

# An Efficiency-Complexity Controllable Rate Adaptive Lossy Source Coding — Hybrid Majority Voting Code

Wensheng Lin, *Student Member, IEEE*, and Tad Matsumoto, *Fellow, IEEE*

**Abstract**—This letter proposes a practical source coding scheme, so-called hybrid majority voting (HMV) code, for lossy compression with discrete memoryless source. Inspired by the coding scheme used in the classic rate-distortion theorem, we find a series of basic MV codes and analyze their rate-distortion performance. We then present an algorithm to find two component MV codes and apply them to lossy compression, group by group, to construct HMV codes. Finally, we show an example of joint source-channel coding based on the HMV code. The performance evaluation indicates that the HMV code makes it possible to easily control efficiency and complexity.

**Index Terms**—Lossy source coding, rate-distortion, discrete memoryless source, rate adaptation, non-linear compression.

## I. INTRODUCTION

In coding theory, data compression is a classical topic including two fundamental categories, i.e., lossless and lossy. The lossless compression has been well studied during the last several decades, e.g., Shannon coding [1], Huffman coding [2] and Lempel-Ziv coding [3], [4]. Regarding lossy compression for continuous source and multimedia data, there are many technologies, such as pulse-code modulation (PCM) for continuous source, moving picture experts group (MPEG) audio layer 3 (MP3) for audio [5], Joint Photographic Experts Group (JPEG) [6] for image, and MPEG-4 [7] for video. Even though the multimedia data is in a digital format, they are basically continuous sources with correlations between information bits.

Nevertheless, the lossy compression for discrete memoryless source is not easy, because the distance between the codewords and the original sequences is considered to be more crucial than the correlations between symbols. Although the optimal performance can be achieved for sufficiently long sequence according to Shannon’s lossy source coding theorem [8], we need significantly huge memory to store the codebook for joint typicality coding [9]. For multiterminal lossy source coding, Wyner and Ziv derive the rate-distortion function for lossy compression with side information in [10]. Berger [11] and Tung [12] independently determine the outer and inner bounds on the achievable rate-distortion region for lossy compression with two sources. However, not only is it difficult to find a theoretically optimal codebook with respect to a specified distortion, but we also have to find diverse codebooks for

different rate or distortion requirements. Therefore, the joint typicality coding scheme is hard to implement in practical systems. In this letter, we are interested in developing a practical lossy source coding algorithm requiring relatively low complexity and exhibiting good performance.

Due to its simplicity, puncturing is frequently used in the design of various source and/or channel coding scheme. Notice that puncturing can be also regarded as a lossy source coding scheme. If there is a coding scheme that has a better performance than puncturing, we can improve the performance by replacing puncturing with some better coding scheme. Consider an  $n$ -bits binary sequence  $X^n$  with  $X \sim \text{Bern}(0.5)$ , if  $X^n$  is punctured with rate  $R$ , the punctured  $n(1-R)$  bits has 0.5 probability of error. Consequently, the expected distortion with puncturing is given by

$$D_p = \frac{0.5 * n(1-R)}{n} = \frac{1-R}{2}, \quad (1)$$

which is obviously linear to the rate  $R$ . Intuitively, lossy source coding can only achieve linear performance with linear algebra method, i.e., multiplying a generation matrix. Puncturing is equivalent to multiplying a diagonal matrix with the diagonal elements for the punctured bit being 0 and other diagonal elements being 1, and hence its performance is linear.

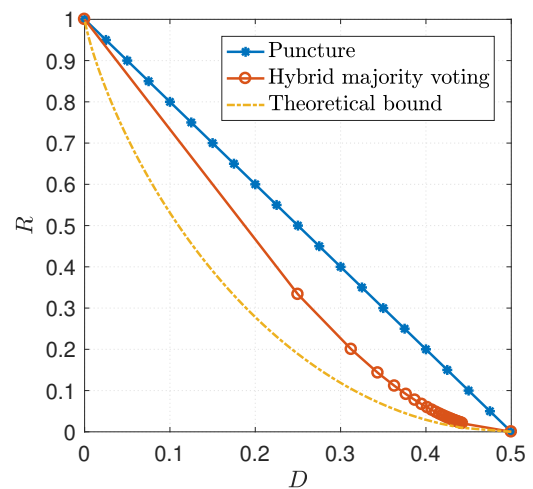


Fig. 1. The performance of hybrid majority voting code.

However, according to Shannon’s lossy source coding theorem, the binary rate-distortion function for  $X^n$  is

$$R = 1 - H_2(D), \quad (2)$$

where  $H_2(\cdot)$  represent the binary entropy function. In the achievability proof of Shannon’s lossy source coding theorem,  $2^n$  sequences  $x^n$  are mapped to  $2^{nR}$  sequences  $\hat{x}^n$  such that

Corresponding author: Wensheng Lin.

This work is funded in part by China Scholarship Council (CSC) and in part by Japan Advanced Institute of Science and Technology (JAIST) Core-to-Core Program.

W. Lin is with the School of Information Science, JAIST, 1-1 Asahidai, Nomi, Ishikawa, Japan 923-1292 (e-mail: linwest@jaist.ac.jp).

T. Matsumoto is with the School of Information Science, JAIST, 1-1 Asahidai, Nomi, Ishikawa, Japan 923-1292 (e-mail: matumoto@jaist.ac.jp). He is also with the Centre for Wireless Communications, University of Oulu, FI-90014 Finland (e-mail: tadashi.matsumoto@oulu.fi).

$(x^n, \hat{x}^n)$  is jointly typical. This joint typicality coding requires extremely huge memory for storing codebook; however, the codebook with non-linear mapping can achieve optimal performance as  $n \rightarrow \infty$ . Therefore, a coding scheme containing non-linear process is the key to make performance closer to the theoretical limit. Inspired by the joint typicality coding scheme, we find a rate adaptive lossy source coding scheme, named hybrid majority voting (HMV) code, which can achieve the performance shown in Fig. 1. The HMV code can make a trade off between efficiency and complexity.

The major contributions of this letter are as follows. We develop a practical lossy compression scheme, i.e., HMV code, with obviously higher efficiency than puncturing, even though its coding and decoding algorithms are very simple. After the theoretical rate-distortion analysis of the HMV code, we perform a series of simulations to make a performance comparison between a joint source-channel coding scheme with the HMV code and that with puncturing. The simulation result verifies that we can achieve significant performance gains with the HMV code instead of puncturing.

The rest of this letter is organized as follows. Section II presents the design and distortion analysis of basic MV code. Subsequently, we introduce the encoding and decoding algorithms of the HMV code in Section III. Section IV evaluate the performance of the HMV code utilized in a source-channel coding scheme through simulation. We finally conclude this letter in Section V.

## II. MAJORITY VOTING CODE

Consider a special case of joint typicality coding, i.e.,  $R = \frac{1}{n}$ , and we have to map  $2^n$  sequences to 2 sequence. It is not difficult to show that two sequences with all zeros and all ones are optimal, because they have maximum Hamming distance  $n$  with each other. Since the distance between all zeros/ones and other sequences are smaller than  $n$ , the distortion is smaller compared to using other codeword as the estimate of sequences. It is also obvious that in order to decrease the distortion, a sequence should be mapped to all ones if the sequence contains more 1 than 0; otherwise, the sequence should be mapped to all zeros. Then, we use only 1-bit codewords “0” and “1” to represent two sequences, i.e., all zeros and all ones, respectively, we have successfully compressed an  $n$ -bits sequence into 1-bit codeword. In summary, we compress an  $n$ -bits sequence into the codeword “1” if the number of ones is more than  $\frac{n}{2}$ ; otherwise, we compress it into “0”. By majority voting, we can simply perform the lossy compression described above, of which the performance becomes closer to the theoretical bound as  $n$  goes large.

Now, we analyze the expected distortion of the MV code. Without loss of generality, we assuming that we receive the 1-bit sequence “0”. For an integer  $l < \frac{n}{2}$ , the number of sequence with  $l$  bits errors is  $\binom{n}{l}$ . For  $l = \frac{n}{2}$  with even  $n$ , only half of the sequences are mapped to all zeros, and hence the number of sequence with  $l$  bits errors is  $\frac{1}{2}\binom{n}{l}$ . Since there are  $2^{(n-1)}$  sequences encoded to “0” and all sequences are generated with

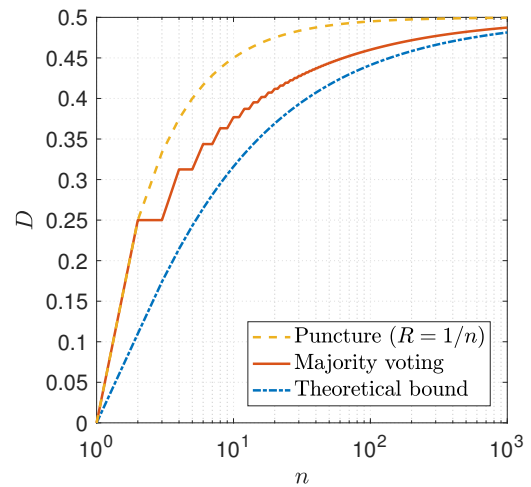


Fig. 2. The performance of majority voting code with diverse sequence length.

the same probability, the expected distortion of the MV code is calculated as

$$D_{MV}(n) = \begin{cases} \frac{1}{2^{n-1}} \sum_{l=0}^{\frac{n-1}{2}} \frac{l}{n} \binom{n}{l}, & \text{for } n \text{ is odd,} \\ \frac{1}{2^{n-1}} \left( \sum_{l=0}^{\frac{n}{2}-1} \frac{l}{n} \binom{n}{l} + \frac{1}{4} \binom{n}{\frac{n}{2}} \right), & \text{for } n \text{ is even.} \end{cases} \quad (3)$$

The performance of the MV code is depicted in Fig. 2, which demonstrates the expected tendency that the distortion of MV code becomes close to the theoretical bound for sufficiently long sequence.

Then, we start lossy compression with arbitrary  $R$  based on MV code. Notice that there are  $nR$  bits in encoded sequence. The most simple way is compressing  $k = n(1 - R) + 1$  bits into 1 bit and keeping the remaining  $nR - 1$  bits the same as origin. By this single-compression with MV code (SMV), we can satisfy the distortion

$$D_{SMV}(n, k) = \begin{cases} \frac{1}{2^{k-1}} \sum_{l=0}^{\frac{k-1}{2}} \frac{l}{n} \binom{k}{l}, & \text{for } k \text{ is odd,} \\ \frac{1}{2^{k-1}} \left( \sum_{l=0}^{\frac{k}{2}-1} \frac{l}{n} \binom{k}{l} + \frac{k}{4n} \binom{k}{\frac{k}{2}} \right), & \text{for } k \text{ is even.} \end{cases} \quad (4)$$

The performance of SMV is shown in Fig. 3 for various values of  $n$ . It is sensible that the performance becomes even worse as  $n$  increases except the case with relatively small  $R$ . The reason for this observation is that the remaining  $nR - 1$  bits have not been exploited in lossy compression, and hence more bits are wasted when  $n$  becomes larger.

## III. HYBRID MAJORITY VOTING CODE

### A. Encoding

For the purpose of utilizing all remaining bits for lossy compression, a reasonable way is to divide the sequence into

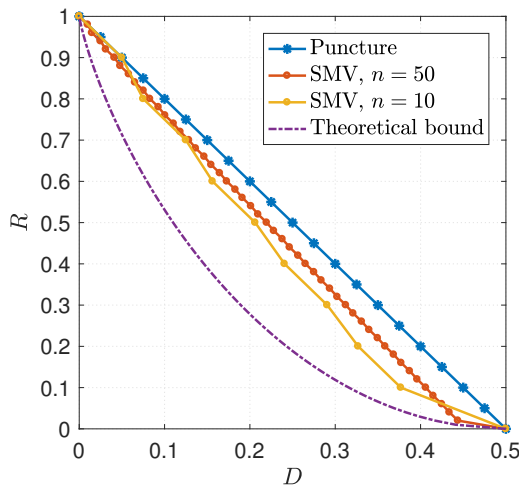


Fig. 3. The performance of single-compression with MV code.

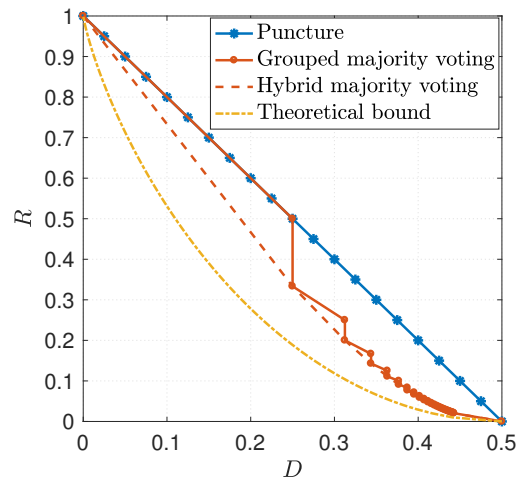


Fig. 4. The performance of grouped MV code.

$nR$  groups and then compress group by group. Assuming that the sequence can be divided into  $nR$  groups with equivalent size  $s = \frac{1}{R}$ , we can obtain the performance curve illustrated in Fig. 4. It should be noticed that the distortion for  $R = \frac{1}{s}$  with even  $s$  is the same as that for  $R = \frac{1}{s+1}$ , i.e., the distortion can be satisfied with lower rate. Therefore, the group with odd size is better than that with even size.

Now, we extend this idea to general cases with  $R$  being arbitrary number from 0 to 1. By mixing two MV codes with different odd sizes  $s$  and  $s+2$ , the average rate between  $\frac{1}{s}$  and  $\frac{1}{s+2}$  can be satisfied. If we use  $i$  groups of the rate  $\frac{1}{s}$  MV code and  $j$  groups of the rate  $\frac{1}{s+2}$  MV code as component codes, the average rate and the average distortion is a linear combination of the rate and distortion of two component codes, respectively. Since each group of the MV code yields 1 bit compressed coding result,  $i+j = nR$ . From the total bits in the MV code groups, we have  $i \cdot s + j \cdot (s+2) = n$ . Consequently, we can obtain  $i = n[(s+2)R - 1]/2$  and  $j = n(1 - sR)/2$ .

The algorithm of the HMV code, consisting of two component MV codes, is summarized in Algorithm 1, of which the gist is to first find appropriate values of  $s$ ,  $i$  and  $j$  given

### Algorithm 1 Hybrid Majority Voting Code

---

**Input:** sequence  $X^n$ ,  $R$   
**Output:** coded sequence  $Y^{nR}$   
 set  $s_{\text{mid}} = \text{floor}(1/R)$ ;  
**if**  $s_{\text{mid}}$  is odd **then**  
     set  $s = s_{\text{mid}}$ ;  
**else**  
     set  $s = s_{\text{mid}} - 1$ ;  
**end if**  
 set  $i = n[(s+2)R - 1]/2$  and  $j = n(1 - sR)/2$ ;  
**for**  $t = 1$  **to**  $i$  **do**  
     encode  $t$ -th group in  $X^n$  to  $Y(t)$  by the rate  $\frac{1}{s}$  MV code;  
**end for**  
**for**  $t = i + 1$  **to**  $i + j$  **do**  
     encode  $t$ -th group in  $X^n$  to  $Y(t)$  by the rate  $\frac{1}{s+2}$  MV code;  
**end for**

---

sequence length  $n$  and compression rate  $R$ . Then, encode group by group.

### B. Decoding

There are two types of decoding algorithm, i.e., hard decoding and soft decoding. The hard decoding algorithm for the HMV code is also very simple, i.e., repeating the coded bit  $s$  times for the first  $i$ -th group and  $(s+2)$  times for the last  $j$ -th group. Consequently, the expected distortion of the HMV code can be expressed as

$$D_{\text{HMV}} = \frac{i \cdot s \cdot D_{\text{MV}}(s) + j \cdot (s+2) \cdot D_{\text{MV}}(s+2)}{n}. \quad (5)$$

The soft decoding yields log-likelihood ratio (LLR) which is useful for exchanging the mutual information in iterative decoding of codes having multiple constituency components [13]. We first transform the received signal into LLR, and then take the expected distortion into account based on the correlation model in [14]. Assuming the *a priori* LLR of received signal is  $\text{LLR}^a$ , the *a posteriori* LLR ( $\text{LLR}^p$ ) of the coded bits can be calculated as

$$\text{LLR}^p = \log \frac{(1-d) \cdot \exp(\text{LLR}^a) + d}{(1-d) + d \cdot \exp(\text{LLR}^a)}, \quad (6)$$

where  $d$  is set at  $D_{\text{MV}}(s)$  and  $D_{\text{MV}}(s+2)$  for the group coded by the rate  $\frac{1}{s}$  and  $\frac{1}{s+2}$  MV codes, respectively. After calculating the  $\text{LLR}^p$  of coded  $(i+j)$ -bits, the  $\text{LLR}_X^p$  of the sequence  $X^n$  is reproduced by performing repetition of the corresponding bit in  $\text{LLR}^p$  for  $s$  and  $(s+2)$  times in the groups of the rate  $\frac{1}{s}$  and  $\frac{1}{s+2}$  MV codes, respectively. In this way, the  $\text{LLR}_X^p$  can be jointly exploited in further iteration of decoding, depending on specified algorithm for exchanging mutual information.

## IV. PERFORMANCE EVALUATION

This section shows a design example of practical joint source-channel coding scheme based on the HMV code and the turbo code. As illustrated in Fig. 5, the encoder first executes source encoding and then channel coding, with the

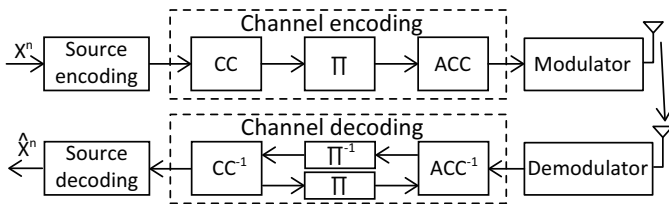


Fig. 5. The simulation system with practical joint source-channel coding scheme.

TABLE I  
BASIC SETTINGS OF SIMULATION PARAMETERS

Parameter	Value
Sequence length	10000 bits
The number of sequence	1000
Generator polynomial of CC	$G = ([3, 2]_3)_8$
Modulation method	BPSK
Maximum iteration time	20

source encoding using the HMV code or puncturing for the performance comparison. In the channel coding, the compressed sequence is encoded by a convolutional code (CC), and concatenated with an accumulator (ACC) [15] via a random interleaver  $\Pi$ . After modulation, the encoded sequence is transmitted through additive white Gaussian noise (AWGN) channel. In the receiver, the demodulated signal is iteratively decoded by the decoder of ACC ( $ACC^{-1}$ ) and the decoder of CC ( $CC^{-1}$ ) with extrinsic information transferred via  $\Pi$  and a deinterleaver  $\Pi^{-1}$ . Finally, the estimate  $\hat{X}^n$  is generated from the hard decision of channel decoding result by the decoding algorithm for the HMV code, or setting the punctured bits at 0.

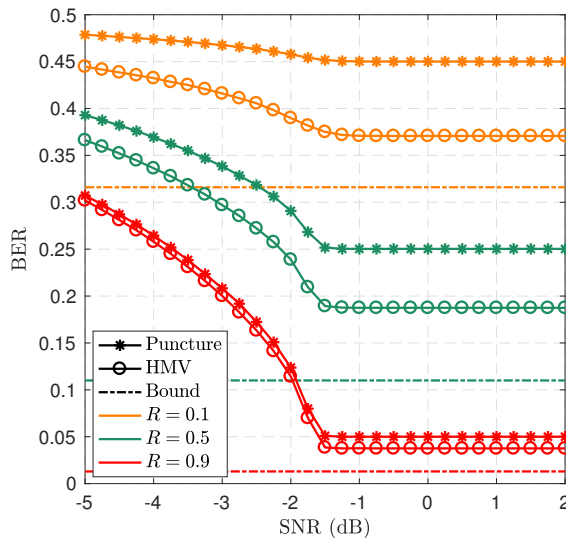


Fig. 6. The performance of joint source-channel coding based on HMV code.

With the basic setting of parameters listed in Table I, we obtain the simulation results depicted in Fig. 6. It is clear that the joint source-channel coding scheme based on the HMV code has lower bit error rate (BER) than puncturing, with the

same signal-to-noise ratio (SNR) value. Furthermore, the SNR floor for the HMV code is also closer to the theoretical bound derived from the rate-distortion function. This observation confirms that a coding scheme can get better performance by replacing the puncturing component with the HMV code.

## V. CONCLUSION

This letter has developed a lossy compression scheme, i.e., the HMV code, with relatively high efficiency and low complexity for practical use. Based on a special and simplest case of the joint typicality coding scheme, we have found the basic MV code and analyze its rate-distortion performance. Then, we exploited the two MV component codes to construct the HMV code, for the purpose of adapting arbitrary compression rate. The encoding and hard/soft decoding algorithms for the HMV code were also presented in detail. Finally, we conducted a series of simulations to compare the performance difference between the HMV code and puncturing. The simulation results verify that the HMV code can achieve better performance than puncturing. To find some simple component code in rate  $\frac{u}{s}$ , i.e., compress an  $s$ -bits sequence group into a  $u$ -bits codeword with better performance and then construct a new type of hybrid code satisfying lower distortion are left as future study.

## REFERENCES

- [1] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, Jul. 1948.
- [2] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098–1101, Sep. 1952.
- [3] J. Ziv and A. Lempel, "A universal algorithm for sequential data compression," *IEEE Transactions on Information Theory*, vol. 23, no. 3, pp. 337–343, May 1977.
- [4] —, "Compression of individual sequences via variable-rate coding," *IEEE Transactions on Information Theory*, vol. 24, no. 5, pp. 530–536, Sep. 1978.
- [5] ISO/IEC International Standard 11172-3, "Coding of moving pictures and associated audio for digital storage media at up to about 15 Mbit/s - Part3: Audio," 1993.
- [6] ISO/IEC International Standard 10918-1, "Information technology – Digital compression and coding of continuous-tone still images: Requirements and guidelines," 1992.
- [7] ISO/IEC International Standard 14496-10, "Information technology – Coding of audio-visual objects – Part 10: Advanced video coding," 2003.
- [8] C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," *IRE Nat. Conv. Rec.*, vol. 4, no. 142-163, p. 1, Mar. 1959.
- [9] A. El Gamal and Y.-H. Kim, *Network information theory*. Cambridge University Press, 2011.
- [10] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Transactions on information Theory*, vol. 22, no. 1, pp. 1–10, Jan. 1976.
- [11] T. Berger, "Multiterminal source coding," in *The Information Theory Approach to Communications*, G. Longo, Ed. New York: Springer-Verlag, 1978, pp. 171–231.
- [12] S. Y. Tung, "Multiterminal source coding," Ph.D. dissertation, Cornell University, Ithaca, New York, 1978.
- [13] J. Hagenauer, "The EXIT chart – introduction to extrinsic information transfer in iterative processing," in *12th European Signal Processing Conference*. Vienna, Austria: IEEE, Sep. 2004, pp. 1541–1548.
- [14] J. Garcia-Frias and Y. Zhao, "Near-Shannon/Slepian-Wolf performance for unknown correlated sources over AWGN channels," *IEEE Transactions on Communications*, vol. 53, no. 4, pp. 555–559, Apr. 2005.
- [15] K. Anwar and T. Matsumoto, "Accumulator-assisted distributed turbo codes for relay systems exploiting source-relay correlation," *IEEE Communications Letters*, vol. 16, no. 7, pp. 1114–1117, Jul. 2012.