VECTOR QUANTIZATION OF DCT COMPONENTS FOR SPEECH CODING

Jianwei Miao
School of Electrical and Computer Engineering
Georgia Institute of Technology

ABSTRACT - In this study a system of design algorithms and experimental results was investigated for vector quantization of discrete cosine transform components in speech coding. The encoding and decoding systems of vector quantization of discrete cosine transform components were developed for digital speech coding. A training sequence consisting of 200,000 speech samples sampled at 8 kHz with 16 bits signed integers was used to design the codebooks. The codebooks were designed using the K-means algorithm. The implementation of a discrete cosine transform vector quantizer used a two-codebook structure, providing different codes for different real component vectors corresponding to different frequency bands. This is a form of subband coding and yields a means of optimizing bit allocations among the subcodes as well as produces a good quality speech at low bit rates. The experimental results of implementing the encoding and decoding systems showed a good quality speech at 7111 BPS with 15.61 dB in signal-to-quantization noise ratio.

INTRODUCTION

The goal of selecting the transformation from the time domain to the frequency domain for digital speech coding is to obtain uncorrelated spectral samples. The Karhunen-Loéve transform (KLT), in this sense, is optimum which it yields uncorrelation of spectral values. However, this KLT is difficult to calculate in general. For instance, given a frame of N points, O(N^4) flops are necessary to compute the KLT (Wintz, 1972). Furthermore, the whole autocorrelation matrix of the frame must be sent to the receiver as side information in order that the transform may be inverted. The discrete Fourier transform (DFT) and discrete cosine transform (DCT) are viable alternatives. The DFT and DCT are not optimum which they do not decorrelate the data, but they have the advantage of not depending on the data statistics and will approximately decorrelate the components when the block length is sufficiently long (Pearl, 1973, Ahmed, et al., 1974, Chang, et al., 1987). Of these two, the DCT yields good performance compared with the KLT and is generally used in practice (Deller, et al., 1993).

The application of vector quantization (VQ) techniques in the DCT components has recently emerged as a powerful means for digital speech compression. Because the DCT of the digital speech data prior to quantization has three potential advantages. First, each sample in the DCT domain depends on many samples in the original domain. Second, DCT vector quantization of the digital speech waveforms supplies apparently better subjective quality than ordinary vector quantization of the digital speech waveforms using codes of comparable complexity. Third, quantization noise can be shaped by allocating bits to the DCT components according to a 'perceptual criterion'.

The purpose of this paper is to investigate the VQ of DCT components for digital speech coding. The implementation of a DCT vector quantizer used a two-codebook structure which provided

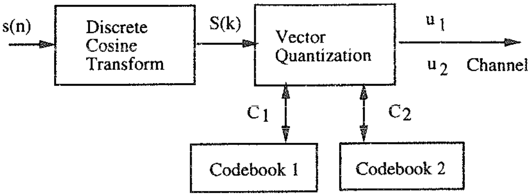

Figure 1: Vector quantization of DCT components-based transmitter.

different codes for different component vectors corresponding to different frequency bands. This is a simple form of subband coding and produces a good quality speech at low bit rates.

SYSTEM DESCRIPTION

Quantization noise introduced at the coder is spread throughout the entire time frame of digital speech. The noise properties may be different across frame boundaries. Therefore, overlapped trapezoidal windows are used to smooth the transitions. Such an overlapped trapezoidal window was employed to segment each block length 252 sampled speech in our studies.

A block diagram of encoding process is shown in Figure 1. Each frame, denoted s(n), is transformed using the DCT to the vector which is represented components. Two codebooks, denoted C_1 and C_2, are designed using the K-means algorithm based on the different frequency bands (or two subblocks) from the components of DCT block. Then, the VQ process maps first frequency band of DCT vector into a channel symbol u_1 and second frequency band of DCT vector into a channel symbol u_2.

Figure 2 shows a block diagram of the VQ of DCT components-based decoding process. The VQ decoder reconstructs the DCT vector from the channel symbols u_1 and u_2. Then, an inverse DCT is taken to rebuild the speech segments $\widehat{s}(n)$.

DESIGN PROCEDURES

The DCT has been shown to have superior performance in speech coding but only have real components. The proposed method in this paper is to quantize the real components by using a two-codebook vector quantizer such that the bit allocation to different frequency bands is obtained. The implementations of a DCT and a two-codebook vector quantizer are discussed in the following sub-sections.

Discrete cosine transform

In this section, we consider one variation known as even symmetrical DCT, which is most often used in speech coding applications. Let s(n) denote an N-point speech sequence that is zero outside. To derive the DCT relationship, it is convenient to relate the N-point speech sequence s(n) to a new 2N-point speech sequence y(n), which is then related to its 2N-point DFT Y(k). Finally, we relate Y(k) to S(k), the N-point DCT of s(n). The definition of DCT of s(n) is described in Equation (1),

$$S(k) = \begin{cases} \sum_{n=0}^{N-1} 2s(n)cos\frac{\pi}{2N}k(2n+1) & 0 \le k \le N-1 \\ 0 & \text{otherwise.} \end{cases} \qquad (1)$$

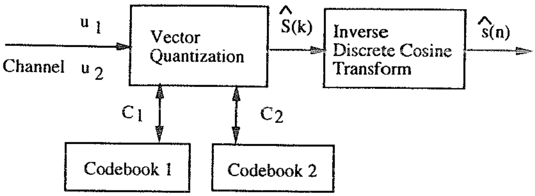

Figure 2: Vector quantization of DCT components-based receiver.

To derive the inverse discrete cosine transform (IDCT) relationship, we first relate S(k) to Y(k), Y(k) to 2N-point speech sequence y(n), and then y(n) to N-point speech sequence s(n). Equation (2) is the definition of IDCT of S(k),

$$s(n) = \begin{cases} \frac{1}{N}\left[\frac{S(0)}{2} + \sum_{n=0}^{N-1} S(k) cos\frac{\pi}{2N}k(2n+1)\right] & 0 \le k \le N-1 \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Fortunately, the computation involved in using the DCT is essentially the same as that involved in using the DFT. Moreover, the DFT and inverse DFT can be computed by using fast Fourier transform (FFT) algorithm. Such a way of efficient to compute DCT pair can be realized in the following:

- The computation of DCT in the transmitter

$$y(n) = s(n) + s(2N-1-n) \tag{3}$$

$$Y(k) = DFT[y(n)] \quad \text{(2N-point DFT computation)} \tag{4}$$

$$S(k) = \begin{cases} W_{2N}^{k/2} Y(k) & 0 \le k \le N-1 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

- The computation of IDCT in the receiver

$$\hat{Y}(k) = \begin{cases} W_{2N}^{-k/2}\hat{S}(k) & 0 \le k \le N-1 \\ 0 & \text{k = N} \\ -W_{2N}^{-k/2}\hat{S}(2N-k) & N+1 \le k \le 2N-1 \end{cases} \tag{6}$$

$$\hat{y}(n) = IDFT[\hat{Y}(k)] \quad \text{(2N-point inverse DFT computation)} \tag{7}$$

$$\hat{s}(n) = \begin{cases} \hat{y}(n) & 0 \le k \le N-1 \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where $W_{2N} = e^{-j(2\pi/2N)}$.

Codebook design

In vector quantization, we need to determine the reconstruction level and corresponding cell. A list of reconstruction levels is called a codebook. The codebook is needed at the transmitter to quantize a DCT vector to one of reconstruction levels and at the receiver to determine the reconstruction level from the receiver codeword. The same codebook should be known to both the transmitter and receiver through prior agreement.

Optimal design of a codebook is a highly nonlinear problem. In practice, we typically use an unsupervised clustering approach, which is a hierarchical or a partitional algorithm, to solve this problem. The hierarchical and partitional clustering approaches contain more general measures of DCT training vector similarity and do not require a set of training vectors with label information. However, when the number of DCT training vectors is large, the hierarchical clustering may not be appropriate. Because the hierarchical methods may lead to resulting data partitions being suboptimal in an agglomerative procedure (Jobson, 1992, Schalkoff, 1992).

A commonly used partitional method is the K-means algorithm which evaluates the proximity between groups using the Euclidean distance between group centroids (Tou & Gonzalez, 1981). This algorithm iteratively subdivides the M DCT training vectors into L clusting (M $\gg$ L) such that the algorithm converges to a locally optimal clustering. The K-means algorithm may be described as follows:

(1) Initialize by setting the iteration number $i = 0$. Choose a set of output vectors $\mathbf{Z}_k(0)$, $1 \leq k \leq L$.

(2) Classify the DCT training vectors $\{\mathrm{S(m)}, 1 \leq m \leq M\}$ into the clusters C_k by

$$\mathrm{S} \in C_k(i) \quad \text{iff} \parallel \mathrm{S} - \mathbf{Z}_k(i) \parallel \leq \parallel \mathrm{S} - \mathbf{Z}j(i) \parallel, \quad \text{for all } k \neq j, \tag{9}$$

where $D(\mathrm{S}, \mathbf{Z}_k) = \parallel \mathrm{S} - \mathbf{Z}_k(i) \parallel$ is a distortion measurement.

(3) Recalculate the output vectors of every cluster by computing the centroid of the DCT training vectors that fall in each cluster,

$$\mathbf{Z}_k(i) = \frac{1}{M_k} \sum_{\mathrm{S} \in C_k} \mathrm{S}(m) \quad 1 \leq k \leq L. \tag{10}$$

Also compute the resulting distortion $D(i)$ at the ith iteration.

(4) If the change $D(i-1) - D(i)$ in the average distortion is relatively small, then stop the K-means algorithm. Otherwise, go to step (2).

Once we have selected the output vector $\{\mathbf{Z}_k, 1 \leq k \leq L\}$, each DCT vector $\mathrm{S}(m)$ is quantized to the output vector that is nearest to it according to the distortion measurement that is adopted.

EXPERIMENTAL RESULTS

A training sequence consisting of 200,000 speech samples sampled at 8 kHz with 16 bits signed integers was used to design the codebooks. Each block length 252 sampled speech is segmented by using an overlapped trapezoidal window. Then, the DCT was applied to each block of samples. Since the high dynamic range of DCT components was located in lower frequency band which included most energy of voiced speech signals and low dynamic range of DCT components was located in higher frequency band which included less energy of voiced speech signals, more bits were assigned for the lower-frequency band and fewer bits were assigned for the higher-frequency band in this studies. Therefore, after implementing a DCT to each block of speech samples, 252 DCT components were divided into first and second subblocks (low frequency band and high frequency band) which contained 84 and 168 frequency components, respectively. The size of codebook C_1 was set to 256 levels × 6 dimensions (8 bits, 6 dimensions per vector) for first subblock and the size of codebook C_2 was set to 16 levels × 6 dimensions (4 bits, 6 dimensions per vector) for second subblock. Figure 3 showed an original striplot of 3500 speech samples sampled at 8 kHz with 128 kbits per second. After executing the VQ of DCT components for speech coding, Figure 4, as a result, showed a corresponding striplot of 3500 speech samples

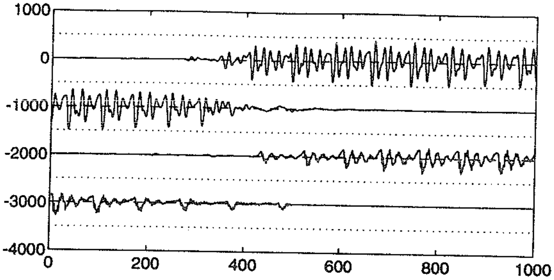

Figure 3: This graph showed an original striplot of 3500 speech samples with 128 kbits per second.

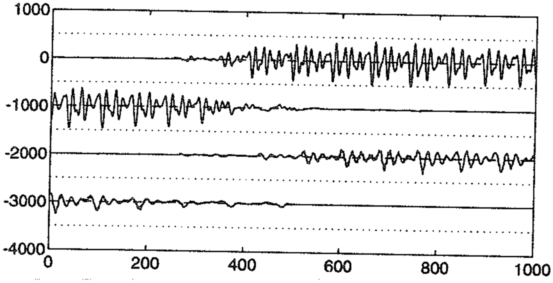

Figure 4: As a result, this graph showed a corresponding striplot of 3500 speech samples with 7.1 kbits per second after executing the VQ of DCT components for speech coding.

with 7.1 kbits per second. As can be seen, the number of kbits per second in Figure 4 is $\frac{1}{18}$ times the number of kbits per second in the original speech signals.

Synthesis results were evaluated using the signal-to-quantization noise ratios denoted by SNR (Rabiner & Schafer, 1978),

$$SNR = \frac{\sigma_s^2}{\sigma_e^2} = \frac{E[s^2(n)]}{E[e^2(n)]} = \frac{\sum_n s^2(n)}{\sum_n e^2(n)}, \tag{11}$$

and

$$SNR(dB) = 10\log_{10}\left[\frac{\sum_n s^2(n)}{\sum_n e^2(n)}\right], \quad e(n) = s(n) - \hat{s}(n). \tag{12}$$

The projected synthesis results based on the VQ of DCT components for speech coding were showed in Table 1.

Table 1. Synthesis Results

BPS	SNR	SNR(dB)
7111	36.3861	15.6094

CONCLUSIONS

In this paper a discrete cosine transform-based coding system was presented for vector quantizing speech at 7111 BPS with 15.61 dB in SNR. The implementation of a DCT vector quantizer used a two-codebook structure which provided different codes for different DCT component vectors corresponding to different frequency bands. An advantage of such an implementation form is that it is a simple means of optimizing bit allocations among the subcodes and yields a good quality speech coding at low bit rates.

Adaptive bit allocation with a dynamic codebook can be applied to enhance the performance and decrease the bit rates of speech coding in the further studies. However, the encoding part will have to compute and transmit side information. On the other hand, the bit rates of speech coding may be reduced by exploiting the redundancy in the transform domain (i.e., by encoding only a subset of the DCT components) and by increasing the number of dimensions per vector. Obviously, these methods will increase the computation complexity and the storage requirement for a system.

REFERENCES

Ahmed, N., Natarajan, T. & Rao, K. (1974) *Discrete cosine transform*, IEEE Trans. Comput., C-23, 90-93.

Chang, P., Gray, R.M & May, J. (1987) *Fourier transform vector quantization for speech coding*, IEEE Trans. on Communications, COM-35, No. 10, 1059-1068.

Deller, J.R., Proakis, J.G., & Hansen, J.H.L. (1993) *Discrete-time processing of speech signals*, (Macmillan Publishing Company: New York).

Jobson, J.D. (1992) *Applied multivariate data analysis: categorical and multivariate methods*, (Springer-Verlag: New York).

Pearl, J. (1973) *On coding and filtering stationary signals by discrete Fourier transforms*, IEEE Trans. Inform. Theory, 229-232.

Rabiner, L.R. & Schafer, R.W. (1978) *Digital processing of speech signals*, (Prentice-Hall, Inc.: New Jersey).

Schalkoff, R. (1992) *Pattern recognition: statistical, structural and neural approaches*, (John Wiley & Sons, Inc.: New York).

Tou, J.T & Gonzalez, R.C. (1981) *Pattern Recognition Principles*, (Addison-Wesley Publishing Company: Massachusetts).

Wintz, P.A. (1972) *Transform picture coding*, Proceedings of the IEEE, vol. 60, 880-920.