Accepted in IEEE International Conference on Image Processing (ICIP 2011), 2011

PIXEL DOMAIN REFERENCELESS VISUAL DEGRADATION DETECTION AND ERROR CONCEALMENT FOR MOBILE VIDEO

Luc Trudeau, Stéphane Coulombe, Steven Pigeon

Department of Software and IT Engineering
École de technologie supérieure, Université du Québec
Montréal, Québec, Canada

ABSTRACT

In mobile video applications, where unreliable networks are commonplace, corrupted video packets can have a profound impact on the quality of the user experience. In this paper, we show that, in a wide range of operating conditions, selectively reusing data resulting from decodable errorneous packets leads to better results than frame copy. This selection is guided by a novel concept that combines motion estimation and a measure of blocking artifacts at block edges to predict visual degradation caused by the decoding of erroneous packets. Simulation results show that, by using the proposed solution, the H.264/AVC JM reference software decoder can select the best option between frame copy and the erroneous frame decoding in 82% of test cases. We also obtain an average gain of 1.95 dB for concealed frames (when they differ from those concealed by the JM decoder).

Index Terms— Error concealment, error detection, H.264, mobile video, pixel domain

1. INTRODUCTION

The capabilities of modern video coding standards like H.264 have paved the way for new consumer market video applications, such as video conferencing, video telephony, and video streaming over wireless networks using mobile devices. Operating conditions in wireless networks are far from ideal for video applications. These network media are unreliable and data loss is unavoidable, which increases the complexity of video-based applications that must recover from the error either by retransmission or by concealment. These limitations must be addressed to ensure the success of wireless video applications.

The impracticality of retransmission in live video makes error concealment the preferred method of error recovery [1]. Error concealment approaches were developed to improve the perceived quality of decoded sequences that have been subject to packet corruption over unreliable networks. However, error concealment algorithms will either conceal the whole slice [2], or perform block-level concealment [3]. The latter requires knowledge of erroneous blocks in the corrupted frame [1].

In H.264, slices are independently decodable groups of blocks. When an erroneous packet is discarded at the network level, the corresponding slice is lost, even if the blocks preceding the error are valid or if the visual impairment resulting from the error is minor. However, in the context of mobile applications, where bit rates are low and resolutions small, dropping a packet entails losing critical parts of a frame, and in some cases the entire frame. Even with state-of-the-art error concealment approaches, this may leads to significant visual degradation which will propagate in subsequent frames.

In [4], Superiori *et al.* have shown that using information from *decodable* (i.e. decoding them will not crash the decoder) erroneous packets can yield beneficial results. However, reusing erroneous packets can lead to the following artifacts: spatially shifted blocks, inverse DCT related noise, empty blocks (green or black), or frame freeze, the effects of which depend on the decoder used.

In cases where corrupted bit streams are decodable, the information they contain, even if it is damaged, once analyzed by the proposed detection approach, could be beneficial in guiding decisions made by a concealment module for either a whole slice concealment method (currently what is avaible in the H.264/AVC JM reference software decoder, and will be presented in this work) or for a block level concealment approach such as the recent findings on outer boundary matching algorithms [3]. The term "slice concealment" describes a family of concealment algorithms that conceal the whole slice, such as frame copy and motion copy [2].

This paper is organized as follows. In section 2, we propose a solution, based on the novel concept of motion-compensated blockiness, to identify visually degraded regions in predicted (P) frames caused by damaged, but decodable, packets. The proposed detection process is intended for block-based codecs. It operates in the pixel domain and is independent of decoder-specific information, such as block size or bitrate. Section 3 presents the concept of selective slice concealment, a low-complexity add-on to modern decoders where motion-compensated blockiness is used to perform referenceless visual degradation detection to guide the choice between using a slice concealment candidate or the erroneous slice when concealing a corrupted frame. In section 4, we present the experimental results of our proposed error detection and concealment solution using H.264, and we present our conclusions in section 5.

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada under research grant #356807-07, and in part by *Le Fonds québécois de la recherche sur la nature et les technologies* (FQRNT) under research grant #141698. Emails: luc.trudeau.1@ens.etsmtl.ca, {stephane.coulombe, steven.pigeon}@etsmtl.ca

2. MOTION-COMPENSATED BLOCKINESS

It is a known fact that a corrupted bit stream can desynchronize a decoder. This will often break the spatial pixel correlation in frame content at the block boundaries. Blockiness, a measure of the visibility of blocking artifacts, is a predominant feature of the visual impairment caused by decoder desynchronization [4]. In [4] and [5], the authors present a no-reference visual impairment detection method based on

$$\mathbf{D} = |\mathbf{F} - \mathbf{P}| \,,$$

the absolute frame difference, computed pixel by pixel (i.e. $\mathbf{D}_{x,y} = |\mathbf{F}_{x,y} - \mathbf{P}_{x,y}|$), between the current frame $\mathbf{F}$ and the previous frame $\mathbf{P}$, assuming that the previous frame is error-free.

To identify visual degradation caused by decoder desynchronization, Ikuno *et al.* [5] measure the block energy and blockiness found in frame $\mathbf{D}$. Values exceeding a certain threshold are considered erroneous. These results are refined with more sophisticated follow-up techniques, such as voting [4, 5] or support vector machines [6].

These refinements are required because $\mathbf{D}$ is not a reliable source of error detection, as it will falsely detect energy originating from changes between $\mathbf{F}$ and $\mathbf{P}$, or blockiness inherent in the content of $\mathbf{F}$ that occurs at block boundaries. Thus, we propose a novel approach in which the absolute difference is replaced with the motion-compensated residual absolute difference.

Let $B \times B$ be the block size, H and W the height and width of the frame respectively, with intervals $\mathcal{I}_m = [0, \frac{W}{B} - 1]$ and $\mathcal{I}_n = [0, \frac{H}{B} - 1]$, and p the half-width of the motion search area. The optimal vector for motion estimation, between $\mathbf{F}$ and $\mathbf{P}$ for a block at block coordinates (m, n) is given by

$$(\mathbf{U}_{m,n}, \mathbf{V}_{m,n}) = \underset{(u,v) \in K}{\arg\min}\ \text{SAD}(\mathbf{F}, \mathbf{P}, mB, nB, u, v) \,, \quad (1)$$

where $K = [-p, p] \times [-p, p]$, $m \in \mathcal{I}_m$, $n \in \mathcal{I}_n$ and

$$\text{SAD}(\mathbf{F}, \mathbf{P}, x, y, u, v) = \sum_{q=0}^{B-1} \sum_{r=0}^{B-1} |\mathbf{F}_{x+q,y+r} - \mathbf{P}_{x+q+u,y+r+v}| \,.$$

$\mathbf{U}$ and $\mathbf{V}$ are matrices of motion vector components (horizontal and vertical respectively) resulting from the motion search of each block of a frame. Considering motion in the error detection process can significantly reduce false error detection by compensating for motion between $\mathbf{F}$ and $\mathbf{P}$. However, motion compensation will not detect errors related to content contained in both the current and the previous frames (e.g. corrupted motion vectors yield blocks in $\mathbf{F}$ that are contained in $\mathbf{P}$, yet often exhibit strong blocking artifacts). To detect these errors, we measure the blockiness of residual blocks resulting from motion estimation. We refer to this measure as motion-compensated blockiness (MCB) and define it as

$$\text{MCB}_d(\mathbf{F}, \mathbf{P}, m, n) = \sum_{l=0}^{B-1} |\mathbf{b}_{d,l}(\mathbf{F}, mB, nB) - \mathbf{b}_{d,l}(\mathbf{P}, mB + \mathbf{U}_{m,n}, nB + \mathbf{V}_{m,n})| \,,$$

for each boundary $d \in \mathcal{B}$, with $\mathcal{B} = \{\mathcal{N}, \mathcal{E}, \mathcal{S}, \mathcal{W}\}$ and $m \in \mathcal{I}_m$, $n \in \mathcal{I}_n$. $\mathbf{U}$ and $\mathbf{V}$ are given by eq. (1). The blockiness vectors $\mathbf{b}_{d,l}$ of all four block boundaries are as follows:

$$\begin{aligned}
\mathbf{b}_{\mathcal{N},l}(\mathbf{F}, x, y) &= \mathbf{F}_{x+l,y} - \mathbf{F}_{x+l,y-1} \,, \\
\mathbf{b}_{\mathcal{E},l}(\mathbf{F}, x, y) &= \mathbf{F}_{x+B,y+l} - \mathbf{F}_{x+B-1,y+l} \,, \\
\mathbf{b}_{\mathcal{S},l}(\mathbf{F}, x, y) &= \mathbf{F}_{x+l,y+B} - \mathbf{F}_{x+l,y+B-1} \,, \\
\mathbf{b}_{\mathcal{W},l}(\mathbf{F}, x, y) &= \mathbf{F}_{x,y+l} - \mathbf{F}_{x-1,y+l} \,,
\end{aligned}$$

with $l \in [0, B-1]$, and where (x, y) are the pixel coordinates of the block. Note that this measure is different than what is proposed in [3], where only external borders are being matched.

The sum of motion-compensated blockiness (SMCB) of a block, at block coordinates (m, n) along its borders $d \in \mathcal{B}$, is given by

$$\text{SMCB}(\mathbf{F}, \mathbf{P}, m, n) = \sum_{d \in \mathcal{B}} \text{MCB}_d(\mathbf{F}, \mathbf{P}, m, n) \,.$$

There remains one significant source of false detection when measuring MCB: blocking artifacts are accounted for in the two blocks that share a blocky edge. To address this issue, we propose a blockiness distribution algorithm which assigns the MCB_d of each block boundary to the adjacent block with the highest SMCB.

The blockiness distribution algorithm is applied, in decreasing order of $\text{SMCB}(\mathbf{F}, \mathbf{P}, m, n)$, to all blocks with $\text{SMCB}(\mathbf{F}, \mathbf{P}, m, n) \geqslant T_b$, where T_b is the threshold that determines whether or not an edge is considered blocky. This threshold is further discussed in section 4. At the end of the process, the motion-compensated blockiness, MCB, of each blocky edge is assigned to a single block. This yields $\text{SDMCB}(\mathbf{F}, \mathbf{P}, m, n)$, the sum of distributed motion-compensated blockiness.

3. SELECTIVE SLICE CONCEALMENT

When decoding corrupted bit streams with the H.264/AVC JM reference software decoder [7], we discovered that a considerable number (up to 44% based on our simulations) of the resulting erroneous frames $\hat{\mathbf{F}}$ exhibited a higher PSNR than the concealment candidate $\mathbf{F}'$, resulting from a whole slice concealment approach used by the decoder. The PSNR being, in both cases, relative to the reference frame $\mathbf{F}_0$ (i.e. relative to the original uncompressed frame from which $\mathbf{F}$ was encoded).

Furthermore, we observed that flexible macroblock ordering and separating frames into multiple slices will contribute to increase the number of frames $\hat{\mathbf{F}}$ with higher PSNR than $\mathbf{F}'$. Based on these observations, we propose a selective slice concealment approach. Motion-compensated blockiness is used to guide the decoder to decide whether to conceal a slice or to use the erroneous data when available.

When an error is detected at the transport layer and the decoder is about to apply a concealment, if the erroneous frame is available, instead of dropping it, we perform a selective slice concealment. This is done by evaluating the SDMCB of both the concealment candidate

$$c_c = \sum_{m=0}^{\frac{W}{B}-1} \sum_{n=0}^{\frac{H}{B}-1} \mathrm{SDMCB}(\mathbf{F}'_{,}\mathbf{P}, m, n) \, ,$$

and the erroneous frame

$$c_e = \sum_{m=0}^{\frac{W}{B}-1} \sum_{n=0}^{\frac{H}{B}-1} \mathrm{SDMCB}(\hat{\mathbf{F}}_{,}\mathbf{P}, m, n) \, ,$$

in order to determine the optimal concealment

$$\mathbf{F}^* = \begin{cases} \hat{\mathbf{F}}, & \text{if } c_e < c_c \\ \mathbf{F}', & \text{otherwise} \end{cases} .$$

4. EXPERIMENTAL RESULTS

Using version 16.2 of the H.264/AVC JM reference software [7], we built a dataset of video subsequences composed of 3 consecutive frames, starting with an I frame followed by P frames (IPP) extracted from a random starting point. This was repeated 5 times for each of the 17 QCIF (176×144) reference video sequence available at [8]. These subsequences were encoded with fixed quantization parameters (16, 20, 24, 28) and exposed to bit error rates of 0.0004, 0.0008, 0.0016, and 0.0032, which resulted in 1360 subsequences that were used to evaluate the proposed solution. We only exposed the third frame (P-frame) to errors according to prescribed bit error rates. Simulation results are presented with respect to this errorneous frame. The type of flexible macroblock ordering used is dispersed, with two-slices (checkerboard). We used the baseline profile encoding parameters, at 30 frames per second, with a QCIF resolution and RTP output format. However, RTP and NAL headers where not exposed to the errors.

The block size used to measure MCB is $B = 16$ pixels and the blockiness threshold T_b was set to 5000.

None of the tested versions (from 15.1 to 16.2) of the H.264/AVC JM reference software decoder provided a functional implementation of motion copy error concealment [9]. This fact was confirmed in several discussion forums on the Web. Therefore, although the proposed method could work with both motion copy and frame copy, we could only test frame copy.

As previously stated, a significant number of erroneous slices decoded using the H.264/AVC JM reference software decoder [7] produced a higher PSNR than their frame copy concealment alternatives, provided that they are decodable. In Fig. 1, we present the measured percentage of successful decodings for different operating conditions.

When the H.264/AVC JM reference software decodes a corrupt bit stream, it passes through three different states. The first is the pre-error state (normal bit stream decoding). When the first error is reached, the decoder enters a desynchronization state. In this second state, unaware that the content is invalid, it inserts degradation into the frame. This might eventually lead to a decoding failure, at which point the decoder reverts to a slice copy algorithm for the remainder of the slice.

According to our observations over all our testing conditions, 44% of successfully decoded erroneous slices resulted in frames with higher PSNR, if they had been concealed with a slice copy algorithm.

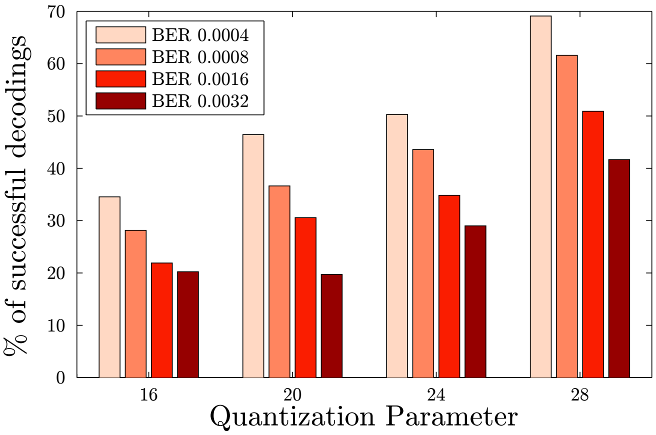

Fig. 1. Percentage of successful decodings (i.e. decodable corrupted frames), based on quantization parameter (QP) and bit error rate (BER) being evenly spread over 1360 sequences.

Compared to frame copy, the PSNR of the erroneous frame is higher while the decoder remains in the first state, than lower in the second, and equal in the third. The number of blocks decoded in each state, the extent of the degradation created in the second state, combined with the changes between the current, and previous frame determine whether or not a correctly decoded erroneous slice will offer a higher PSNR than a slice copy concealment candidate.

Enabling the decoder to select between slice copy and the erroneous slice is the basis of our proposed selective slice copy approach. With SDMCB as our discriminant, we achieved the results shown in Fig. 2. Selective slice copy produced an average gain of 0.72 dB. In contrast, a reference-based selective algorithm (i.e. an ideal, but impossible to realize, method that selects the highest PSNR value between a frame copy and the erroneous frame when compared against $\mathbf{F}_0$) produced an optimal result of 0.94 dB.

We observed that selective slice concealment makes an optimal choice in 82% of the cases (compared to the reference-based method). This allows the selective decoder to outperform the conventional decoder in 32% of cases, for an average gain of 1.95 dB. In contrast, the reference-based selective algorithm produced an optimal gain of 2.12 dB.

The histogram in Fig. 3 indicates that selective slice concealment results in fewer poor concealments ($<$ 30 dB) and a larger number of frames for every interval over 35 dB. This means that not only does the proposed approach reduce the number of low-quality concealments, these having greater impact on the quality of the user experience, it also offers a higher quality concealment without requiring additional bandwidth or a high-complexity algorithm.

Furthermore, SDMCB's ability to choose the favorable concealment could be exploited in future works for block level concealment such as [3]. The advantages are twofold, by first identifying erroneous blocks that need to be concealed (it has been shown that not all erroneous blocks require a concealment), and second by offering a referenceless validation

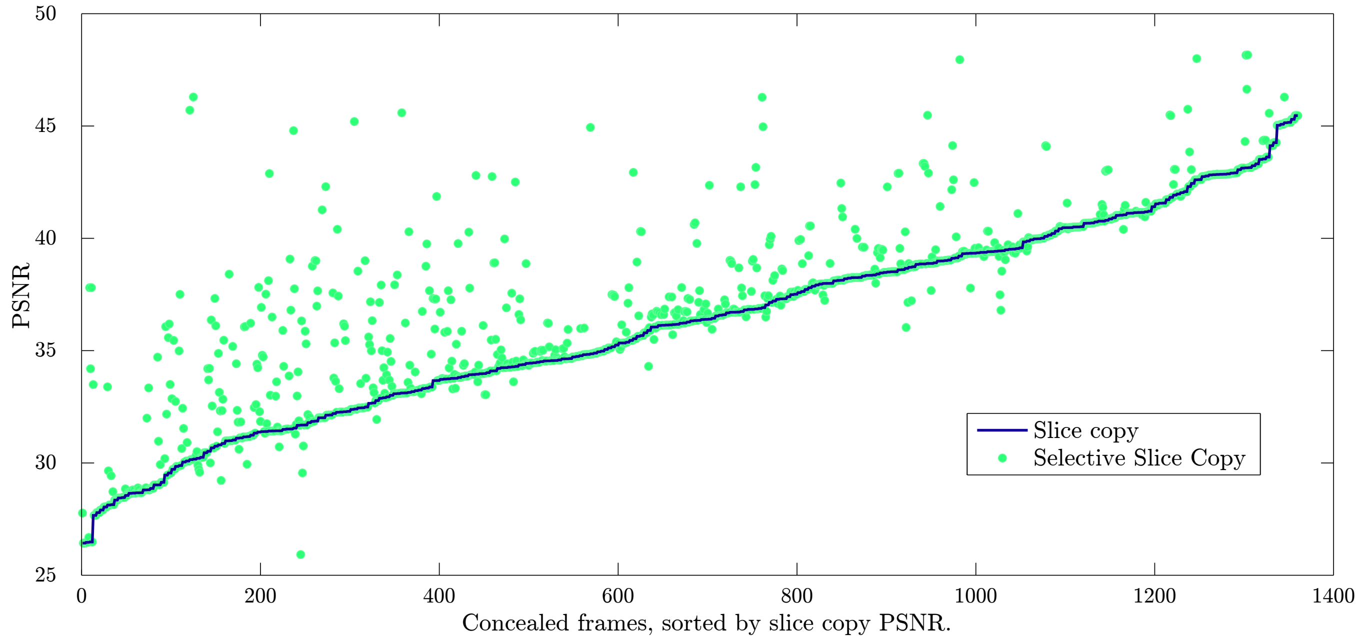

Fig. 2. Comparison of selective slice copy and the slice copy algorithms over the 1360 sequences of our test dataset.

mechanism for concealment candidates, as some erroneous blocks will exhibit very little visual degradation.

5. CONCLUSION

In this paper, we described a novel visual degradation detection approach based on the sum of distributed motion-compensated blockiness, SDMCB. We proposed using this information for a low-complexity decoder add-on operation that allows the decoder to decide whether to use the erroneous slice or slice copy. Our simulations show interesting ratios of decodable sequences based on varying quantization parameters and bit error rates, and improved concealment results when SDMCB is used to guide the decoder in selecting between the erroneous slice and applying frame copy. These results constitute interesting avenues for decoders, players, and no-reference video quality analysis tool makers.

6. REFERENCES

[1] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: a review," *Procs. of the IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.

[2] Z. Wu and J. Boyce, "An error concealment scheme for entire frame losses based on H.264/AVC," in *Procs. IEEE Int. Symposium on Circuit and Systems (ISCAS)*, 2006, p. 4.

[3] T. Taipanich, P.-H. Wu, and C.-C. J. Kuo, "Low-complexity mobile video error concealment using OBMA," in *Consumer Electronics, 2008. ICCE 2008. Digest of Technical Papers. International Conference on*, January 2008, pp. 1–2.

[4] L. Superiori, O. Nemethova, and M. Rupp, "Detection of visual impairments in the pixel domain of corrupted H.264/AVC packets," *Procs. of the Picture Coding Symposium (PCS 2007), Lisboa, Portugal*, pp. 7–9, November 2007.

[5] J. C. Ikuno, "Performance of an error detection mechanism for damaged H.264/AVC sequences," M.S. thesis, December 2007.

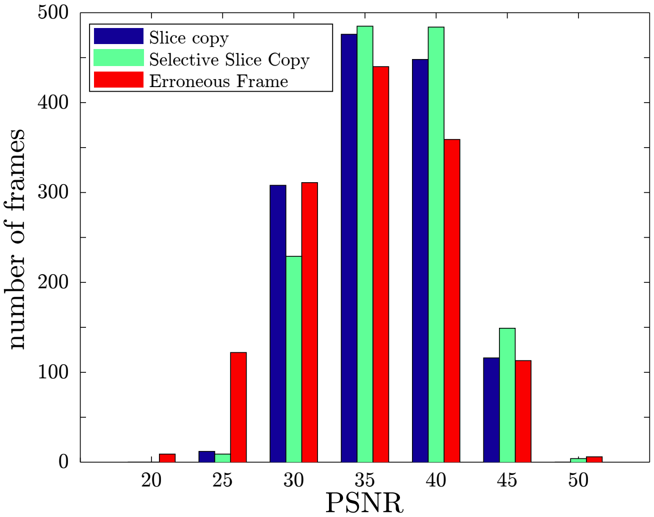

Fig. 3. Frame PSNR histogram, with intervals of 5 dB centered at every 5 dB increment after 20 dB, for 1360 erroneous frame concealments.

[6] R. A. Farrugia and C. J. Debono, "A robust error detection mechanism for H.264/AVC coded video sequences based on support vector machines," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 18, no. 12, pp. 1766–1770, December 2008.

[7] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, "H.264/AVC JM reference software," 2010, http://iphome.hhi.de/suehring/tml/.

[8] Arizona State University, "YUV video sequences," December 2010, http://trace.eas.asu.edu/yuv/.

[9] Mantis JVT JM H.264/AVC reference software project, "0000189: Crash in JM 16.1 Decoder," September 2009, https://ipbt.hhi.fraunhofer.de/mantis/view.php?id=189.