Wireless Sensor Network, **2009, 1, 334-339**
doi:10.4236/wsn.2009.14041 Published Online November 2009 (http://www.scirp.org/journal/wsn).

Distributed Video Coding Using LDPC Codes for Wireless Video

P. APARNA, Sivaprakash REDDY, Sumam DAVID[*]
Department of Electronics and Communication Engineering,
National Institute of Technology Karnataka, Surathkal, India
Email: *sumam@ieee.org*

Received May 19, 2009; *revised June* 16, 2009; *accepted June* 23, 2009

Abstract

Popular video coding standards like H.264 and MPEG working on the principle of motion-compensated predictive coding demand much of the computational resources at the encoder increasing its complexity. Such bulky encoders are not suitable for applications like wireless low power surveillance, multimedia sensor networks, wireless PC cameras, mobile camera phones etc. New video coding scheme based on the principle of distributed source coding is looked upon in this paper. This scheme supports a low complexity encoder, at the same time trying to achieve the rate distortion performance of conventional video codecs. Current implementation uses LDPC codes for syndrome coding.

Keywords: Syndrome Coding, Cosets, Distributed Source Coding, Distributed Video Coding (DVC).

1. Introduction

With the proliferation of various complex video applications it is necessary to have advanced video and image compression techniques. Popular video standards like ISO MPEG and ITU-H.26x have been successful in accomplishing the requirements in terms of compression efficiency and quality. However these standards are pertinent to downlink friendly applications like video telephony, video streaming, broadcasting etc. These conventional video codecs work on the principle of motion compensated prediction which increases the encoder complexity due to the coexistence of the decoder with the encoder. Also motion-search algorithm makes the encoder computationally intensive. The downlink friendly architectures belong to the class of *Broadcast* model, where in high encoder complexity is not an issue. The encoder of a *Broadcast* model resides at the base-station where power consumption and computational resources are not an issue. However this *Broadcast* model of video is not suitable for uplink friendly applications like mobile video cameras, wireless video sensor networks, wireless surveillance etc which demands a low power, low complexity encoder. These uplink friendly applications which belong to *wireless-video* model demands a simple encoder since the power and the computational resources are of primary concern in the wireless scenario. Based on the information theoretic bounds established in 1970's by Slepian-Wolf [1] for distributed lossless coding and by Wyner-Ziv [2] for lossy coding with decoder side information, it is seen that efficient compression can also be achieved by exploiting source statistics partially or wholly at the decoder. Video compression schemes that build upon these theorems are referred as distributed video coding which befits uplink friendly video applications. Distributed video coding shifts the encoder complexity to the decoder making it suitable for wireless video model. Unlike conventional video codecs distributed coding exploits the source statistics at the decoder alone, thus interchanging the traditional balance of complex encoder and simple decoder. Hence the encoder of such a video codec is very simple, at the expense of a more complex decoder. Such algorithms hold great promise for new generation mobile video cameras and wireless sensor networks. In the design of a new video coding paradigm, issues like compression efficiency, robustness to packet losses, encoder complexity are of prime importance in comparison with conventional coding system. In this paper we present the simulation results of distributed video coding with syndrome coding as in PRISM [3], using LDPC codes for coset channel coding [4].

2. Background

2.1. Slepian-Wolf Theorem for Lossless Distributed Coding [1]

Consider two correlated information sequences X and Y.

[*] SMIEEE.

Encoder of each source is constrained to operate without the knowledge of the other source while the decoder has access to both encoded binary message streams as shown in Figure 1. The problem that Slepian-Wolf theorem addresses is to determine the minimum number of bits per source character required for encoding the message stream in order to ensure accurate reconstruction at the decoder. Considering separate encoder and the decoder for X and Y, the rate required is $R_X \geq H(X)$ and $R_Y \geq H(Y)$ where $H(X)$ and $H(Y)$ represents the entropy of X and Y respectively. Slepian-Wolf [1] showed that good compression can be achieved with joint decoding but separate encoding.

For doing this an admissible rate region is defined [6] as shown in Figure 2 given by:

$$R_X + R_Y \geq H(X,Y) \quad (1)$$

$$R_X \geq H(X/Y),\ R_Y \geq H(Y) \quad (2)$$

$$R_X \geq H(X),\ R_Y \geq H(Y/X) \quad (3)$$

Thus Slepian-Wolf [1] showed that Equation (1) is the necessary condition and Equation (2) or Equation (3) are the sufficient conditions required to encode the data in case of joint decoding. With the above result as the base, we can consider the distributed coding with side information at the decoder as shown in the Figure 3. Let X be the source data that is statistically dependent to the side information Y. Side information Y is separately encoded at a rate $R_Y \geq H(Y)$ and is available only at the decoder. Thus as seen from Figure 2 X can be encoded at a rate $R_X \geq H(X/Y)$.

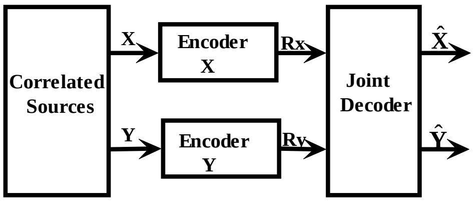

Figure 1. Compression of correlated sources by separate encoder but decoded jointly.

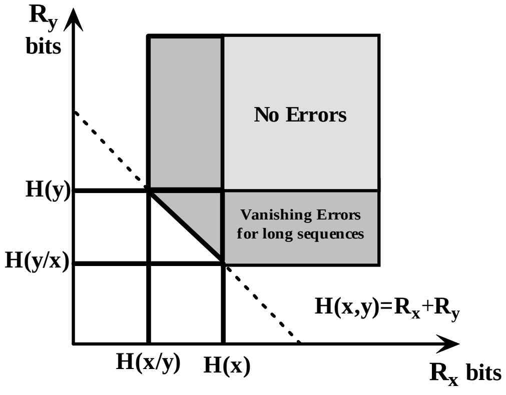

Figure 2. Admissible rate region [5].

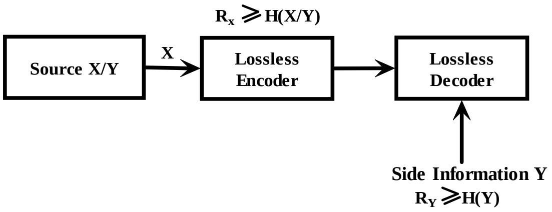

Figure 3. Lossless decoder with side information.

2.1. Wyner-Ziv Rate Distortion Theory[2,6]

Aaron Wyner and Jacob Ziv [2,6] extended Slepian-Wolf theorem and showed that conditional Rate-MSE distortion function for X is same whether the side information is available only at the decoder or both at encoder and decoder; where X and Y are statistically dependent Gaussian random processes. Let X and Y be the samples of two random sequences representing the source data and side information respectively. Encoder encodes X without access to side information Y as shown in Figure 4.

Decoder reconstructs $\hat{X}$ using Y as side information. Let D = E [d ($\hat{X}$, X)] is the acceptable distortion. Let $R_{X/Y}(D)$ be the rate required for the case where side information is available at the encoder also and $R^{WZ}_{X/Y}(D)$ represent the Wyner-Ziv rate required when encoder doesn't have access to side information. Wyner-Ziv proved that Wyner-Ziv rate distortion function $R^{WZ}_{X/Y}(D)$ is the achievable lower bound for the bitrate for a distortion D

$$R^{WZ}_{X/Y}(D) - R_{X/Y}(D) \geq 0 \quad (4)$$

They also showed that for Gaussian memoryless sources

$$R^{WZ}_{X/Y}(D) - R_{X/Y}(D) = 0 \quad (5)$$

As a result source sequence X can be considered as the sum of arbitrarily distributed side information Y and independent Gaussian Noise.

Distributed video coding is based on these two fundamental theories, specifically works on the Wyner-Ziv coding considering a distortion measure. In such a coding system the encoder encodes each video frame separately

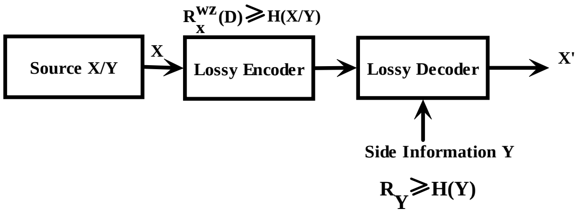

Figure 4. Lossy decoder with side information.

with respect to the correlation statistics between itself and the side information. The decoder decodes the frames jointly using the side information available only at the decoder. This video paradigm is as opposed to the conventional coding system where the side information is available both at the encoder and decoder as shown in Figure 5.

2.2. Syndrome Coding [5]

Let X be a source that is to be transmitted using least average number of bits. Statistically dependent side information Y, such that $X = Y + N$ is available only at the decoder. The encoder must therefore encode X in the absence of Y, whereas the decoder jointly decodes X using Y. Distributed source encoder compresses X in to syndromes S with respect to a Channel code C [7]. Decoder on receiving the syndrome can identify the coset to which X belongs and using side information Y can reconstruct back X.

2.3. Correlation Channel and the Channel Codes [4]

The performance of the channel codes is the key factor of the distributed video coding system in both error correcting and data compression. Turbo and LDPC codes are two advanced channel codes which have astonishing performance near the Shannon Capacity limit. The use of LDPC codes for syndrome coding was first suggested by Liveris in [4], where the message passing algorithm was modified to take syndrome information in to account.

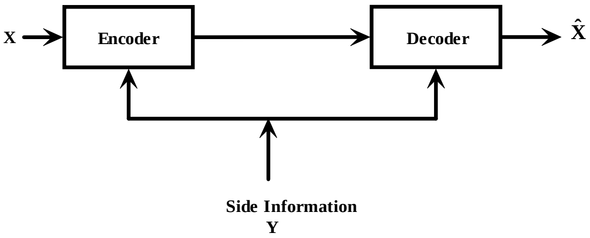

Figure 5. Lossless decoder with side information.

The correlation between binary sources $X = [X_1, X_2.....,X_n]$ and $Y = [Y_1, Y_2, ..., Y_n]$ is modeled using a binary symmetric channel. We consider X_i and Y_i to be correlated according to $Pr\ [X_i \neq Y_i] = p < 0.5$. The rate used for Y is its entropy $R_Y = H(Y)$, therefore the theoretical limit for lossless compression of X is given by

$$nR_x \geq nH(X_i/Y_i) = nH(p) = n(-plog_2p-(1-p)log_2(1-p)) \tag{6}$$

The compressed version of X is the syndrome S which is the input to the channel. The source Y is assumed to be available at the decoder as side information. Using a linear (n,k) binary block code, it is possible to have 2^{n-k} distinct syndromes, each indexing a set of 2^k binary words of length n. This compression results in mapping a sequence of n input symbols into $(n-k)$ syndrome symbols.

3. Implementation

3.1. Encoder

The encoder block diagram is shown in the Figure 6. The video frames are divided into blocks of 8x8 and each block is processed one by one. Block DCT (Discrete Cosine Transform) is applied to each 8x8 block (or 16x16) and the DCT coefficients are zig-zag scanned so that they are arranged as an array of coefficients in order of their importance. Then the transformed coefficients are uniform quantized with reference to target distortion measure and desired reconstruction quality. After quantization a bitplane is formed for each block as shown in Figure 7 [3]. Main idea behind distributed video coding is to code source X assuming that the side information Y is available at the decoder such that $X = Y + N$, where N is Gaussian random noise. This is done in the classification step where bitplane for each coefficient is divided into different levels of importance. Classification step strongly rely on the correlation noise

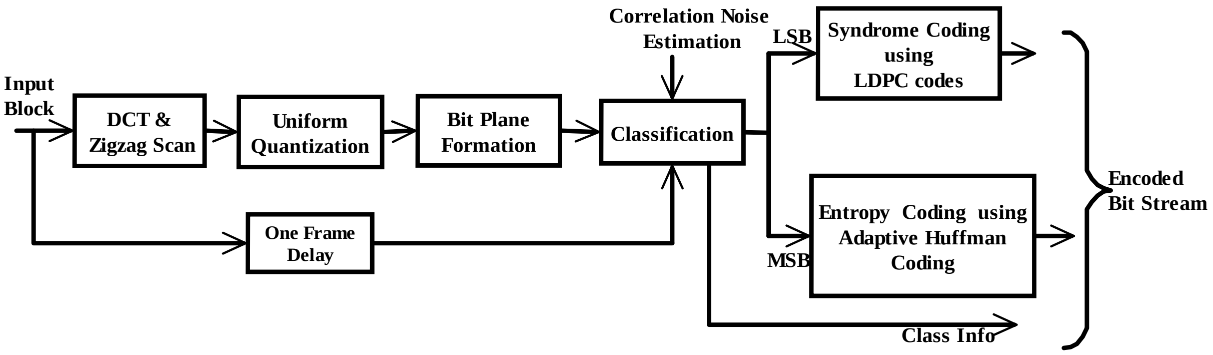

Figure 6. Video encoder.

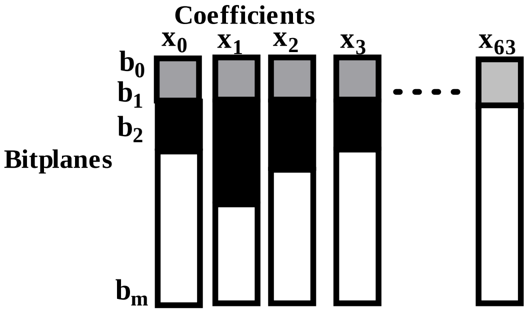

Figure 7. Bit planes for each coefficient blocks.

structure N between the source block X and the side information block Y. Less is the correlation noise between X and Y, more is the similarity and hence less number of bits of X can be transmitted to the decoder. In order to classify the bitplanes offline training is done for different types of video files without any motion search. On the basis of offline process 16 types of classes are formed, where each class considers different number of bitplanes for entropy coding and syndrome coding for each coefficient in the block. In the classification process, MSE (mean square error) for each block is computed with respect to the zero motion blocks in the previous frame. Based on the MSE and the offline process appropriate class for that particular block is chosen. As a result some of the least significant bit planes are syndrome coded and some of the bitplanes that can be reconstructed from side information are totally ignored. The syndrome coding bitplanes shown in black and gray in Figure 7 and skip planes shown in white in Figure 7. Skip planes can be reconstructed back using side information at the decoder and hence need not be sent to the decoder. The important bits of each coefficient that cannot be determined by side information has to be syndrome coded [3]. In our implementation we code two bitplanes using coset channel coding and the remaining syndrome bitplanes using Adaptive Huffman coding. Among the syndrome coding bitplanes we code the most significant bit planes using Adaptive Huffman coding. The number of bitplanes to be syndrome coded is directly used from class information that is hard coded. Hence we need not send four-tuple data (run, depth, path, last) as in PRISM [3]. Rest of the least significant bitplanes is coded using coset channel coding. This is done by using a parity check matrix H of a (n,k) linear channel code. Compression is achieved by generating syndrome bits of length $(n\text{-}k)$ for each n bits of data. These syndrome bits are obtained by multiplying the source bits with the parity check matrix H such that

$$S = Hb_X$$

where S represents the syndrome bits. H represents the parity check matrix of linear channel code. b_X represents the source bits.

These syndromes identify the coset to which the source data belongs to. In this implementation we have considered two biplanes for coset coding marked gray in the Figure 7. We have implemented this using irregular 3/4 rate LDPC coder [4].

3.2. Decoder

The Decoder block diagram is shown in the Figure 8. The entropy coded bits are decoded by an entropy decoder and the coset coded bits are passed to the LDPC decoder. In this implementation, previous frame is considered as the side information required for syndrome decoding. Once the syndrome coded bits are recovered they identify the coset to which X_i belongs and hence using the side information Y_i we can correctly decode the entire bits of X_i. The quantized codeword sequence is then dequantized and inverse transformed to get the original coefficients.

4. Simulation Results

Video Codec is designed for a single camera scenario which is an application to wireless network of video camera equipped with cell phones. The video codec is simulated and tested with a object oriented approach

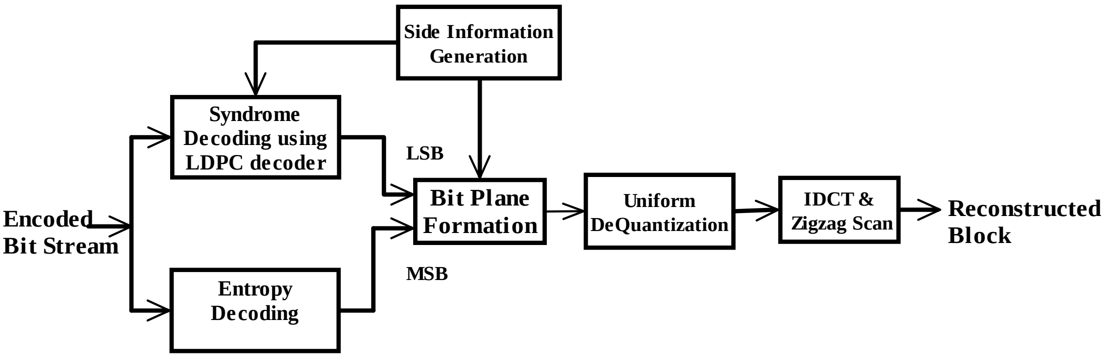

Figure 8. Video decoder.

Table 1. Filename: foreman. QCIF, frame rate=30fps.

	Luma PSNR (dB) for different Methods		
BitRate (Mbps)	DVC Implementation	H.263+Predictive Coder	IntraCoder (Motion JPEG)
2.57	31.357	34.72	30.092
2.67	33.554	35.03	32.863
3.55	35.534	35.86	34.92

Table 2. Filename: football. QCIF, frame rate=30fps.

	Luma PSNR (dB) for different Methods		
BitRate (Mbps)	DVC Implementation	H.263+ Predictive Coder	IntraCoder (Motion JPEG)
3.52	30.724	25.62	30.07
3.67	31.834	25.76	30.92
4.87	34.005	26.59	33.80

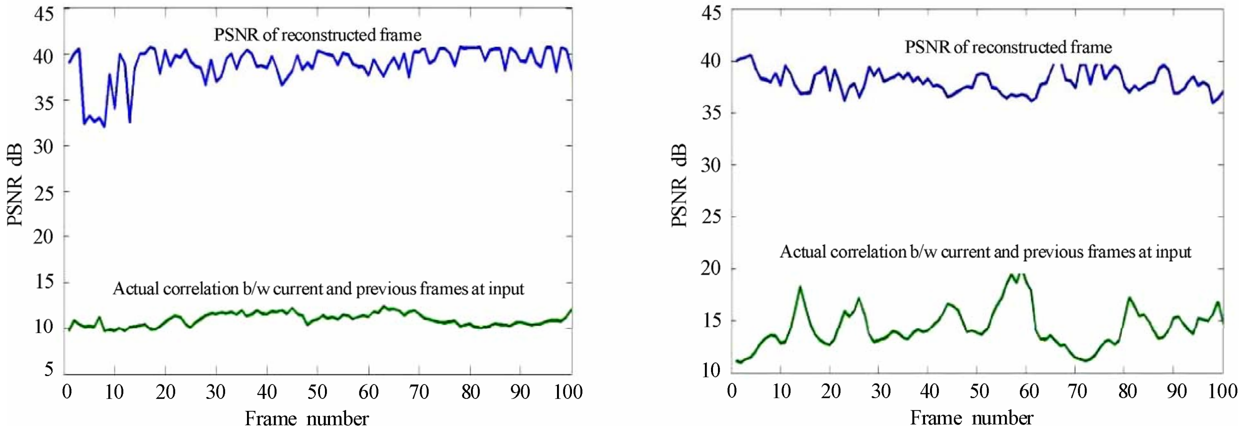

Figure 9. a) Error resilience characteristics of DVC, 4th, 10th, 20th frames are lost for football; b) Error resilience characteristics of DVC, 4th, 10th, 20th frames are lost for foreman.

using C++ in gcc. The program processes frames one by one and within each frame, block wise processing is done. The input to the encoder is a QCIF video file (Quarter Common Intermediate Format). Encoder allows the storage of one previous frame. Objective performance evaluation of the system is done by measuring the Compression Ratio (CR), MSE and the Peak Signal to Noise Ratio (PSNR) between the original and the reconstructed video. The PSNR and CR for various video sequences is computed. These are compared with that of H.263+ Intra and H.263+ Predictive video codec [8]. The encoder and decoder block as shown in Figure 6 and Figure 8 respectively are implemented and some preliminary simulation results are presented in this paper for two video files *Football* and *Foreman* in *QCIF resolution* with a frame rate of 30 *fps*. The rate distortion performance and the error resilience characteristics of the distributed video coder is presented in this paper. As seen from the Table 1, for the same bitrate distributed video coder has better PSNR than DCT based intraframe coder and but is slightly inferior to H.263+ predictive coder [8] for *Foreman* file. As seen from Table 2 distributed video coder has better PSNR than DCT based intraframe coder and H.263+ predictive coder for *Football* file. With some enhancements to the current coding scheme such as accurate modeling of correlation statistics between the source data and the side information, proper motion search module for side information generation etc, better rate-distortion performance can be achieved with a low complexity encoder model.

Error Resilience characteristics of Distributed video scheme is as shown in Figure 9a for *Football* and Figure 9b for *Foreman*. Effect on the quality of the reconstructed video sequence is seen by dropping 4th, 10th, 20th frames at the decoder in our implementation. It is seen that distributed video coder recovers quickly. In Distributed video scheme, decoding is dependent on the side information *Y* that is universal for all source data *X* as long as correlation structure is satisfied.

5. Conclusion

In this paper we have tried PRISM [3] like implementation using LDPC coset channel coding. By proper modeling of correlation structure of source and the side information for video we can achieve better compression performance with better quality of reconstructed video sequence. However the main aim of distributed video coding scheme is to reduce encoder complexity to conform with *wireless-video* model, which seems to be satisfied. Distributed codec is more robust to packet /frame loss due to the absence of prediction loop in the encoder. In a Predictive coder accuracy of decoding is strongly dependent on a single predictor from the encoder, loss of which results in erroneous decoding and error propagation. Hence Predictive coder can recover from packet or frame loss by only some extent. The quality of the reconstructed signal for the same CR can be improved by performing more complex motion search. However it is seen that the current implementation operates well in high quality (PSNR of order of 30dB) regime. The extension to lower bit rates without any compromise in the quality so that it is comparable with the conventional codecs will be the next part of the work.

6. References

[1] J. D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," IEEE Transactions on Information Theory, Vol. IT-19, pp. 471–480, July 1973.

[2] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," IEEE Transactions on Information Theory, Vol. IT-22, No. 1, pp. 1–10, January 1976.

[3] R. Puri, A. Majumdar, and K. Ramachandran, "PRISM: A video coding paradigm with motion estimation at the decoder," IEEE Transactions on Image Processing, Vol. 16, No. 10, October 2007.

[4] A. D. Liveris, "Compression of binary sources with side information at the decoder using LDPC codes," IEEE Communication Letters, Vol. 6, No. 10, October 2002.

[5] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): Design and construction," Proc. IEEE Data Compression Conference, Snowbird, UT, pp. 158–167, March 1999.

[6] A. D. Wyner, "Recent results in the Shannon theory," IEEE Transactions on Information Theory, Vol. 20, No. 1, pp. 2–10, January 1974.

[7] R. Puri and K. Ramchandran, "PRISM: A new robust video coding architecture based on distributed compression principles," Proc. Allerton Conference on Communication, Control and Computing, Allerton, IL, October 2002.

[8] G. Cote, B. Erol, M. Gallant, and F. Kosssentini, "H.263+: Video coding at low bitrates," IEEE Transactions. Circuits Sys. Video Technology, Vol. 8, No. 7, pp. 849–866, November 1998.

[9] B. Girod, A. M. Aaron, S. R. and D. Rebollo-Monedero, "Distributed video coding," Proceedings of the IEEE, Vol. 93, No. 1, pp. 71–83, January 2005.