An Adaptive Transcoding Method for H.264 Video Coding

Soon-kak Kwon†, Seong-woo kim†, Jung-hwa Lee†, Jong-min Lee†

Dept. of Computer Software Engineering, Dongeui University, Pusan, KOREA

Summary
In this paper, we adopt a linear relative formula of a logarithmic function between bitrate and quantization step-size for homogeneous transcoding. Also we propose a novel transcoding method to convert the bitrates between H.264 coded bitstreams. The proposed transcoding method updates the model parameters given for the previous picture or slice by using an approximated relationship between bitrate and quantization step-size. In order to meet a lower target bitrate, it finds a target quantization step-size, and then generates a bitstream by a simple re-quantization process. From the simulation, this paper illustrates that the designated lower target bitrate can be obtained by the re-quantization process after finding the proper quantization step-size with simple implementation only without a complex bitrate control, for the four test sequences with different scene characteristics.
Key words:
Transcoding, Bitstream conversion, Bitrate control, Quantization step-size

1. Introduction

With the increase of extensive applications using compressed video, a bitstream transcoding method has been introduced in the inter-network communications. Bitstream transcoding is the conversion operation from a compressed video bitstream with specific syntax to other compressed bitstreams that have the same or different syntax. The bitstream transcoding algorithms [1-7], can be classified by the homogeneous and heterogeneous methods. The homogeneous bitstream transcoding maintains an identical coding standard or syntax between the networks of a different transmission environment, but converts the bitrates, picture rates and resolutions. The heterogeneous bitstream transcoding converts the coding standard or syntax, that is, from MPEG-2 to MPEG-1 or MPEG-2 to MPEG-4.
In this paper, a homogeneous bitstream transcoding is treated with a H.264 encoding standard, MPEG-4 part 10[8-10]. The research on the homogeneous bitstream transcoding was activated due to a VoD(Video on Demand) application. Because the VoD data is stored as a bitstream with high bitrate for good picture quality, the bitrate reduction may be needed when a final user cannot accept the bitrate of the originally stored bitstream. This bitrate reduction is necessary for a network with a limited bandwidth range or congested network node. The bitrate reduction algorithm converts from high bitrate video(a few Mega bits per second) to lower rate video signal(a few Kilo bits per second). Because the use of mobile networks and satellite links is increasing, a bitrate conversion from high bitrate to lower bitrate is becoming gradually more important.
There are two types of bitrate down conversions. The first one is full decoding and re-encoding. This method accomplishes a re-ordering of the reconstructed pictures and a re-estimation of motion vectors for the motion compensated coding, thus the delay time and the implementation complexity for the transcoding are increased. The second one is to achieve the target bitrate only with the re-quantization process in DCT (Discrete Cosine Transform) domain. The bitstream is partially decoded only by both the inverse VLC (Variable Length Code) and inverse quantization. That is, the DCT coefficients are obtained from an inverse quantization with each quantization step-size. For reducing the video bitrate, the DCT coefficients are re-quantized with a larger quantization step-size. Finally, the coefficients generated by the re-quantization are converted into a new bitstream with a lower bitrate after the VLC encoding process. This bitstream transcoding operation does not include the re-ordering and re-estimation for the motion compensation. Therefore, the bitstream transcoding by re-quantization process requires lower complexity and a little more processing time.
For the bitrate down conversion, the optimal schemes [11,12,13] have been proposed in order to solve the Lagrangian optimization based on each rate-distortion formulation. The critical drawback of the optimization is the heavy complexity required to measure the rate-distortion data on all possible quantization settings. In [14] and [15], accurate rate control methods for transcoding are presented based on the linear relationship between the number of bits generated from the quantized DCT coefficients and the number of non-zero DCT coefficients. Also, a rate control algorithm based on a piece-wise linearly decreasing model of bitrate and quantization parameter has been proposed [16]. This method can obtain the quantization parameter to result in a desired bitrate of a bitstream for MPEG-1 or MPEG-4 video coding.

Manuscript received December 5, 2008
Manuscript revised December 20, 2008

We also have proposed a model of bitrate and quantization step-size for transcoding among MPEG-2 coded bitstreams [17]. However, in this paper, we investigate another relationship between bitrate and quantization step-size for transcoding among H.264 coded bitstreams. The model is modified to be applied the wider range of bitrates. We also propose a novel transcoding method which updates the model parameters given for the previous picture or slice using an approximated relationship between bitrate and quantization step-size according to the coded picture-type, and finds a target quantization step-size, and then meets the target bitrate by a simple re-quantization process. Therefore, the proposed method converts to the target bitrate by simple implementation without a complex bitrate control.

2. Relation between bitrate and quantization step-size for bitstream transcoding

For the bitstream transcoding, a relationship between the bitrate and quantization step-size for an encoding system must be illustrated first. A scalar quantizer is used in H.264. The scalar quantizer is rounding off decimal point numbers to the nearest integer. In the H.264 quantizer[25], a total of 52 quantization step-sizes are supported in standard case and a step-size is used in the indexed value matched by the quantization parameter, that is, the value of a step-size is 0.625 for the index 0 of a table of the quantization parameter and 225 for the index 51, and so on.

In the existing papers, the related formulas between bitrate and quantization step-size have been introduced, these are dependent on the quantizers and application areas.

For an MPEG-1,2 encoding system, we have proposed a formula based on a linear relation between bitrate (*R*) and log function of a quantization step-size (*Qstep*) in a small range of bitrate [17].

$$R = b + a \cdot \log(\frac{1}{Qstep}) \tag{1}$$

where, *b* and *a* are constants. Also, for an application to a wider range of bitrates, the above formula was improved as [18,19].

$$R = b + \frac{a}{Qstep^{\gamma}} \tag{2}$$

where, $0 < \gamma \leq 2$. The other relative formula having two *Qstep* parameters was also proposed [20,21],

$$R = \alpha \cdot Qstep^{-1} + \beta \cdot Qstep^{-2} \tag{3}$$

where, α and β are constants. Also, a linear relative formula was proposed as a logarithmic function between bitrate and quantization step-size [16, 22],

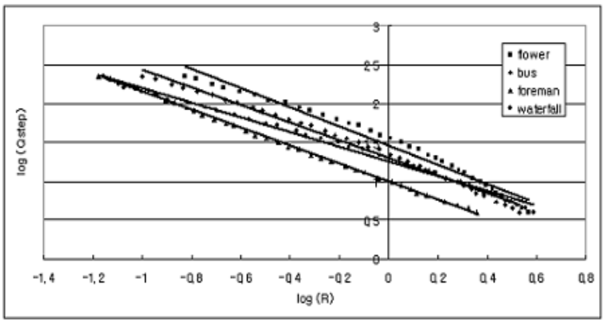
$$\log Qstep = b + a \cdot \log R \tag{4}$$

where, *b* and *a* are dependent on the video characteristics.

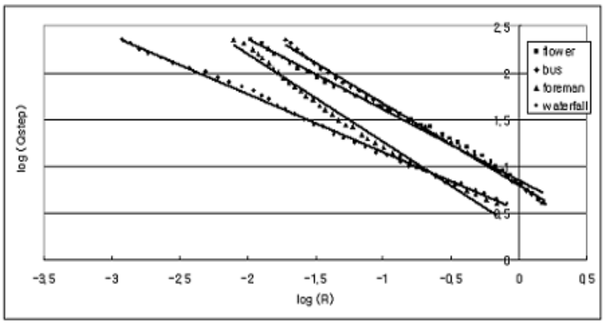
In this paper, in order to find a relative formula between bitrate and quantization step-size for the bitstream transcoding, a simulation was accomplished. For the simulation, JM 9.5 [23,24] of H.264 was used, test sequences were 'Flower Garden', 'Bus', 'Foreman', 'Waterfall' which have the same resolution of horizontal 352 pixels and vertical 288 pixels. The sequences were coded by H.264 baseline profile[8,9,10], I-picture was repeated by 15 pictures interval within 45 pictures in total. One picture contains 18 slices of the same size.

In order to find the relationship between the bitrate and quantization step-size required for the bitstream transcoding, the video sequence reconstructed from a bitstream coded with a particular bitrate should be re-encoded by re-quantization processing. Fig. 1 shows the relationship between the bitrate and quantization step-size. The original sequence before re-encoding was encoded by setting the quantization parameter of H.264 as 15 which indexes 3.5 as the value of a quantization step-size. Re-encodings for the transcoding were carried out by increasing the quantization parameter from 16 to 51. The solid lines of Fig. 1 were obtained by fitting an algorithm of a least-square-error approximate method, based on the model formula of Eq. (4).

Table 1 shows the values of *a*, *b* as the parameters of the model formula. These parameters are dependent on the characteristics of the video.



(a) I-picture



(b) P-picture

Fig. 1 Observed the relationship of *R-Qstep* by the re-quantization, a sequence before transcoding was coded by quantization parameter as 15 (quantization step-size = 3.5) and then re-quantized by increasing the quantization parameter from 16 to 51.

Table 1: The model parameters *a*, *b* of Fig. 1.

Test sequence	I-picture		P-picture	
	a	*b*	*a*	*b*
Flower Garden	-1.23	1.45	-0.76	0.85
Bus	-1.14	1.30	-0.87	0.79
Foreman	-1.15	0.99	-0.93	0.32
Waterfall	-0.94	1.25	-0.62	0.52

Also, we observed the relationship for the sequence of another bitrate, that is, a sequence before transcoding was coded by quantization parameter as 25 (quantization step-size = 11) and then re-quantized by increasing the quantization parameter from 26 to 51. Table 2 shows the obtained values of *a*, *b* as the parameters of the model formula.

Table 2: The model parameters *a*, *b* in the case of setting a quantization parameter as 25 before transcoding.

Test sequence	I-picture		P-picture	
	a	*b*	*a*	*b*
Flower Garden	-1.33	1.51	-0.81	0.91
Bus	-1.20	1.35	-0.93	0.82
Foreman	-1.12	1.01	-0.91	0.27
Waterfall	-1.04	1.32	-0.68	0.55

3. Proposed bitstream transcoding method

The proposed bitstream transcoding is based on the re-quantization which adjusts quantization step-size by the slice in the picture. In the re-quantization, a target quantization step-size to match a target bitrate should be obtained. If bitrate R_1 encoded by quantization step-size $Qstep_1$ is given and a target bit rate R_t is fixed, then we can obtain a target quantization step-size $Qstep_t$ from an approximated model formula for bitstream transcoding. At some narrow bitrate range, the approximation can be allowed for simple implementation [26]. By using the approximated model, a re-encoded bitrate can be different with the target bitrate R_t. The difference between both the target bitrate and the re-encoded bitrate will be compensated for in the next slice. That is, if the re-encoded bitrate is generated as a larger value than the target bitrate in the current slice, then the target bitrate in the next slice will be assigned with a smaller value. Otherwise, the target bitrate will be assigned with a larger value.

3.1 Calculation of target quantization step-size

A target bitrate in bits per pixel, $R_t(bpp)$ is proportional to a target bitrate in bits per second, $BR_t(bps)$, and already an encoded bitrate $BR_1(bps)$ in bits per second, it appears as follows:

$$R_t = \frac{BR_t}{BR_1} \cdot R_1 \qquad (5)$$

where, $R_1(bpp)$ is in bits per pixel, obtained from pixels in a slice before the transcoding.

If R_1 and $Qstep_1$ are given, *R-Qstep* model parameter *b* is calculated by using the model of Eq. (4) as follows.

$$b = \log Qstep_1 - a \cdot \log R_1 \qquad (6)$$

The arrangement between R_t and $Qstep_t$ of the relation formula Eq. (4) is as follows:

$$Qstep_t = 10^{b + a \cdot \log R_t} \qquad (7)$$

If we substitute Eq. (5) and Eq. (6) into Eq. (7), $Qstep_t$ is expressed as follows:

$$Qstep_t = 10^{\log Qstep_1 + a \cdot \log \frac{BR_t}{BR_1}} = Qstep_1 \cdot \left(\frac{BR_t}{BR_1} \right)^a \qquad (8)$$

From the above equation, $Qstep_t$ can be obtained by a fixed *a*. The following section explains a method to find the value of the model parameter *a*.

3.2 Calculation of model formula slope *a*

The value of slope *a* will be different in each picture within a video sequence. However the difference among the sequences is not so noticeable as shown in Fig 1. Therefore, we can assume that the slope *a* is almost constant within same sequence for simple implementation. To compensate for the small difference between the actual and approximated values of slope *a*, we adopt a predicting method for slope *a* which is deduced differently according to the picture-coding types, (I-picture or P-/B-pictures).

• P-/B-picture (predicted as the value of the previous picture)

Because the characteristics of the slices in the spatially same position of adjacent P-/B-pictures are very similar, the value of slope *a* is predicted as a value which is obtained by Eq. (4) with two pairs of coded results. The first pair is bitrate R_1 and quantization step-size $Qstep_1$ before transcoding and second pair is target bit rate R_t and target quantization step-size $Qstep_t$ after transcoding, obtained from a slice of the same position in the picture, coded previously as similarly coded picture-type. Thus, the predicted value of *a* for i^{th} slice of n^{th} P-/B-picture, $\overline{a}_i(n)$, is obtained as follows:

$$\overline{a}_i(n) = a_i(n-1) \qquad (9)$$

where, $a_i(n\text{-}1)$, the value of *a* for i^{th} slice of $(n\text{-}1)^{th}$ P-/B-picture is calculated by substituting the given $Qstep_1$, R_1 and the calculated $Qstep_t$, R_t after re-quantization into the formula Eq. (4).

• I-picture (predicted as the average value of the previous picture)

Because the interval between adjacent I-pictures is normally more than 12 pictures, and the characteristics between slices in the spatially same position from adjacent I-pictures can be greatly different from each other, the value of slope a is predicted as an average value of all slices within the previous I-picture. That is, the predicted value of a for i^{th} slice of n^{th} I-picture, $\overline{a}_i(n)$ is obtained as follows:

$$\overline{a}_i(n) = \frac{1}{N}\sum_{i=1}^{N} a_i(n-1) \quad (10)$$

where, N is the total number of slices in a picture.

3.3 Maximum limit of target quantization step-size

If the approximated value of slope a is used instead of an exact one, the quantization step-size, $Qstep_t$, calculated from Eq. (8) will possibly have a largely different value, compared to the quantization step-sizes of adjacent slices.

Because the change of the values of quantization step-sizes can directly affect the subjective picture quality, we set a limitation of a maximum change of quantization step-sizes among adjacent slices. A variable $T(k)$ is defined as: The accumulated change range, $T(k)$ is calculated by summing all the differences between quantization step-sizes ($Qstep_1$) before bitstream transcoding and quantization step-sizes ($Qstep_t$) after the bitstream transcoding of all slices before current k^{th} slice in a picture. The difference is defined as follows:

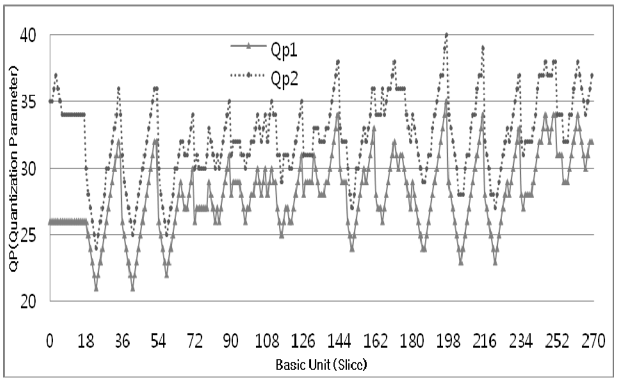
$$T(k) = \sum_{i=1}^{k-1}(Qstep_t(i) - Qstep_1(i))/(k-1), \quad k > 1 \quad (11)$$

In order to avoid the abrupt change of quantization step-sizes between slices before and after the transcoding, also among adjacent slices, we set a limitation as follows. If, $Qstep_t(k) - Qstep_1(k) > T(k) + T(k)/2$ then we fix an upper limitation such as $Qstep_t(k) = Qstep_1(k) + T(k) + T(k)/2$ and if $Qstep_t(k) - Qstep_1(k) < T(k) - T(k)/2$, then we fix a lower limitation such as $Qstep_t(k) = Qstep_1(k) + T(k) - T(k)/2$.

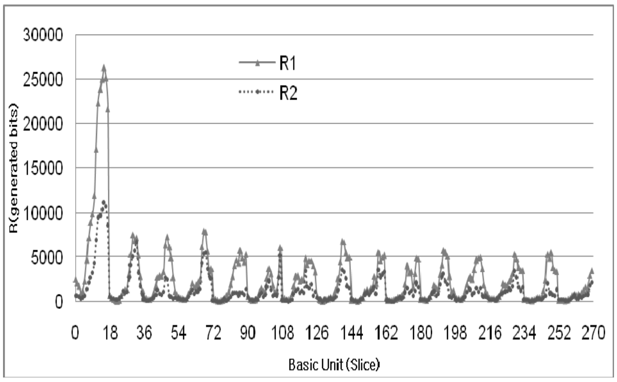
4. Simulation Results

For the simulation, JM 9.5 of H.264 was used, and the test sequences have the same resolution of horizontal 352 pixels and vertical 288 pixels within 4:2:0 color video signals. The sequences were coded by the H.264 baseline profile, I-picture was repeated by a 15-picture interval within 30 pictures, and each picture contains 18 slices of the same size.

Fig. 2, Fig. 3, Fig. 4, and Fig. 5 show the simulation results in the case where a bitstream, coded with the bitrate of 1.5Mbps, is transcoded to the bitrate of 0.8Mbps. The figures compare the quantization parameter QP1, bitrate R1 before transcoding with the quantization parameter QP2, bitrate R2 after transcoding, controlled by a slice unit for 15 pictures in the 2nd GOP (Group Of Pictures) of each sequence.



(a) quantization parameter



(b) number of generated bits

Fig. 2 Transcoding results for ‘Flower Garden’ test sequence, 1.5Mbps bitstream (R1, QP1) is transcoded to 0.8Mbps bitstream (R2, QP2).

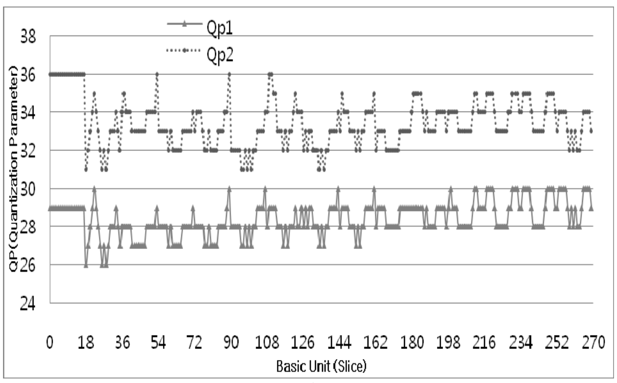


Fig. 3 Transcoding results for ‘Bus’ test sequence, 1.5Mbps bitstream (R1, QP1) is transcoded to 0.8Mbps bitstream (R2, QP2).

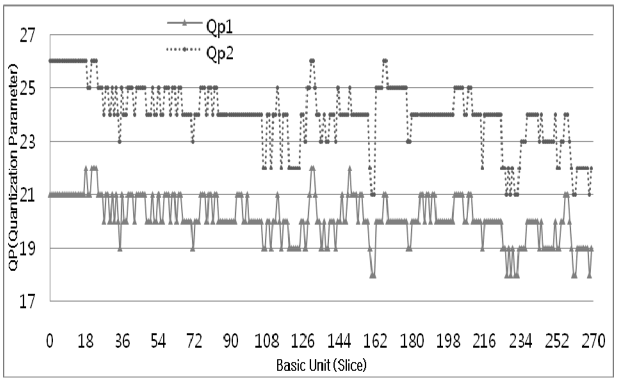


Fig. 4 Transcoding results for ‘Foreman’ test sequence, 1.5Mbps bitstream (R1, QP1) is transcoded to 0.8Mbps bitstream (R2, QP2).

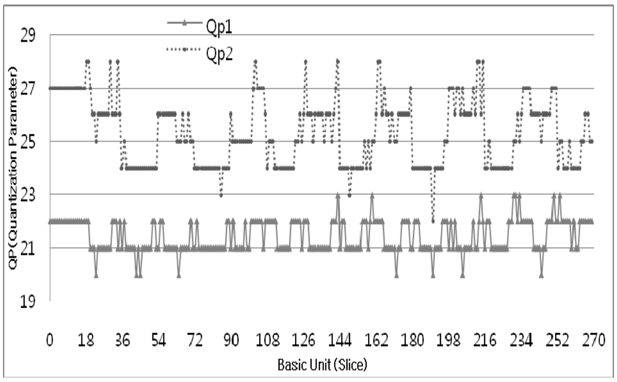
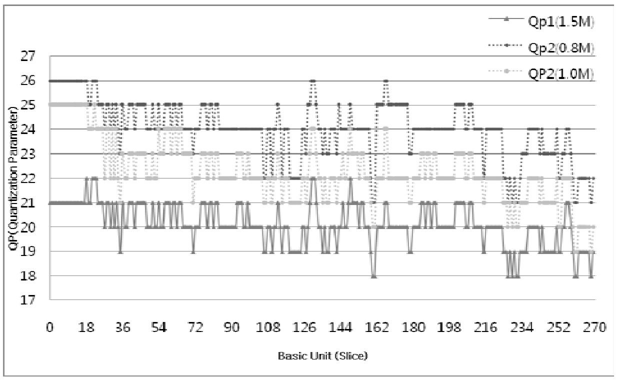


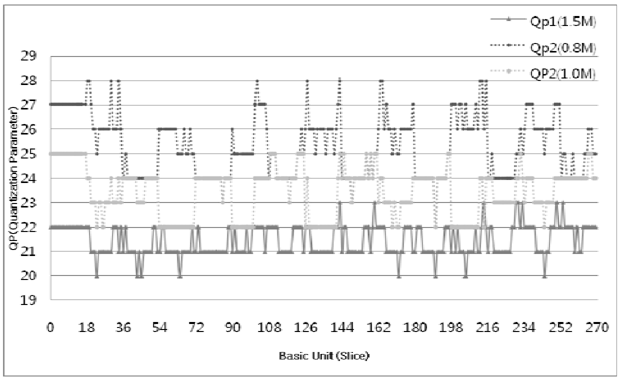
Fig. 5 Transcoding results for ‘Waterfall’ test sequence, 1.5Mbps bitstream (R1, QP1) is transcoded to 0.8Mbps bitstream (R2, QP2).

We can see that the proposed method generates the exact target bitrate for the transcoding, regardless of the video characteristics, the complex texture sequences such as ‘Flower Garden’ and ‘Bus’, or the simple sequences such as ‘Foreman’ and ‘Waterfall’. Also, the variation range between the quantization step-sizes before and after transcoding is similar in slice unit to all sequences.

Fig. 6 shows the other simulation results in the cases where the bitrate of 1.0Mbps was added for transcoding. Also for the target bitrate 1.0Mbps, we can see that the variation range between quantization step-sizes before and after transcoding is similar in slice unit to all sequences. For the ‘Flower Garden’ sequence, the average quantization parameter is 30.5 in the case of bitrate 1Mbps; 32.4 in the case of bitrate 0.8Mbps; compared with 28.1 in the case of bitrate 1.5Mbps before transcoding. Also, the ‘Bus’ sequence gets 31.4.9 for 1Mbps, 33.4 for 0.8Mbps, and 28.4 for 1.5Mbps. The ‘Foreman’ sequence gets 22.3 for 1Mbps, 24.0 for 0.8Mbps, and 20.1 for 1.5Mbps. Finally the ‘Waterfall’ sequence gets 23.5 for 1Mbps, 25.3 for 0.8Mbps, and 21.5 for 1.5Mbps.



(a) ‘Foreman’ test sequence



(b) ‘Waterfall’ test sequence

Fig. 6 Quantization parameters before and after transcoding, where 1.5Mbps bitstream (QP1) is transcoded to 1Mbps bitstream (QP2(1.0M)) and 0.8Mbps bitstream (QP2(0.8M)).

5. Conclusion

This paper proposed an adaptive bitstream transcoding method for the H.264 encoding system and showed a relationship between bitrate and quantization step-size for bitstream transcoding. A particular parameter of the model formula is approximated to convert a bitstream to another bitstream by slice unit. The model parameter for a slice within a P-/B-picture encoded by motion compensation is predicted according to the value of model parameter of slice in the spatially same position as the previous picture of the same picture-type. The model parameter for a slice within I-picture is predicted according to the average value of all slices of the previous I-picture. Also, the limitation of the change range of quantization step-size among adjacent slices is set in order to avoid the picture quality deviation. From the simulation, this paper illustrates that the target lower bitrate is obtained by the re-quantization process after finding the proper quantization step-size with simple implementation only.

References

[1] I.-M. Pao and M.-T. Sun, "Encoding Stored Video for Streaming Applications", *IEEE Trans. Circuits and Syst. Video Technology*, Vol. 11, No. 2, pp. 199-209, Feb. 2001.

[2] T. Shanableh and M. Ghanbari, "Heterogeneous Video Transcoding to Lower Spatio-temporal Resolutions and Different Encoding Formals", *IEEE Trans. Multimedia*, Vol. 2, No. 2, pp. 101-110, June 2000.

[3] N. Bjork and C. Christopoulos, "Transcoder Architectures for Video Coding", *IEEE Trans. Consumer Electronics*, Vol. 44, No. 1, pp. 88-98, Feb. 1999.

[4] H. Sun, W. Kwok, and J. Zdepski, "Architectures for MPEG Compressed Bitstream Scaling", *IEEE Trans. Circuits and Syst. Video Technology*, Vol. 6, No. 2, pp. 191-199, Apr. 1996.

[5] K. D. Seo, S. H. Lee, J. K. Kim, and J. S. Koh, "Rate Control Algorithm for Fast Bit-rate Conversion Transcoding", *IEEE Trans. Consumer Electronics*, Vol. 46, No. 4, pp. 1128-1136, Nov. 2000.

[6] S. Acharya and B. Smith, "Compressed Domain Transcoding of MPEG", *Proc. IEEE Int. Conf. Multimedia Computing and Systems, Austin, Texas*, pp. 295-304, July 1998.

[7] T. Qian, J. Sun, D. Li, X. Yang, and J. Wang, "Transform Domain Transcoding From MPEG-2 to H.264 with Interpolation Drift-Error Compensation," *IEEE Trans. Circuit Syst. Video Technology*, Vol. 16, No. 4, pp. 523-534, April 2006.

[8] ISO/IEC 14496-10 and ITU-T Rec., "H.264 Advanced Video Coding", *Joint Video Team of ISO/IEC MPEG & ITU-T VCEG*, May 2003.

[9] T. Widegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard", *IEEE Trans. Circuit Syst. Video Technology*, Vol. 13, No. 7, pp. 560-576, July 2003.

[10] S.-k. Kwon, A. Tamhankar, and K. R. Rao, "Overview of H.264/MPEG-4 Part 10", *Journal of Visual Communications and Image Representation*, Vol. 17, No. 2, pp. 186-216, Apr. 2006.

[11] P. Assuncao and M. Ghanbari, "Transcoding of MPEG-2 video in the Frequency Domain," *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, 1997.

[12] M. Xia, A. Vetro, B. Liu, and H. Sun, "Rate-distortion Optimized Bit Allocation for Error Resilient Video Transcoding," *Proc. IEEE Int. Symp. Circuits and Systems*, Vol. 3, pp. 945-948, May 2004.

[13] W. Lie, M. Tsai, and T. Lin, "Rate-distortion Optimized DCT-domain Video Transcoder for Bit-rate Reduction of MPEG Videos," *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Vol. 5, pp. 969-972, May 2004.

[14] L. Chen, L. Cheng, and S. Yu, "Accurate Rate Control Method in Transcoding," *Electron. Lett.*, Vol. 40, No. 1, pp. 16-18, Jan. 2004.

[15] J. Lei and N. Georganas, "Accurate Bit Allocation and Rate Control for DCT Domain Video Transcoding," *Proc. IEEE Canadian Conf. Electrical and Computer Engineering*, Vol. 2, pp. 968-973, May 2002.

[16] K. Seo, S. Heo, and J. Kim, "Adaptive Rate Control Algorithm Based on Logarithmic R-Q model for MPEG-1 to MPEG-4 Transcoding," *Signal Process. Image Commun.*, Vol. 17, No. 10, pp. 857-875, Nov. 2002.

[17] S.-k. Kwon, J.-k. Kim, and Y.-d. Park, "Bit-rate Control for Uniform Picture Quality within Composed Video Sequence", *Optical Engineering*, Vol. 37, No. 11, pp. 3053-3060, Nov. 1998.

[18] W. Ding and B. Liu, "Rate Control of MPEG Video Coding and Recording by Rate-quantization Modeling", *IEEE Trans. Circuit Syst. Video Technology*, Vol. 6, No. 1, pp. 12-20, Feb. 1996.

[19] L.-J. Lin and A. Ortega, "Bit-Rate Control Using Piecewise Approximated Rate-Distortion Characteristics", *IEEE Trans. Circuit Syst. Video Technology*, Vol. 8, No. 4, pp. 446-459, Aug. 1998.

[20] T. Chiang and Y.-Q. Zhang, "A New Rate Control Scheme Using Quadratic Rate Distortion Model", *IEEE Trans. Circuit Syst. Video Technology*, Vol. 7, No. 1, pp. 246-250, Feb. 1997.

[21] H.-J. Lee, T. Chiang, and Y.-Q. Zhang, "Scalable Rate Control for MPEG-4 Video", *IEEE Trans. Circuit Syst. Video Technology*, Vol. 10. No. 9, pp. 878-894, Sep. 2000.

[22] J. Katto and M. Ohta, "Mathematical Analysis of MPEG Compression Capability and its Application to Rate Control", *Proc. Int. Conf. Image Processing*, Washington, D.C., pp. 555-558, Oct. 1995.

[23] Institute of Computing Technology, "JVT-H017 : Proposed Draft of Adaptive Rate Control", *Joint Video Team of ISO/IEC MPEG & ITU-T VCEG,* May, 2003.

[24] H.264/AVC JM Software, http://iphome.hhi.de/suehring/tml/download.

[25] I. E. G. Richardson, H.264 and MPEG-4 Video Compression, Wiley, 2003.

[26] S.-k. Kwon, K. R. Rao, O.-J. Kwon, and T.-S. Kim, "Joint Bandwidth Allocation for User Required Picture Quality Ratio Among Multiple Video Sources," *IEEE Trans. Broadcasting*, Vol. 51, No. 3, pp. 287-295, Sep. 2005.

Soon-kak Kwon received the B.S. degree in Electronic Engineering from Kyungpook National University, in 1990, the M.S. and Ph.D. degrees in Electrical Engineering from Korea Advanced Institute of Science and Technology (KAIST), in 1992 and 1998, respectively. From 1998 to 2000, he was a team manager at Technology Appraisal Center of Korea Technology Guarantee Fund. Since 2001, he has been a faculty member of Dongeui University, where he is now an associate professor in the Department of Computer Software Engineering. From 2003 to 2004, he was a visiting professor of electrical engineering in the University of Texas at Arlington. His research interests are in the areas of image processing, video coding, and video transmission.

Seong-woo Kim received his B.S., M.S., and Ph.D. degrees in Electrical Engineering from Korea Advanced Institute of Science and Technology (KAIST), in 1991, 1993, and 1999 respectively. He worked at Electronics and Telecommunications Research Institute (ETRI) from 1999 to 2001. Since 2002, he has been a faculty member of Dongeui University, where he is now an associate professor in the Department of Computer Software Engineering. His research interests are in the areas of embedded computing, real-time system.

Jung-hwa Lee received his B.S., M.S., and Ph.D. degrees at the Department of Computer Science from Pusan National University, Korea, in 1992, 1995, and 2001, respectively. He is now an associate professor at the Department of Computer Software Engineering, Dongeui University in Korea. His research interests include multimedia system and database.

Jong-min Lee received the B.S. degree in computer engineering from Kyungpook National University, Korea, in 1992, and the M.S. degree and the Ph.D. degree in computer science from Korea Advanced Institute of Science and Technology in 1994 and 2000, respectively. From 1999 to 2002, he worked as a senior engineer at Mobile Communication Division, Samsung Electronics Co., Ltd. Since 2002 he has been a faculty member of Dongeui University, where he is an associate professor in the Department of Computer Software Engineering. From 2005 to 2006, he visited the University of California, Santa Cruz as a research associate. His research interests include mobility management and multimedia service in wireless networks.