DIXIEME COLLOQUE SUR LE TRAITEMENT DU SIGNAL ET SES APPLICATIONS

NICE du 20 au 24 MAI 1985

Vector quantization with extraction and separate encoding of features for waveform coding

LU Luzheng, GODLEWSKI P., COHEN G.

Département Systèmes et Communications
Ecole Nationale Supérieure des Télécommunications

46,Rue Barrault, 75013 Paris, France

RESUME

Nous proposons une nouvelle méthode de quantification vectorielle ayant une complexité réduite, impliquant l'extraction et le codage séparé de certaines caractéristiques du signal telles que la moyenne et la variance. La comparaison est faite avec les quantificateurs à recherche exhaustive, sur des critères de complexité et de performance.

Au vu des simulations de codage direct de parole, on observe un comportement dégradé mais encore satisfaisant, avec une complexité de recherche et une taille mémoire sensiblement réduite.

SUMMARY

A new vector quantization structure for reducing the complexity of the vector quantization is proposed. This method involves extraction and separate encoding of features of the signal such as average, variance, etc.... The basic idea is described, and two cases, the extraction of the average, or the extraction of the average and the variance, are studied. The comparison with the full search quantizers is done, based on the complexity and performance.

According to results of our experimental simulations for speech waveform coding, using such a new technique with better combinations, both the search complexity and the memory requirements can be substantially reduced with a degraded but still satisfactory encoding performance in comparison with the corresponding full search vector quantization at the same rate. An improvement of the performance can be obtained using such a new technique with a greater dimension at the same level of complexity.

Vector quantization with extraction and separate encoding of features for waveform coding

1. Introduction

A K-dimensional vector quantizer (VQ) is a mapping of K-dimensional Euclidean space R^K into a finite subset of R^K. It assigns to each input vector $\mathbf{x}$ a reproduction vector $\mathbf{y}=q(\mathbf{x})$ drawn from this finite set of N reproduction vectors, called the codebook which is denoted by $CB=\{\mathbf{y}_i; i=1,2,\ldots,N\}$, where $\mathbf{y}_i$ is called a codeword for each i. If we define a distrotion $d(\mathbf{x},\mathbf{y})$ which measures the cost of reproducing a vector $\mathbf{x}$ by $\mathbf{y}$, then the best mapping or the optimal encoding is one which selects as the reproduction vector for $\mathbf{x}$ the particular codeword $\mathbf{y}$ that minimizes $d(\mathbf{x},\mathbf{y})$. The study of the vector quantization is to design the optimal codebook in the sense that minimizes the average distortion and to find the associated effective optimal encoding rule.

A very important and effective method for optimal vector quantizers design is the LBG algorithm which was proposed and extensively studied by Lind, Buzo, and Gray [1]. The problem of existing techniques is the complexity: very large randomly generated codebook which must be stored, and an exhaustive search through such a codebook, or so called full search encoding process which is necessary to find out the best codeword. The computational complexity of the search can be evalueted by the numbers of additions n_+, the numbers of multiplications n_X, and the numbers of comparisons required per dimension. For the rate r bits/dimension, dimension K (codebook size $N=2^{rK}$) full search quantizer we have [2]

$$n_X = N = 2^{rK} \qquad (1.1)$$
$$n_+ = 2N = 2^{(1+rK)} \qquad (1.2)$$
$$n_{COM} = (N-1)/K = (2^{rK}-1)/K \qquad (1.3)$$

The memory (real numbers) denoted by M_e for the storage of the codebook is

$$M_e = K.N = K.2^{rK} \qquad (1.4)$$

Several attempts have been made to reduce the complexity [2-7]. In this paper we propose a new codebook structure which reduces the codebook size and then reduces the search computations.

2. Basic idea

When features such as average and variance are extracted from a signal, the dynamic range of this signal is reduced, one may expect to quantize this signal with a smaller size codebook. That proposes another vector quantization technique which may greatly reduce both the memory and the encoding complexity.

This technique can be illustrated by Fig.2.1. The features $u_1,\ldots,u_m$ of the input vector $\mathbf{x}$ of a signal to be quantized are first extracted. After the extraction operation, a new vector, denoted by $\mathbf{x}_r$, is formed, then each feature and the new vector are quantized by a scalar quantizer and a LBG vector quantizer respectively. The reproduction vector is produced from the reproduction vector of the new vector and the reproductions of features by a combination function which is a inverse procedure of the extraction operation.

Such a quantizer requires m scalar codebooks for encoding m features $u_1,\ldots,u_m$; and a vector codebook for encoding the new vector $\mathbf{x}_r$. Suppose that a vector training sequence $\{\mathbf{x}_j; j=1,\ldots,L\}$ is provided, a new vector training sequence $\{\mathbf{x}_r^{(j)}\}$ and m scalar feature's sequences $\{u_1^{(j)}\},\ldots,\{u_m^{(j)}\}$ can then be formed by the extraction operation. Each scalar codebook $CB_1,\ldots,CB_m$ is designed on its proper feature's training sequence by the LBG algorithm of K=1 or a optimal scalar quantizer design algorithm, while the vector codebook CB_r is designed on the new vector training sequence by the LBG algorithm, as described in Fig.2.2.

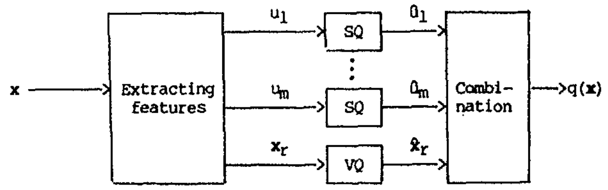

Fig.2.1. VQ with extraction and separating encoding of features: structure I (SQ-Scalar quantizer; VQ-Vector quantizer)

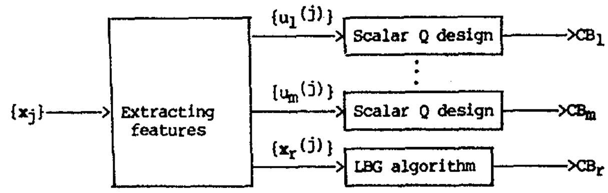

Fig.2.2. Codebook design of a vector quantizer with extracting features: structure I (Q-quantizer)

When m >1, it is possible to use another structure, that is, to design two vector quantizers: one for the m-dimensional feature vector $\mathbf{u}=(u_1,\ldots,u_m)$; another for the new vector $\mathbf{x}_r$. That is, m scalar quantizers in Fig.2.1 are replaced by a vector quantizer, as illustrated in Fig.2.3. For convenience, we call the structure of Fig.2.1 structure I and that of Fig.2.3 structure II. Two vector codebooks are required for this structure: CB_u for the feature vector, CB_r for the new vector. The design procedure is depicted in Fig.2.4. In section 4, we will use these two schemes and compare their performances.

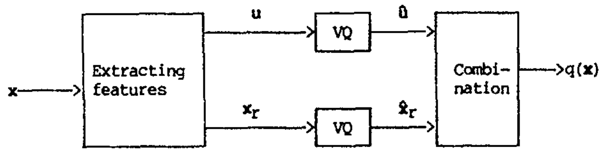

Fig.2.3. VQ with extraction and separating encoding of features: structure II (VQ-Vector quantizer)

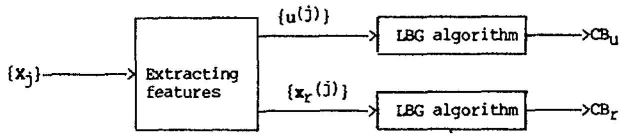

Fig.2.4. Codebook design of a vector quantizer with extracting features: structure II

Features which are extracted and separetely encoded in our method may be the average, the energy, the variance, etc. In this paper we only study two cases: (1) Vector quantization with extraction of the average; (2) Vector quantization with extraction of the average and the variance.

Generally, neither codebook nor encoding operation are optimal. The reduced complexity may be worth the performance degradation.

There are simultaneous, independent and related works by Sabin, Baker, and Gray [5],[6].

The basic idea of our technique, is that the features and the new vector are throughout separated and are processed independently, codebooks are designed individually using LBG algorithm and a scalar optimal algorithm on each proper training sequence. This is also the principal difference with [5],[6].

3.VQ with extraction of the average(VQEA)

First, we conside the simplest structure: vector quantization with extraction of the average(VQEA).

3.1. Quantizer structure

We define the average of a vector $\mathbf{x}=(X_1,X_2,\ldots X_K)$ as

$$C=(\sum_{i=1}^{K} X_i)/K \qquad (3.1)$$

and the new vector, i.e., the centred vertion of the vector as

$$\mathbf{a}=\mathbf{x}-\mathbf{c}=(a_1,a_2,\ldots,a_K)=(X_1-C,X_2-C,\ldots,X_K-C) \qquad (3.2)$$

where $\mathbf{c}=C\mathbf{1}$, $\mathbf{1}=(1,1,\ldots,1)$. Obviously we have

$$\sum_{i=1}^{K} a_i=0 \qquad (3.3)$$

The encoding operation with such a vector quantizer for a given input vector $\mathbf{x}$ begins by extracting the average from it. The new vector $\mathbf{a}$ and the average C are then encoded by a vector encoder and a scalar encoder respectively using the nearest neighbor or the minimum distortion rule. A reproduction vector $\hat{\mathbf{a}}$ which minimizes the distortion between $\mathbf{a}$ and $\hat{\mathbf{a}}$ and a reproduction average $\hat{c}$ which minimizes the distortion between C and $\hat{c}$ are selected. The index i which represents the reproduction of C and the index j which represents the reproduction vector of $\mathbf{a}$ are transmitted. The decoder selects the reproduction pair $(\hat{c},\hat{\mathbf{a}})$ from the ROM according to the index pair (i,j), and the final reproduction vector is

$$\hat{\mathbf{y}}=\hat{\mathbf{c}}+\hat{\mathbf{a}}=(\hat{a}_1+\hat{c},\hat{a}_2+\hat{c},\ldots,\hat{a}_K+\hat{c}) \qquad (3.4)$$

where $\hat{\mathbf{c}}=\hat{c}\mathbf{1}$.

For designing such a quantizer, a training vector sequence $TS=\{\mathbf{x}_j;\ j=1,2,\ldots,L\}$ is provided and the average of each vector of this vector sequence is extracted. A scalar training sequence TS_c which consists of averages and a new vector training sequence TS_a which consists of new vectors are formed:

$$TS_c=\{C_j;\ j=1,2,\ldots L\} \qquad (3.5(a))$$

$$TS_a=\{\mathbf{a}_j;\ j=1,2,\ldots,L\} \qquad (3.5(b))$$

Run the LBG algorithm on these two new training sequences, or, run the LBG algorithm on the new vector training sequence TS_a and run a optimal scalar quantizer desing algorithm on the new scalar training sequence TS_c, two corresponding codebooks CB_c with size N_c and CB_a with size N_a are designed.

Now we demonstrate that although such a quantizer is not optimal, the encoding operation described above is optimal for such a given codebook structure for the weighted squared error distortion measure with some condition in the sense that the pair $(\hat{c},\hat{\mathbf{a}})$ minimizes the distortion $d(\mathbf{x},\mathbf{y})$.

In fact, for the weighted squared error distortion measure, the distortion between the input vector $\mathbf{x}$ and its reproduction vector $\mathbf{y}$ for such a vector quantizer is

$$\begin{aligned} d(\mathbf{x},\mathbf{y}) &= (\mathbf{x}-\mathbf{y})'\mathbf{W}(\mathbf{x}-\mathbf{y}) \\ &= ((\mathbf{c}+\mathbf{a})-(\hat{\mathbf{c}}+\hat{\mathbf{a}}))'\mathbf{W}((\mathbf{c}+\mathbf{a})-(\hat{\mathbf{c}}+\hat{\mathbf{a}})) \\ &= ((\mathbf{c}-\hat{\mathbf{c}})+(\mathbf{a}-\hat{\mathbf{a}}))'\mathbf{W}((\mathbf{c}-\hat{\mathbf{c}})+(\mathbf{a}-\hat{\mathbf{a}})) \end{aligned}$$

Where "'" denotes the transpose. $\mathbf{W}$ is a symmetric, positive definite matrix. Expanding this equation, we have

$$\begin{aligned} d(\mathbf{x},\mathbf{y}) &= (c-\hat{c})^2 \sum_{i=1}^{K}\sum_{j=1}^{K} w_{i,j} \\ &\quad +2(c-\hat{c})\Big(\sum_{i=1}^{K}\sum_{j=1}^{K} w_{j,i}a_i-\sum_{i=1}^{K}\sum_{j=1}^{K} w_{j,i}\hat{a}_i\Big) \\ &\quad +(\mathbf{a}-\hat{\mathbf{a}})'\mathbf{W}(\mathbf{a}-\hat{\mathbf{a}}) \end{aligned} \qquad (3.6)$$

From (3.3) we have $\sum a_i=0$. We also notice that $\hat{a}_i$ is a codeword of codebook CB_a, and for our codebook design structure described above, this codebook is designed on a vector training sequence TS_a of (3.5(b)) using the LBG algorithm. In the training sequence TS_a, each vector $\mathbf{a}_j$ is extracted from $\mathbf{x}_j$ of the training sequence TS and satisfies formula (3.3). Thus, according to the LBG algorithm and the appendix of [8], we have

$$\sum_{i=1}^{K} \hat{a}_i = \sum_{i=1}^{K}\Big(\sum_{j=1}^{n} a_i^{(j)}/n\Big)=(1/n)\sum_{j=1}^{n}\Big(\sum_{i=1}^{K} a_i^{(j)}\Big)=0 \qquad (3.7)$$

where n is the number of training vectors in the corresponding partition. If we constrain the general matrix $\mathbf{W}$ such that

$$\sum_{s=1}^{K} w_{i,s}=\sum_{s=1}^{K} w_{s,i}=\text{constant}=t_0 \qquad (3.8)$$

The second term of (3.6) equals zero and the distortion is then

$$d(\mathbf{x},\mathbf{y})=Kt_0(c-\hat{c})^2+(\mathbf{a}-\hat{\mathbf{a}})'\mathbf{W}(\mathbf{a}-\hat{\mathbf{a}}) \qquad (3.9)$$

Thus, the optimal pair$(\hat{c},\hat{\mathbf{a}})$ that minimizes the distortion can be found by selecting $\hat{c}$ to minimize $(c-\hat{c})^2$ and selecting $\hat{\mathbf{a}}$ to minimize

$$(\mathbf{a}-\hat{\mathbf{a}})'\mathbf{W}(\mathbf{a}-\hat{\mathbf{a}})$$

in the same time. This is what we do in the encoding operation described above. That is, the encoder described above is optimal for such a given codebook design structure for the weighted squared error distortion measure if the condition (3.8) is satisfied.

For the squared-error distortion measure, $\mathbf{W}=I$(an identity matrix), the condition of (3.8) is always satisfied, the distortion for such a quantizer is

$$d(\mathbf{x},\mathbf{y})=K(c-\hat{c})^2+\sum_{i=1}^{K}(a_i-\hat{a}_i)^2 \qquad (3.10)$$

Thus, for the squared error distortion measure the encoder that we propose here is always optimal for such a given codebook structure.

3.2. Complexity

Now we consider the complexity of such a quantizer which has the vector dimension K, the average codebook size N_c, and the centred version codebook size N_a. We devote R_c bits to the average and R_a bits/vector to the new vector, and we suppose that

$$N_c=2^{Rc},\ N_a=2^{Ra} \qquad (3.11)$$

The memory requirement for the storage of codebooks for such a new quantizer is

$$Me=N_c+KN_a=2^{Rc}+K2^{Ra} \qquad (3.12)$$

The separation of the average requires 2K additions(difference), and 1 multiplication(division) per vector. Encoding of the average can be performed by comparing value c with that of codewords in codebook CB_c. This requires (N_c-1) comparisons per vector. Encoding of the centred version can be performed by a usual full search. This involves KN_a multiplications, $2KN_a$ additions, and (N_a-1) comparisons per vector. Thus, we have

$$n_x = N_a + 1/K = 2^{Ra} + 1/K \quad (3.13)$$
$$n_+ = 2N_a+2 = 2(1+2^{Ra}) \quad (3.14)$$
$$n_{COM} = (N_c+N_a-2)/K \quad (3.15)$$

The reduction of the complexity (memory requirement and encoding computations) is apparent when one compares formulae of (1.1-1.4) with that of (3.12-3.15) for the same dimension and same information rate, and many numerical examples are shown in section 5.

4. VQ with extraction of the average and the variance (VQEAV)

Now, we consider a more complicated structure: vector quantization with extraction of the average and the variance(VQEAV).

4.1. Quantizer structure

With the same definition of the average C of a vector $\mathbf{x}=(X_1,X_2,\ldots,X_K)$ as (3.1), its variance is defined as

$$V^2=\sum_{i=1}^{K}(X_i-C)^2 \quad (4.1)$$

We define the new vector as

$$\mathbf{z}=(\mathbf{x}-\mathbf{c})/V=(Z_1,Z_2,\ldots,Z_K)$$
$$=(1/V)(X_1-C,X_2-C,\ldots,X_K-C) \quad (4.2)$$

and the vector can be expressed as

$$\mathbf{x} = \mathbf{z}V + \mathbf{c}=(Z_1V+C,Z_2V+C,\ldots,Z_KV+C) \quad (4.3)$$

According to our basic idea, the encoding operation for such a quantizer for a given input vector $\mathbf{x}$ begins by extracting the average C and the variance V from it. Then C, V and the new vector $\mathbf{z}$ are encoded by two scalar encoder and a vector encoder respectively using the nearest neighbor or the minimum distortion rule. A reproduction vector $\hat{\mathbf{z}}$ which minimizes the distortion between $\mathbf{z}$ and $\hat{\mathbf{z}}$, a reproduction average $\hat{c}$ which minimizes the distortion between C and $\hat{c}$ and a reproduction variance $\hat{v}$ which minimizes the distortion V and $\hat{v}$ are selected. The index i which represents the selected $\hat{c}$, the index j which represents the selected $\hat{v}$ and the index s which represents the selected $\hat{\mathbf{z}}$ are transmitted. The decoder selects the reproductions $\hat{c}$, $\hat{v}$, and $\hat{\mathbf{z}}$ from the ROM according to the (i,j,s), and the final reproduction vector of $\mathbf{x}$ is

$$\mathbf{y}=\hat{v}\hat{\mathbf{z}} + \hat{\mathbf{c}}$$
$$=(\hat{v}\hat{z}_1+\hat{c},\hat{v}\hat{z}_2+\hat{c},\ldots,\hat{v}z_K+\hat{c}) \quad (4.4)$$

For designing such a quantizer, a training vector sequence $TS=\{\mathbf{x}_j; j=1,2,\ldots,L\}$ is provided and the average and the variance of each vector of this vector sequence is extracted. Then two scalar training sequence TS_c and TS_v which consist of averages and variances respectively and a vector training sequence TS_z which consists of new vectors are formed:

$$TS_c=\{C_j;\ j=1,2,\ldots,L\}$$
$$TS_v=\{V_j;\ j=1,2,\ldots,L\} \quad (4.5)$$
$$TS_z=\{\mathbf{z}_j;\ j=1,2,\ldots,L\}$$

By running the LBG algorithm on these new training sequence, or, by running the LBG algorithm on new vector training sequence and running an optimal scalar quantizer design algorithm on two new scalar training sequences TS_c and TS_v, three corresponding codebooks CB_c CB_v and CB_z with sizes N_c, N_v and N_z respectively are designed.

Another structure, structure II of such a new quantizer, processes jointly parametres C and V as a two-dimensional vector. Two scalar codebooks CB_c and CB_v in the structure I are replaced by a vector codebook CB_{cv} with size N_{cv} which is designed on a vector training sequence $TS_{cv}=\{(c_j,v_j);\ j=1,2,\ldots,L\}$ using the LBG algorithm if the vector training sequence $TS=\{\mathbf{x}_j;\ j=1,2,\ldots,L\}$ is provided. Two scalar encoders in the structure I are replaced by a vector encoder which reprensents the vector (C,V) by a codeword of the codebook CB_{cv} using the nearest neighbor or minimum distortion rule.

4.2. Complexity

For such a quantizer of structure I with dimension K, the average's codebook size N_c, and its corresponding bit rate R_c bits$(N_c=2^{Rc})$, the variance's codebook size N_v and its corresponding bit rate R_v $(N_v=2^{Rv})$, and the new vector's codebook size N_z and its corresponding bit rate R_z per vector$(N_z=2^{Rz})$, the memory requirement for the storage of codebooks is

$$Me=N_c + N_v + KN_z=2^{Rc} + 2^{Rv} + K2^{Rz} \quad (4.6)$$

The analysis of the encoding complexity is similar to that of section 3.2, but the extraction requires more computations: 3K additions, 2K+2 multiplications per vector. Then we have

$$n_x=N_z +2 +(2/K)=2^{Rz}+2+(2/K) \quad (4.7)$$
$$n_+=2N_z+3=2^{(Rz+1)}+3 \quad (4.8)$$
$$n_{COM}=(N_c+N_v+N_z-3)/K \quad (4.9)$$

For a quantizer of the structure II with the dimension K, codebook size $N_{cv}=2^{Rcv}$ of CB_{cv}, codebook size $N_z=2^{Rz}$ of CB_z, the memory requirements for the storage of codebooks is

$$Me=2N_{cv} + KN_z=2^{(Rcv+1)} + K2^{Rz} \quad (4.10)$$

Same computations for extracting C and V as that of the structure I are required. A two-dimensional and a K-dimensional vectors are required for encoding. So the encoding complexity can be expressed by

$$N_x=N_z+2+2(N_{cv}+1)/K \quad (4.11)$$
$$N_+=3+2N_z+4N_{cv}/K \quad (4.12)$$
$$N_{COM}=(N_z+N_{cv}-2)/K \quad (4.13)$$

The reduction of the complexity is then apparent when one compares (1.1-1.4) with (4.6-4.13) for the same dimension and same information rate, and many numerical examples are shown in section 5.

5. Experimental simulations

The performance is studied by the application of such a new technique into the direct speech waveform coding. A set of simulations is organized as follows: for a given bit rate r bits per sample or R=rK bits per vector, the performance of a full search vector quantizer with dimension K and $N=2^{rK}$ is computed, and performances of all possible combinations ($R_a+R_c=R$ in VQEA, $R_z+R_c+R_v=R$ in the structure I of VQEAV, and $R_z+R_{cv}=R$ in the structure of VQEAV) of the corresponding new vector quantizer with same dimension are computed; then several cases with different K or $N=2^{rK}$ are considered.

Vector quantization with extraction and separate encoding of features for waveform coding

The design of all codebooks was on a long training sequence of the real french speech, which consists of 645120 samples sampled at 1000hz using a 14 bits linear PCM from three speakers (two men and one woman). The splitting technique is used for the initial guess of the design and 0.005 is used as the threshold. The squared error distortion is used, and only the case of one bit per sample is considered. Using these codebooks, we have encoded two french sentences by the full search and the new method respectively: one is inside of the training sequence that is used for producing the codebook; another is outside of the training sequence. Some results of the performance in these experimental simulations with computations of M_e, n_X, n_+ and n_{COM} are given in tables 5.1-5.4. More results are given in the doctor dissertation of LU Luzheng. Here the signal to (quantization) noise ratio (SQNR,db) is defined as $R(0)/D(q)$, $R(0)$ is the input signal variance: $R(0)=E[(\mathbf{x}-E\mathbf{x})^2]$; and $D(q)$ is the expectation of the minimum distrotion: $D(q)=Ed_{min}(\mathbf{x},q(\mathbf{x}))$, where $d(.)$ is the distrotion between the input vector and the codeword of the codebook, and the minimum is taken over all codewords in the codebook.

All the quantized speech sequences were decoded and synthesized, all the decoded sequences from the full search quantizers and most of the decoded sequences from new quantizers are intelligible, and some better combinations of this new structure have the subjective quality very near to that of the corresponding full search case.

From these objective and subjective results we have some conclusions:

(1) In comparison with corresponding full search cases at the same bit rate, if the vector dimension is same and $R=R_a+R_c$ for VQEA, or $R=R_z+R_c+R_v$ for structure I of VQEAV, or $R=R_z+R_{cv}$ for structure II of VQEAV, using such a new quantization structure (VQEA or VQEAV), both the search complexity and memory are substantially reduced. The performance is degraded and for some combinations the performance degradation is acceptable.

(2) For each given vector dimension, there is a "optimal" combination with which the performance of such a new quantizer is the nearest to that of the corresponding full search case, and this "optimal" performance is improved with the increment of the vector dimension. For VQEA, in the cases of K=6 or 7 of our simulations, the "optimal" point is nearly the middle point of all combinations and both the average and the centred version of the vector are important in these cases. When the dimension is smaller, the "optimal point" moves toward one terminal where the avarage has more bits. In these cases, the average is more important than the new vector for such a new technique. When the vector dimension is getting larger, the "optimal point" moves toward another terminal where the new vector has more bits. For VQEAV, features are more important, because more features are extracted. In our simulations, even in case of K=8, the "optimal" point is at the side where the features have more bits. For the structure I of VQEAV, the average is more important than the variance for the encoding performance. For each given dimension, the inside "optimal" performance is always better than the outside "optimal" performance.

(3) In comparison with corresponding full search cases at the same bit rate, if the level of the complexity is nearly the same,(In this case, generally, $N_a=N$ for VQEA, or $N_z=N$ for VQEAV holds, the number of multiplications per dimension is nearly equal, and we can consider they have same order of encoding complexity.) such a new structure with the best combination has a performance improvement in comparison with corresponding full search case because of the increment of the dimension.

(4) In comparison with the system of VQEA, for each best combination with same vector dimension(K>=4), the performance of VQEAV is worse by 0.5-1.3 db. Generally speaking, these two systems are comparable.

(5) For same vector dimension and same bit rate per dimension, the best combination of the structure I and that of the structure II of VQEAV are comparable. The structure I is less complex.

Table 5.1. Results of full search vector quantizers (r=1 bit/sample, 10kb/s, K-dimension, R-rate bit per vector, n_X-numbers of multiplications per sample, n_+-numbers of additions per sample, n_{COM}-numbers of comparisons per sample, Me-memory, SQNR-signal to quantization noise ratio)

K=R	N	n_X	n_+	n_{COM}	Me	SQNR(db)	
						inside	outside
1	2	2	4	1	4	2.42	2.80
2	4	4	16	1.5	8	6.53	6.06
3	8	8	16	2.3	24	8.12	7.52
4	16	16	32	3.8	64	8.72	8.12
5	32	32	64	6.2	160	9.54	8.67
6	64	64	128	10.5	384	10.05	8.93
7	128	128	256	18.1	896	10.61	9.11
8	256	256	512	31.9	2048	11.69	9.66

Table 5.2. Results of vector quantizers of extracting the average ($R=R_a+R_c$, r=R/K=1 bit/sample, 10kb/s, Ra-bit rate/vectorfor the new vector, Rc-bit rate per vector for the average,Na-codebook size for the new vector, N_b-codebook size for the average.)

K=R	(R_a,R_c)	(Na,Nc)	n_X	n_+	n_{COM}	Me	SQNR(db)	
							inside	outside
5	(4,1)	(16,2)	16.2	34.0	3.2	82	3.18	3.31
	(3,2)	(8,4)	8.2	18.0	2	44	6.79	6.11
	(2,3)	(4,8)	4.2	10.0	2	28	8.36	7.99
	(1,4)	(2,16)	2.2	6.0	3.2	26	7.00	7.08
6	(5,1)	(32,2)	32.2	66.0	5.3	194	3.43	3.54
	(4,2)	(16,4)	16.2	34.0	3	100	6.95	6.33
	(3,3)	(8,8)	8.2	18.0	2.3	56	8.76	8.10
	(2,4)	(4,16)	4.2	10.0	3	40	8.46	8.24
	(1,5)	(2,32)	2.2	6.0	5.3	44	6.46	6.67
7	(6,1)	(64,2)	64.1	130.0	9.1	500	3.74	3.83
	(5,2)	(32,4)	32.1	66.0	4.9	228	7.52	6.57
	(4,3)	(16,8)	16.1	34.0	3.1	120	9.52	8.43
	(3,4)	(8,16)	8.1	18.0	3.1	72	9.08	8.34
	(2,5)	(4,32)	4.1	10.0	4.9	60	8.15	7.89
	(1,6)	(2,64)	2.1	6.0	9.1	78	6.22	6.26
8	(7,1)	(128,2)	128.1	258.0	16	1026	4.18	4.01
	(6,2)	(64,4)	64.1	130.0	8.3	516	7.85	6.71
	(5,3)	(32,8)	32.1	66.0	4.8	264	9.69	8.54
	(4,4)	(16,16)	16.1	34.0	3.8	144	9.44	8.72
	(3,5)	(8,32)	8.1	18.0	4.8	96	8.57	8.04
	(2,6)	(4,64)	4.1	10.0	8.3	96	7.36	7.41
	(1,7)	(2,128)	2.1	6.0	16	144	5.49	5.67

Vector quantization with extraction and separate encoding of features for waveform coding

Table 5.3. Results of vector quantizer with extracting the average and the variance (structure I) ($R=R_z+R_c+R_v$, $r=R/K=1$ bit/sample, 10Kb/s)

K=R	(Rz,Rc,Rv)	(Nz,Nc,Nv)	n_X	n_+	n_{COM}	Me	SQNR(db) inside	SQNR(db) outside
6	(1,4,1)	(2,16,2)	4.3	7	2.8	30	7.70	7.84
	(1,3,2)	(2,8,4)	4.3	7	1.8	24	8.02	7.61
	(1,2,3)	(2,4,8)	4.3	7	1.8	24	6.29	5.73
	(1,1,4)	(2,2,16)	4.3	7	2.8	30	2.96	3.04
	(2,3,1)	(4,8,2)	6.3	11	1.8	34	7.52	7.46
	(2,2,2)	(4,4,4)	6.3	11	1.5	32	6.37	5.88
	(2,1,3)	(4,2,8)	6.3	11	1.8	34	3.09	3.17
	(3,2,1)	(8,4,2)	10.3	19	1.8	54	6.06	6.01
	(3,1,2)	(8,2,4)	10.3	19	1.8	54	3.23	3.43
	(4,1,1)	(16,2,2)	18.3	35	2.8	100	2.92	3.32
7	(1,5,1)	(2,32,2)	4.3	7	4.7	48	7.27	7.35
	(1,4,2)	(2,16,4)	4.3	7	2.7	34	8.14	7.73
	(1,3,3)	(2,8,8)	4.3	7	2.1	30	7.95	7.24
	(1,2,4)	(2,4,16)	4.3	7	2.7	34	6.22	5.50
	(1,1,5)	(2,2,32)	4.3	7	4.7	48	3.03	3.07
	(2,4,1)	(4,16,2)	6.3	11	2.7	46	7.64	7.54
	(2,3,2)	(4,8,4)	6.3	11	1.9	40	8.15	7.50
	(2,2,3)	(4,4,8)	6.3	11	1.9	40	6.54	5.77
	(2,1,4)	(4,2,16)	6.3	11	2.7	46	3.20	3.24
	(3,3,1)	(8,8,2)	10.3	19	2.1	66	7.74	7.59
	(3,2,2)	(8,4,4)	10.3	19	1.9	64	7.01	6.24
	(3,1,3)	(8,2,8)	10.3	19	2.1	66	3.56	3.56
	(4,2,1)	(16,4,2)	18.3	35	2.7	120	6.14	5.98
	(4,1,2)	(16,2,4)	18.3	35	2.7	118	3.49	3.67
	(5,1,1)	(32,2,2)	34.3	67	4.7	228	3.06	3.50
8	(1,6,1)	(2,64,2)	4.3	7	8.1	82	6.40	6.89
	(1,5,2)	(2,32,4)	4.3	7	4.4	52	7.43	7.34
	(1,4,3)	(2,16,8)	4.3	7	2.9	40	7.66	7.35
	(1,3,4)	(2,8,16)	4.3	7	2.9	40	7.25	6.81
	(1,2,5)	(2,4,32)	4.3	7	4.4	52	5.82	5.29
	(1,1,6)	(2,2,64)	4.3	7	8.1	82	3.06	3.02
	(2,5,1)	(4,32,2)	6.3	11	4.4	66	6.97	7.26
	(2,4,2)	(4,16,4)	6.3	11	2.6	52	7.97	7.67
	(2,3,3)	(4,8,8)	6.3	11	2.1	48	7.81	7.26
	(2,2,4)	(4,4,16)	6.3	11	2.6	52	6.25	5.63
	(2,1,5)	(4,2,32)	6.3	11	4.4	66	3.29	3.22
	(3,4,1)	(8,16,2)	10.3	19	2.9	82	7.52	7.90
	(3,3,2)	(8,8,4)	10.3	19	2.1	76	8.57	8.06
	(3,2,3)	(8,4,8)	10.3	19	2.1	76	7.15	6.31
	(3,1,4)	(8,2,16)	10.3	19	2.9	82	3.77	3.62
	(4,3,1)	(16,8,2)	18.3	35	2.9	138	7.23	7.55
	(4,2,2)	(16,4,4)	18.3	35	2.6	136	6.97	6.39
	(4,1,3)	(16,2,8)	18.3	35	2.9	138	3.86	3.75
	(5,2,1)	(32,4,2)	34.3	67	4.4	262	5.98	6.13
	(5,1,2)	(32,2,4)	34.3	67	4.4	262	3.80	3.89
	(6,1,1)	(64,2,2)	66.3	131	8.1	516	3.21	3.61

Table 5.4. Results of vector quantizers with extracting the average and the variance (structure II) ($R=R_a+R_{cv}$, $r=R/K=1$ bit/sample, 10Kb/s)

K=R	(R_z,R_{cv})	(Nz,Ncv)	n_X	n_+	n_{COM}	Me	SQNR(db) inside	SQNR(db) outside
5	(4,1)	(16,2)	19.5	36.6	3.2	84	2.00	2.25
	(3,2)	(8,4)	12	22.2	2.0	48	4.80	4.43
	(2,3)	(4,8)	9.6	17.4	2.0	36	6.21	5.91
	(1,4)	(2,16)	10.8	19.8	3.2	42	7.95	7.00
6	(5,1)	(32,2)	35	68.3	5.3	196	1.63	1.70
	(4,2)	(16,4)	19.6	37.7	3.0	104	4.40	4.35
	(3,3)	(8,8)	13	24.3	2.3	64	6.12	5.81
	(2,4)	(4,16)	11.7	21.7	3.0	56	7.25	6.74
	(1,5)	(2,32)	15	28.3	5.3	76	8.08	7.47
7	(6,1)	(64,2)	66.9	132.1	91	452	1.38	1.51
	(5,2)	(32,4)	35.4	69.3	4.9	232	4.14	4.05
	(4,3)	(16,8)	20.3	39.6	3.1	128	5.89	5.54
	(3,4)	(8,16)	14.9	28.1	3.1	88	7.53	6.88
	(2,5)	(4,32)	15.4	29.3	4.9	92	7.93	7.27
	(1,6)	(2,64)	22.6	43.6	9.1	142	8.24	7.42
8	(7,1)	(128,2)	130.8	260	16	1028	1.47	1.60
	(6,2)	(64,4)	67.3	133	8.3	520	4.20	3.85
	(5,3)	(32,8)	36.3	71.0	4.8	272	5.86	5.65
	(4,4)	(16,16)	22.3	43.0	3.8	160	7.30	6.72
	(3,5)	(8,32)	18.3	35.0	4.8	128	8.46	7.76
	(2,6)	(4,64)	22.3	43.0	8.3	160	7.99	7.50
	(1,7)	(2,128)	36.3	71.0	16	272	7.70	7.30

Acknowledgment

We would like to thank associate professor YU Tie-cheng for several helpful discussions. Our special thanks go to the staff of the computer centre of the laboratory SYC for their help in simulations.

References

[1] Y.Lind, A.Buzo, and R.M.Gray, "An algorithm for vector quantizer design," IEEE Trans.Commun.,Vol.COM-28, pp.84-95, Jan.1980.

[2] R.M.Gray, and Y.Lind,"Vector quantizer and predictive quantizers for Gauss Markov sources," IEEE Trans.Commun.,Vol.COM-30, pp.381-389,Feb.1982.

[3] B.H.Juang and A.H.Gray,Jr.," Multiple stage vector quantization for speech coding,"Proc. ICASSP., pp.597-600, Paris, 1982.

[4] D.Y.Chen, A.Gersho, B.Ramamurthi, and Y.Shoram, "Fast search algorithms for vector quantization and pattern matching,"Proc.ICASSP.,pp.9.11.1-9.11.4,1984.

[5] M.J.Sabin and R.M.Gray, "Product vector quantizers for waveform coding and voice coding,"IEEE Tran. ASSP,Vol ASSP-32, No.3, June 1984.

[6] Baker,R.L., and Gray,R.M.," Differetial vector quantization of achromatic imagery,"Proceeding of the International Picture Coding Symposium, March 1983.

[7] Cohen G., Godlewski Ph., and LU Luzheng,"Vector quantization with pre-searching," to be published in IASTED International Symposium,Paris,France.June 1985

[8] Abut,H., Gray,R.M, and Rebolledo, G., "Vector quantization of speech and speech-like waveforms," IEEE Trans.ASSP., ASSP-30, pp.423-435, June 1982.