INDEXING LATTICE VECTORS IN A JOINT WATERMARKING AND COMPRESSION SCHEME

Ludovic Guillemot, Jean-Marie Moureaux

CRAN – CNRS UMR 7039
Université Henri Poincaré, Nancy 1, BP 239,
F-54506 Vandœuvre-les-Nancy Cedex
Tel.: +33 3 83 68 44 74, Fax: +33 3 83 68 44 61
e-mail: {ludovic.guillemot,jean-marie.moureaux}@cran.uhp-nancy.fr

ABSTRACT

The problem of the design of a joint watermarking and compression (JWC) scheme allowing an efficient variable rate coding is addressed here. We have proposed in previous works a method, called *modulated lattice vector quantization* (MLVQ), based on dither modulation and lattice vector quantization (LVQ) and have shown experimentally its good performances. In this paper, we first show theoretically that the use of a multidimensional lattice codebook must be privileged in JWC. To take benefit from this property, an indexing method dedicated to the MLVQ codebook is proposed. It is based on the use of the geometrical properties of the MLVQ codebook.

1. INTRODUCTION

The robustness requirements of a data hiding scheme depend on the applications. However, among all the potential attacks, lossy compression is one of the most destructive and is common to the main part of data hiding applications. Furthermore, compression is most of the time *unavoidable* since most of these marked data have to be transmitted and / or stored. A compressed signal could be advantageously seen like a state of the watermarking channel known at the coder. This point of view leads to methods where compression and watermarking are processed jointly (JWC). Thus, compression is no more an attack. A practical realization of JWC has several requirements corresponding to the properties of the most popular compression schemes: first, the method must be compatible with entropy coding and, second, it must be efficient in a transform domain.

In [4], we have proposed a scheme[1], called modulated lattice vector quantization (MLVQ), based on dither modulation [2] for embedding and lattice vector quantization [1] (LVQ) for coding which takes into account these requirements. MLVQ was applied with success in terms of rate distortion performances in the field of image compression using discrete wavelet transform.

The contribution of this paper is double. Firstly, we study the asymptotic performances of MLVQ and prove the interest of using a multidimensional codebook in JWC schemes to reach a good rate distortion trade-off. Secondly, we take benefit from the low coding rate bound pointed out in the theoretical study, by designing an efficient indexing method for MLVQ codebook vectors. Experimental results are given to show the efficiency of the whole designed system.

[1]Note that a JWC scheme based on fixed length scalar quantization has been proposed in [8].

2. CODING RATE LIMIT OF MODULATED LATTICE VECTOR QUANTIZATION

2.1. Modulated lattice vector quantization

In modulated lattice vector quantization, information embedding and quantization are jointly performed using m dithered uniform quantizers [2]:

$$Q_i(X) \triangleq mQ\left(\frac{X - i\frac{\gamma}{m}}{\gamma}\right) + i \qquad (1)$$

where Q is the uniform quantizer in $\mathbb{Z}^n$, $i \in \{0, 1, ..., m-1\}$ and γ the scaling factor.

The corresponding reconstruction point is given by:

$$\tilde{X} = \frac{\gamma}{m} Q_i(X) \qquad (2)$$

Let f be the embedding function of the m-ary message $M = (M_1, ..., M_L)$ of length $L = \frac{N}{n}$ (N corresponds to the size of the source). Watermarking and quantization are jointly performed in MLVQ using f as follows:

$$f_{X^j}(M_j) = \frac{\gamma}{m} Q_{M_j}\left(X^j\right) \qquad (3)$$

with X^j the j^{th} source vector .

The MLVQ codebook corresponding to (1) is called modulated lattice $\mathbb{Z}_m^n$. It is equal to the union of the m cosets S_i of $\mathbb{Z}^n$:

$$\mathbb{Z}_m^n \triangleq \bigcup_{i=0}^{m-1} S_i \tag{4}$$

The codebook has a pyramidal shape (see figure 1), as it yields a good trade-off [1] [3] between compression performances and arithmetic complexity. We denote S_i the coset of vectors quantized by Q_i:

$$S_i \triangleq \{m\mathbb{Z}^n + [i]\} \tag{5}$$

with $[i] = \begin{pmatrix} i \\ ... \\ i \end{pmatrix}$.

In this paper, we only consider the case of a binary message. Consequently, S_0 and S_1 correspond, by (5), to the set of vectors with even components and odd components, respectively. Because of the regular structure of lattices, each vector is located on a surface of constant radius with respect to the origin (*i.e.* with constant norm). This particularity enables to assign a product code $I(Y)$ to any lattice vector Y, the prefix r corresponding to the norm and the suffix p to its position on the corresponding surface: $I(Y) = (r, p), (r, p) \in \mathbb{N}^2$. r will be entropy coded whereas p will be fixed length coded.

The total bit rate R_{total} of MLVQ is then:

$$R_{tot} = R_{norm} + R_{position} = \\ -\sum_{r=0}^{r_T} P(r) \left[\log_2(P(r)) - \log_2(N(r))\right] \text{ bits/vector} \tag{6}$$

where r_T is the truncation radius, $P(r)$ the discrete probability of the radius r and $N(r)$ the number of vectors belonging to the shell of radius r. Note that a vector dead zone

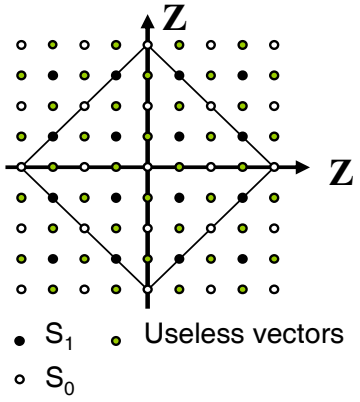

Fig. 1: Example of modulated lattice : $\mathbb{Z}_2^2$

is added to the modulated codebook [4] to exploit sparsity of data.

vector size	1	2	4	8	16
$\mathcal{H}_n^\infty$ (bits/sample)	1.5	1.25	0.94	0.84	0.83

Table 1: Lower bound of the entropy of a MLVQ quantized source for several size of vector.

2.2. Asymptotic performances of JWC based on dither modulation

In classical source coding, the entropy of the quantized source tends to 0 when $\gamma \to \infty$ (*i.e.* all the source vectors are quantized by zero). This is no more the case in a JWC scheme. Indeed, when the source is infinitely scaled, only the shells of radius 0 and n of the codebook are used (corresponding to the insertion of bits 0 and 1 respectively). Consequently, using (6) we have:

$R_{total} \underset{\gamma \longrightarrow \infty}{\longrightarrow} \mathcal{H}_n^\infty =$
$-\left[P_0 log_2(P_0) + P_n log_2(P_n)\right] + P_0 log_2(N(0)) + P_n log_2(N(n))$
with $P_0 = P(\|Y\| = 0)$ and $P_n = P(\|Y\| = n)$.

The message is supposed to be equiprobable: $P_0 = P_n = \frac{1}{2}$. Finally, as $N(0) = 1$, we have:

$$\mathcal{H}_n^\infty = \frac{1}{n}\left[1 + \frac{\log_2(N(n))}{2}\right] \text{ bits/sample} \tag{7}$$

Formula (7) shows that whatever the distribution of the MLVQ quantized source is, its entropy cannot be lower than $\mathcal{H}_n^\infty$. This value only depends on the dimension n of the codebook. Figure 2 illustrates the effect of the coding limit on the rate distortion (R-D) function of the scalar version of MLVQ,*i.e.* two scalar dither uniform quantizers and entropy coding. The function admits an asymptote in 1.5 bits/sample (corresponding to the coding limit (7) for $n = 1$) which leads to poor coding performances. Table 1 shows that this limit decreases significantly when the size of vector increases. For example, $\mathcal{H}^\infty$ is equal to 0.8 bits/sample for vectors of size 8, that is to say a decreasing of 44 percents of the coding limit. This point suggests that the use of a multidimensional codebook is more suitable for joint watermarking and compression.

However, to exploit this property, the indexing process must take into account vectors of the modulated lattice only. For example, 87.5 percents of vectors of $\mathbb{Z}^4$ do not belong to $\mathbb{Z}_2^4$ (see [4] for details). Consequently, the use in MLVQ of an indexing method dedicated to $\mathbb{Z}^n$ would increase dramatically the length of the suffix codewords.

3. INDEXING IN THE MODULATED LATTICE

In lattice coding, the suffix indexing remains a difficult problem for which some methods [3][5][6] [7] have been proposed for different codebook shapes and lattices. In the case of the modulated lattice, our method consists in enumerating separately odd and even vectors within a given shell.

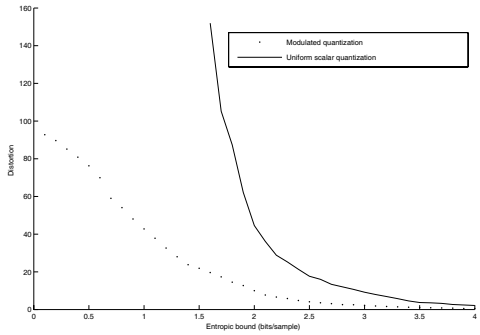

Fig. 2: Rate distortion function of both uniform scalar quantizer and modulated scalar quantizer for a gaussian source (zero mean and standard deviation of 10).

3.1. Principles

Let us give first some definitions.

$\mathcal{I}^{mod}(Y)$ is the index of a vector $Y \in \mathbb{Z}_2^n$. If $Y \in S_i$, we denote $\mathcal{I}^i(.)$ its index within S_i, with $i = 0, 1$.

$C^i(r)$ with $i \in \{0, 1\}$ corresponds to the set of vectors of the shell of radius r belonging to S_i. It is defined by: $C^i(r) = \{Y \in S_i / \|Y\| = r\}$. $C(r)$ corresponds to the shell of radius r in $\mathbb{Z}^n$.

The index $\mathcal{I}^{mod}(Y)$ is deduced from $\mathcal{I}^0$ and $\mathcal{I}^1$ using the following relation:

$$\mathcal{I}^{mod}(Y) = \begin{cases} \mathcal{I}^0(Y), \text{ if } Y \in C^0(r) \\ card\left(C^0(r)\right) + \mathcal{I}^1(Y), \text{ if } Y \in C^1(r) \end{cases} \quad (8)$$

where $card(.)$ stands for cardinality .

The index $\mathcal{I}^0(.)$ and $\mathcal{I}^1(.)$ in formula (8) are computed differently. One can show that a map exists between S_0 and $\mathbb{Z}^n$. It yields the definition of a bijection g between $C^0(r)$ and $C\left(\frac{r}{2}\right)$: $g(Y) = \frac{Y}{2}$. Thus, $\mathcal{I}^0(.)$ can be deduced from the index $\mathcal{I}(.)$ in $\mathbb{Z}^n$ (see [3][5][6][7]) by:

$$\mathcal{I}^0(Y) = \mathcal{I} \circ g(Y) \quad (9)$$

Unfortunately, formula (9) cannot be used to evaluate $\mathcal{I}^1(.)$ because no such map exists between S_1 and $\mathbb{Z}^n$. However, the computing of $\mathcal{I}^1(.)$ can be derived from the method described in [6].

3.2. Enumerating odd vectors

Each set $C^1(r)$ can be divided into vector subsets, called *orbits*. The orbit $\mathcal{O}(Y)$ of a vector Y is the following set :

$$\mathcal{O}(Y) = \{X \in S_1 / \exists p / X = p(Y)\} \quad (10)$$

with *p* a signed permutation.

Each orbit (10) admits a unique vector $l = (l_1, ...l_n)$, called *leader*, such that $0 \leq l_1 \leq ... \leq l_n$.

The key point of the method is the following property: any vector of an orbit is the image by a signed permutation of its leader. Figure 3 represents this property in the case of $\mathbb{Z}_2^2$.

Given a vector Y, its index $\mathcal{I}^1(Y)$ is deduced from the index of its leader $l(Y)$ and its index $\mathcal{I}^1_{\mathcal{O}(Y)}$ within $\mathcal{O}(Y)$:

$$\mathcal{I}^1(Y) = \mathcal{I}^1(l(Y)) + \mathcal{I}^1_{\mathcal{O}(Y)}(Y) \quad (11)$$

where $\mathcal{I}^1_{\mathcal{O}(Y)}(Y)$ is computed using the following rule:

$$\mathcal{I}^1_{\mathcal{O}(Y)}(Y) = N(Y) \times 2^n + S(Y) \quad (12)$$

with N the number (in the lexicographical order) of permutations *p* such that $p(Y) = l(Y)$ and S the number of symmetries. The index of leaders are computed off-line and

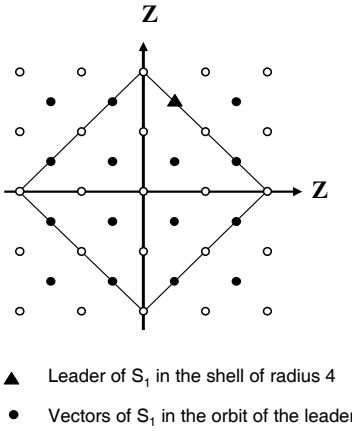

Fig. 3: Leaders and corresponding orbits on the shell of radius 4.

stored in a look-up table (the coding table) as detailed in the following. First, given the leader l^k, compute the next leader l^{k+1} in the lexicographical order. The index $\mathcal{I}^1\left(l^{k+1}\right)$ is determined by:

$$\mathcal{I}^1\left(l^{k+1}\right) = \mathcal{I}^1\left(l^k\right) + card\left(\mathcal{O}\left(l^k\right)\right)$$

Each vector of $\mathcal{O}\left(l^k\right)$ is a signed permutation of l^k, consequently we have:

$$card\left(\mathcal{O}\left(l^k\right)\right) = 2^n \frac{n!}{w_1!...w_k!} \quad (13)$$

where w_i is the number of times the ith value occurs in the vector. Finally, the index of leaders are stored in the coding table using the bijection $b(l_1) = \frac{l_1 - [1]}{2}$ between $L^1(r)$, the set of leaders of $C^1(r)$, and $L(r)$ the set of leaders of $C(r)$.

Note that formulas (12) and (13) differ from those in [6] because there is no null component in odd vectors. The goal of the use of the function b is to minimize the size of the coding table, *i.e.* the storage cost. Indeed, the norm of the image is more than twice lower than the norm of the leader itself: $||l|| = \left|\left|\frac{l_1-[1]}{2}\right|\right| = \frac{||l_1||-n}{2}$.

3.3. Experimental results

Figure 4 represents the codeword length of the suffix for vectors of size 8. The modulated indexing permits to decrease significantly the codeword length compared to indexing in lattice $\mathbb{Z}^n$ (about 2 bits and 6 bits for vectors of size 4 and 8, respectively). The total complexity of the modulated indexing does not increase significantly compared to classical indexing methods in $\mathbb{Z}^n$. Indeed, on the one hand, even vectors are indexed using a $\mathbb{Z}^n$ method, and on the other hand, the indexing of odd vectors is low cost and well suited to high parallelism. Figure 5 permits to compare the performances of two JWC schemes based on wavelet coding of the Lena image: (A) two dither uniform quantizers without modulated indexing; (B) MLVQ. As we can see, the first image is dramatically degraded by the coarse quantization whereas the second has a good quality (PSNR equals to 33.1 dB).

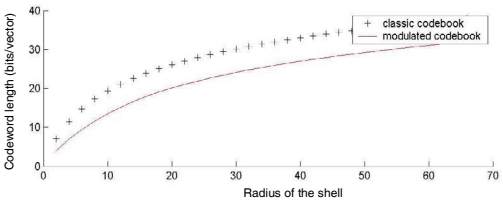

Fig. 4: Suffix codeword length for classical indexing in $\mathbb{Z}^n$ and modulated indexing (vectors of size 8).

Fig. 5: Image Lena coded at 0.36 bits/pixel using (A) DM-QIM: PSNR=16.8 dB; 8.6 kbits embedded. (B) MLVQ: PSNR=33.1 dB; 3.8 kbits embedded.

4. CONCLUSION

In this paper we have first given the asymptotic coding performances of joint watermarking and compression coders based on dither modulation and entropy coding. In particular we have shown that the existence of a coding limit justifies the choice of a multidimensional codebook in joint watermarking and compression.

Then, we have described our proposed indexing method permitting to take benefit from the good properties of the multidimensional codebooks in terms of rate distortion performances. By dividing the codebook into appropriate subsets, our method permits to decrease significantly the length of the suffix codewords and, at the same time, leads to the same complexity as existing methods on lattice $\mathbb{Z}^n$.

We have shown that MLVQ remains an efficient coder in terms of rate distortion trade-off despite of the large amount of information embedded. Future works will concern the robustness of the scheme against other attacks than compression.

5. REFERENCES

[1] M. Barlaud, P.Solé, T. Gaidon, M. Antonini, and P. Mathieu, "Pyramidal lattice vector quantization for multiscale image coding," IEEE Trans. Image Processing, vol. 3, pp. 367-381, July 1994.

[2] B.Chen and G. Wornell, "Quantization Index Modulation: A Class of Provably Good Methods for Digital Watermarking and Information Embedding", IEEE Transaction on Information Theory, vol. 47, NO. 4, May 2001.

[3] T.R. Fischer, "A pyramid vector quantizer", IEEE Transactions Information Theory, vol. 32, NO. 4 pp. 568-583, July 1986.

[4] L. Guillemot and J.M. Moureaux, "Hybrid transmission, compression and data hiding: quantisation index modulation as source coding strategy", Electronics letters, Vol. 40, No. 17, pp. 1053-1055, august 2004.

[5] P. Loyer, J.-M. Moureaux and M. Antonini, "Lattice codebook enumeration for generalized gaussian source", in IEEE Transactions on Information Theory, VO. 49, NO. 2, February 2003.

[6] J.M. Moureaux, P. Loyer and M. Antonini: Low-Complexity Indexing Method for $\mathbb{Z}^n$ and $\mathbb{D}^n$ lattice quantizers, IEEE Transactions on Communications, vol. 46, 12, pp. 1602-1609, december 1998.

[7] P. Rault and C. Guillemot, "Lattice vector quantization with reduced or without look-up table", in Proc. SPIE Electronic Imaging, Santa Clara, Fl, january 1998.

[8] G. Wu, E.H. Yang, "Joint watermarking and compression using scalar quantization for maximizing robustness in the presence of additive gaussian attacks", IEEE transactions on Signal Processing, vol. 53, N0. 2, pp. 834-844, February 2005.