

## Research Article

# Network-Aware Reference Frame Control for Error-Resilient H.264/AVC Video Streaming Service

**Hui-Seon Gang, Goo-Rak Kwon, and Jae-Young Pyun**

*Department of Information and Communication Engineering, Chosun University, 309 Pilmun-daero, Dong-gu, Gwangju 61452, Republic of Korea*

Correspondence should be addressed to Jae-Young Pyun; [jyppyun@chosun.ac.kr](mailto:jyppyun@chosun.ac.kr)

Received 7 August 2015; Revised 21 December 2015; Accepted 15 February 2016

Academic Editor: Qi Wang

Copyright © 2016 Hui-Seon Gang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To provide high-quality video streaming services in a mobile communication network, a large bandwidth and reliable channel conditions are required. However, mobile communication services still encounter limited bandwidth and varying channel conditions. The streaming video system compresses video with motion estimation and compensation using multiple reference frames. The multiple reference frame structure can reduce the compressed bit rate of video; however, it can also cause significant error propagation when the video in the channel is damaged. Even though the streaming video system includes error-resilience tools to mitigate quality degradation, error propagation is inevitable because all errors can not be refreshed under the multiple reference frame structure. In this paper, a new network-aware error-resilient streaming video system is introduced. The proposed system can mitigate error propagation by controlling the number of reference frames based on channel status. The performance enhancement is demonstrated by comparing the proposed method to the conventional streaming system using static number of reference frames.

## 1. Introduction

Today, high-quality video content is a basic requirement of multimedia services and is becoming important in mobile communication systems. Because of the low cost of powerful processors and the advancement of mobile communication services, consumers are able to use high-definition multimedia streaming services on their hand-held devices. These multimedia streaming data have been compressed for storage and transmission. Even though many service providers have developed and provided advanced mobile communication services, it remains difficult to reliably transmit high-quality video streams because of the varying channel conditions and limited available bandwidth of wireless channels.

The current streaming video system generally uses motion estimation and compensation procedure at encoder and decoder, respectively, for a high coding efficiency feature. This system considerably reduces the number of bits to encode because it utilizes multiple reference frames to remove temporal redundancy. Because of its high coding efficiency, H.264/AVC and H.265/HEVC are suitable for the streaming

system transmitting high-quality video sequences in the environments that have limited channel capacity [1].

However, if the encoded sequences are damaged by channel errors, the damage can be propagated to neighboring macroblocks (MBs) and frames. Even though motion estimation using multiple reference frames can significantly decrease the number of data bits that must be encoded, the compressed sequence can be vulnerable to error propagation. To mitigate the impact of error propagation, the streaming video system includes error-resilience tools. Error-resilience tools preprocess the video data either by reordering each macroblock's coding sequence or by inserting redundant data, such that the damaged blocks can be spreaded out (especially in the case of burst errors). That is, the damaged video is improved by error-resilience tools. These tools can make the encoded video sequence more robust to errors, but coding efficiency will be decreased because of the additional bits [2]. Among the typical error-resilient methods, intrarefresh (IR) algorithm is used often to avoid error propagation in a distorted video sequence over an error-prone network. When IR algorithm is used as an error-resilience method, multiple

reference frame structure used in H.264/AVC motion compensation has recently been found to reduce the received video quality in the presence of transmission errors [3, 4]. This effect occurs because the blocks refreshed by IR coding at the decoder may not be used for further motion compensation of the next frames in multiple reference frame structures; thus, the propagated distortions are not always removed.

In this paper, a new network-aware streaming video system controlling the number of reference frames is proposed for a reliable video transmission system over an error-prone network. The proposed streaming video system uses both the multiple reference frame structure and error-resilience tools for ensuring both error robustness and coding efficiency. To demonstrate the trade-off between error resilience and coding efficiency, various IR and reference frame conditions are used in the performance evaluation.

This paper is organized as follows: Section 2 describes the typical video streaming system. The proposed error-resilient system is explained in Section 3 and the experimental results are presented in Section 4. Finally, Section 5 presents our conclusions.

## 2. Typical Streaming Video System

*2.1. Motion Estimation with Multiple Reference Frames.* The key principle of video compression is the elimination of redundancy. Typically, a video encoder compresses a video sequence by removing the temporal, spatial, and statistical redundancies. Specifically, motion estimation and compensation within the compression strategy increase the coding efficiency by removing temporal redundancy between frames. To remove more temporal redundancies, a typical streaming video system based on H.264/AVC and H.265/HEVC uses multiple reference frames for motion compensation and chooses the best reference frame among them based on rate-distortion optimization (RDO). Next, it searches for motion vectors and encodes the residual data in each MB [5]. Motion information regarding blocks is separately encoded and transferred over the network; it is then used for the reconstruction of the original blocks.

However, this compressed motion may be distorted on the unreliable channel. If there are errors caused by packet losses on an unreliable channel, the errors can be propagated into the following frames by the motion compensation procedure even though there are intrarefreshed blocks that are inserted for the error resilience [2]. Furthermore, error propagation becomes more severe as the number of reference frames increases. As a result, motion estimation using multiple reference frames has the advantage of increasing the coding efficiency; however, it also makes the transferred video sequence less robust to errors.

*2.2. Error Resilience against Transmission Error.* To mitigate quality degradation caused by error propagation, the streaming video system includes error-resilience tools. These tools run the preprocessing in the encoder to make encoded data more robust to errors. In arbitrary slice ordering (ASO), each

slice group can be sent in any order and can (optionally) be decoded in order of receipt instead of in the usual scan order; flexible MB ordering (FMO) also reorders the coding sequence of MBs. Even though slices and MBs are consecutively damaged, these errors are scattered and localized in neighboring coding units because these units are reordered at the decoder [1]. Additionally, data partitioning (DP) provides the ability to separate more important and less important syntax elements into different packets of data. Furthermore, redundant slices (RS) are added when the encoder inserts additional picture data such as redundant slice, thus making it more robust to errors [6].

IR coding method is one of the typical error-resilience tools that are widely used in conventional streaming video systems. When the streaming encoder performs motion estimation to code MBs, it decides their coding mode (intra, inter, or skip) using RDO. However, the IR method forces MBs of each frame to be encoded with intracoding mode. The intracoded MBs reduce error propagation and improve the robustness to transmission errors without significantly increasing the RD cost. Additionally, to guarantee that all MBs are eventually refreshed and errors do not propagate indefinitely, random intrarefresh (RIR), which selects MBs randomly in a cyclic mode, can be used. However, the RIR method has some limitations, because RIR randomly decides the locations of intracoded MBs. That is, RIR does not recognize the generated bit-rate difference between moving objects and background area in video sequences [7, 8]. Additionally, the error refresh capabilities can be weakened when multiple reference frames are used in motion compensation, because the damage is propagated into the following frames, even though MBs in the previous frame have been refreshed via the IR method [3, 4].

*2.3. Video Streaming over RTP and RTCP.* After encoding an input video stream via the video coding layer, the H.264/AVC encoder packetizes the encoded bits into RTP packets for network transmission. RTP is a transport layer protocol that has been developed to carry the encoded video sequence on top of IP and UDP [9]. RTP's companion protocol, RTCP, is used to monitor the transmission status of the media data and provide feedback information including the reception quality [10, 11]. The streaming video server multicasts RTP video packets with sender report (SR) type of RTCP to all clients and the clients reply with receiver reports (RRs) to inform the sender and other receivers about the quality of service. In this way, RTCP RR packets can provide end-to-end feedback information about delay jitter and packet-loss performance [12, 13]. Based on this feedback channel information, the encoder can change its coding strategy to reduce errors and adapt to changing network conditions.

## 3. Proposed Network-Aware Streaming Video System

*3.1. Requirement of Streaming Video System.* Typically, the number of reference frames is set to be large for the high coding efficiency that is set up during the initial video

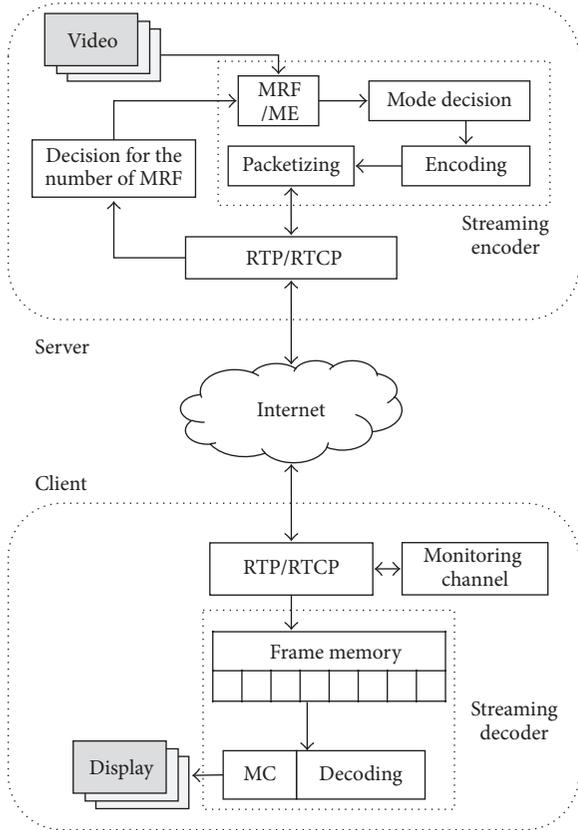


FIGURE 1: Proposed streaming server and client system.

encoding. This multiple reference frame structure is preferred to single reference frame in the recent video coding methods, even though it requires a large amount of frame memory and computation power for motion estimation and compensation. However, the multiple reference frame coding method combined with typical error-resilience function such as RIR has been found to make the overall video quality worse in an erroneous network [3, 4].

Typical error-resilience methods, that is, FMO [2], ASO, and DP, can not maintain their resilience features under the multiple reference frame structure. Therefore, there should be a strategy combining multiple reference frame structure for high coding efficiency and resilience function for strong error-resilience feature. The strategy is introduced in this paper.

**3.2. Proposed Network-Aware Error Resilience.** In this section, a network-aware error-resilient streaming video system is proposed. The proposed system monitors RTCP feedback messages (including the channel status) delivered by the client and manages the number of reference frames so as to mitigate error propagation. Additionally, the number of MBs to be intracoded forcefully per frame can be added in this procedure.

Figure 1 shows the proposed streaming server and client system. The streaming server has additional functions to monitor the channel status delivered in the form of RTCP

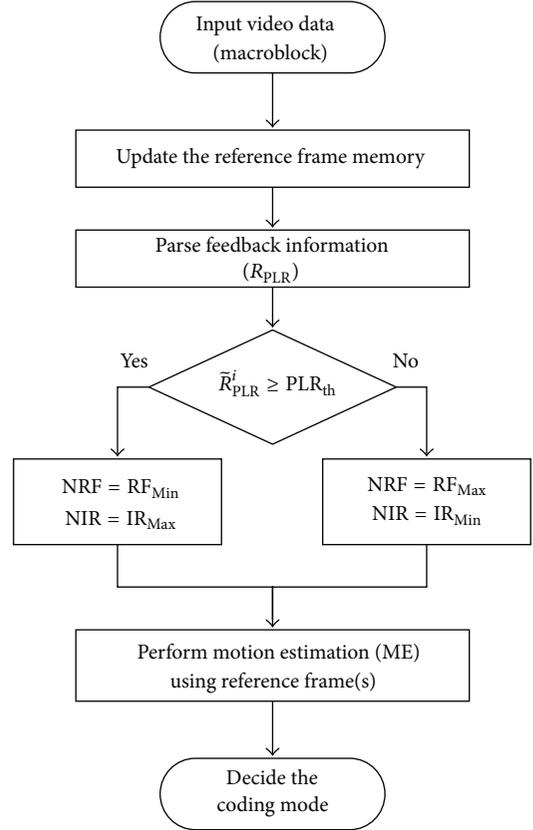


FIGURE 2: Network-aware decision for the number of reference frames and MBs used in intrarefresh coding.

packets and to change the number of reference frames and intracoded MBs being inserted into a frame based on the channel status. After deciding on a suitable number of reference frames and MBs, the streaming server encodes the video and assembles it into RTP packets. The size of a packetized unit is decided by the input parameters of the encoder; then, the packet is delivered to network [14]. At the streaming client, the quality of the decoded video sequence should be observed and returned to the streaming server to control the error resilience. However, it is difficult to measure the quality of the decoded video because there are no undamaged reference frames at the client side. Therefore, the proposed streaming client measures the refined packet loss ( $R_{PLR}$ ) instead of the decoded video quality.  $R_{PLR}$  is smoothed by exponential weighted moving average (EWMA) method that is defined as

$$\tilde{R}_{PLR}^i = (1 - \alpha) \tilde{R}_{PLR}^{i-1} + \alpha R_{PLR}, \quad (1)$$

where  $\alpha$  is a weighting factor that defines the acceleration of averaging and  $\tilde{R}_{PLR}^i$  is the estimated packet-loss ratio (PLR). The delivered  $\tilde{R}_{PLR}$  value in RTCP packet is compared to the predefined PLR threshold  $PLR_{th}$  at the streaming server to determine the number of reference frames. The procedure for PLR comparison and controlling the number of reference frames and intracoded MBs is shown in Figure 2.

TABLE 1: Encoder parameters.

Profile level	Baseline (66)
GOP structure	IPPP...
Bit rate	400 kbps (CBR)
Num. of reference frames	1, 7 (1~7)
Num. of intrarefresh MBs	6, 18
Entropy coding	CAVLC
Packet type	RTP
Reference S/W ver.	JM 17.2 [15]

We present a strategy for achieving high error robustness in the multiple reference frame based streaming video encoding system. The proposed streaming server reduces the number of reference frames (NRF) to  $RF_{\min}$  which is the smallest number of reference frames, when  $\bar{R}_{\text{PLR}}$  is greater than  $\text{PLR}_{\text{th}}$ . When there are video frames that are found damaged, the streaming video system makes the number of reference frames small. Then, refreshing feature of the intracoded MBs can be effective [3, 4]. That is, more blocks refreshed by RIR coding will be used for further motion compensation of the next frames in multiple reference frame structures. After the transmission channel becomes stable, that is,  $\bar{R}_{\text{PLR}}$  being less than  $\text{PLR}_{\text{th}}$ , the streaming video encoder increases the number of reference frames up to  $RF_{\max}$ , which is the largest number of reference frames. Additionally, the higher number of MBs for intracoded mode, NIR, can be used together with the proposed system to achieve higher error resilience because the intrarefresh will be more effective when the error propagation has been mitigated.

In this way, the proposed system can strike a balance between high coding efficiency (for the stable channel) and error robustness (for the unstable channel).

## 4. Performance Evaluation

*4.1. Experimental Setup.* The proposed network-aware reference frame control system is expected to achieve both coding efficiency and error robustness by monitoring the channel status and making suitable adjustments.

For the experiments, reference software of H.264/AVC standard named JM is used. As it is well known, JM includes encoder, decoder, and rtp\_loss model. Specifically, the proposed error-resilience method has added to the encoder of JM for the proposed streaming video system. Table 1 shows the encoder parameters used in the experiments. Both the proposed network-aware streaming video system and the conventional streaming system use the same encoding parameters. Additionally, both systems encode the same test sequences shown in Table 2 in baseline profile and packetize the compressed video sequences into RTP packets. To compare the performance of the proposed system under varying conditions, input video sequences depicting different motion activities are used. For example, the *akiyo* and *carphone* sequences have slow motion and a static background. Thus, they are less damaged than the high-activity video sequence

TABLE 2: Test video sequences.

Input video	<i>akiyo</i>	<i>carphone</i>	<i>football</i>	<i>soccer</i>
Total frames	240 frames			
Frame rate	30 fps			
Resolution	CIF (352 × 288)			
Motion activity	Slow	Slow	Fast	Fast

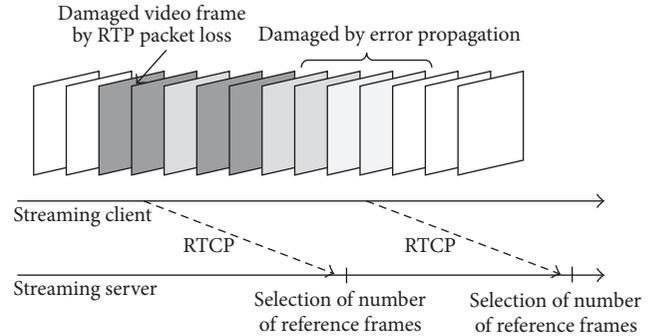


FIGURE 3: Simulation scenarios for the proposed network-aware reference frame control system.

when video packets are lost. In contrast, the *football* and *soccer* sequences consist of fast motion.

To simulate the various error-prone wireless networks, both a bursty pattern and a random pattern for packet losses are created as shown in Figure 3. Also, in order to observe the performance of error resilience reducing error propagation, the severe channel error conditions should be considered. That is, 20% packetized frames are lost in the forms of burst error and random error over one second at the simulation time of 2 seconds after starting the streaming video. To decode and analyze the quality of damaged video stream, decoder of JM is used. If the transferred video packets are lost, the frame copy is used for error concealment at the decoder. The  $\bar{R}_{\text{PLR}}$  observed at the streaming client is delivered to the server and is expected to be received after a transmission time of  $T_{\text{Trans}}$ .  $T_{\text{Trans}}$  is determined to have a uniform distribution between 1 s and 2 s. Therefore, the high coding efficiency is required before channel error starts, while error-resilience feature is more expected after channel errors. In our experiment, we set the the minimum and maximum numbers of reference frames,  $RF_{\min}$  and  $RF_{\max}$ , to 1 and 7, respectively. At the same time, 6 and 18 intracoded MBs per CIF picture have been forcefully generated for RIR.

In this experiment, the proposed network-aware reference frame control system using either  $RF_{\min}$  or  $RF_{\max}$  based on channel condition, named NARF, is compared to two conventional streaming systems, NRF1 and NRF7. Here, NRF1 and NRF7 refer to the conventional system using the static number of reference frames of 1 and 7, respectively.

*4.2. Experimental Results and Analysis.* For the first error-resilience experiment, we applied a burst error pattern to simulate a worse situation like handover in the mobile network. As shown in Figure 3, the damage to the test sequences is observed for a duration of 1 s at the simulation

TABLE 3: Average PSNR values observed for test sequences under the burst error conditions.

	<i>akiyo</i>		<i>carphone</i>		<i>football</i>		<i>soccer</i>	
	RIR6	RIR18	RIR6	RIR18	RIR6	RIR18	RIR6	RIR18
	Phase 1 (error-free)							
NRF1	43.97	43.43	37.71	37.44	27.76	27.61	33.06	32.82
NRF7	44.20	43.67	38.28	38.00	27.83	27.74	33.21	32.93
NARF	44.20	43.67	38.28	38.00	27.83	27.74	33.21	32.93
	Phase 2 (after burst error)							
NRF1	31.74	36.11	26.23	25.87	26.74	28.20	28.40	27.00
NRF7	32.23	34.07	25.32	27.22	21.04	24.92	27.33	22.54
NARF	32.88	36.90	26.43	27.83	22.89	26.79	27.83	27.54

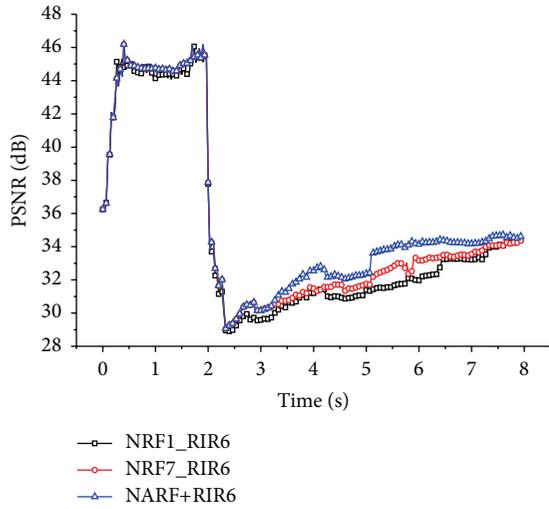


FIGURE 4: *akiyo* sequence under the burst error condition with RIR6.

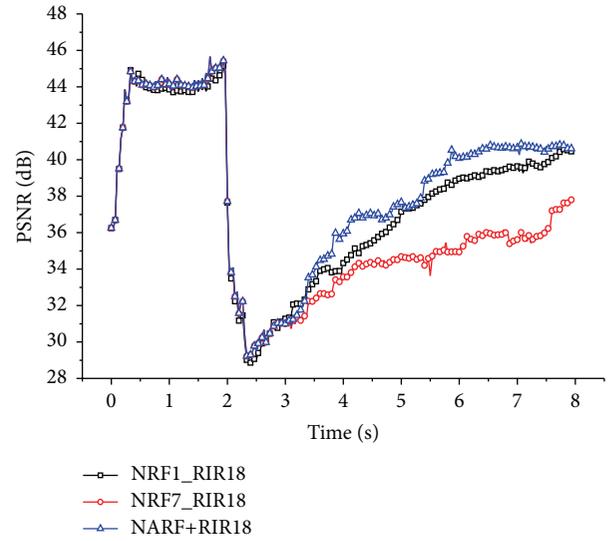


FIGURE 5: *akiyo* sequence under the burst error condition with RIR18.

time of 2 s. All PSNR results of test sequences are shown in Figures 4–11. Before the packet losses occur, the conventional system of NRF7 shows the more enhanced PSNR result at all test sequences than NRF1 (as shown in Table 3). However, the PSNR result of NRF7 can be worse as the video damage is accumulated and propagated, even though 6 and 18 intrarefresh MBs are inserted. That is, the error-resilience feature of RIR does not work properly under the condition of multiple reference frame structure. However, NRF1 tends to generate the better error recovery feature because its single reference frame structure can reduce the error propagation. Specifically, NRF1 in both *football* and *soccer* sequences shows the stronger error resilience than that in both *akiyo* and *carphone* sequences, because *football* and *soccer* sequences consisting of faster motions have more serious error propagation than others. Also, we could measure another trade-off between NRF1 and NRF7. When small number of previous frames, that is, one or two frames, are damaged, NRF7 partially shows better PSNR performance than NRF1 because NRF7 can perform the motion compensation in the current frame with the blocks in the far-located (undamaged) frames in the multiple reference frame. Therefore, NRF1 does not always show better PSNR than NRF7 even after the packet losses are detected.

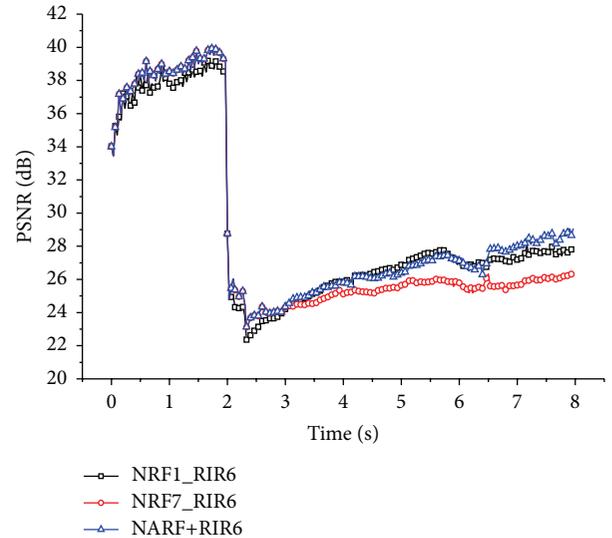


FIGURE 6: *carphone* sequence under the burst error condition with RIR6.

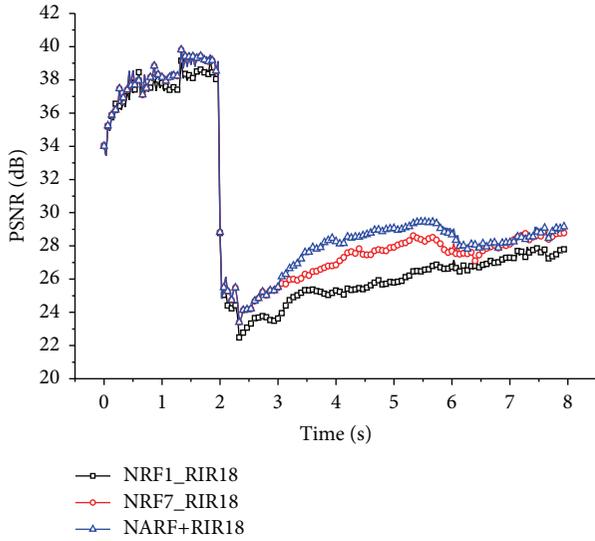


FIGURE 7: *carphone* sequence under the burst error condition with RIR18.

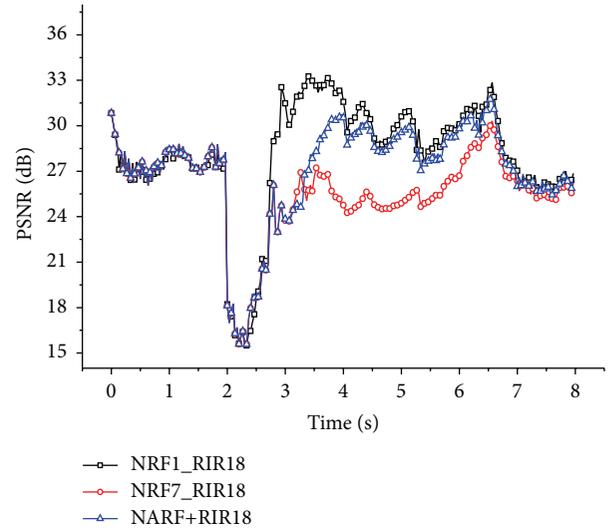


FIGURE 9: *football* sequence under the burst error condition with RIR18.

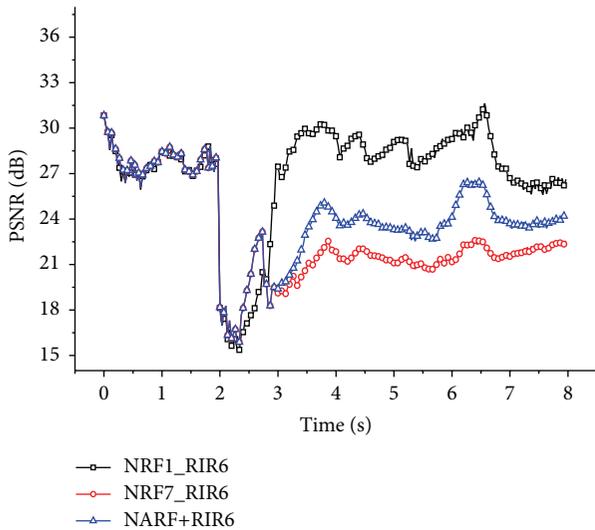


FIGURE 8: *football* sequence under the burst error condition with RIR6.

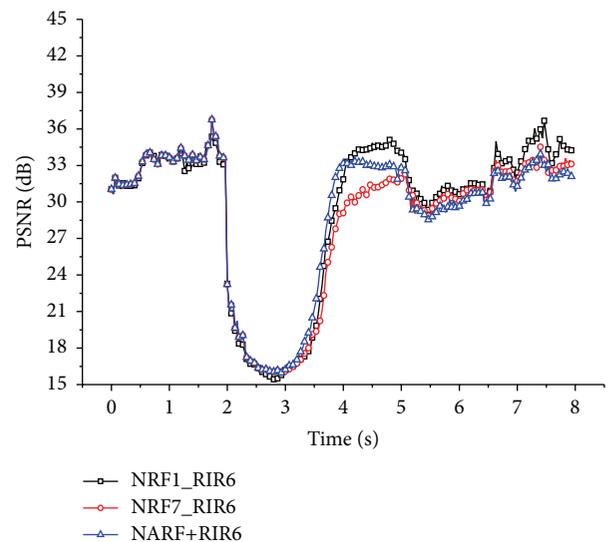


FIGURE 10: *soccer* sequence under the burst error condition with RIR6.

On the other hand, the proposed NARF system exhibits stronger error robustness than NRF7 for all test sequences because of its channel adaptiveness. The proposed NARF uses  $RF_{Max}$  for the higher coding efficiency at the error-free condition, whereas it uses  $RF_{Min}$  for the error restoration performance after recognition of channel errors. Indeed, NARF maintains  $RF_{Max}$  during the time period when channel errors are notified to the streaming server. This coding balance between  $RF_{Max}$  and  $RF_{Min}$  makes the proposed NARF perform better than NRF7 all the time and better than NRF1 in some error conditions. It can be effective at both coding efficiency and error robustness. Figure 12 presents representative images of the *soccer* sequence encoded under the various reference frame conditions. Additionally, the average PSNR values observed for the test sequences are

shown in Table 3. Here, phase 1 implies the error-free period from 0 s to 2 s at the simulation time, while phase 2 indicates the erroneous period from 2 s to the end of simulation time. During phase 1, NRF7 and NARF using  $RF_{Max}$  perform better than NRF1. However, during phase 2, NRF1 and NARF using  $RF_{Min}$  perform better than NRF7. That is, the proposed NARF manages its encoding to work like NRF7 at the error-free time and NRF1 at the erroneous time.

For the second error-resilience experiment, we applied the random error pattern to the simulation which is created by random function in the standard C library with `rtp_loss` model in the JM reference software. Here, the same encoding conditions as in the first experiment (shown in Tables 1 and 2) are used.

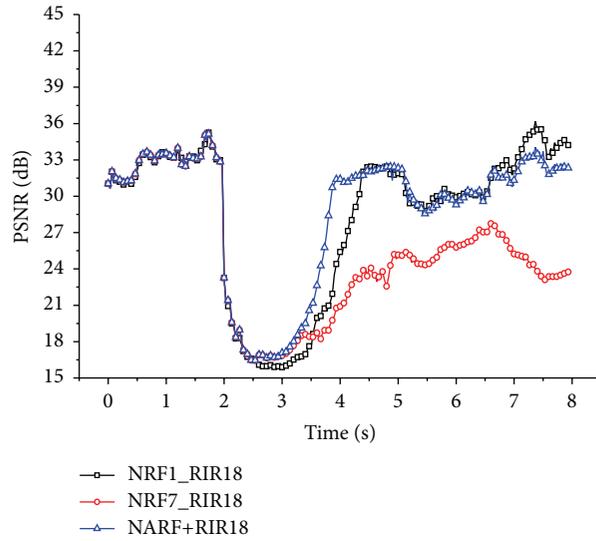


FIGURE 11: *soccer* sequence under the burst error condition with RIR18.



FIGURE 12: Visualized *soccer* sequence under the burst error conditions, (a) no damage, (b) NRF1, (c) NRF7, and (d) NARF.

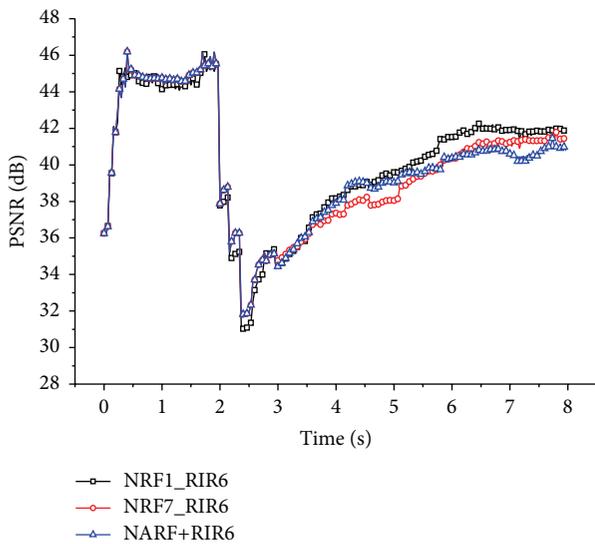
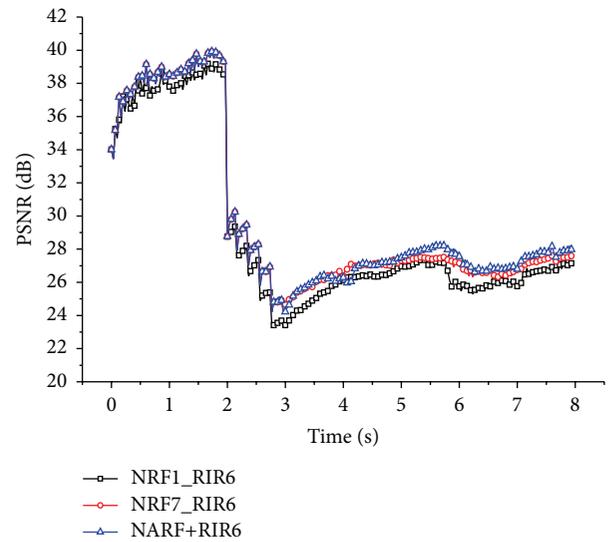
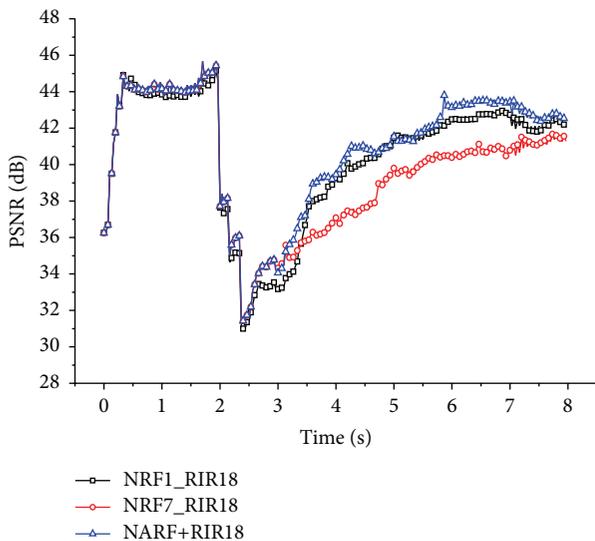
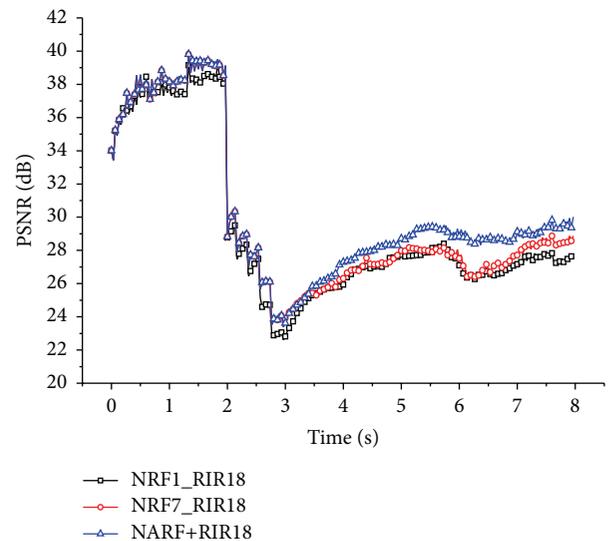
Experimental results of NARF and conventional streaming systems performed under the random errors are shown in Figures 13–20. From the simulation time of 2 s, random number of frames are lost during 1 s. As the same with the first simulation, the NARF system shows better coding efficiency and error resilience than the cases of NRF1 and NRF7 in both cases of RIR6 and RIR18. Figure 21 presents representative

images of *football* sequence under various reference frame conditions. The average PSNR values observed for the test sequences are shown in Table 4.

The observation results indicate that the proposed streaming method is effective at achieving more reliable transmission of streaming video because it strikes a balance between coding efficiency and error-resilience features.

TABLE 4: Average PSNR values observed for test sequences under the random error condition.

	<i>akiyo</i>		<i>carphone</i>		<i>football</i>		<i>soccer</i>	
	RIR6	RIR18	RIR6	RIR18	RIR6	RIR18	RIR6	RIR18
	Phase 1 (error-free)							
NRF1	43.97	43.43	37.71	37.44	27.76	27.61	33.06	32.82
NRF7	44.20	43.67	38.28	38.00	27.83	27.74	33.21	32.93
NARF	44.20	43.67	38.28	38.00	27.83	27.74	33.21	32.93
	Phase 2 (after random error)							
NRF1	38.98	39.49	26.18	26.69	22.92	28.26	24.96	27.11
NRF7	38.44	38.37	26.93	27.13	22.87	26.53	24.59	27.34
NARF	38.51	40.13	27.11	27.94	23.51	27.86	28.56	27.22

FIGURE 13: *akiyo* sequence under the random error conditions with RIR6.FIGURE 15: *carphone* sequence under the random error conditions with RIR6.FIGURE 14: *akiyo* sequence under the random error conditions with RIR18.FIGURE 16: *carphone* sequence under the random error conditions with RIR18.

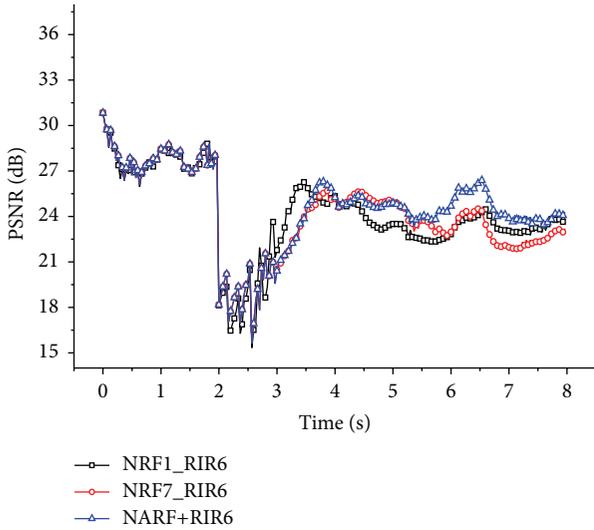


FIGURE 17: *football* sequence under the random error conditions with RIR6.

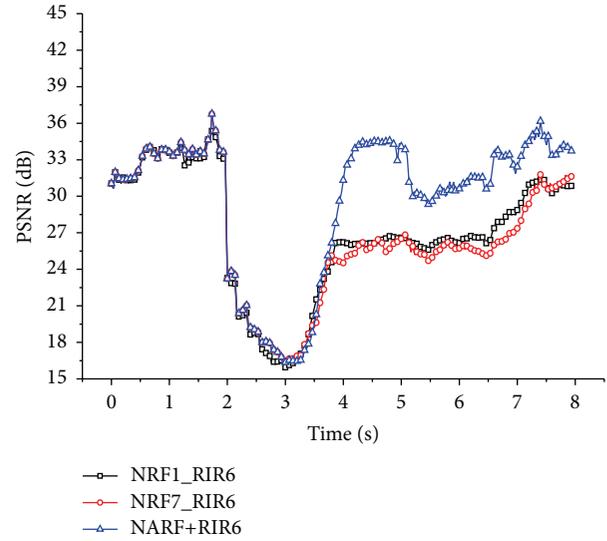


FIGURE 19: *soccer* sequence under the random error conditions with RIR6.

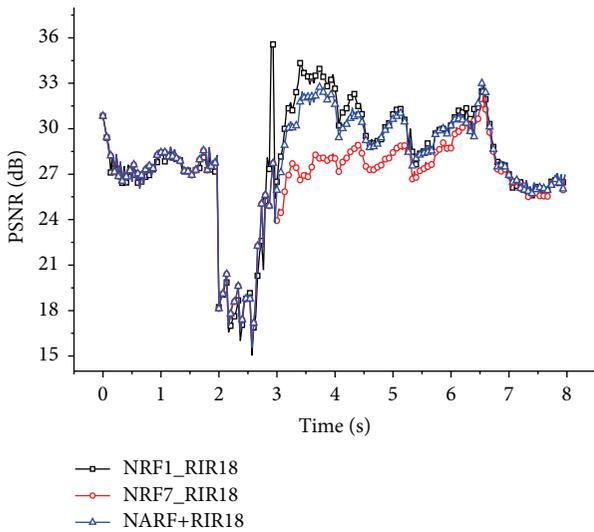


FIGURE 18: *football* sequence under the random error conditions with RIR18.

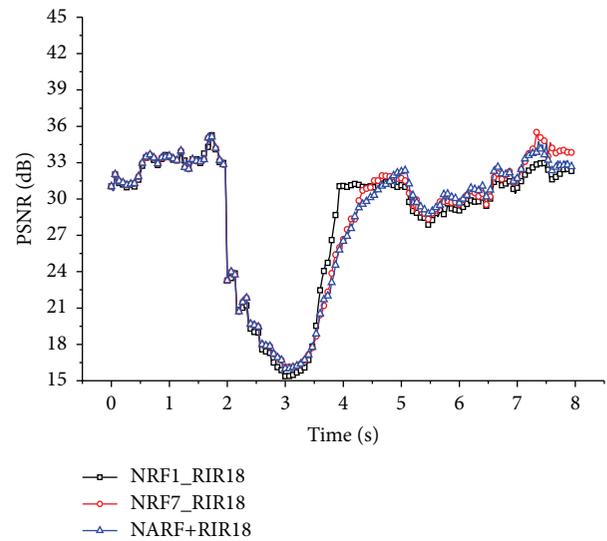


FIGURE 20: *soccer* sequence under the random error conditions with RIR18.

### 5. Conclusion

The requirements of reliable real-time streaming video services in the recent mobile and wireless communication environments are enormous. However, these channels remain unreliable while consumer demand for streaming video increases. Therefore, video coding standards typically include both error-resilience tools to cope with the error propagation and multiple reference frame structures to achieve higher coding efficiency. However, error-resilience tools decrease coding efficiency. In addition, the multiple reference frame structure used for higher coding efficiency in motion estimation and compensation interferes with conventional error-resilience tools. In this paper, we propose a network-aware

reference frame control system that keeps a balance between coding efficiency and error resilience based on the channel status. Experimental results show that the proposed video streaming system provides better PSNR approximately from 0.2 to 0.5 dB than conventional video streaming system (NRF1) using single reference frame when no video frames are damaged and better PSNR from 0.3 to 3 dB than conventional system (NRF7) using 7 reference frames when the video frames are damaged. That is, the proposed streaming video system maintains the high video quality by controlling the number of reference frames based on the channel status. In addition, the proposed system can adopt the channel-adaptive intrarefresh methods to increase the error robustness.

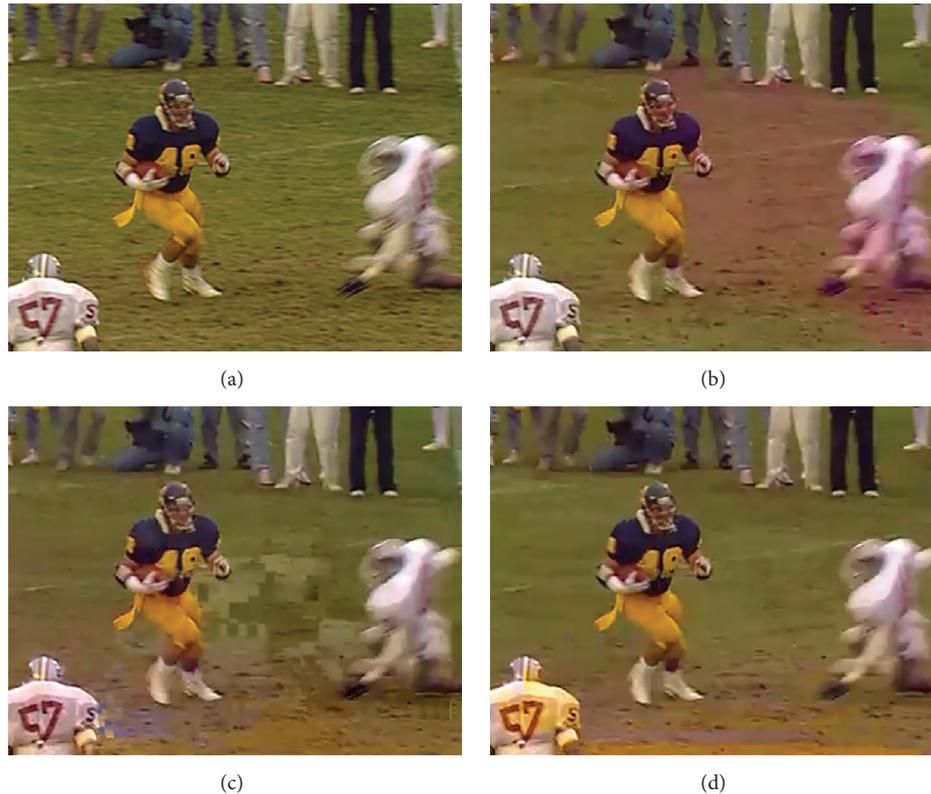


FIGURE 21: Visualized football sequence under the random error conditions, (a) no damage, (b) NRF1, (c) NRF7, and (d) NARF.

## Competing Interests

The authors declare that they have no competing interests.

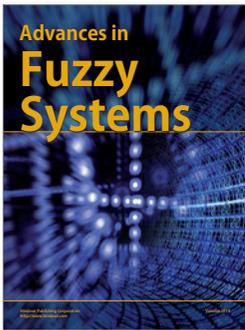
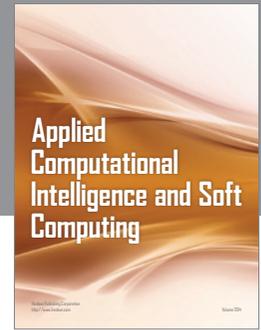
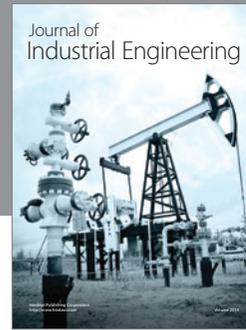
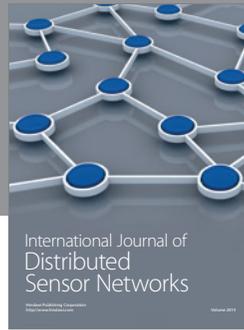
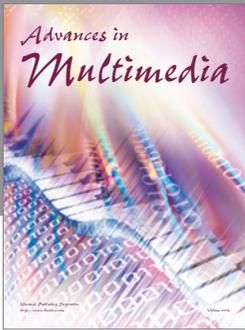
## Acknowledgments

This study was supported by research funds from Chosun University, 2015.

## References

- [1] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.
- [2] T. H. Vu and S. Aramvith, "An error resilience technique based on FMO and error propagation for H.264 video coding in error-prone channels," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '09)*, pp. 205–208, New York, NY, USA, July 2009.
- [3] S. Moiron, I. Ali, M. Ghanbari, and M. Fleury, "Limitations of multiple reference frames with cyclic intra-refresh line for H.264/AVC," *Electronic Letters*, vol. 47, no. 2, pp. 103–104, 2011.
- [4] S. I. Chowdhury, J.-N. Hwang, P.-H. Wu, G.-R. Kwon, and J.-Y. Pyun, "Error resilient reference selection for H.264/AVC streaming video over erroneous network," in *Proceedings of the IEEE International Conference on Consumer Electronics (ICCE '13)*, pp. 418–419, IEEE, Las Vegas, Nev, USA, January 2013.
- [5] P. Nunes, L. D. Soares, and F. Pereira, "Error resilient macroblock rate control for H.264/AVC video coding," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '08)*, pp. 2132–2135, San Diego, Calif, USA, October 2008.
- [6] Y. Xu and C. Zhu, "End-to-end rate-distortion optimized description generation for H.264 multiple description video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 9, pp. 1523–1536, 2013.
- [7] R. M. Schreier and A. Rothermel, "Motion adaptive intra refresh for the H.264 video coding standard," *IEEE Transactions on Consumer Electronics*, vol. 52, no. 1, pp. 249–253, 2006.
- [8] X. Wang, H. Cui, and K. Tang, "Attention-based adaptive intra refresh method for robust video coding," *Tsinghua Science and Technology*, vol. 17, no. 1, pp. 67–72, 2012.
- [9] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: a transport protocol for real-time applications," Tech. Rep. RFC 3550, 2003.
- [10] M. Sivabalakrishnan and D. Manjula, "Analysis of decision feedback using RTCP for multimedia streaming over 3G," in *Proceedings of the International Conference on Computer and Communication Engineering (ICCCCE '08)*, pp. 1023–1026, IEEE, Kuala Lumpur, Malaysia, May 2008.
- [11] H. Gharavi, K. Ban, and J. Cambiotis, "RTCP-based frame-synchronized feedback control for IP-video communications over multipath fading channels," in *Proceedings of the IEEE International Conference on Communications*, vol. 3, pp. 1512–1516, June 2004.
- [12] N. Baldo, U. Horn, M. Kampmann, and F. Hartung, "RTCP feedback based transmission rate control for 3G wireless multimedia streaming," in *Proceedings of the 15th IEEE International*

- Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '04)*, vol. 3, pp. 1817–1821, IEEE, September 2004.
- [13] L.-L. Wang and J.-D. Lu, “IPTV quality monitoring system based on hierarchical feedback of RTCP,” in *Proceedings of the International Conference on Electronics, Communications and Control (ICECC '11)*, pp. 870–873, Ningbo, China, September 2011.
- [14] S. Wenger, “H.264/AVC over IP,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 645–656, 2003.
- [15] K. Sühring, *H.264/AVC Reference Software—JM 17.2*, 2010, <http://iphome.hhi.de/suehring/tml/>.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

