Channel-adaptive Image Compression for Wireless Transmission

Yun-Gu Lee[1,*] and Ki-Hoon Lee[2]

[1] Department of Computer Software, Kwangwoon University / Seoul, Korea
[2] Department of Computer Engineering, Kwangwoon University / Seoul, Korea

* Corresponding Author: Yun-Gu Lee, yglee96@kw.ac.kr

Received July 28, 2017; Accepted August 21, 2017; Published August 30, 2017

** Short Paper*

Abstract: This paper presents computationally efficient image compression for wireless transmission of high-definition video, to adaptively utilize available channel bandwidth and improve image quality. The method indirectly predicts an unknown available channel bandwidth by monitoring encoder buffer status, and adaptively controls a quantization parameter to fully utilize the bandwidth. Experimental results show that the proposed method is robust to variations in channel bandwidth.

Keywords: Image coding, Channel adaptive image coding, Channel adaptive rate control

1. Introduction

The Wi-Fi Alliance recently announced Miracast, which is a peer-to-peer wireless screencasting standard formed via Wi-Fi Direct connections [1]. This technology enables us to wirelessly stream audiovisual data from one device to other devices, including televisions, desktops, laptops, tablets, and mobile phones, over Wi-Fi. Nowadays, such real-time wireless video streaming scenarios have become popular. Miracast adopts international video coding standards such as H.264. Rate control plays a key role in meeting constraint bitrates in video coding [2, 3].

One interesting application for wireless transmission of video is to replace a wired high-definition multimedia interface (HDMI) between two devices with wireless HDMI. Miracast might be a solution for wireless HDMI, but there are limitations before it can be widely utilized. The first issue is latency. While many real-time rate control methods have been proposed to reduce latency [4-6], the delay is one frame or across several frames [7]. For applications requiring extremely fast response to user input, latency becomes a critical problem. For example, consider a digital television wirelessly connected with a game console. To the best of our knowledge, the existing video encoders hardly satisfy the low latency requirement. The second issue is implementation cost. A video codec usually requires external memory, such as DDR, to store previous frames, and a system bus that supports high bandwidth throughput to access the external memory. A video codec with extra external memory is relatively expensive for low-cost devices like wireless HDMI.

Image compression can resolve the above latency and cost issues by sacrificing compression efficiency. While image compression is less efficient than video compression, recent wireless technology (like Wi-Fi) provides bandwidth high enough to overcome the low efficiency of image compression. So WirelessHD [8] emerged, which is an industry-led effort to define specifications of a new digital network interface for wireless transmission of high-definition content to consumer electronics products [9]. The consortium currently has over 40 adopters, including Broadcom, Intel, LG, Panasonic, NEC, Samsung, SiBEAM, Sony, Philips, and Toshiba. Wireless HD allows either lightly compressed or uncompressed digital transmission of high-definition video and audio and data signals, essentially making it equivalent to wireless HDMI [9]. Since the real channel is not ideal due to noise and obstacles between transmitter and receiver, the bandwidth in real environments may not always be enough to send uncompressed high-quality video. Thus, the lightly compressed transmission is preferred.

On the other hand, the above transmission system usually assumes a 1:1 dedicated connection within a small area so the transmission system can fully utilize the available channel bandwidth. Since wireless channel bandwidth is so dynamic, channel adaptation is one of the

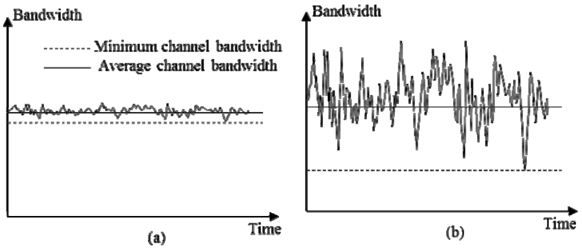

Fig. 1. Channel bandwidth with (a) small, (b) high variances.

key features in image compression for wireless video transmission systems, as well as low implementation cost and low latency. Although many image compression methods have been studied, such as JPEG [10], JPEG-LS [11, 12], JPEG2000 [13], and JPEG-XR [14], they do not consider channel adaptation. Therefore, this paper proposes channel adaptive image compression with low latency and a low implementation cost, based on previous work [15], to reduce bus traffic in a system on chip (SoC). Experimental results demonstrate that the proposed image compression provides better performance than existing methods for dynamically varying channel bandwidth.

2. Proposed Method

2.1 Overview

Many image coding tools have been proposed to reduce redundancies in images. While simple coding tools have low implementation costs, their coding efficiencies are not usually optimal. Advanced coding tools may provide the best coding performance with high computational costs. Hence, there is a trade-off between complexity and coding efficiency in the tools, and coding tools should be selected depending on applications. Let us consider important requirements for wireless transmission of video. Fig. 1 illustrates channel bandwidth with small and high variations along the time line. In Fig. 1(a), the difference between the minimum and average channel bandwidth is small. Although a target bitrate for an image coder is set to the minimum channel bandwidth, the channel resource can be used efficiently. Since the available channel bandwidth is almost constant along the time line, the target bitrate (or compression ratio) can easily be defined. In this case, the most important requirement for improving system performance is to achieve high coding efficiency using advanced image coding tools. In Fig. 1(b), while the average channel bandwidth is similar to Fig. 1(a), the minimum channel bandwidth in Fig. 1(a) is much lower than that of Fig. 1(b). When a target bitrate is set to the average channel bandwidth, a system will suffer from buffer overflow in an encoder or underflow in a decoder. If the target bitrate is set to the minimum channel bandwidth, the channel resource cannot be efficiently utilized. Coding efficiency can be increased by adopting advanced image coding tools. However, the improvement is not enough to overcome the problem of low channel utilization. In this scenario, the most important requirement is not to adopt an advanced coding tool, but to increase channel utilization. Therefore, the proposed image coder focuses on how to improve channel utilization by introducing a channel adaptive rate control method. Simple image coding tools are adopted for the remaining parts, in order to minimize image coder complexity. This approach will gain some advantages for an image coder for wireless video transmission. Simple image coding tools can significantly reduce the implementation cost and can be applied easily. By fully utilizing the available channel resource, image quality is dramatically improved. Accordingly, the proposed low-cost image coder can provide high performance for wireless video transmission applications.

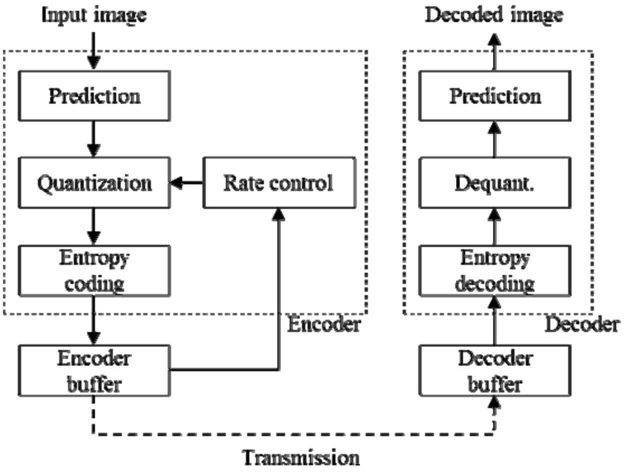

Fig. 2. Overview of the system.

Fig. 2 illustrates an overview of the proposed coding system. An encoder compresses an input image and generates compressed bits. The bits are temporarily stored in an encoder buffer and transmitted to a decoder buffer. Then, the decoder decompresses the encoded bits to reconstruct a decoded image.

2.2 Proposed Image Coder

The proposed method divides an image into 1 × 4 blocks, and each block is sequentially coded in raster scan order. Each color component is handled independently. The basic structure of the proposed image coder is given in Fig. 2. An encoder consists of prediction, quantization, rate control, and entropy coding. As mentioned in the previous subsection, since the most important feature for a channel adaptive image coder is rate control, the proposed coder simply adopts existing methods for prediction, quantization, and entropy coding. A current pixel is predicted from its neighboring pixels, like the method in JPEG-LS [11, 12]. Then, the difference between the current and predicted pixels is quantized. Quantization is a simple division operation as follows:

$$q(m)=\begin{cases} o(m)/(Q+1) & \text{if } o(m)>0 \\ -\left|o(m)\right|1/(Q+\ ) & \text{otherwise} \end{cases} \tag{1}$$

Dequantization is as follows:

$$r(m) = \begin{cases} q(m) \times (Q+1) + (Q/2) & \textit{if } q(m) > 0 \\ -\left(|q(m)| \times (Q+1) + (Q/2)\right) & \textit{otherwise} \end{cases}$$

Here, $o(m)$, $q(m)$, and $r(m)$ denote an input, and its quantized and dequantized values, respectively. Q is a quantization parameter (QP).

The basic concept for coding quantized coefficient $q(m)$ is similar to previous work [15]. The coder first determines the maximal size of four pixels within a block. For example, let significant bit lengths for four pixels within the block be 5, 4, 5, and 6. Then, the maximal size is 6, and a code word length of four pixels becomes 6. The details were given by Lee et al. [15].

2.3 Rate Control Method

The proposed rate control considers an encoder buffer status to control QP. A straightforward method to fully utilize channel bandwidth is as follows. When buffer usage is greater than T_{Full}, the method increases the QP value to reduce generated bits and prevent buffer overflow. When buffer usage is less than T_{Empty}, the scheme decreases the QP value to enhance the image quality. However, this approach has the following problem. Once the coder encounters a complex image region to encode, the amount of encoded bits increases, and buffer usage suddenly becomes much more than T_{Full}. Then, the rate control increases the QP value by 1. However, the buffer usage cannot immediately become less than T_{Full}. In this case, although the current QP value is the optimal one, if buffer usage is still more than T_{Full}, the rate control increases the QP value again in the next processing. Sometimes, the QP value is increased again and again. Since the QP value becomes big compared to the optimal one, buffer usage is rapidly decreasing. Eventually, the buffer becomes empty, and buffer usage is less than T_{Empty}. Now the QP value is decreasing, and buffer usage becomes full again. Accordingly, the QP value sometimes oscillates out of control. Inversely, when the rate control encounters a very flat region, the buffer suddenly becomes empty, and the QP value is abruptly decreased a lot. This fluctuation of the QP value significantly decreases subjective and objective visual quality, as well as coding efficiency

In order to solve the above problem, the proposed algorithm divides the buffer status into safe and unsafe states. When the buffer is in the safe state, the QP value is changing within limited values. In other words, the QP value has upper and lower bounds within the safe state, and the bounds make the visual quality consistent within a frame. The upper and lower bounds are predicted from the QP values in the previous frame. The basic assumption is that since successive frames are highly correlated with each other, the QP value for the current frame can be predicted from the QP values in the previous frame. The details are as follows:

$$Q_C1 = \begin{cases} \max(Q_R - 2, Q_c - 1) & \textit{if } B_U < T_{E1} \\ \max(Q_R - 1, Q_c - 1) & \textit{if } B_U < T_{E2} \\ \min(Q_R + \quad Q_c) & \textit{if } T_{E2} \le B_U < T_{F1} \\ \min(Q_R + 2, Q_c + 1) & \textit{if } T_{F1} \le B_U < T_{F2} \\ Q_c + 1 & \textit{Otherwise} \end{cases}$$

Here, Q_R is the average QP value in the previous frame. If buffer usage is less than T_{F2}, the rate control regards the buffer status as being in the safe state. In detail, when the buffer status is less than T_{E1} (or T_{E2}), the rate control decreases the QP value by 1. However, the QP value has a lower bound at ($Q_R - 2$) or ($Q_R - 1$). The rate control keeps the current QP value for $T_{E2} \leq B_U < T_{F1}$, but there is an upper bound at (Q_R + 1). For $T_{F1} \leq B_U < T_{F2}$, the rate control increases the QP value by 1 to an upper bound of (Q_R + 2). If $B_U \geq T_{F2}$, the rate control determines that the encoder buffer should be strictly controlled to prevent encoder buffer overflow. In this case, there is no limitation in controlling the QP value. Thresholds for T_{E1}, T_{E2}, T_{F1}, and T_{F2} are B_S/100, B_S/10, B_S/4, and B_S × 15/16, respectively, which are empirically chosen. Here, B_S is the size of an encoder buffer.

3. Simulation Results

Experiments were performed to evaluate the proposed algorithm. Fig. 2 depicts the structure of the simulator. After an encoder generates encoded bits for each 1×4 block, the bits are immediately stored in an encoder buffer. Then, a virtual transmitter moves the bit stream (BS) from the encoder buffer to the decoder buffer. Latency between encoder and decoder is set to 1 msec. The size of the BS is set to (average bitrate × latency). Typically, the BS is 0.5 M bits for 500 Mbps.

In the experiments, channel utilization and the performance of coding efficiency were evaluated. Table 1 demonstrates the channel utilization of the proposed algorithm. A channel transfer rate used in the simulation is given in Fig. 3. The transfer rate is not constant, to consider the wireless medium. The minimum and average transfer rates are 346 Mbps and 500 Mbps, respectively. Most existing methods support only a fixed compression ratio or do not guarantee the compression ratio. Hence, this paper chooses the rate control of previous work [15] as a conventional method. In order to compare only the performance of rate control algorithms, the same coding tools for prediction, quantization, and entropy coding were utilized for the conventional and proposed methods. In the table, the performance of the constant channel bandwidth (CCB) of 500 Mbps with the proposed method is similar to that of varying channel bandwidth (VCB) with the proposed method. It illustrates how the proposed algorithm efficiently utilizes the available channel bandwidth. Note here that the average channel bandwidth in Fig. 3 is 500 Mbps. Also the proposed method outperforms the conventional method for the same CCB condition:

Table 1. Simulation results for channel utilization of the proposed algorithm in terms of PSNR (dB) and bitrate (Mbps) with sequences of BQTerrace (BQ), Kimono1 (K), BasketBallDrive (BBD), Tennis (T), and ParkScene (PS). CCB and VCB stand for constant channel bandwidth and varying channel bandwidth, respectively.

Sequence name	Conventional method [15]				Proposed method			
	CCB: 346 Mbps		CCB: 500 Mbps		CCB: 500 Mbps		VCB as in Fig. 3	
	PSNR	Bitrate	PSNR	Bitrate	PSNR	Bitrate	PSNR	Bitrate
BQT	**32.50**	345	39.00	501	42.37	501	42.19	500
K	39.65	347	48.18	501	49.51	501	49.55	501
BBD	39.14	347	48.21	501	49.31	501	49.30	501
T	42.16	347	46.59	501	48.79	501	48.61	501
PS	36.42	436	44.73	501	46.34	501	46.37	501

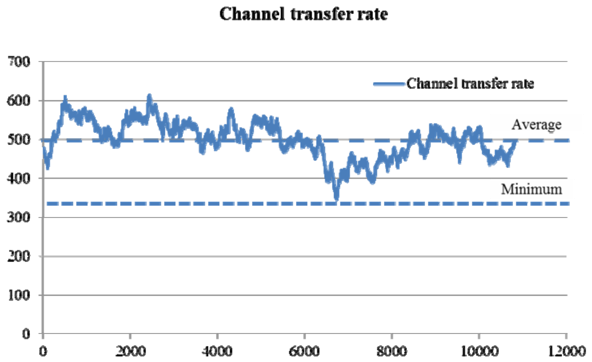

Fig. 3. Channel transfer rate for the experiments.

Table 2. Performance comparison of the proposed algorithm under JPEG-LS with sequences of BQTerrace (BQ), Kimono1 (K), BasketBallDrive (BBD), Tennis (T), and ParkScene (PS).

Sequence name	Compression ratio for a lossless mode	
	Proposed	JPEG-LS
BQT	**1.61**	**2.08**
K	2.13	2.63
BBD	2.04	2.38
T	2.04	2.50
PS	1.86	2.17

500 Mbps. Since a conventional rate control method does not support channel adaptiveness, if a target bitrate is set to 500 Mbps, an encoder buffer will overflow. Hence, for safety, the target bitrate may be set to the minimum value for the given channel, which is 346 Mbps in this example. Then, the peak signal-to-noise ratio (PSNR) performance is significantly degraded. This simulation shows that channel adaptive rate control is very important for real-time wireless video transmission scenarios.

Table 2 shows the performance comparison of the proposed algorithm under JPEG-LS without rate control. Since the proposed algorithm adopts very simple coding tools, its coding efficiency is less than the coding efficiency of JPEG-LS. However, it should be noted that JPEG-LS is not suitable for real-time video transmission services. While the proposed algorithm does not provide the best coding efficiency, compared with existing methods, it maximizes channel utilization by adopting the proposed rate control method in order to further improve image quality.

4. Conclusion

This paper introduces a new image coder to adaptively utilize available channel bandwidth for real-time video transmission services. The proposed method adopts simple image coding tools to minimize the implementation costs. A new rate control method enables the image coder to fully utilize the available channel bandwidth. Consequently, while the proposed image coder is quite simple, it can significantly improve image quality for wireless transmission by maximizing channel utilization.

Acknowledgement

This present research has been conducted by the Research Grant of Kwangwoon University in 2016.

References

[1] Wi-Fi Certified Miracast: Extending the Wi-Fi experience to seamless video display, Wi-fi.org, Oct. 2013. Article (CrossRef Link)

[2] X. Jing, L.-P. Chau, and W.-C. Siu, “Frame complexity-based rate-quantization model for H.264/AVC intraframe rate control,” *IEEE Signal Processing Letters*, vol. 15, pp. 373-376, 2008. Article (CrossRef Link)

[3] D. K. Kwon, M-Y. Shen, C-C. J. Kuo, “Rate control for H,264 video with enhanced rate and distortion models,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, issue 5, pp. 517-529, 2007. Article (CrossRef Link)

[4] Y. Liu, Z.G. Li, and Y. C. Soh, “Novel rate control scheme for low delay video communication of H.264/AVC standard,” *IEEE Transactions on Circuits Systems Video Technology*, vol. 17, no. 1, pp. 68-78, Jan. 2007. Article (CrossRef Link)

[5] J. Tasi, “Rate control for low delay video using a dynamic rate table,” *IEEE Transactions on Circuits*

Systems and Video Technology, vol. 15, no. 1, pp. 133-137, Jan. 2005. Article (CrossRef Link)

[6] Y. G. Lee and B. C. Song, "An intra-frame rate control algorithm for ultralow delay H.264/Advanced Video Coding (AVC)," *IEEE Transactions on Circuits Systems and Video Technology*, vol. 19, no. 5, pp. 747-752, May 2009. Article (CrossRef Link)

[7] C. Gong and X.Wang, "Adaptive transmission for delay-constrained wireless video," *IEEE Transactions on Wireless Communications*, vol. 13, issue 1, pp. 49-61, Jan. 2014. Article (CrossRef Link)

[8] http://www.wirelesshd.org/about/

[9] http://en.wikipedia.org/wiki/WirelessHD

[10] G. K. Wallace, "The JPEG Still Picture Compression Standard", *Communications of the ACM*, vol. 34, pp.30 -44 1991 Article (CrossRef Link)

[11] ISO/IEC, Information Technology-Lossless and near-lossless compression of continuous-tone images-Baseline, ITU-T Recommendation T.87, 14495-1, 1998 Article (CrossRef Link)

[12] M. J. Weinberger, G. Sapiro, G. Seroussi, "The LOCO-I lossless image compression algorithm: principle and standardization into JPEGLS," *IEEE Transactions on Image Processing*, vol. 9, no. 8, pp. 13091324, 2000. Article (CrossRef Link)

[13] ISO/IEC 15444-1, Information technology JPEG 2000 image coding system Part 1: Core coding system, 2000. Article (CrossRef Link)

[14] F. Dufaux, G. J. Sullivan, and T. Ebrahimi, "The JPEG XR image coding standard", *IEEE Signal Processing Magazine*, vol. 26, issue 6, pp. 195199, 2009. Article (CrossRef Link)

[15] Y. G Lee, B. C. Song, N. H. Kim, T. H. Kim, and W. H. Joo, "Low complexity near-lossless image coder for efficient bus traffic in very large size multimedia SoC," *IEEE International Conference on Image Processing 2009*, pp. 2329-2332, Nov. 2009. Article (CrossRef Link)

Yun-Gu Lee received his BSc, MSc, and PhD in electrical engineering from the Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea, in 2000, 2002, and 2006, respectively. From 2006 to 2013, he was in the Media Processing Laboratory, DMC Research and Development Center, Samsung Electronics, Republic of Korea, where he was a principal engineer. In 2013, he joined the Department of Computer Software, Kwangwoon University, Seoul, Republic of Korea, where he is currently an associate professor. His current research interests include the general areas of image and video coding, image and video processing, image stabilization, computer vision, and three-dimensional display systems.

Ki-Hoon Lee received a BSc (2000), an MSc (2002), and a PhD (2009) in Computer Science from the Korea Advanced Institute of Science and Technology (KAIST). He is currently an associate professor in the Department of Computer Engineering at Kwangwoon University.