcoder. Some experimental results are showed in Section 4. A video multiplex system is proposed in Section 5 for some NVOD applications. We draw a brief conclusion in Section 6.

2. THE TRANSCODER DESIGN

As shown in Figure 1, a transcoder can generally be considered as a combination of a decoder and an encoder. First, an original bitstream with bitrate R_1 flow through the decoder to get the original video information, and then it enters the encoder for the desired R_2 .

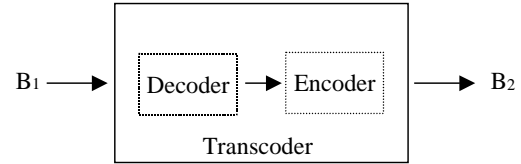


Figure 1: A transcoder can be a decoder followed by an encoder

In order to design an efficient transcoder system, we need to consider the following three issues: (1) How to satisfy the time constraints of a real time video broadcast system, (2) How to control the bitrate, and (3) How to maintain the video quality.

2.1. Time constraints of the real time system

Considering the various processes in the encoder and the decoder for MPEG2, we find that Discrete Cosine Transformation (DCT), Inverse Discrete Cosine Transformation (IDCT), and Motion Prediction and Compensation (MPC) are time consuming processing. Hence we will use the existing DCT, IDCT, and MPC results to avoid complicated computation in the transcoder design.

2.2. Bitrate control

Basically, the bitrate can be decreased by increasing the quantizer scale in the macroblock. Since the relation between the quantizer scale and the bitrate is not linear, one can not directly multiply the quantizer scale by the ratio of the desired bitrate to the original bitrate in order to produce the bitstream with the desired bitrate. In order to keep the output bitrate of transcoded bitstream equal to or below the desired bitrate, one can use the buffer fullness to ceiling the current output bitrate, so as to adjust the quantizer scale.

2.3. Maintain the quality

Because the transcoder processes the coded bitstream frame by frame, we need only to consider the frame quality. Basically, the

larger the quantization step size is, the lower the macroblock quality will be. Hence, we would like to adjust the quantizer scale of each macroblock carefully to make sure that the necessary quality of each image macroblock.

3. IMPLEMENTATION OF THE TRANSCODER

By considering the three issues discussed in last section, we proposed the transcoder as in Figure 2. The transcoder consists of a Variable Length Decoder (VLD), a De-quantizer (DQ), a Quantizer (Q), a Variable Length Coder (VLC), a Quantizer Scale Predictor (QSP), and a Delay Buffer (DB). B_1 is an input coded bitstream with bitrate R_1 , and B_2 is a output transcoded bitstream with bitrate R_2 . DB is used to delay several macroblock time in order to look ahead the video bitstream. QSP is used to predict the quantizer scale in order to decrease the bitrate of the video bitstream. After the video bitstream flowed through VLD and DQ, DCT coefficients can be obtained. Then the Quantizer uses DCT coefficients and the new quantizer scale predicted by QSP to produce the new quantized levels which is used by VLC to generate the coded video bitstream with the new bitrate. Because the transcoder, as shown on Figure 2, does not contain DCT processes in either decoder or encoder, the computing time of the transcoder is short enough to conform the time constraints of a real-time system. The most important element of proposed transcoder is QSP.

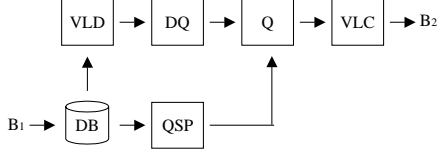


Figure 2: The architecture of the real time transcoder

If QSP can predict a comfortable quantizer scale of each macroblock, the transcoded bitstream will have the desired bitrate and the proper video quality.

3.1. The control of the bitrate

For a given coded frame, we can obtain the target number of bits in a transcoded frame by:

$$T = \frac{R_2}{R_1} \cdot B^{prev} \quad (1)$$

where R_1 is the bitrate of input coded frame and R_2 is the desired bitrate in the output transcoded frame. B^{prev} is the number of bits at input coded frame. The value of T is the target number of bits in output transcoded frame. We can also obtain the target number of bits in a transcoded macroblock as in Eq. (2).

$$T_j^{mb} = \frac{C_j}{C_1 + C_2 + \dots + C_m} T, 1 \leq j \leq m \quad (2)$$

where m is the number of macroblocks in a frame, and T is the target number of bits in the frame. C_j is the complexity estimation of the macroblock j in the frame which can be estimated by Eq. (3).

$$C_j = q_j \cdot B_j^{prev}, j = 1 \dots m \quad (3)$$

where q_j is the quantizer scale of the macroblock j , and B_j^{prev} is the number of bits in macroblock j in the input coded frame.

Because the nonlinear relation between the number of bits in a macroblock and the quantizer scale of a macroblock, we can not determine the new quantizer scale of the macroblock j directly using the target number of bits T_j . In order to adjust the quantizer scale to maintain the desired output bitrate, we used the buffer fullness coefficient to trace the output number of bits in current macroblock. The fullness coefficient d_j can be updated as:

$$d_j = d_{j-1} + B_{j-1}^{mb} - T_{j-1}^{mb}, \quad (4)$$

d_j is the size of fullness coefficient when the j th macroblock is going to be transcoded. B_{j-1}^{mb} is the actual number of bits which is output in the $(j-1)$ th macroblock, and T_{j-1}^{mb} is the desired number of bits which is output in the $(j-1)$ th macroblock. The initial fullness coefficient d_0 is assigned by:

$$d_0 = q_{seed} \cdot \frac{r}{31} \quad (5)$$

where q_{seed} is computed as Eq. (6) [7], and r is computed as Eq. (7).

$$q_{seed} = q_1 \cdot \exp\left(\frac{R_1 - R_2}{\beta}\right) \quad (6)$$

where q_1 is the quantizer scale of the first macroblock in the first frame, and β is the coefficient related to q_1 .

$$r = 2 \cdot \frac{bit_rate}{frame_rate} \quad (7)$$

As shown in Eq. (4), d_j is incremented when the actual number of bits B_{j-1}^{mb} exceeds the desired number of bits T_{j-1}^{mb} . d_j will continually be increased until B_{j-1}^{mb} is small than T_{j-1}^{mb} . Every time a frame is transcoded, its final fullness coefficient will be used to the initial fullness coefficient of the next frame.

The discussing above is focused on I-frame, however when the input coded frame is P-frame or B-frame, the calculation of its fullness coefficient can be achieved similarly.

3.2. Neural network based QSP

After the fullness coefficient and the target number of bits in each macroblock are obtained, QSP will predict the suitable quantizer scale of each macroblock. By considering the issues presented in Section 2.3, we see the following relation:

$$q_i^{optimal} = f(d_{i-1}, mb_i, MB_i^{neighbor}) \quad (8)$$

where $q_i^{optimal}$ is the optimal quantizer scale of macroblock i , d_{i-1} is the fullness coefficient, mb_i are the features of macroblock i , $MB_i^{neighbor}$ are the features of macroblocks which are the neighbors of macroblock i . Since the quantization function f in (8) is not linear, we propose to use a neural network to implement the quantizer scale predictor. As shown in Figure 3, the neural network quantizer scale predictor (NNQSP) is constructed by a 2-layer MLP. The output NNQSP is the normalized predictive quantizer scale of the current macroblock and the inputs are the current fullness coefficient (d_{i-1}), the target output number of bits (mb_i) in the current macroblock, and the target output number of bits ($MB_i^{neighbor}$) in the following macroblocks.

The training data for the NNQSP can be achieved by collecting the quantizer scale of each macroblock which were manually tuned

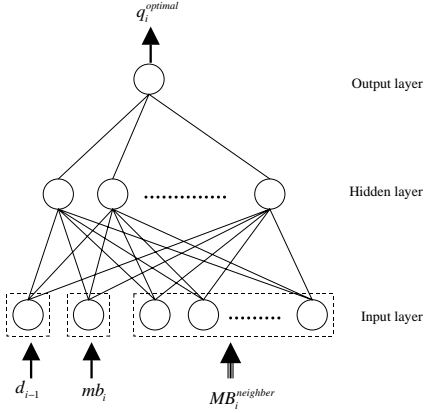


Figure 3: The structure of NNQSP

such that the actually output bitrate be the optimal output bitrate, and also the image quality to be acceptable.

After the network is trained, the quantizer scale of the current macroblock will be obtained easily by using the feature values of the current macroblock to the network. Since the prediction of quantizer scale value is a forward computing in NNQSP, the computing time can be constant, and within the limits for a real-time system.

4. EXPERIMENTAL RESULTS AND EVALUATION

We simulated the proposed transcoder on a Pentium processor based personal computer. The library of the MPEG2 encoder and the MPEG2 decoder is from Berkeley Multimedia Research Center [2]. The experimental MPEG2 bitstream is from Tektronix Corp. [3]

4.1. Processing speed evaluation

The processing time of two bitstreams tens_80 and tens_15 are shown in Table 1. The tens_80 and the tens_15 have same content but different bitrate. The bitrate of the tens_80 is 8 Mbits/s, and the bitrate of the tens_15 is 1.5 Mbits/s. Both of two bitstream contain 499 frames.

Table 1: The transcoding time and average frame rates of the transcoded bit streams (bitrate to be 1 Mbits/sec) from the video with same contents but different bitrates.

	Output bitrate (Mbits/s)	Transcoded time (second)	Average transcoded frame per second
tens_80	6	105	4.2
	4	96	4.6
	1.5	86	5.2
tens_15	1.0	21	21.4
	0.5	18	24.9
	0.2	17	26.4

Table 2 shows the transcoding time and average frame rates of the transcoded bit streams (bitrate to be 1 Mbits/sec) from the video with same contents but different bitrates.

Table 2: The transcoding time and average frame rate of transcoding a video stream with different bitrates to bit streams with a bitrate of 1 Mbits/s

	Original bitrate (Mbits/s)	Transcoded time (second)	Average transcoded frame per second
tens_120	12	102	4.4
tens_80	8	84	5.3
tens_40	4	53	8.5
tens_15	1.5	21	21.4

The transcoding time can be greatly reduced in half, if the transcoding program is optimized for processing efficiency.

4.2. Bitrate control

Figure 4 shows the time-bitrate relation for the benchmark video tens_080.m2v to be transcoded from bitrate 8 Mbits/s to 1.5 Mbits/s. In this figure, we can see that the prototype transcoder properly processes the input bitstream to the desired bitrate.

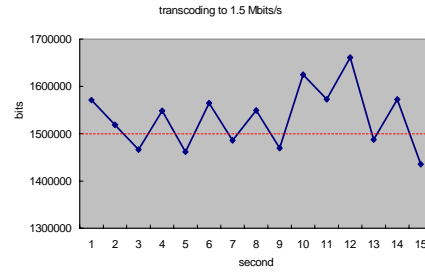


Figure 4: The time-bitrate relation for the transcoding of video tens_080.m2v at 1.5 Mbits/sec.

4.3. Image quality

In the Figures 5 and 6, two corresponding snapshots are captured from the original and transcoded bitstream tens_080.m2v. We can see that there are some distortion in the transcoded bitstream.

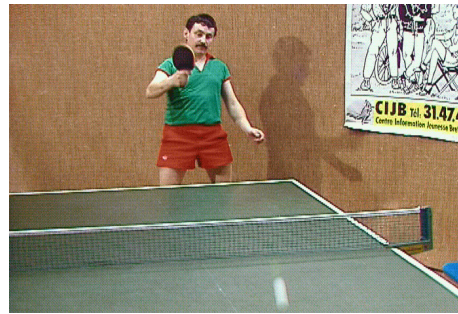


Figure 5: The input bitstream with bitrate 8 Mbits/s.



Figure 6: The output transcoded bitstream with bitrate 1.5 Mbits/s.

5. APPLICATIONS OF THE DESIGNED TRANSCODER

For a given video sequence, not all scenes within a video program, however, need to be encoded at the same, constant bit rate in order to achieve picture quality equal to that of the best rendered scenes. Easy scenes, such as conversations between people, require much fewer bits to encode than complex scenes, such as explosions. When the same, constant bit rate is assigned equally to all different types of scenes regardless of content, bandwidth and channel capacity is wasted.

In reality, a program needs far fewer bits than the constant ceiling to achieve uniformly high picture quality. The required bit rate varies considerably during the course of a program. The peaks in bitrate occur at the complex, fast-action scenes, and the valleys in bit rate occur during easy scenes. In typical video, complex scenes requiring large peaks occur very rarely. Clearly, if one can capitalize on this varying need for bit rate to eliminate any extra information redundancy to encode movies at an average of 1 Mbit/sec, thus achieving the ability to send 24 digital movies in the space of a single analog TV channel (cable, satellite, fiber, or wireless). When peaks occur at the same time, causing potential overload of channel capacity, the proposed multiplexer, as shown in Figure 7 process the actual encoded bitstream to reduce the peaks and hide potential artifacts in areas of the picture where they will go unnoticed. Bit rates for programs vary with content (news, sports, movies) as well, but the proposed system always produces a 6~8 x improvement in the number of channels that can be transmitted in a single analog TV channel in comparison to native MPEG2.

In Figure 7, the whole multiplexer can be implemented in a LAN based personal computer system. The Q-processor is a single board computer with RAM as I/O buffer and CPU as a transcoder. By plugging 3 single broad computers into a PC, the host computer (Q-Mux) can be programmed as a 3:1 stat-multiplexer by using a Ether network based LAM switch and a personal computer, a 8:1 Multiplexer can be implemented to construct a 24-video program multiplexer over a single analog TV channel.

With the channel efficiency of the proposed multiplexer, service providers can effectively offer hundreds of channels of Near Video-On-Demand (NVOD) service, enhanced PPV, and any services that can be created from compressed video stored on video servers, tape, or some other storage medium. The proposed video system allows the scheduling of stored movies in 24 start-time increments in a single analog television channel, giving an average access time of 2-1/2 minutes for NVOD movies using only one copy of the movie on disk.

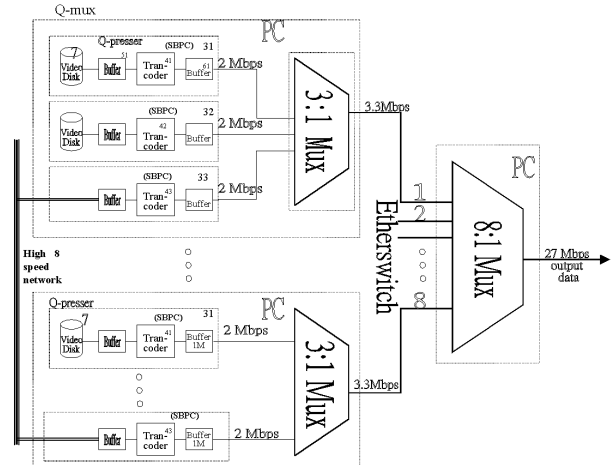


Figure 7: A personal computer based 24:1 video multiplexer over a 6MHz TV channel.

6. DISCUSSION

In this paper, we only adjust the quantizer scale to control the bitrate in a MPEG2 based system. The transcoder efficiency is very high, and the image quality after transcoding is quite acceptable. It is true that the transcoded image quality can not be as good as the original image, this is trade off between image quality and processing and bandwidth. In the future we intent to include the re-quantization in the motion compensation to achieve better dynamic pictorial quality.

7. REFERENCES

- [1] United states Patent, patent number: 5,216,503 Jun. 1993.
- [2] Available, <http://bmrc.berkeley.edu/ftp/pub/mpeg2/software/mpeg2v12.zip>.
- [3] MPEG-2 bitstream available, <ftp://ftp.tek.com/tv/test/streams/Element/MPEG-Video>.
- [4] Liang-Jin Lin and Antonio Ortega, "Bit-Rate Control Using Piecewise Approximated Rate-Distortion Characteristics," *IEEE Transactions on Circuits and Systems for Video Technology*, Accepted for publication, Feb 1998.
- [5] Jinho Choi and Daechul Park, "A Stable Feedback Control of the Buffer State Using the Controlled Lagrange Multiplier Method," *IEEE Transactions on Image Processing*, vol 3, no. 5, pp.546-558, September 1994.
- [6] P.N. Tundor and O.H. Werner (BBC), "Real-time transcoding of MPEG-2 video bitstreams," *IEE Conference Publication*, pp.286-301, September 1997.
- [7] H.-M. Hang and J.-J. Chen, "Source model for transform video coder and its application," *IEEE Trans. on CAS for video tech.*, vol. 7, no. 2, pp. 287-311, Apr. 1997.