Trellis Vector Residual Quantization

Giovanni Motta  
Volcan Center for Complex Systems  
Brandeis University  
Waltham, MA 02254, USA  
gim@cs.brandeis.edu

Bruno Carpentieri  
Dip. di Informatica ed Applicazioni  
Universita di Salerno  
84081 Baronissi (SA), Italy  
bc@udsab.dia.unisa.it

ABSTRACT

Vector Quantizers (or VQs) have the property of encoding sources while achieving asymptotically the best theoretical performance. Unfortunately, conventional optimal VQs require exponentially growing computational and memory resources. Nevertheless, the quality achieved by the VQs is frequently desirable in applications where only a limited amount of resources is available.

In this paper, we present a sub-optimal vector quantizer that combines in an innovative way, trellis coding and residual quantization. Our Trellis Coded Vector Residual Quantizer (or TCVRQ) is a general-purpose sub-optimal VQ with low computational costs and small memory requirement. Despite its good performance, TCVRQ permits considerable memory savings when compared to traditional quantizers.

We propose new methods for computing quantization levels, and experimentally analyze the performances of our TCVRQ in the case of Linear Prediction speech coding and still image coding.

Our experiments confirm that our TCVRQ is a good compromise between memory/speed requirements and quality, and that it is not sensitive to codebook design errors.

INTRODUCTION

Vector Quantization (or in short VQ) is a source coding technique that, as Shannon proved in his "Source Coding Theorem", has the property of achieving asymptotically the best theoretical performance. Although Shannon's theorem guarantees the existence of vector quantizers that give nearly optimal performance, the theory provides no methods of determining such quantizers and, as was recently demonstrated by Lin[8], the design of an optimal VQ is an NP-complete problem.

In 1980 a vector quantizer code book design method (named LBG for the authors) was introduced by Linde, Buzo and Gray[2], this method designs locally optimal code books that have no natural order or structure. As a consequence of the lack of structure, the memory needed for the code book grows exponentially and the encoding of every source vector, requires an exhaustive search to locate a code word that minimizes a given distortion measure. In the following, we will refer to this kind of quantizer as Exhaustive Search Vector Quantizer.

Imposing a structure on the VQ has been suggested to hold down both the memory and computation costs[6]. A very interesting structure is the so called Cascade or Residual Quantization; combining this structure with a trellis graph preserves the memory saving and allows a "virtual increase" of the quantization levels.

DEFINITIONS

In this section we give a formal definition of the proposed system and we present an extension of the LBG algorithm to design the quantization levels.
Definition 1: Let $x$ be a random vector in $n$-dimensional Euclidean space $\mathbb{R}^n$; an $N$-level Exhaustive Search Vector Quantizer (or ESVQ) of $\mathbb{R}^n$ is a triple $Q = (A, Q, P)$ where:
1. $A = \{y_1, y_2, \ldots, y_N\}$ is a finite indexed subset of $\mathbb{R}^n$ called codebook, of code vectors $y_i$;
2. $P = \{S_1, S_2, \ldots, S_N\}$ is a partition of $\mathbb{R}^n$ where the equivalence classes (or cells) $S_j$ of $P$ satisfy:
   $$\bigcup_{j=1}^{N} S_j = \mathbb{R}^n,$$
   $$S_j \cap S_k = \emptyset \text{ for } j \neq k;$$
3. $Q : \mathbb{R}^n \mapsto A$ is a mapping that defines the relationship between the codebook and partitions as:
   $$Q(x) = y_j \text{ if and only if } x \in S_j.$$

Definition 2: A Residual Quantizer consists of a finite sequence of ESVQs $Q_1, Q_2, \ldots, Q_K$ such that $Q_i$ quantizes the input $x = x_i$ and each $Q_i$, $1 < i \leq K$ encodes the error (or residual) $x_i = x_{i-1} - Q(x_{i-1})$ of the previous quantizer $Q_{i-1}$, $1 < i \leq K$.

The output is obtained as the sum of all the code words:
$$y = \sum_{i=1}^{K} Q_i(x_i)$$

Definition 3: A multistage (or layered) graph is a pair $G = (V, E)$ with the following properties:
1. $V = \{v_1, v_2, \ldots, v_n\}$ is a finite set of vertices such that $V = \bigcup_{k=1}^{K} V_k$, $V_k \subset V$ for $1 \leq k \leq K$ and $V_i \cap V_j = \emptyset$ for $i \neq j, 1 \leq i, j \leq K$;
2. $E = \{(v_i, v_j) : v_i \in V_k, v_j \in V_{k+1}, 1 \leq k < K\}$ is a finite set of edges.

The number $K$ is the number of stages.

If a residual vector quantizer is associated to each node of a multistage graph and each layer of this graph is not "fully connected" to its successor (as in an Ungerboeck's trellis[4]), clearly a bit saving can be achieved. Each vector is fully specified by the path on the graph and by the indexes of the code words in the quantizers associated to the nodes along the path. Using, for example, the trellis showed in the Figure 1, if each $Q_i(j)$ is a $N$-level VQ, an output vector is specified giving $K + 1$ bits for the path and $K \cdot \log_2(N)$ bits for the code words. In the "equivalent" residual quantizer, each stage has $4N$ levels and $K \cdot \log_2(4N)$ bits are necessary, so the trellis configuration allows a "virtual doubling" of the available levels.

A formal definition of the TCVRQ is the following:

Definition 4: A Trellis Coded Vector Residual Quantizer is a pair $T = (G, Q)$ where:
1. $G = (V, E)$ is a Trellis[4] multistage graph with $|V| = n$ and $K$ stages;
2. $Q = (Q_1, Q_2, \ldots, Q_K)$ is a finite set of ESVQs, $|V| = |Q|$ and each $Q_i \in Q$ is associated to the vertex $v_i \in V$;
3. The ESVQ $Q_i$ encodes the residual of $Q_j$ if and only if $(v_i, v_j) \in E$.

The design of the quantization levels for the VQs associated to each node of the trellis is performed in sequence, from stage 1 to stage $K$, using the LBG algorithm on the residuals generated by the connected nodes.

This design is not optimal; nevertheless it respects the structure of the quantizer and, for a limited number of stages, achieves interesting performance.

The optimality conditions for the residual quantizers stated by Barnes and Frost in [6] and in [8], can be adapted to our TCVRQ, we do not use them here due to the excessive
complexity of the algorithm.

The best sequence of residual quantizers is determined using the Viterbi[1] algorithm.

In this particular framework, Viterbi search is not optimal in fact it behaves like a greedy algorithm. Nevertheless the performance is not degraded because of the decreasing error introduced by the residual structure.

**IMAGE CODING**

Several experiments were made to assess the performance of our TCVRQ on a natural source. A comparison was made between an ESVQ and a TCVRQ in quantizing 28 gray-levels images commonly used as a reference. These images can be found on internet at the ftp address: "ftp://links.uwaterloo.ca" in the directory "/pub/BragZone".

The training and the test sets were composed respectively by 12 and 16 images 512x512 pixels, 256 gray levels, divided in blocks (vectors) of 3x3 and 4x4 pixels. The measure used for the quantization error was the Signal to Quantization Noise Ratio or SQR.

The results are shown in the Figure 2. For low bit rates the error of TCVRQ is very close to the optimum; increasing the bit rate (and the number of stages) the performance of the TCVRQ decreases due to the sequentially optimum design.

Using the same training and test sets, we compared the performance of our TCVRQ to Jill Goldschneider's VQ package freely available on internet at the ftp address: "ftp://isdl.ee.washington.edu/pub/VQ/code". This package consists of two different kinds of tree quantizers (fixed and variable rate) and an ESVQ that uses the same code book of the tree quantizers.

As is clear from Figure 3, for low bit rates, our quantizer outperforms the package in terms of SQR. Due to their structure, the tree quantizers use two times the memory of the ESVQ that grows exponentially.
Our TCVRQ uses only an amount of memory that grows linearly with the vector dimension.

**LOW BIT RATE SPEECH CODING**

TCVRQ is a general-purpose VQ, with low computational costs and small memory requirements. This makes it very appealing for low bit-rate speech coding applications. A high quality low bit-rate codec is needed when speech signal must be encoded on a narrow-band channel preserving voice intelligibility and speaker identification. We have evaluated the performances of our TCVRQ in a low bit-rate Linear Prediction based speech codec.

The implemented low bit-rate speech codec (see Figure 4) follows a well-known scheme due to Atal and Remde[5].

It is a hybrid single-pulse codebook excited codec where voice signal, sampled at 8KHz with 16 bits per sample, is analyzed by using Linear Prediction. Every voice frame (80 samples) is classified as "Voiced" or "Unvoiced" thresholding the peak of the autocorrelation function and, for the voiced frames, the main pitch period is estimated.

Every frame is synthesized at 2400 bits per second using a singlepulse or a stochastic excitation vector. The best excitation signal is chosen depending on the Voiced/Unvoiced classification and evaluating the *Squared Error* between the original and the reconstructed frame.

Our TCVRQ quantizes LP parameters, represented in terms of *Line Spectrum Frequencies* (or LSFs) that change every three frames (i.e. 240 samples). With this kind of codec, the quality of the synthesized signal strongly depends on the quantization of the LP parameters.

Quantization was performed using a 10-stages trellis quantizer to encode LSFs with a bit-rate of 1.9 - 2.4 bits per LP parameter.

The Figure 5 shows the distribution of the *Cepstral Distance* (or CD) between the original and the quantized parameters; CD is a perceptually motivated distortion measure widely used for the speech coding. The results were obtained by our TCVRQ when experimenting with the speech files in the test set; the histogram shows that the average value is approximately 1 dB and the percentage of frames with a CD greater than 2 dB is quite small.

As it is confirmed from the informal listening tests too, the conditions for a
transparent quantization expressed by Paliwan in [9] are satisfied and a transparent quantization can be performed with our TCVRQ using only 1.9 - 2.4 bits per LP parameter.

CONCLUSIONS

In this paper, we extended the results stated in [14], giving a formal definition of the Trellis Coded Vector Residual Quantizer and showing the result of some experiments in low bit rate speech and still image coding.

Used for the direct quantization of gray levels still images our TCVRQ performs very close to a (locally) optimal ESVQ and outperforms a popular VQ package; for low coding rates ranging from 0.3 to 1 bits per pixel we obtained better results in terms of SQNR, speed and memory required.

The results obtained using the TCVRQ in an LP based speech codec confirm that the performance is good even in a transform codec and that nearly transparent quantization can be performed with our system at a rate of 1.9 - 2.4 bits per LP parameter.

More experiments are in progress to explore the optimality and the effect of different search algorithms.

ACKNOWLEDGMENTS

We wish to thank Prof. Martin Cohn for the fruitful discussions and the patient review of the manuscript.

REFERENCES


