A Robust H.264 Video Streaming Scheme for Portable Devices

Xiaosong Zhou, Wei-ying Kung and C.-C. Jay Kuo
Department of Electrical Engineering and Integrated Media Systems Center
University of Southern California, Los Angeles, CA 90016
E-mails: xiaosonz@usc.edu, weiyingk@ieee.org, cckuo@sipi.usc.edu

***Abstract*—We propose a novel error resilient video coding and streaming scheme specially designed for the streaming of H.264 coded video to portable devices in this work. The scheme is based on the error feedback and tracking techniques, and it utilizes the SP/SI coding method introduced in H.264. During the video streaming process, the sender identifies macroblocks that have been affected by previous errors based on the feedback information from the receiver. Then, it re-encodes these macroblocks into secondary SP macroblocks with predictions from concealed reference frames and uses them to replace the originally coded version in the bit stream. As proven by experimental results, the scheme is capable of stopping or reducing the error propagation effect caused by transmission errors. It demands a small bit rate overhead if there is a transmission error, and no bit rate overhead, otherwise. Experimental results are provided to demonstrate the error resilient performance of the proposed scheme.**

I. INTRODUCTION

The number of portable devices in the market and their computational capability are growing very fast these days. Devices such as pocket PCs and multi-functional cellular phones have started to include the function of playing video contents with acceptable quality. The increasing bandwidth of wireless networks has made video streaming to these devices popular. As the H.264 video coding standard [1] is shown to outperform other existing video coding standards in terms of coding efficiency over a wide range of bit rates, it is expected that H.264 will find numerous applications, which include the streaming of video contents to portable devices.

However, the characteristics of wireless channels are not friendly to the transmission of highly compressed H.264 video. Transmission errors and corrupted or dropped data packets may appear frequently, they will affect not only the current video frame but also subsequent frames. This phenomenon, called error propagation, may significantly degrade the quality of received video for a long period of time. Therefore, it is important to develop an effective error resilient tool to stop or reduce the error propagation effect in video streaming.

Our scheme aims at completely stopping or largely reducing propagation errors. It is specially designed for H.264 video streaming to portable devices. It is able to overcome the challenges from the varying and long delay of wireless networks and the limited computation and memory resources of portable devices. The scheme is proven to work effectively in improving the error resilience performance of H.264 coded video, and it can be adapted to work for a wide range of video streaming applications. This work is an extension of our previous work on error resilient H.264 video coding [2].

The rest of this paper is organized as follows. A brief overview of error resilient tools including our previous work is given in Section II. In Section III, the proposed scheme is described in detail. We present a number of experimental results to demonstrate the performance of the proposed scheme in Section IV. Concluding remarks are given in Section V.

II. ERROR RESILIENT TOOLS FOR H.264

Various error resilient tools have been introduced before. Error resilient techniques of earlier video standards such as Intra macroblock refreshing, data partitioning and data embedding, etc. have been adapted to the context of H.264. The H.264 standard also includes a number of error resilient features such as the FMO (Flexible Macroblock Ordering) coding method. Error concealment tools [4] can also be implemented in the H.264 decoder to enhance the video quality of corrupted frames. As discussed in [2], these techniques provide good error resilient performance but may fail to completely stop or effectively reduce propagation errors. Some of them may not be suitable for video streaming applications.

Error propagation is mainly caused by the mismatch between the encoder and the decoder since the reference frames used at the decoder may be corrupted by errors and, therefore, the subsequent frames cannot be decoded correctly and errors start to propagate in these frames. In order to compensate the mismatch, several error resilient schemes designed for previous standards based on the error feedback have been proposed [5-7]. In these schemes, the sender (encoder) obtains the error feedback information from the receiver (decoder) so that it can encode subsequent frames accordingly to avoid the mismatch. But all of them require the image sequence to be encoded in real time. For many scenarios, image sequences (especially those for commercial video contents) are encoded off-line before the streaming takes place. Furthermore, real-time H.264 encoding demands great computational power and can be expensive. Therefore, these existing schemes may not work well for H.264 video streaming applications.

In [2], we proposed a novel error resilient scheme specifically designed for H.264. It utilizes SP/SI encoding [3] and works for off-line coded image sequences. Specifically, for each encoded macroblock, we encode additional versions of it with different reference frames and/or different predictions and save them as secondary SP/SI macroblocks, called alternatively coded versions. During transmission, these alternative SP/SI macroblocks are used to replace the originally coded versions in the output video stream if they are affected by previous errors detected by the receiver.

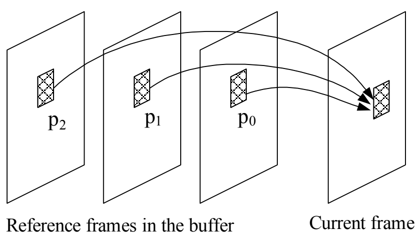

Fig. 1: Example of alternative SP coding.

An example of alternative SP coding is shown in Fig. 1. Suppose that three reference frames can be used for motion estimation of the current frame and macroblock M is coded using one of these references, which we call the originally coded version. The other two predictions of macroblock M are used to encode two different versions of secondary SP macroblocks, where the reconstruction values of the originally coded version are used as input. During video streaming, the sender receives error feedback information from the receiver, if such information indicates that the prediction used in the originally coded version is affected by errors, the sender can send either of the two alternatively coded versions instead. Due to the special secondary SP encoding and decoding method introduced in [2], the decoder is able to reconstruct exactly identical or very close reconstruction values to the originally coded version from any of alternative SP macroblocks. Therefore, no significant mismatch will be introduced and error propagation is successfully stopped.

The scheme is proven to effectively stop error propagation and can be used in general video streaming applications. It works well with varying but relatively short network delays. However, if the network delay is long, when the sender receives the feedback from the receiver, error propagation has affected all available reference frames. Then, the above method may fail since all of alternatively coded versions are also affected by errors. On the other hand, due to the limited computational power and memory storage of portable devices, the decoder may only allow a very small number of reference frames for motion compensation, which exaggerates the problem described above.

To solve this problem, as discussed in [2], we can create multiple predictions from a single reference frame to encode SP macroblocks. These predictions have to be un-correlated to avoid them being affected by the same errors. An example of these multiple predictions is shown in Fig. 2. The problem of this approach is that the SP macroblocks will consume a lot of bits since it may be coded with bad predictions. Another way to solve the problem is to encode alternative SI macroblocks. However, it will introduce big bit rate overhead as well. The scheme proposed in this work aims to solve the problem while maintaining high coding efficiency of the coded bit stream.

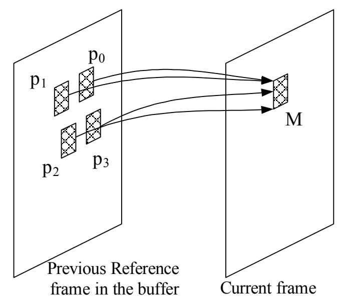

Fig. 2: Multiple predictions in a single frame.

III. PROPOSED ERROR RESILIENT STREAMING SCHEME

Although the error concealment tools implemented at the decoder cannot be used to completely stop error propagation, such tools still help reducing artifacts of corrupted video frames and improving the PSNR of received video. Corrupted video frames concealed by the decoder are stored as reference for subsequent frames. Since these concealed frames are different from the reference frames used at the encoder, the mismatch is introduced and the subsequent frames become vulnerable of propagation errors. However, if the encoder obtains the exact concealed frames based on the error feedback information and the knowledge of error concealments tools used by the decoder, such a mismatch can be avoided using the following coding and streaming methods.

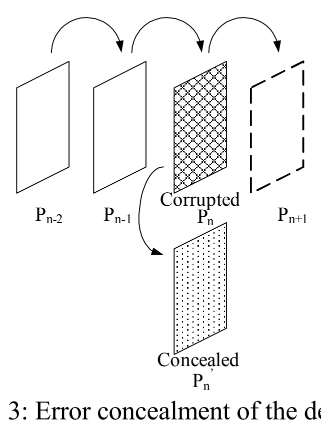

Fig. 3: Error concealment of the decoder.

Fig. 3 shows the video frames received at the decoder during the streaming of encoded H.264 stream. Suppose that a transmission error occurs during transmission of frame P_n and corrupts the received frame. Upon detection of the transmission error, the decoder conceals the corrupted frame using error concealment tools and stores the concealed P_n' as a reference frame in its buffer. At the same time, the error information is sent back to the sender using the feedback channel. After receiving the feedback for frame P_n, the sender is able to generate the concealed frame P_n', to conceal motion vectors and other parameters exactly the same as at the decoder, using the same error concealment tool.

By calculating the dependency between frames P_n and P_{n+1}, the sender can identify the macroblocks in frame P_{n+1} that are affected by the mismatch between P_n and P_n'. These macroblocks will be re-encoded in real time using the concealed reference frame P_n'. As shown in Fig. 4, they are encoded into secondary SP macroblocks with the reconstruction of their originally coded version as input, and their predictions are obtained from motion estimation within concealed frame P_n'. These SP macroblock will then be used to replace their originally coded version in the bit stream and sent to the receiver. No primary SP macroblocks are encoded. As discussed in [2], by appropriately choosing the quantization step size, the decoder can generate an identical or very close reconstruction as the originally coded version from the SP macroblock as shown in Fig. 4. Thus, no significant mismatch will be introduced when these macroblocks are used as the references, the rest of the encoded bit stream will not be affected by the replacements and can be decoded correctly.

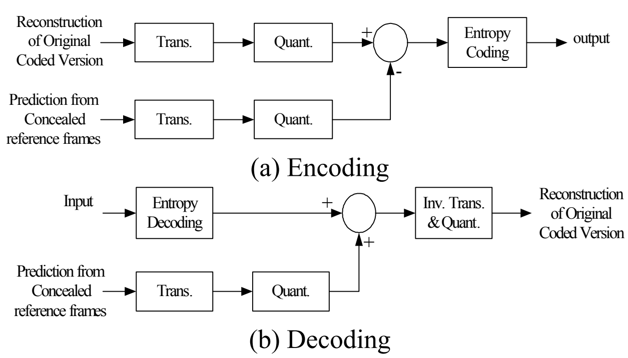

Fig. 4: SP encoding and decoding.

If proper error concealment tools are used to conceal the corrupted frame, it is expected that the motion estimation within the concealed video frame can provide a good prediction for the macroblock being coded. But these SP macroblocks generally still consume more bits than the originally coded version due to the lower coding efficiency of SP macroblocks. However, since these SP macroblocks are only encoded and transmitted when transmission errors occur and macroblocks are found to be affected by errors, the overall bit rate overhead will be small in most cases.

The scheme is designed for the streaming of off-line coded H.264 image sequences. The sender has to calculate and identifies affected macroblocks, re-generate the concealed frames and, when needed, encode the SP version for those affected macroblocks. The overall computational complexity is much lower than a real-time H.264 encoder that encodes the whole video stream on the fly. The scheme does not require multiple reference frames to be used in motion compensation, which saves the computational power and memory consumptions at the decoder and fits portable devices well.

The above example demonstrates how the scheme works under short network delay as the sender receives the error feedback immediately after the receiver detects an error. When the network delay becomes longer, the scheme has to be adjusted to work effectively.

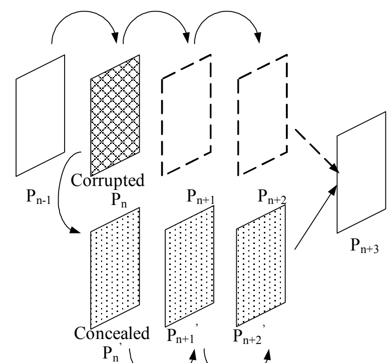

Fig. 5: The long delay scenario.

As shown in Fig. 5, when the sender receives the feedback regarding the transmission error occurring at frame P_n, two subsequent frames P_{n+1} and P_{n+2} have already been transmitted to the receiver and it is expected that errors will propagate in these two frames. In this case, the encoder still generates the concealed version of P_n as P_n'. Furthermore, based on P_n' and the coding parameters, it also generates frame P_{n+1}' and P_{n+2}' that are identical to the reconstructed frames of P_{n+1} and P_{n+2} at the decoder. Then, using these re-generated reference frames, the sender encodes the affected macroblocks in P_{n+3} into SP macroblocks and sends them in the output bit stream to stop error propagation. By following this method, our scheme works effectively under long network delays.

IV. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed scheme in wireless networks, we simulated the network packet loss conditions by generating burst errors patterns. In our test, the error concealment tools described in [4] and implemented in the reference codec (JM7.6) are used to conceal corrupted frames. FMO is enabled as it improves the performance of the error concealments tools when encountering burst errors.

In [2], we showed that our combined alternative SP/SI coding scheme outperforms other error resilient tools such as intra refreshing for the one reference frame case. In order to prove the performance of the proposed scheme in this work, we compare the R-D performance of the scheme with that of the combined SP/SI approach under different error rates.

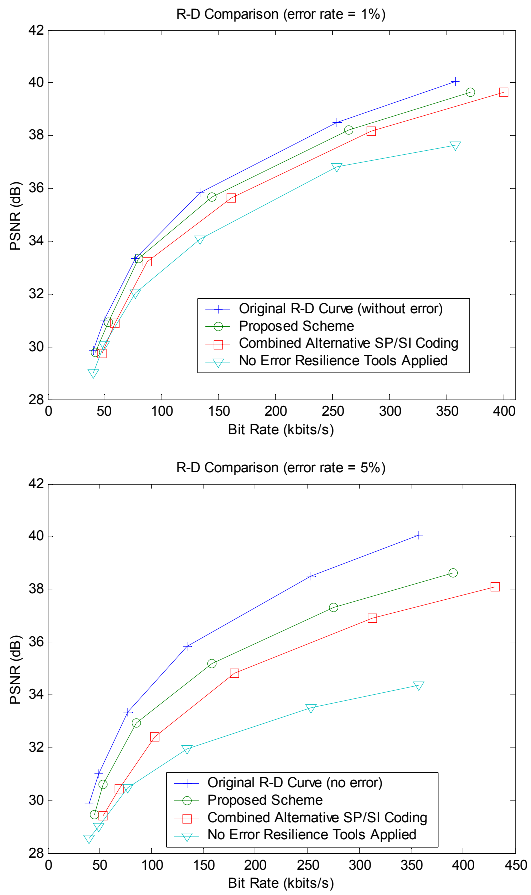

Fig. 6: The R-D performance of the proposed error resilient scheme (Foreman QCIF sequence, short delay) with two error rates 1% and 5%.

A couple of experimental results are shown in Fig. 6, where the original R-D curve without errors and the R-D curve with no error resilient scheme applied (only the concealment is used) are also shown. The proposed scheme demonstrates excellent performance. When the error rate is relatively lower, the PSNR values of the proposed scheme and the combined SP/SI approach are similar and very close to the original value. This is because that both of the two approaches are able to stop error propagation effectively and the slight PSNR degradation mostly comes from the concealment of corrupted frames. When the error rate becomes higher, the proposed scheme is still able to maintain the PSNR level. Besides, it outperforms the combined SP/SI approach significantly in bit rate savings.

We also test the proposed scheme under different network delays. The results are shown in Fig. 7. The delays are measured by the time eclipsed (in frame intervals) when the sender receives the error feedback after transmitting the frame. As shown in these figures, the scheme provides good error resilient performance under different network delays. The PSNR degradation is due to the fact that errors already propagate in frames transmitted before the sender receives the error feedback.

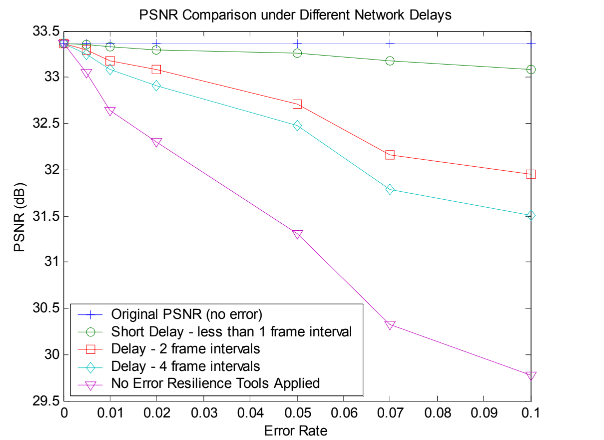

Fig. 7: PSNR under different delays (Foreman, QCIF)

V. CONCLUSION

The proposed error resilient video streaming scheme can stop error propagation caused by transmission errors effectively under different network delays as shown in Section IV. The proposed scheme works with off-line coded H.264 video, and it saves computational power and memory consumption at the decoder significantly, which makes it very suitable for portable devices. The scheme can be implemented for a wide range of video streaming applications to improve the error resilient performance. Especially, it fits the video streaming applications in portable devices through wireless communication networks.

REFERENCES

[1] ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC, Document JVT-G050, 8th Meeting: Geneva, Switzerland, May 2003.

[2] X. Zhou, W. Y. Kung and C.-C. Kuo, "Error resilient H.264 video with SP/SI coded macroblocks", *The 14th International Packet Video Workshop*, Irvine, CA, December 2004.

[3] M. Karczewicz and R. Kurceren,"The SP- and SI-frames design for H.264/AVC", *IEEE Trans. On Circuits and Systems for Video Technology*, vol. 13, no. 7, pp.637-644, July 2003

[4] Y. K. Wang, M. M. Hannuksela, V. Varsa, A. Hourunranta and M. Gabbouj, "The error concealment feature in the H.26L test model", *IEEE Int. Conf. on Image Processing*, pp. 729-732, September 2002

[5] T. C. Wang, H. C. F and L. G. Chen, "Low-delay and error-robust video transmission for video communcations", *IEEE Trans. On Circuits and Systems for Video Technology*, vol. 12, pp 1049-1058, December 2002

[6] E. Steinbach, N.Farber and B.Girod, "Standard compatiable extension of H.263 for robust video transmission in mobile environments", *IEEE Trans. On Circuits and Systems for Video Technology*, vol. 7, pp 872-881, December 1997

[7] W. Wada, "Selective recovery of video packet loss using error concealment", *IEEE J. Select. Areas Commun.*, vol. 7, pp. 807-814, June 1989.