SUCCESSIVE APPROXIMATION TRELLIS-CODED VECTOR QUANTIZATION FOR EMBEDDED CODERS

Alessandro J. S. Dutra

Depto. de Eng. Elétrica
Instituto Militar de Engenharia
Praça General Tibúrcio, 80
Rio de Janeiro - RJ
22290-270, BRAZIL
a.dutra@ieee.org

Eduardo A. B. da Silva *Gelson V. Mendonça*

PEE/COPPE/DEL/EE
Universidade Federal do Rio de Janeiro
Cx. P. 68504
Rio de Janeiro, RJ
21945-970, BRAZIL
eduardo@lps.ufrj.br gelson@lps.ufrj.br

ABSTRACT

Embedded bit-plane-based and Trellis Coded Quantization-based (TCQ) methods are among the state-of-the-art for wavelet image coding. In the JPEG 2000 Verification Model, these two methods have been successfully combined in an Embedded TCQ Wavelet Coder. In it, TCQ is first applied to the coefficients and then its indexes are bit-plane encoded. In this paper, we propose a different philosophy of combining the aforementioned methods. Instead of encoding the TCQ indexes by bit-planes, we would apply TCQ to each bit-plane. However, since in scalar successive approximation processes, at every pass just a symbol from a two-valued source is transmitted (e.g., $\{+\delta, -\delta\}$), TCQ is not suitable for such scenario. We have circumvented this problem by using *vector* bit-planes as defined in [1]. Each vector bit-plane can then be encoded using Trellis Coded Vector Quantization (TCVQ). Lattice-based codebooks were investigated and modified in order to yield efficient TCVQ schemes. Simulations have shown that the proposed method outperforms the Embedded Wavelet Vector Bit-plane encoder (SA-W-VQ), from [1]. This indicates that such approach has the potential of increasing the coding performance of bit-plane-based methods.

1. INTRODUCTION

Scalar bit-plane encoders [2, 3] are notable for the good performance they achieve in a low encoding complexity framework. Moreover, they can provide us with an embedded source code, a feature which is desirable in many applications. Although these methods are highly successful, it is natural to consider the possibility of using Vector Quantization [4] as a means of obtaining even better performance results. This is exactly what has been done by da Silva and Craizer in [1], where it was shown that vector bit-plane encoding is possible provided that some conditions are met.

Embedded methods employing TCQ are among the state-of-the-art and the one presented by Sementelli *et al.* [5] is part of the *Verification Model* (VM) for the JPEG-2000 standard. This method applies TCQ to the coefficients of the wavelet transform of an image and then bit-plane encodes its indexes. As a result, the successive encoding of bit-planes of TCQ indexes, from MSB to LSB, provides the successive approximation of the wavelet coefficients.

The most general framework, however, is the one presented by Brunk and Farvardin [6], in which different stages in the refining process employ different trellises (Fig. 1(b)). This happens because every source description in the k^{th} stage must include the source description in the $(k-1)^{th}$ stage. This is accomplished by substituting each branch of the actual trellis either by the base trellis (the one employed at the first stage) or by any other suitable trellis, recursively.

A novel way of encoding vector bit-plane significance data by using Trellis-Coded Vector Quantization (TCVQ) is presented here. In our approach, for each vector bit-plane to be encoded, a trellis path and a sequence of reconstruction symbols are generated. This allows us to minimize each bit-plane distortion with respect to the input sequence *as a whole*, instead of individually minimizing each sample distortion.

This paper is organized as follows: section 2 provides a review on vector bit-plane encoding, and so does section 3 on TCQ. Section 4 briefly reviews embedded TCQ-based existing methods. Section 5 outlines the proposed embedded trellis coded vector quantization

0-7803-5467-2/99/ $ 10.00 © 1999 IEEE

scheme, as well as the determination of "good" codebooks for it. Simulation results are presented in section 6. In section 7, our conclusions and suggestions for future improvements are presented.

2. BIT-PLANE ENCODING

A scalar quantity $-1 \leq c \leq 1$ can be described using a bit-plane decomposition by the sequence $\{b_1, \dots, b_n, \dots\}$, which is equivalent to:

$$c = \sum_{i=1}^{\infty} b_i 2^{-i} \tag{1}$$

where $b_i \in \{-1, 1\}$.

Da Silva and Craizer [1] showed that a generalization of vector bit-planes from scalar bit-planes is possible and that some improvements are obtained by employing vector codebooks in successive approximation coding algorithms. More precisely, a vector $\mathbf{v}$ can be represented as:

$$\mathbf{v} = \sum_{i=1}^{\infty} \mathbf{u}_{n_i} \alpha^i \tag{2}$$

where $\mathbf{u}_{n_i} \in C_N$, is an orientation codebook composed by unitary vectors on an hyper-sphere and $0 < \alpha < 1$.

Assuming worst case convergence conditions we have that if an orientation codebook $C_N = \{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_M\}$ such that $||\mathbf{u}_i|| = 1$, $\forall i$ is given, then there exists a representation as in eq. (2) such that convergence is achieved for all $\mathbf{v} \in \mathbb{R}^N$, $||\mathbf{v}|| \leq 1$, if:

$$\frac{1}{2\cos[\Theta(C_N)]} \leq \alpha < 1, \quad \Theta(C_N) \leq 45^\circ \tag{3}$$

$$\sin[\Theta(C_N)] \leq \alpha < 1, \quad \Theta(C_N) \geq 45^\circ \tag{4}$$

where $\Theta(C_N)$ is the maximum possible angle between any vector $\in \mathbb{R}^N$ and its nearest neighbor $\in C_N$. More precisely,

$$\Theta(C_N) = \cos^{-1}\{\min_{\mathbf{x} \in \mathbb{R}^N}\{\max_{\mathbf{u}_i \in C_N}\{\frac{\mathbf{x} \cdot \mathbf{u}_i}{||\mathbf{x}||||\mathbf{u}_i||}\}\}\} \tag{5}$$

A decomposition as in eq. (2) can be found by a greedy procedure: at each pass, the residual error is computed and $\mathbf{u}_{n_i}$ is the vector in C_N closest to it.

Hence, good codebook choices for vector bit-plane encoding should jointly have the following characteristics: a) its size should be as small as possible; and b) $\Theta(C_N)$ should also be as small as possible. This implies that codebooks should be "uniformly" distributed over an N-dimensional hyper-sphere.

3. TRELLIS-CODED QUANTIZATION

Trellis coded quantization [7] is a coding scheme, based on Ungerboeck's Trellis-Coded Modulation [8], which provides good rate-distortion performance at a low implementation complexity.

A TCQ with rate R bits per source sample is implemented by using a dictionary of 2^{R+1} symbols divided into 4 subsets, according to Ungerboeck's mapping by set partitioning rules. A typical trellis design is presented in Fig. 1(a).

Each subset is assigned to a group of trellis branches. The set of possible state transitions is defined by a convolutional code and the Viterbi algorithm is used to determine the best source encoding path within the trellis.

4. EMBEDDED TCQ-BASED ENCODING

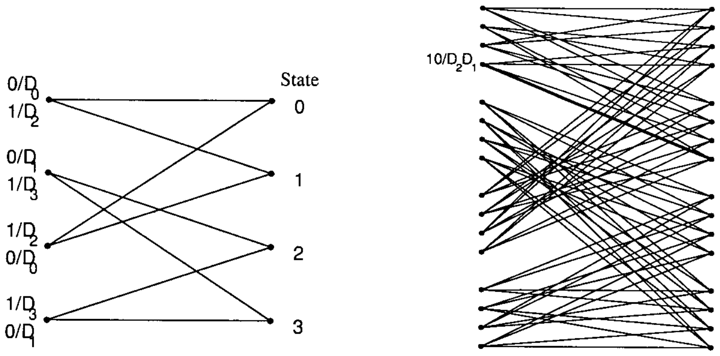
In this section, we highlight the main features of two of the embedded TCQ-based encoders found in the literature.

We begin with the Embedded Trellis-Coded Quantization [6] method, proposed by Brunk and Farvardin. Fig. 1 shows us two different trellises: the conventional one, also called *base trellis* and the trellis that results when two stages of the E-TCQ method are employed.

In the latter, the assignment of input bits and selected subsets is carried out following the same rules for the conventional trellis encoding. For each stage, however, the information regarding the current transition is appended to the end of the existing symbol, which already describes all previous transitions. So, in Fig. 1(b), the best path included the transition between states 0 and 1 for the first stage, and the transition between states 3 and 3 for the second stage. Considering that transitions may be represented by the pair (input bit, selected subset), it means that the first stage corresponds to the (1,D_2) pair while the second one is represented by (0,D_1). The two-stage transition would then be represented by (10, D_2D_1).

There are two ways of determining the best path through the trellis in the E-TCQ method:

1. The distortion minimization may be performed globally, i.e., a n-stage trellis is defined and then the path that provides the lowest distortion between the input and output sequences is determined. Proceeding in this way, all the paths from the lower-stage trellises are implicitly defined. The major disadvantage of this approach is that the number of trellis states grows very



(a) Conventional Trellis

(b) Embedded Trellis

Figure 1: A 4-state Ungerboeck trellis and a two-stage embedded trellis derived from the former.

fast, resulting in highly complex search for the best path;

2. The minimization may be carried out once at every stage, i.e., the algorithm searches for the best path for each stage trellis and uses this path as a basic path for the next stage search, being *progressively* embedded. Although this method results in a lower complexity search problem, the final minimization is not optimal.

Another characteristic of this method, when performing the global minimization, i.e., when searching for the best available path, is that the whole input sequence must be encoded prior to its transmission, which may result in long delays.

The JPEG-2000 VM [5] applies TCQ to the wavelet transform coefficients of an image. Then, the indexes are transmitted on a bit-plane basis, and the sign information is appended to the first non-zero bit of the coefficient. As both the encoder and the decoder have the mapping from indexes to reconstruction values, this method corresponds to approximating the coefficients from the indexes.

The number of necessary bit-planes is determined by the maximum magnitude index, $i_{\max}$, and is given by $n = \log_2 i_{\max}$. Initially, all the bits that define the path to be followed within the trellis are transmitted, what tells the decoder exactly the sequence of subsets that will be used for decoding. As this method employs scalar quantization, the indexes are directly associated to the reconstruction values. Hence, the successive approximation of indexes corresponds to approximating the coefficients, and it is carried out by transmitting each bit of the index, starting from its most significant and going through up to the least significant.

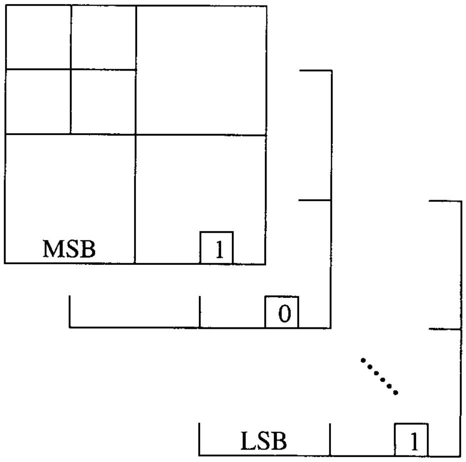


Figure 2: Coefficient bit-planes - JPEG-2000 Verification Model

Fig. 2 shows a representation of the indexes bit-planes to be transmitted. Each plane is scanned such that a higher resolution band transmission begins only after all lower resolution ones are finished. The coefficient sign is transmitted right after its first non-zero bit, so as to avoid unnecessary information transmission.

5. A NEW TCQ-BASED ENCODING PHILOSOPHY

A major disadvantage shared by both of the aforementioned methods is the fact that they need to completely encode the source data before they become able to transmit it. In order to overcome this drawback, a new way of TCQ-encoding is proposed.

5.1. The Encoding Scheme

In order to improve the rate-distortion performance of embedded vector bit-plane coders one has to consider whether the codevectors indexes are being efficiently encoded. Our approach to this problem is the following:

1. at every step of the successive approximation process a list of vectors to be encoded is generated;
2. TCVQ is applied to the list yielding a source description;
3. the source description is arithmetically coded to produce the bit-stream.

The codebooks used in our experiments are based on the lattices that provide the best sphere packings in their dimensions [9]. However, some modifications were necessary in order to make them suitable for use in TCVQ. That included obtaining the doubled alphabet by finding a conveniently rotated version of the original shell, and then separating the two codebooks in what would be considered the four TCQ subsets. The separation algorithm used was the one presented in [10].

5.2. Codebook Determination

Two codebooks have been investigated: a doubled-D_4 shell and a doubled-E_8 shell. Some interesting characteristics of the mentioned lattice shells are exploited.

The properly scaled D_4 shell-2 vectors are located exactly at those places in the 4-dimensional hyper-sphere that are mostly separated of the shell-1 points, which is used as the codebook basic block. It means that the angle between any D_4 shell-2 vector and its closest neighbor in D_4 shell-1 is exactly $\Theta(D_4$ shell-1), see eq. (5).

A similar behavior is exhibited by the E_8-based codebook. However, in this case there are nine times more vectors in shell-2 than in shell-1, and all of them are placed at an angle of $\Theta(E_8$ shell-1) with each closest vector of shell-1. Hence, to obtain a doubled-E_8 shell codebook, the E_8 shell-2 was divided into its nine cosets. One of these cosets, properly scaled, was used as the codebook completion.

Simulations were performed using a modified version of the SA-W-VQ coder [1]. A zero codeword was added to each of the subsets due to the algorithm's refinement pass requirement.

6. EXPERIMENTAL RESULTS

Simulations show that, for the $\alpha \in [0.5, 1)$ range, the proposed method outperforms the SA-W-VQ algorithm using the corresponding codebooks, as can be seen from Fig. 3, where results for the Lena 256×256 image are shown for a bit-rate of 0.5 bpp, considering the $E8$-based codebook. The largest PSNR value of 32.31 dB was obtained for a 64-state TCQ and $\alpha = 0.6$.

Similar results are also obtained when using the D_4-based codebook, as can be seen from Fig. 4. In this case, a PSNR of 32.23 dB was obtained, employing a 4-state TCQ and $\alpha = 0.59$, which was found as the optimum encoding alpha.

It is interesting to notice that these optimum values of α are the same ones as for the original SA-W-VQ algorithm for both $D4$ and $E8$-based codebooks.

In Table 1, the SA-W-TCVQ results are compared to the ones obtained by the SA-W-VQ algorithm for the Lena 256×256 image, for a bit-rate of 0.5 bpp ($\alpha_{optimum}$ was used for both codebooks).

Method	Codebook	
	D4	E8
SA-W-VQ	31.89 dB	32.14 dB
4-state SA-W-TCVQ	32.23 dB	32.29 dB
128-state SA-W-TCVQ	32.28 dB	32.31 dB

Table 1: Comparison between the performance of the SA-W-VQ and SA-W-TCVQ. $\alpha = \alpha_{optimum}$, bit-rate=0.5 bpp

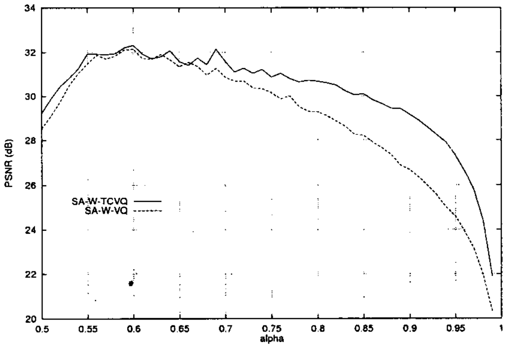


Figure 3: Comparison between the $\alpha \times$PSNR performance of the SA-W-TCVQ and SA-W-VQ algorithms for the Lena 256×256 image, E-8 codebook, for a bit-rate=0.5 bpp

In Fig. 5, a comparison between the rate-distortion performance of SA-W-VQ and the new method is shown. Once again, it is clear that the use of TCVQ provides

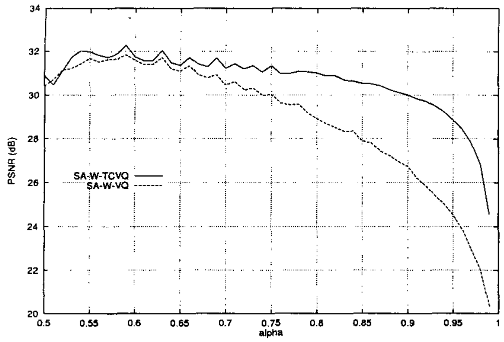


Figure 4: Comparison between the $\alpha \times$PSNR performance of the SA-W-TCVQ and SA-W-VQ algorithms for the Lena 256×256 image, D-4 codebook, for a bit-rate=0.5 bpp

the vector bit-plane method with some improvement in its performance.

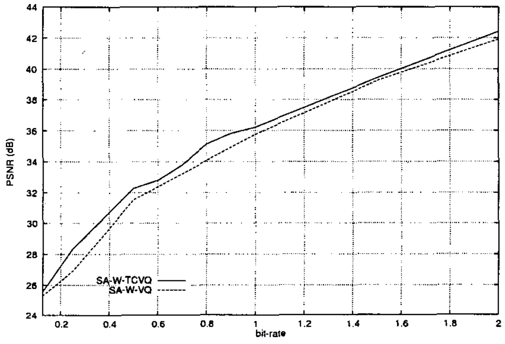


Figure 5: Rate-distortion performance for the Lena 256×256 image for the SA-W-TCVQ and SA-W-VQ algorithms with $D4$-based codebooks

7. CONCLUSIONS

We presented a novel method for embedded trellis encoding. In this method, which we call Successive Approximation Trellis Coded Vector Quantization (SA-W-TCVQ), vector bit-planes are trellis coded quantized generating an embedded bit-stream. We have also investigated methods for obtaining good TCVQ subsets from lattices related to the solution of the Sphere Packing problem. It is important to note that the use of subsets based on lattice codebooks simplifies a great deal the TCVQ design. Simulations show that the proposed approach can be used to provide increased performance in existing bit-plane-based wavelet coders. As suggestions for further improvements, we could include the use of higher dimensional codebooks, having as basis the Barnes-Wall Λ_{16} and the Leech Λ_{24} lattices, as well as the use of more general uniform codebooks, based on training sequences.

8. REFERENCES

[1] E. A. B. da Silva and M. Craizer, "Generalized bit-planes for embedded codes," in *1998 IEEE International Conference on Image Processing*, (Chicago, IL), October 1998.

[2] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 41, pp. 3445–3462, December 1993.

[3] A. Said and W. A. Pearlman, "A new, fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, pp. 243–250, June 1996.

[4] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression.* New York: Kluwer Academic Publishers, 1991.

[5] P. J. Sementelli, A. Bilgin, J. H. Kasner, and M. W. Marcellin, "Wavelet TCQ: Submission to JPEG 2000 (invited paper)," in *Proc. of Applications of Digital Image Processing, SPIE*, (San Diego, CA), July 1998.

[6] H. Brunk and N. Farvardin, "Embedded trellis coded quantization," *IEEE Transactions on Communications.* Submitted.

[7] M. W. Marcellin and T. Fischer, "Trellis coded quantization of memoryless and gauss-markov sources," *IEEE Transactions on Communications*, vol. 38, pp. 82–93, January 1990.

[8] G. Ungerboeck, "Trellis-coded modulation with redundant signal sets - parts I and II," *IEEE Communications Magazine*, vol. 25, pp. 5–21, February 1987.

[9] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups.* New York: Springer-Verlag, 1988.

[10] H. S. Wang and N. Moayeri, "Trellis coded vector quantization," *IEEE Transactions on Communications*, vol. 40, pp. 1273–1276, August 1992.