ERROR RESILIENCE PERFORMANCE EVALUATION OF A DISTRIBUTED VIDEO CODEC

Claudia Tonoli, Marco Dalai, Pierangelo Migliorati and Riccardo Leonardi

Department of Electronics for Automation,
University of Brescia, Italy.
Email: {*name.surname*}@ing.unibs.it

ABSTRACT

Distributed Video Coding (DVC), one of the most active research field in the video coding community, is based on the combination of Slepian-Wolf coding techniques with the idea of performing the prediction at the decoder side rather than at the encoder side. Besides its main property, which is flexible allocation of computational complexity between encoder and decoder, the distributed approach has other interesting properties. One of the most promising DVC characteristics is its intrinsic robustness to transmission errors. In this work we have evaluated the error resilience performance of a video codec based on the DVC scheme proposed by Stanford, and we have carried out a preliminary comparison with traditional H.264 encoding, showing that at high error probabilities and high bitrates the distributed approach can also outperform the traditional one.

1. INTRODUCTION

Distributed Video Coding has been studied extensively in the last years, being probably one of the most challenging research field in video coding, due to potential new application perspectives with respect to traditional coding techniques. The main idea underlying the DVC paradigm is to apply some fundamental results of information theory dating back to the '70s, studied under the so called Distributed Source Coding (DSC) field pioneered by Slepian-Wolf and Wyner-Ziv works [1], [2]. The basic idea of DVC is the exploitation of the correlation of a video sequence in the decoding phase rather than in the encoding phase. So, no motion compensated prediction is used in the encoding phase. DSC principles are instead applied, compressing the information by transmitting the parity bits of a systematic channel code, which are extracted from the frames that need to be sent. At the decoder, the redundancy of the video sequence is exploited, by performing motion compensation of already received data, in order to use the received parity bits to recover the original information from the compensated signal.

The effect of this reverse approach is twofold. A first consequence is that in DVC, there is a shift of the computational complexity from encoder to decoder. The decoding phase requires in fact complex operations, that are conceptually and computationally analogous to motion estimation performed by the encoder on traditional video codecs . The encoding phase involves only very simple operations. For this reason, DVC is particularly well-suited for applications that need simple, cheap, low power encoding devices. A second promised consequence of using a distributed coding approach is its likely intrinsic resilience to transmission errors. In fact, given that no prediction loop is used in the encoding phase, distributed coding should be more resilient to errors because it is not affected by the typical drift problems which may occur in the case of traditional predictive systems.

While the advantages in terms of reduced computational complexity may be partially mitigated by hardware technological advances, the expected intrinsic error resilience may turn out quite attractive in application scenarios prone to continuously varying channel characteristics. In the literature, various works that address the topic of error resilience and distributed coding from different points of view have been presented. For example in [11] and [10], Wyner Ziv coding is used to generate a supplementary bitstream and achieve error protection on H.264.
In this paper we provide some indications of the error resilience properties of distributed coding, considering a video codec based on the DVC scheme described for example in [3], [4], [5], [6] and further developed within DISCOVER European funded project.

The paper is organized as follows. In Section 2 we briefly describe the architecture of the considered coding scheme. In Section 3 we present in some detail the different tests performed and the associated performance evaluation. Concluding remarks are provided in Section 4.

2. CODEC ARCHITECTURE

In this section we briefly describe the architecture of the considered distributed video coding system. The block diagram of the codec is shown in Figure 1. We do not describe in detail the single component blocks of the codec and we refer the reader to [5] and [7] for a more complete description of the single components of the scheme.

The encoder works independently on each frame of the video. Even indexed frames, that are referred to as *key-frames*, are traditionally encoded using an H.264 encoder operating in intra-mode, i.e., without using inter-frame prediction. The odd indexed frames, called *Wyner-Ziv* (WZ) frames, are Wyner-Ziv encoded. The frames are first transformed, with a block based DCT, and then quantized with proper quantization matrices. The quantized coefficients are then encoded, bitplane by bitplane, with the use of a turbo code. In particular, each bitplane is fed to a turbo code with rate 1/3; while information bits are discarded, the parity bits are stored. The encoded stream is thus composed of two different parts: the H.264 intra coded key-frame stream and the WZ parity bits stream. The key-frame information is entirely sent to the decoder, while parity bits are only partially sent, depending on decoder requests through a feedback channel, as will be now clarified.

The decoding of a WZ frame is performed by first generating the Side Information (SI), i.e., an estimate of the missing frame, by motion compensated interpolation between two adjacent key-frames, which are completely known at the decoder. This estimate is used to extract a first approximation of the information bits of the original quantized frame. These bits are then corrected using a turbo decoder with the parity bits received from the encoder. Parity bits are not sent all at once, but are iteratively requested by the decoder, through the feedback channel, until the estimated error probability on the decoded bitplane reaches a given threshold. In this way, the decoder always achieves the target quality during the decoding phase, and the impact is seen in the total rate sent by the encoder. This fact is important as far as error resilience is concerned, because it will be partially responsible for the tradeoff between rate and quality.

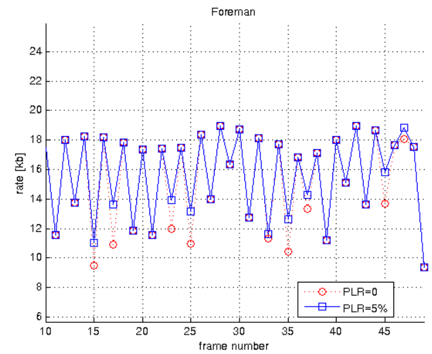
3. ERROR RESILIENCE TESTS

In this section we describe the tests we have performed and present the results we have obtained.

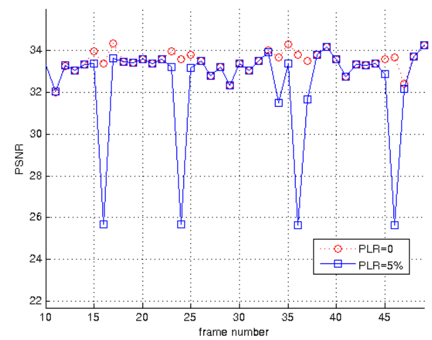
3.1. Evaluation of the behaviour of the codec in presence of channel errors

We have first analyzed how the codec reacts in presence of transmission errors on the received encoded stream. We have studied the effect of errors on the key-frames, assuming WZ bits were received correctly, in order to investigate the reaction to bad quality side information. As described in Section 2, key-frames are encoded in H.264 intra mode. This implies that the study of the distributed codec performance on error-prone channels involves the error resilience features of the H.264 codec.

H.264 has various error concealment strategies, mainly based on the flexible reordering of the frame macroblocks inside the slices at the encoder, and on the estimation of lost slices at the decoder, as described for example in [9]. The H.264 implementation used in our experiments corresponds to JM11.0. We have enabled the Flexible Macroblock Ordering (FMO) feature, in dispersed mode; in this way each slice is composed of non adjacent macroblocks. We have also enabled the error concealment tool at the decoder.



(a) Frame by frame rate.



(b) Frame by frame PSNR.

Fig. 2. Frame by frame rate and PSNR for the *"Foreman"* sequence.

We have analyzed the behaviour of the codec considering jointly the plots in Figure 2(a) and 2(b), depicting, respectively, the rate and the PSNR frame by frame, for the *Foreman* sequence, encoded at an average PSNR of 34.5 dB . The key-frames that are affected by packet loss are characterized by a PSNR much lower than the correctly received ones (e.g., frame n. 36). Let's consider the encoded rate of the WZ frames that are adjacent to a corrupted keyframe (e.g., frames n. 35 and 37). We can notice that such frames require more bits than they do in the case of no transmission errors. Their decoded quality is much higher than the one of the corrupted key-frames; this means that the decoder can react to a bad side information by asking more parity bits to the encoder.

In Figure 3(a) the rate-distortion plot for a Packet Loss Rate of 5% is shown, for the WZ frames of the *Foreman* sequence. We can notice that at low bit-rates the performance

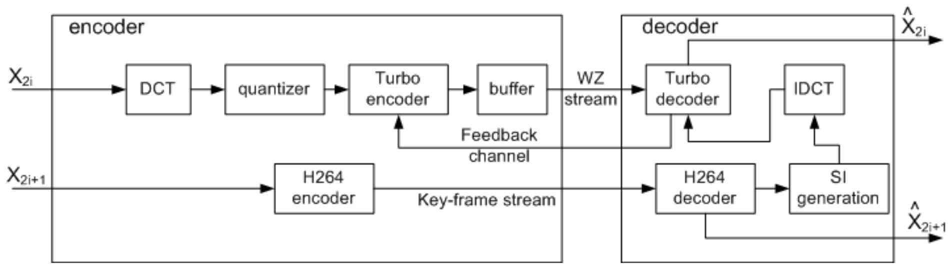


Fig. 1. Architecture of the considered video codec.

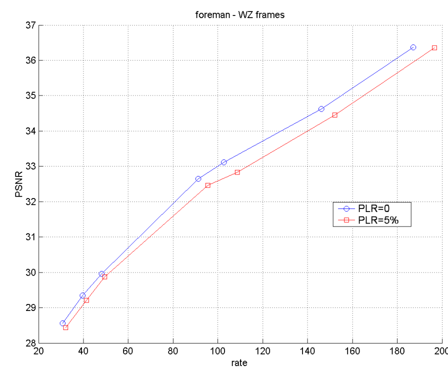
loss is about 1dB. The plot in Figure 3(b) shows that the PSNR loss of the corrupted key-frames at high bitrates is much lower than the one at low bitrates. Thus it can be deduced that the feedback request to the encoder can compensate quite well the presence of low quality side information.

3.2. Preliminary comparison with traditional coding

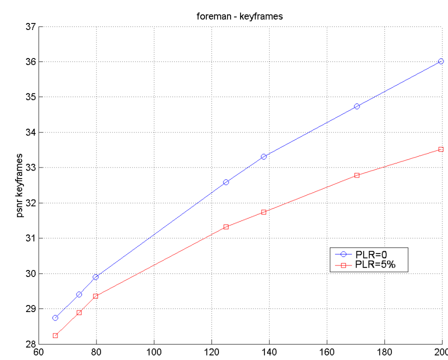
In order to help the evaluation of the error resilience performance that are currently obtainable with a distributed coding approach with respect to a traditional one, we present the results obtained by simulating the transmission over a noisy channel of the distributed stream, and of an H.264 stream. We have chosen to use H.264 as a reference because it represents the state-of-art of traditional coding, and we have chosen to use that standard in a scenario reproducing the features of the most common ones. This is the reason why we accept some unfairness in the comparison, in the sense that a feedback channel is given to the WZ part of the distributed codec. We also remark that the WZ feedback requires very few bits: in fact it just signals whether new WZ bits are required or not. Nevertheless, if H.264 is given a feedback channel with ARQ, it can perform as well as if there were no packet loss.

In our tests, we have considered the transmission over a packet network, characterized by a known Packet Loss Rate. This scenario is realistic according to modern transmission networks characteristics. We have introduced a basic Forward Error Correction protection on the H.264 encoded stream, that has been performed according to the scheme, derived from [10], depicted in Figure 4. A (n, k) Reed-Solomon code (in our case, a (15, 12) code) has been applied across a group of k slices, to obtain $(n - k)$ groups of parity bytes. Each group of parity symbols is supposed to be sent as a different packet. As a (n, k) Reed-Solomon code can correct up to $(n - k)$ erasures, with this scheme the loss of at most $(n - k)$ packets can be compensated. H.264 is constrained to work in GOP8 IPP mode.

This error protection scheme is also applied to the key-frame component of the distributed stream, while the WZ packets are supposed to be sent unprotected. The errors patterns have been generated using a uniform random distribution; the considered slice size is 200 byte. The Reed-Solomon



(a) WZ frames.



(b) Keyframes.

Fig. 3. Rate-distortion on the WZ frames and key-frames, for the *"Foreman"* sequence.

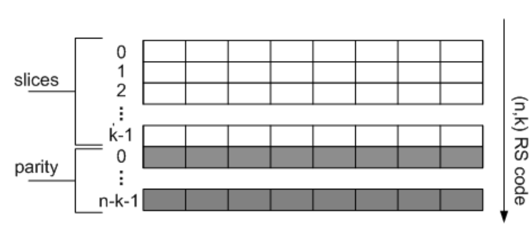
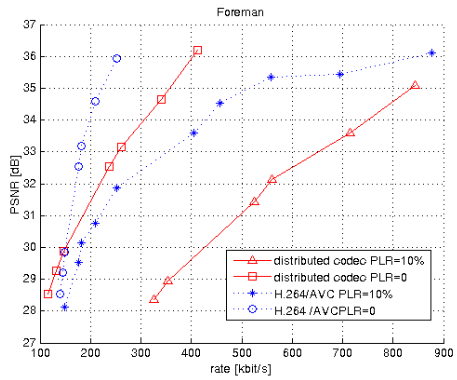
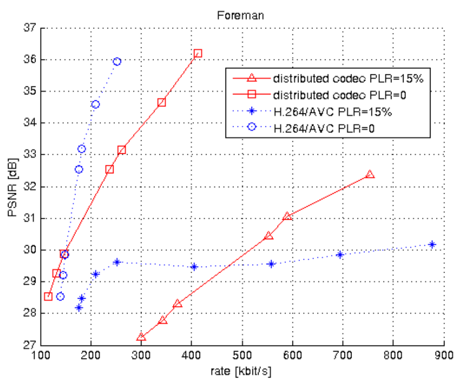


Fig. 4. Reed-Solomon code application scheme.



(a) *Foreman* sequence, PLR = 10%.



(b) *Foreman* sequence, PLR = 15%.

Fig. 5. PSNR curves for a distributed and a traditional codec

code rate is $4/5$.

The plots in Figures 5(a) and 5(b) show the results we have obtained for a Packet Loss Rate of 10%, and 15% respectively, for the *Foreman* sequence. Similar results have been obtained for other video sequences. We can notice that, at a low PLR (Figure 5(a)) H.264 can compensate the channel errors with a quality loss of about 1 dB at low bitrates and 4 dB at higher rates, while the quality loss of the distributed codec is higher. On the other hand, for a higher PLR value (Figure 5(b)) H.264 gives worst results, especially at high bitrates. The H.264 quality increases very slowly and hardly reaches 30 dB, due to the fact that the number of lost slices is often higher than the correcting capability of the channel code. On the contrary, the quality obtained with the distributed codec increases and, at high bitrates, is 2 dB higher than that obtainable with H.264.

4. CONCLUSION

In this work we have evaluated the error resilience behavior of a distributed video codec. In particular, we have first analyzed how the distributed codec reacts to the presence of channel errors, considering the behavior in the case of corruption of the key-frame only, showing that distributed coding can compensate quite well the presence of low quality side information. We have then presented a comparison with respect to traditional H.264 encoding, showing that at high error probabilities and high bitrates the distributed approach can also outperform the traditional one.

5. ACKNOWLEDGMENTS

The work presented in this paper was developed within DISCOVER (Distributed Coding for Video Services), a European Commission funded Future and Emerging Technologies (FET) project. DISCOVER software started from the IST_WZ software developed at the Image Group from Instituto Superior Técnico (IST) of Lisbon by Catarina Brites, João Ascenso and Fernando Pereira.

6. REFERENCES

[1] D. Slepian and J.K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, 19: 471–480, 1973.

[2] A. D. Wyner and J. Ziv, "The rate distortion function for source coding with side information at the receiver," *IEEE Trans. Inform. Theory*, 22: 1–11, 1976.

[3] B.Girod, A. Aaron, S.Rane, D. Rebollo-Monedero, "Distributed Video Coding", Proceedings of the IEEE, vol. 93, no. 1, pp. 71-83, January 2005.

[4] A. Aaron, B. Girod, "Compression with Side Information Using Turbo Codes", *Proc.IEEE Data Compression Conference*, Snowbird, Utah, April 2002.

[5] A. Aaron, R. Zhang, and B. Girod, "Wyner-ziv coding for motion video", *Asilomar Conference on Signals, Systems and Computers*, November 2002.

[6] A. Aaron, S. Rane, E. Setton and B. Girod, "Transform-Domain Wyner-Ziv Codec for Video", *Visual Communications and Image Processing Conference*, San Jose, CA, USA, January 2004.

[7] C.Brites, J. Ascenso, F. Pereira, "Improving transform domain Wyner-Ziv video coding performance", *ICASSP*, Toulouse, France, May 2006.

[8] Sunil Kumar, Liyang Xu, Mrinal K. Mandal and Sethuraman Panchanathan, "Error Resiliency Schemes in H.264/AVC Standard", *Elsevier J. of Visual Communication and Image Representation*, Special issue on Emerging H.264/AVC Video Coding Standard, Vol. 17, April 2006.

[9] Thomas Wiegand, Gary J. Sullivan, Gisle Bjontegaard, and Ajay Luthra, "Overview of the H.264 AVC Video Coding Standard", *IEEE Transaction on circuits and systems for video technology*, July 2003.

[10] P. Baccichet, S. Rane and B. Girod, "Systematic Lossy Error Protection based on H.264/AVC Redundant Slices and Flexible Macroblock Ordering", *Packet Video Workshop (PV 2006)*, Hangzhou, China.

[11] S. Rane, P. Baccichet and B. Girod, "Modeling and Optimization of a Systematic Lossy Error Protection System based on H.264/AVC Redundant Slices", *Proc. Picture Coding Symposium, PCS-2006*, Beijing, China, April 2006.

[12] A. Sehgal, A. Jagmohan, and N. Ahuja, "Wyner-Ziv Coding of Video: An Error-Resilient Compression Framework", *IEEE Transaction on Multimedia*, vol.6, no.2, April 2004.