A Proposal to Suppress the Training Stage in a Coset-Based Distributed Video Codec[1]

Xavi Artigas[†], *Marco Tagliasacchi*[‡], *Luis Torres*[†], *Stefano Tubaro*[‡]

[†]Technical University of Catalonia
{xavi, luis}@gps.tsc.upc.edu

[‡]Politecnico di Milano
marco.tagliasacchi@polimi.it stefano.tubaro@elet.polimi.it

ABSTRACT

Distributed Video Coding (DVC) is a coding paradigm that gives the decoder the task to exploit the source statistics to achieve efficient compression. Many approaches to the DVC problem have recently appeared in the literature, including the PRISM codec. Instead of encoding the deterministic quantized prediction error residual, PRISM partitions the quantization lattice into cosets and sends the index of the coset each quantized coefficient belongs to. Estimating the number of cosets is of crucial importance to achieve good coding efficiency. In PRISM, this is determined during an offline training phase. The present work aims at being a starting point for the suppression of the training stage of PRISM at the cost of sending the number of cosets for each DCT coefficient. The statistics of the number of cosets are analyzed to figure out the maximum compression efficiency achievable by entropy coding. Furthermore the paper discusses some techniques that might be used to lower the amount of transmitted bits. Based on these results, directions for future works are proposed.

1. INTRODUCTION

So far, research activities on video coding as well as standardization efforts have adopted a video coding paradigm where it is the task of the encoder to explore the source statistics, leading to a complexity balance where complex encoders interact with simpler decoders. This paradigm is motivated by applications such as broadcasting, video on demand, and video streaming. Distributed Video Coding, a video coding paradigm built on top of Distributed Source Coding (DSC) principles, adopts a completely different approach by partially moving at the decoder the task to exploit the source statistics. This change in paradigm shifts the encoder-decoder complexity balance, allowing the provision of efficient compression solutions with simple encoders and complex decoders. Emerging applications might take advantage of DVC enabled solutions, such as wireless video cameras and wireless low-power surveillance networks, disposable video cameras, medical applications, sensor networks, multi-view image acquisition, networked camcorders, etc., where low complexity encoders are a must because memory, computation, and energy are a scarce resource.

Although the theoretical bases for Distributed Source Coding were established thirty years ago with the work by Slepian and Wolf [1] (for the lossless case) and Wyner and Ziv [2] (for the lossy case), it has been only recently that research on this topic has taken momentum. This has been encouraged by the rise of some new practical codec designs developed at UC Berkeley [3] and Stanford University [4].

The rest of this paper is organized as follows. First of all, the theoretical foundations of DVC are summarized in Section 2. The PRISM codec, introduced in [3], will be described in some detail in Section 3. Section 4 illustrates the coset generation algorithm used in the experiments, and Section 5 shows the obtained results. Finally, Section 6 gives some directions on the intended future work.

2. THEORETICAL FOUNDATIONS

It is well known that the minimum lossless rate at which a signal X can be transmitted is $H(X)$, the signal's entropy. If two statistically dependent signals X and Y are to be transmitted, the best thing that can be done is to encode them jointly, in order to exploit their statistical dependency, by achieving a minimum lossless rate equal to $H(X, Y)$, their joint entropy. Slepian and Wolf showed in 1973 [1] that this lower bound for the lossless joint transmission rate is also achievable when the signals X and Y are encoded *separately*, that is, when the encoder for X does not have access to Y, and vice versa. No coding efficiency loss is observed when the correlation is exploited at the decoder only (joint decoding), if a arbitrary small probability of decoding error can be tolerated.

[1] ACKNOWLEDGMENT: The work presented was developed within VISNET, a European Network of Excellence (http://www.visnet-noe.org), funded under the European Commission IST FP6 programme.

It is the ability of encoding X and Y separately that makes Distributed Source Coding so attractive, because encoders for separate signals do not have to search for inter-correlations among signals, and therefore require fewer computations. In a DSC setting these inter-correlations are exploited only at the decoder, thus shifting most of the complexity of the coder at the decoder end.

3. DESCRIPTION OF THE PRISM CODEC

The present work is partially inspired to the PRISM codec. This section briefly describes its main features.

The PRISM codec, developed at UC Berkeley [3], takes advantage of DSC principles to allow flexible distribution of the computational complexity between encoder and decoder and to achieve inbuilt robustness to drift caused by channel loss. All of this, without losing in terms of compression efficiency.

The core compression mechanism in PRISM is coset encoding. Conventional predictive coding schemes encode the quantized difference between the signal and its motion-compensated predictor. Conversely, with PRISM, the encoder sends only the least significant bits (LSB) of each quantized DCT coefficient, which is equivalent to the coset of the quantization lattice the coefficient belongs to. (in this case, the number of cosets is 2 raised to the number of transmitted LSB). At the decoder side, different motion-compensated predictors from previously reconstructed frames are tested in order to *fill-in* the missing most significant bits (MSB), i.e. to identify the right quantization index within the signaled coset. The rationale behind this scheme is that only those bits that cannot be obtained from the side information at the decoder (the LSB) are actually transmitted. Figure 1 shows an example of coset encoding.

A key parameter to be determined in the coset encoding algorithm is the number of cosets used to partition the source codebook or, in other words, the number of LSB to send. The fewer the transmitted bits, the lower the bitrate, but at the same time more bits are left for the decoder to estimate, and therefore the probability of making a decoding error increases. If the encoder is constrained not to perform any motion search (or little motion search), finding the right number of LSB to send is not an easy task, since the encoder does not have access to the decoder's side information, i.e. the best motion-compensated predictor.

In the earlier version of PRISM described in [3], each block is compared with the co-located block in the previous frame (zero-motion prediction) to find MSE_0, the MSE of the difference between the two. By thresholding MSE_0, the block is then classified into one of many classes ranging from the SKIP class, where the block difference is so small that it is not encoded at all, to the INTRA class, where there are so many differences that the block is encoded in intra-frame mode. Each of the classes in between specifies the

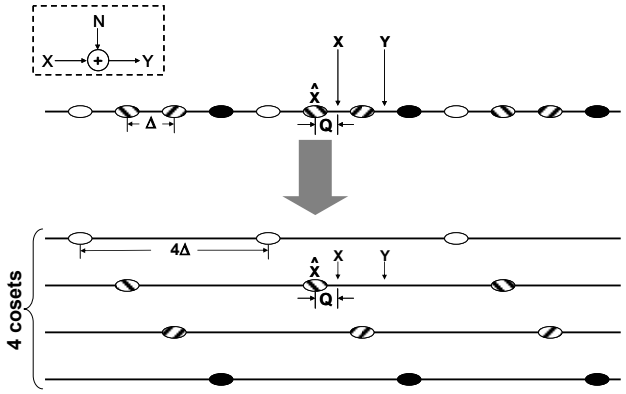

Figure 1 Coset encoding example with 4 cosets (2 LSB). Encoder sends the index of the coset ($\log_2 4$ bits). Decoder finds X based on Y and the signaled coset.

number of LSB that need to be encoded for each DCT coefficient.

Therefore, for each block, the class identifier and the associated LSBs need to be transmitted. A checksum in the form of a 16-bit Cyclic Redundancy Check (CRC) is also transmitted per block, to aid the decoder's search for the correct predictor.

At the receiving end, the block is decoded by testing several predictors from previously decoded frames, until the decoded block matches the CRC sent by the encoder.

In [3] the rate-distortion performance of PRISM is reported to be in between H.263+ intra and inter modes, when the motion search is completely shifted at the decoder and a lossless channel is assumed.

4. COSET GENERATION ALGORITHM

In [3], the number of cosets to be encoded for each DCT coefficient is uniquely determined by the class the block belongs to. Offline training on several test sequences allows to obtain an estimate of these numbers. Adopting this scheme, the correlation noise is being estimated at the block level, and individual coefficients might end up receiving more or less LSB than they actually require.

In this paper an alternative strategy is considered for coset generation which is close to the spirit of the version of the PRISM codec recently appeared in [5].

The basic idea is to calculate the correlation noise for each coefficient, thus sending the necessary LSB for each coefficient, thus avoiding offline training. The drawback is that two pieces of information need to be sent for each coefficient: the number of LSB and the actual value of the LSB. While the last term was already there in the PRISM codec described in Section 3, the first tends to be in general more costly than the simple block class index.

In [5] it is explained that the motion estimation task can be flexibly shifted from the encoder to the decoder depending on the state of the transmission channel. Intuitively, the higher is the channel noise, the less is the

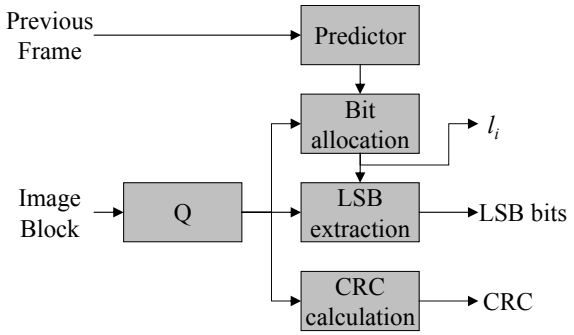

Figure 2 Encoder scheme. "Bit allocation" calculates the necessary amount of bits to safely encode the quantized coefficients given the predictor, according to equation (1). "LSB extraction" simply returns the least significant l_i bits.

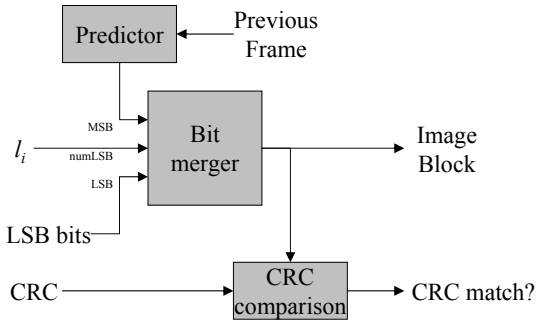

Figure 3 Decoder scheme. "Bit merger" joins the l_i LSB bits with the remaining MSB bits.

work that needs to be done at the encoder, since little value is gained by accurately modeling the motion.

Therefore, in this section, we show how the number of cosets depends upon the level of motion estimation carried out at the encoder side. The number of LSB to transmit for DCT coefficient X_i (in zigzag scan order) is called l_i. Equivalently, the source codebook is partitioned into 2^{li} cosets. Let us denote with Y_i the best predictor available at the encoder. In general, Y_i depends on the amount of motion search carried out at the encoder. If $N_i = X_i - Y_i$ is the correlation noise observed at the encoder, the number of levels l_i can be obtained as:

$$l_i = \begin{cases} 0 & 0 \le |N_i|/Q < 0.5 \\ 1 & 0.5 \le |N_i|/Q < 1 \\ 1 + \lceil \log_2 (|N_i|/Q) \rceil & 1 \le |N_i|/Q \end{cases} \quad (1)$$

Where N_i is the difference between the real coefficient i and the zero-motion predicted one. Q is the quantization step, which is the same for all coefficients, to keep things simple.

The decoding process follows the same steps as in the original version of PRISM. For each tested predictor block and for each DCT coefficient, the decoder chooses the value belonging to the coset signaled by the encoder that is closest to the predictor. Once all coefficients are decoded, the CRC of the block is calculated and compared with the received one. If they match, decoding is declared successful, otherwise, another block is tried as a predictor. Figure 2 and Figure 3 show the encoder and decoder schemes respectively.

5. ANALYSIS OF THE COSET STATISTICS

In this paper, we explore the distribution of the coset statistics as this represents a valuable information for the design of an optimized arithmetic codec for this kind of source. Particularly, the entropy of the l_i and the LSB has been studied to find a lower bound for the attainable bitrates. No channel errors have been simulated. Blocks are composed of 8x8 DCT coefficients.

In the conducted experiments, the l_i LSB calculated by equation (1) are transmitted. The decoder can perfectly recover the original data, so the only degradation comes from quantization. The statistics of l_i and LSB are then examined to find out the attainable bitrates.

The resulting probability density function of the LSB is almost flat, meaning that these bits are completely random and therefore very difficult to compress. TABLE I and TABLE II show the total number of bits that would be necessary to encode the LSB of a coefficient block, for different quantization steps and different levels of motion estimation performed at the encoder. The tables show that, generally, the more motion estimation is performed at the encoder, the less LSB need to be sent, and this is so because more motion search means that predictors closer to the original block can potentially be found.

When little or no motion search is performed at the encoder, more LSB than strictly needed might be transmitted. However, these bits are useful when the transmission channel is lossy as they contribute to increase the robustness of the system.

The entropy of each individual l_i is also calculated to estimate how many bits would be necessary to encode a whole block if the l_i were to be independently encoded. TABLE III and TABLE IV show the results, for different quantization steps and different levels of motion estimation performed at the encoder. As with the LSB, it can be observed that increased motion estimation at the encoder reduces the necessary bitrate.

In order to benefit from possible inter-coefficient correlations, the encoding of the l_i conditioned to neighboring l_i has been studied. TABLE V shows the obtained results. It can be seen that the highest compression gain comes from using 3 neighboring l_i as context, yielding a 15% reduction in the bitrate.

6. CONCLUSIONS AND FUTURE WORK

This work has presented some techniques aimed at improving the PRISM approach in Distributed Video

TABLE I. Average number of bits necessary to losslessly encode all the LSB of a block of coefficients.
Foreman QCIF – 400 frames

		No motion search	Coarse motion search (Search step)				Full motion search
			16	8	4	2	
Q. Step	8	33.28	32.00	31.36	30.08	26.88	23.04
	16	17.28	16.64	16.00	14.72	13.44	10.88
	32	7.68	7.04	7.04	6.40	5.76	5.12
	64	3.20	2.56	2.56	2.56	2.56	1.92

TABLE II. Average number of bits necessary to losslessly encode all the LSB of a block of coefficients.
Mother & Daughter QCIF – 400 frames

		No motion search	Coarse motion search (Search step)				Full motion search
			16	8	4	2	
Q. Step	8	6.464	6.528	6.656	6.720	6.592	6.272
	16	2.688	2.816	2.880	3.008	2.944	3.008
	32	0.960	1.152	1.280	1.408	1.536	1.664
	64	0.256	0.512	0.704	0.832	0.960	0.960

TABLE III. Average number of bits necessary to losslessly encode all the l_i of a block of coefficients.
Foreman QCIF – 400 frames

		No motion search	Coarse motion search (Search step)				Full motion search
			16	8	4	2	
Q. Step	8	65.28	64.00	62.72	62.72	60.80	56.96
	16	40.96	39.68	39.04	39.04	37.76	35.20
	32	22.40	21.76	21.12	21.76	21.12	20.48
	64	10.24	9.60	9.60	10.88	10.88	10.88

TABLE IV. Average number of bits necessary to losslessly encode all the l_i of a block of coefficients.
Mother & Daughter QCIF – 400 frames

		No motion search	Coarse motion search (Search step)				Full motion search
			16	8	4	2	
Q. Step	8	22.016	22.080	22.208	22.784	22.848	22.784
	16	10.944	11.200	11.392	12.224	12.544	13.568
	32	4.608	5.120	5.568	6.592	7.680	8.640
	64	1.600	2.432	3.008	4.288	5.120	5.440

Coding and in particular to suppress its training stage. First results show that the approach is promising although more work is needed as explained below.

First of all, a mode decision mechanism like the one used in PRISM [3] should be implemented. In this way, blocks very correlated to the previous co-located block could be simply skipped, and blocks very uncorrelated to the encoder's prediction could be intra-encoded

Another way to lower the bitrate would be to transmit fewer bits than those calculated by equation (1). Since the decoder has access to a better prediction than the encoder, the l_i calculated at the encoder are, in general, greater than strictly needed by the decoder.

A complete codec that exploits the studied statistics should also be implemented, so that rate-distortion curves can be generated and compared to other codecs like state-of-the-art predictive coding standards.

The next step should then be the simulation under noisy channel conditions, since this is one point where DVC is very likely to outperform predictive coding.

TABLE V. Average number of bits necessary to losslessly encode all the l_i of a block of coefficients, when using conditional encoding with different contexts.
Foreman QCIF – 400 frames

		Quantization Step			
		8	16	32	64
Contexts	No context: independent encoding	65.28	40.96	22.40	10.24
	Left coefficient	58.88	37.12	20.48	9.60
	Left & upper coefficient	55.04	34.56	19.20	8.96
	Left, upper & and left-upper coefficient	53.12	33.28	18.56	8.96
	Previous coefficient in zigzag scan order	59.52	37.12	21.12	9.60
	Previous 2 coefficients in zigzag scan order	57.60	36.48	20.48	9.60
	Previous 3 coefficients in zigzag scan order	56.32	35.84	19.84	9.60

7. REFERENCES

[1] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources", *IEEE Trans. Inform. Theory*, vol. 19 pp. 471-480, July 1973.

[2] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder", *IEEE Trans. Inform. Theory*, vol. 22, pp. 1-10, January 1976.

[3] R. Puri and K. Ramchandran. "PRISM: A new robust video coding architecture based on distributed compression principles". *Proc. of 40th Allerton Conf. on Comm., Control, and Computing*, Allerton, IL, Oct. 2002.

[4] B. Girod, A. Aaron, S. Rane and D. Rebollo-Monedero, "Distributed video coding", P*roc. of the IEEE*, vol. 93, no. 1, January 2005

[5] A. Majumdar, R. Puri, P. Ishwar and K. Ramchandran, "Complexity/performance trade-offs for robust distributed videocoding", *Internation Conference on Image processing (ICIP),* Genova, Italy, September 2005.