WYNER-ZIV CODING OF MULTIVIEW IMAGES WITH UNSUPERVISED LEARNING OF DISPARITY AND GRAY CODE

David Chen, David Varodayan, Markus Flierl, Bernd Girod

Information Systems Laboratory, Stanford University, Stanford, CA 94305
{dmchen, varodayan, mflierl, bgirod}@stanford.edu

ABSTRACT

Wyner-Ziv coding of multiview images avoids communications between source cameras. To achieve good compression performance, the decoder must relate the source and side information images. Since correlation between the two images is exploited at the bit level, it is desirable to map small Euclidean distances between coefficients into small Hamming distances between bitwise codewords. This important mapping property is not achieved with the binary code but can be achieved with the Gray code. Comparing the two mappings, it is observed that the Gray code offers a substantial benefit for unsupervised learning of unknown disparity but provides limited advantage if disparity is known. Experimental results with multiview images demonstrate the Gray code achieves PSNR gains of 2 dB over the binary code for unsupervised learning of disparity.

Index Terms— stereo vision, multiview images, Gray code

1. INTRODUCTION

Multiview images captured by a camera array are very similar, and exploiting these similarities is desirable for efficient compression. The conventional approach requires a joint encoder, but this method is not practical if the cameras do not communicate with one another. Distributed coding has emerged as an alternative, with separate encoders and a joint decoder. The information theoretic Slepian-Wolf and Wyner-Ziv theorems suggest that distributed coding can be as efficient in coding performance as conventional joint compression [1][2].

A Wyner-Ziv multiview image codec in [3] proposes decoder-side unsupervised learning of disparity between two multiview images using the Expectation Maximization (EM) algorithm. It is a lossy extension of a lossless Slepian-Wolf codec reported in [4]. The Wyner-Ziv encoder converts quantized transform coefficients of the source image X into a low-density parity check (LDPC) bitstream [5]. Using the received segments of the LDPC bitstream and a side information image Y, the Wyner-Ziv decoder progressively learns the disparity between X and Y, disparity-compensates Y, and decodes a lossy reconstruction of X.

An inefficiency in the existing Wyner-Ziv system of [3] is the usage of the two's complement binary code as the bit representation of transform coefficients. The binary code does not consistently map small Euclidean distances between coefficients to small Hamming distances between the corresponding codewords. Even two consecutive integers may have codewords whose Hamming distance is much larger than one. When small Euclidean differences between the source and side information coefficients result in unnecessarily large Hamming distances, decoding efficiency is reduced. Comparing just the source and side information bitstreams, the decoder can be misled into wrong conclusions about the proximity of the side information to the source.

The Gray code is well known to have the desirable property of mapping two consecutive integers into two codewords separated by a Hamming distance of one [6]. Compared to the binary code, the Gray code much more consistently maps small Euclidean distances into small Hamming distances. This is a highly beneficial property for distributed coding. The decoder is now limited by the accuracy of the side information and not by any inherent deficiency of the bit representation. Previously, the Gray code has been applied in [7] to improve distributed video coding, but there conditional bitplane coding is used rather than joint bitplane coding as in [3] and no unsupervised learning at the decoder is employed.

In this paper, we improve the system in [3] by replacing the binary code with the Gray code and show that the Gray code has significant impact on unsupervised learning of unknown disparity. Section 2 reviews the existing disparity learning codec that uses the binary code. Section 3 presents the improved codec with the Gray code. A new joint bitplane encoder (decoder) is constructed in the Wyner-Ziv encoder (decoder). In Section 4, experimental results with actual multiview images demonstrate that the Gray code enables PSNR gains of 2 dB over the binary code. With the improved bit representation, the performance gap to a decoder that receives optimal disparity information from an oracle is significantly reduced.

2. EXISTING CODEC WITH THE BINARY CODE

The Wyner-Ziv image codec from [3] can be summarized by the block diagram in Fig. 1. One image Y is transmitted by conventional coding. The other image X is encoded independently of Y but decoded using Y as side information.

2.1. Wyner-Ziv Encoder

To exploit spatial correlation in the pixel domain, the Wyner-Ziv encoder transforms X through an 8-by-8 blockwise discrete cosine transform (DCT) and quantizes the DCT coefficients X^T. The integer-valued quantized coefficients X^{TQ} are then converted into a source bitstream X_{bin}^{TQ} using an eight-bit two's complement binary code. The source bitstream is further encoded using an LDPC accumulate (LDPCA) code [5]. An LDPCA code enables small segments of the syndrome bitstream S_{bin} to be incrementally sent.

2.2. Wyner-Ziv Decoder

The Wyner-Ziv decoder iteratively learns the quantized coefficients X^{TQ} from the received segments of the syndrome bitstream S_{bin} and the side information image Y. When X^{TQ} cannot be losslessly recovered, another small segment of S_{bin} is requested by the decoder

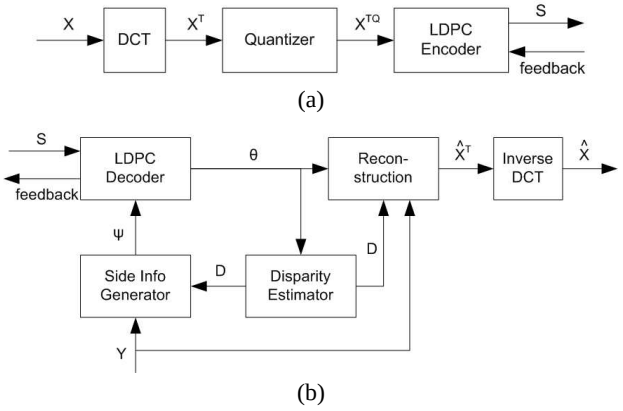

Fig. 1. Wyner-Ziv image (a) encoder and (b) decoder.

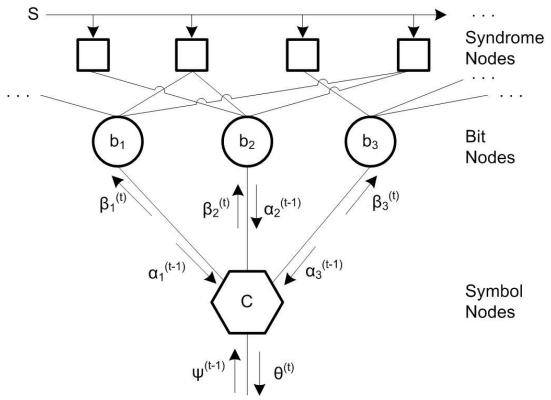

Fig. 2. LDPC joint bitplane decoder graph.

via a feedback channel. A significant bit rate saving over conventional lossless transmission of X^{TQ} occurs when the disparity D between X and Y is accurately estimated and only a short portion of S_{bin} needs to be sent. After X^{TQ} is decoded, the coefficients X^T are reconstructed as $\hat{X}^T$ and image X is reconstructed as $\hat{X}$.

The loop comprising the LDPC decoder, the disparity estimator, and the side information generator is an instance of the EM algorithm, as formally presented in [4]. Given θ, which is a probabilistic estimate of X_{bin}^{TQ}, the disparity estimator calculates a distribution on D. Then, given the distribution on D, the side information generator synthesizes ψ, which is a probabilistic estimate of the quantized transform coefficients of disparity-compensated Y. Finally, given ψ and S_{bin}, the LDPC decoder updates θ by joint bitplane decoding [4].

In this system, joint bitplane decoding is preferred over conditional bitplane decoding because disparity estimation requires having a distribution over all bitplanes simultaneously. Joint bitplane decoding can be visualized using the graph in Fig. 2. First, the syndrome nodes are filled with the received bits of the syndrome bitstream S_{bin}. Second, the bit nodes contain probabilistic distributions of the source bitstream X_{bin}^{TQ}. Third, the symbol nodes contain probabilistic distributions of the quantized coefficients of the disparity-compensated side information. Belief propagation between the syndrome and bit nodes represents classical LDPC decoding subject to syndrome constraints, while belief propagation between the bit and symbol nodes represents joint bitplane decoding. The additional level of symbol nodes enables the relationship between bits of the same coefficient to be exploited.

An example of joint bitplane decoding for a three-bit binary code (eight levels of coefficient values) is presented in [4]. The bottom two levels of Fig. 2 show a close-up view of the interaction between three bit nodes and their common symbol node. Before iteration t of the EM algorithm, bit node b_i contains a single probability $\alpha_i^{(t-1)} \equiv P_{prior}^{(t-1)}\{b_i = 1\}$, an *a priori* belief remaining from previous message passing between b_i and connected syndrome nodes. The symbol node contains $\psi_{\text{bin}}^{(t-1)}$, an eight-leveled *a priori* distribution derived from the disparity-compensated side information,

$$\psi_{\text{bin}}^{(t-1)} \equiv (p_{000}, p_{001}, p_{010}, p_{011}, p_{100}, p_{101}, p_{110}, p_{111}), \quad (1)$$

where the binary codewords 000 through 111 count consecutively from 0 to 7 in most significant bit (MSB) first notation. In Fig. 2, b_1 is the MSB. Passing information from the symbol node to, say, bit node b_2, enables us to calculate an *a posteriori* belief $\beta_2^{(t} \equiv P_{post}^{(t)}\{b_2 = 1\}$. The update is

$$\begin{aligned} L_k &:= p_{0k0} \cdot \left(1-\alpha_1^{(t-1)}\right)\left(1-\alpha_3^{(t-1)}\right) \\ &+ p_{0k1} \cdot \left(1-\alpha_1^{(t-1)}\right)\alpha_3^{(t-1)} \\ &+ p_{1k0} \cdot \alpha_1^{(t-1)}\left(1-\alpha_3^{(t-1)}\right) \\ &+ p_{1k1} \cdot \alpha_1^{(t-1)} \cdot \alpha_3^{(t-1)}, \quad k \in \{0,1\}, \\ \beta_2^{(t)} &:= \frac{L_1}{L_1 + L_0}. \end{aligned} \quad (2)$$

Here, we do not use $\alpha_2^{(t-1)}$ to avoid recycling information. The indices 010, 011, 110 and 111 of $\psi_{\text{bin}}^{(t-1)}$ contribute to L_1, corresponding to the event $\{b_2 = 1\}$, while the indices 000, 001, 100, and 101 of $\psi_{\text{bin}}^{(t-1)}$ contribute to L_0, corresponding to the event $\{b_2 = 0\}$. Analogous updates are defined for $\beta_1^{(t)}$ and $\beta_3^{(t)}$.

To pass messages in the other direction, from bit nodes to the symbol node, a similar update is defined. The *a posteriori* coefficient distribution is $\theta_{\text{bin}}^{(t)}$, which is also eight-leveled and of the form

$$\theta_{\text{bin}}^{(t)} \equiv (q_{000}, q_{001}, q_{010}, q_{011}, q_{100}, q_{101}, q_{110}, q_{111}). \quad (3)$$

The update for $\theta_{\text{bin}}^{(t)}$ is defined to be:

$$\begin{aligned} q_{000} &:= p_{000} \cdot \left(1-\alpha_1^{(t-1)}\right) \cdot \left(1-\alpha_2^{(t-1)}\right) \cdot \left(1-\alpha_3^{(t-1)}\right), \\ q_{001} &:= p_{001} \cdot \left(1-\alpha_1^{(t-1)}\right) \cdot \left(1-\alpha_2^{(t-1)}\right) \cdot \alpha_3^{(t-1)}, \\ &\vdots \\ q_{111} &:= p_{111} \cdot \alpha_1^{(t-1)} \cdot \alpha_2^{(t-1)} \cdot \alpha_3^{(t-1)}, \end{aligned} \quad (4)$$

where $\theta_{\text{bin}}^{(t)}$ is properly normalized after (4). The updates in (2) and (4) are repeated for each group of three bit nodes and their common symbol node. Generalization of joint bitplane decoding for where eight bit nodes share a common symbol node follows in a straightforward fashion. In Section 3, we revisit this example and develop joint bitplane decoding for the Gray code.

3. IMPROVED CODEC WITH THE GRAY CODE

The same block diagram of Fig. 1 applies for the proposed codec. Important differences lie in the LDPC encoder and decoder, where the Gray code replaces the binary code to improve the efficiency of the bit representation.

Table 1. Coefficient values around zero, binary codewords, and Gray codewords.

Coefficient	Binary Code	Gray Code
-2	11111110	10000001
-1	11111111	10000000
0	00000000	00000000
1	00000001	00000001
2	00000010	00000011

3.1. Wyner-Ziv Encoder

The binary code used in the existing Wyner-Ziv codec often maps transform coefficients separated by small Euclidean distances into codewords separated by large Hamming distances. Consider some coefficient values near zero, such as those listed in the first column of Table 1, which are highly probable values for AC transform coefficients. The second column shows the corresponding binary codewords. Suppose the coefficient for the source image is 0 and the corresponding coefficient for the side information image is -1. The Euclidean distance between these coefficients is only 1, indicating a close match, but their binary codewords have Hamming distance of 8, which misleadingly suggests a larger mismatch than actually exists. Gray encoding solves this problem.

Among the different flavors of Gray codes, we chose the original binary-reflected Gray code [6]. If n_{bin} is the binary codeword for integer n, then the Gray codeword n_{Gray} is calculated as

$$n_{\text{Gray}} = n_{\text{bin}} \ \text{XOR} \ (n_{\text{bin}} \gg 1), \tag{5}$$

where the XOR is bitwise and $\gg$ represents rightward bitshift. The Gray codewords are shown in the third column of Table 1. For values 0 and -1, the Hamming distance of their Gray codewords reduces to 1 as desired, significantly better than the binary-coded case. Smaller Hamming distance implies a lower bit error rate (BER) in correlating the source and side information bitstreams and improved coding efficiency [7]. In the proposed codec, the encoder described in Sec. 2.1 is modified so that the Gray code replaces the binary code for the bit representation of the transform coefficients. The Gray-coded bitstream X_{Gray}^{TQ} is then LDPCA-encoded into a syndrome bitstream S_{Gray}.

3.2. Wyner-Ziv Decoder

A new joint bitplane decoder must be created, similar to that described in Sec. 2.2. In the decoding graph of Fig. 2, the syndrome and symbol nodes retain their original meanings, but the bit nodes have a new interpretation. Instead of representing bits in binary codewords, the bit nodes now represent bits in Gray codewords. The example of three-bit codewords from Sec. 2.2 is revisited to illustrate the key changes.

Each symbol node still contains an *a priori* eight-leveled distribution, now of the form

$$\psi_{\text{Gray}}^{(t-1)} \equiv (p_{000}, p_{001}, p_{011}, p_{010}, p_{110}, p_{111}, p_{101}, p_{100}) \tag{6}$$

which resembles (1) except for the different ordering of codewords. The Gray codewords 000 to 100 count consecutively from 0 to 7. Message passing from the coefficient node to bit node b_2 is the same as in (2), except the meaning of terms such as p_{010} has changed. Whereas p_{010} previously represented the probability for a coefficient value of $2 = 010_{\text{bin}}$, p_{010} now represents the probability for a coefficient value of $3 = 010_{\text{Gray}}$. The update steps for bit nodes b_1 and b_3 are modified analogously.

In the other direction, source-to-coefficient message updates the *a posteriori* distribution $\theta_{\text{Gray}}^{(t)}$, which has the form

$$\theta_{\text{Gray}}^{(t)} \equiv (q_{000}, q_{001}, q_{011}, q_{010}, q_{110}, q_{111}, q_{101}, q_{100}) \tag{7}$$

which is similar to (3) except with a new ordering of the codewords that reflects the change from the binary code to the Gray code. Update of $\theta_{\text{Gray}}^{(t)}$ is the same as in (4), with the proper reinterpretation of codewords. From an algorithmic point of view, joint bitplane decoding for the Gray code very much resembles joint bitplane decoding for the binary code. Decoding efficiency, however, is significantly improved, as demonstrated in Sec. 4.

(a) (b)

Fig. 3. Image X from multiview sets (a) *Teddy* and (b) *Barn*, each of size 176-by-144 and bit depth 8.

4. EXPERIMENTAL RESULTS

The performance of the disparity-learning codecs with the binary code and the Gray code are evaluated using two sets of multiview images from [8], which we name *Teddy* and *Barn*. One image from each set is shown in Fig. 3(a-b) and takes the role of the image X which is Wyner-Ziv coded. One other image in each set is chosen to serve as the reference image Y. We assume a high-quality version of Y resides at the decoder, having been previously transmitted using conventional coding. To simplify the experiment, X and Y are approximately related by a horizontal, integer-pel disparity field, measured at block resolution and limited to the range [-5,5]. Our system also works for larger, non-horizontal, non-integer disparities, but at the expense of longer decoding times.

The rate-distortion performance of different codecs are compared. First, an impractical system called *disparity oracle* allows the decoder to know the optimal blockwise disparities between X and Y. The disparity oracle is meant to measure an upper performance bound for practical *disparity learning* codecs. Second, the disparity learning codec with the binary code is tested. Last, the disparity learning codec substitutes the Gray code for the binary code. At the decoder, the EM algorithm is permitted to run for 50 iterations at each incremental rate of the LDPCA code. If after 50 iterations convergence is not achieved, the LDPC decoder requests additional bits to advance to the next higher incremental rate.

Rate-PSNR plots for *Teddy* and *Barn* are presented in Fig. 4. For *Teddy*, the learning system with the Gray code achieves 2 dB increase in PSNR over the disparity learning system with the binary code. Similarly for *Barn*, the Gray code performs better than the binary code, resulting in 2.5 dB higher PSNR for disparity learning. At high rates, disparity learning with the Gray code even outperforms

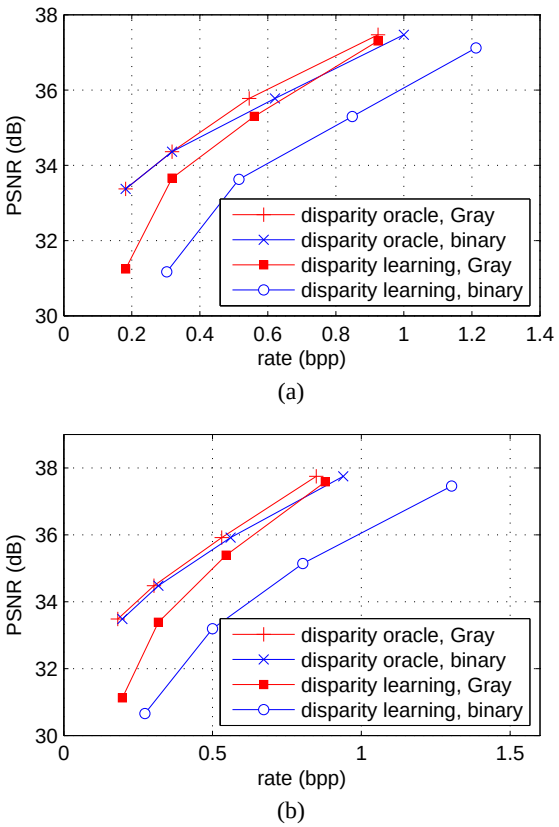

Fig. 4. Rate-PSNR performances for three different Wyner-Ziv decoders, evaluated for test image sets (a) *Teddy* and (b) *Barn*.

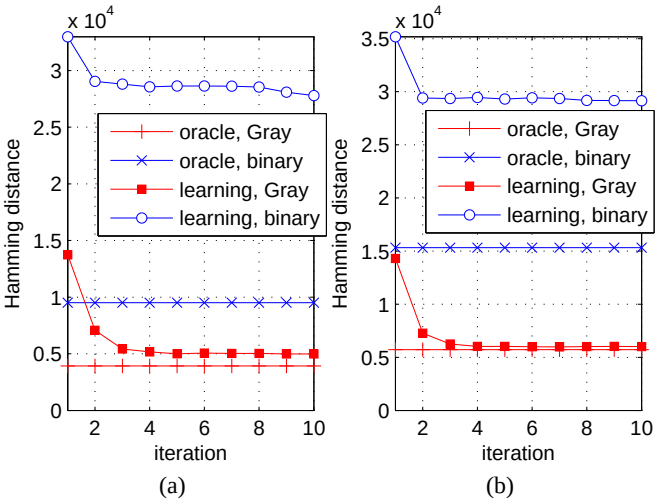

Fig. 5. Hamming distance between bitstreams of source X^{TQ} and disparity-compensated side information Y_D^{TQ}, evaluated for first 10 iterations of EM and for test image sets (a) *Teddy* at rate = 0.68 bpp, and (b) *Barn* at rate = 0.88 bpp.

disparity oracle with the binary code. The application of the Gray code significantly narrows the gap between disparity learning and oracle, which is not accomplished with the binary code.

The reason for the superior performance of the Gray code over the binary code, particularly for disparity learning, is the property of mapping small Euclidean distances into small Hamming distances. Fig. 5 shows the Hamming distance between bitstreams of source coefficients X^{TQ} and disparity-compensated side information coefficients Y_D^{TQ}, for the first 10 iterations of EM, at high rate. When disparity D is known perfectly, as for disparity oracle, the Euclidean distance between X^{TQ} and Y_D^{TQ} is relatively small. When disparity D is unknown, however, as in the early iterations of EM for disparity learning, the Euclidean distance between X^{TQ} and Y_D^{TQ} can be very large. As a consequence, the gap between the binary and Gray disparity learning curves in Fig. 5 is much greater than the gap between the binary and Gray disparity oracle curves. Thus, Hamming distance savings achieved with the Gray code are much more evident in disparity learning than in disparity oracle. This explains why in Fig. 4 the Gray code outperforms the binary code by only 0.2 dB for disparity oracle but by 2 dB for disparity learning.

5. CONCLUSION

This paper has presented an improved Wyner-Ziv multiview image codec that uses a Gray code instead of the binary code for bit repesentation of transform coefficients. The Gray code exhibits a desirable property of mapping small Euclidean distances between coefficients into small Hamming distances between codewords. This property is highly beneficial for unsupervised learning of unknown disparity in distributed coding. For actual multiview images, the Gray code produces 2 dB gain in PSNR compared to the binary code and at high rates enables disparity learning to perform nearly as well as disparity oracle.

6. REFERENCES

[1] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Info. Theory*, vol. 19, no. 4, 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 Trans. Info. Theory*, vol. 22, no. 1, pp. 1–10, Jan. 1976.

[3] D. Varodayan, Y.-C. Lin, A. Mavlankar, M. Flierl, and B. Girod, "Wyner-Ziv coding of stereo images with unsupervised learning of disparity," in *Proc. Picture Coding Symposium*, Lisbon, Portugal, Nov. 2007.

[4] D. Varodayan, A. Mavlankar, M. Flierl, and B. Girod, "Distributed grayscale stereo image coding with unsupervised learning of disparity," in *Proc. Data Compr. Conf.*, Snowbird, Utah, March 2007, pp. 143–152.

[5] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive distributed source coding using low-density parity-check codes," in *Proc. Asilomar Conf. on Signals, Systems, Comput.*, Pacific Grove, CA, Nov. 2005, pp. 1203–1207.

[6] J. R. Bitner, G. Ehrlich, and E. M. Reingold, "Efficient generation of the binary reflected gray code and its applications," *Commun. ACM*, vol. 19, no. 9, pp. 517–521, Sept. 1976.

[7] Z. He, L. Cao, and H. Cheng, "Correlation estimation and performance optimization for distributed image compression," in *Proc. Visual Commun. and Image Process.*, San Jose, California, Jan. 2006.

[8] D. Scharstein and R. Szeliski, "Middlebury stereo vision page," http://vision.middlebury.edu/stereo.