FAST VQ ENCODING ALGORITHMS USING ANGULAR CONSTRAINT

Kousuke Imamura, Ahmed Swilem, Hideo Hashimoto

Kanazawa University, Japan

ABSTRACT

In this paper, we propose two fast encoding algorithms for vector quantization. The first algorithm uses three significant features of a vector, that is , the norm, its projection angle to a reference line and the mean, to reduce the search area and accelerate the search process. The second algorithm has feature of using a suitable hyperplane to partition the codebook and image data. These algorithms allow significant acceleration in the encoding process. Experimental results are presented on image block data. These results confirm the effectiveness of our proposed algorithms.

1. INTRODUCTION

A standard vector quantization (VQ) is an efficient compression technique for which many variants are known [1]. It is defined as a mapping Q from a k-dimensional Euclidean space R^k to a finite set $Y=\{y_1,y_2,...,y_N\}$ of vectors in R^k called the codebook. Each representative vector y_i in the codebook is called a codeword. A complete description of vector quantization process includes three phases: codebook design, encoding and decoding. The objective of codebook design is to construct a codebook Y from a set of training vectors using clustering algorithms like the generalized Lloyd algorithm (GLA) [2]. This codebook is used in both the encoder and the decoder. The encoding phase is equivalent to finding the vector $Q(x)=y_i \in Y$ minimizing the distortion $d^2(x,y_i)$ defined as the squared Euclidean distance between the vector x and y_i. The decoding phase is simply a table look-up procedure that uses the received index i to deduce the reproduction codeword y_i, and then uses y_i to represent the input vector x .

The computational cost of finding the closest codeword in the encoding of VQ imposes practical limits on the codebook size *N*. When *N* becomes larger, the computational complexity problem for full codebook search occurs. To avoid such an exhaustive search through the codebook, many fast algorithms [3]-[7] have been proposed. These algorithms reduce the computational complexity by performing some simple tests before computing the distortion between the input vector and each codeword, and then rejecting those codewords that fail in the tests.

This paper introduces two new algorithms to reduce the computational complexity of the encoding process. The first one achieves equivalent performance to the full search VQ. It uses three constraints; the annular constraint, the angular constraint and the mean constraint. This method uses the projection angles on a reference line in the space of input vectors and codewords. It searches a smaller number of codewords than the previously known methods. The second method uses a hyperplane partitioning rule, which separates the codebook and the input vectors into two parts, and searches in only one part according to the vector feature. The searching in this method speeds up the encoding process with a negligible small sacrifice of encoding quality.

2. EQUAL-AVERAGE EQUAL-VARIANCE NEAREST NEIGHBOR SEARCH (EENNS)

Lee and Chen [5] introduced an efficient codeword search method , which uses the mean and the variance of the vector for two tests to reject the codewords. In the mean test, the central line *l* in an Euclidean space R^k is defined as a line on which the coordinates of any point have the same value. For a given input vector x with mean value m_x, and a current best codeword y_i with distance $d_{\min}=d(x,y_i)$, any codeword that is closer to x than y_i has to be located inside the hypersphere centered at x with radius $d_{\min}$. Two boundary points $L_{\max}=(m_{\max},m_{\max},...,m_{\max})$ and $L_{\min}=(m_{\min},m_{\min},...,m_{\min})$ can be obtained by projecting the hypersphere on the central line, where

$$m_{\max}=m_x+d_{\min}/\sqrt{k}\ , \tag{1}$$

and

$$m_{\min}=m_x-d_{\min}/\sqrt{k}\ . \tag{2}$$

Thus, by (1) and (2) only the codewords that are bounded by the two hyperplanes S_1 and S_2 which are perpendicular to the central line and pass through $L_{\max}$ and $L_{\min}$, respectively, will be searched.

In the variance test, the squared root of variance of the vector x, v_x, is used as the distance $d(x,L_x)$ between x and its projection point L_x on the central line. The closest codeword y_j with squared root of variance v_{y_j} will satisfy the following inequality:

$$(v_x-v_{y_j})^2<d_{\min}^2\ . \tag{3}$$

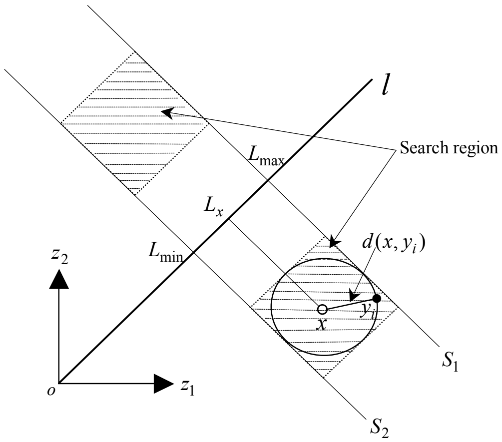

Fig. 1. Geometrical interpretation of the EENNS method for 2-dimensional case.

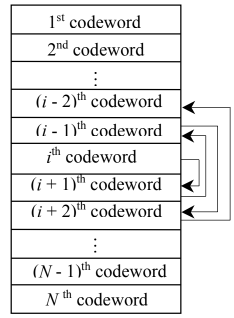

Fig. 2. Searching path.

By using the constraints in (1), (2) and (3), the search region will be reduced to the two shaded squares shown in Fig. 1. Every codeword not contained in this region is eliminated. To speed up the search process, the codebook is sorted according to increasing order of the codeword means and the search is performed up and down iteratively as shown in Fig. 2.

3. PROPOSED METHODS

3.1. Angular and mean constraints method (AMC)

In this method we use the annulus constraint, the angular constraint and the mean constraint. In the annulus constraint, the annulus is centered at the origin o. For a given input vector *x* of distance $\|x\|$ from the origin and a current best codeword y_i with distance $d_{\min} = d(x, y_i)$, any closer codeword y_j to *x* than y_i will satisfy the following relationships:

$$\| y_j \| < \| x \| + d(x, y_i), \tag{4}$$

and

$$\| y_j \| > \| x \| - d(x, y_i), \tag{5}$$

where $\| y_j \|$ is the Euclidean distance of y_j from the origin. Thus, any codeword y_j satisfying (4) and (5) must be contained in the annulus defined by $\|x\| + d(x, y_i)$ and $\|x\| - d(x, y_i)$.

In the angular constraint, let *l* be a reference line in the search space and it contains a unit vector $u = (1,1,...,1)/\sqrt{k}$ on it. For any vector *z*, we define the angle between *z* and the reference line *l* as:

$$\alpha_z = \cos^{-1} \frac{u\, z^T}{\| z \|}. \tag{6}$$

Because the values of all vector components are nonnegative, then the angle $\alpha \in [0, \frac{\pi}{4}]$. The angle α is called the projection angle to the reference line *l*. We define another angle between the input vector *x* and the tangent from the origin to the hypersphere centered at *x* with radius $d(x, y_i)$, where y_i is the current best codeword, as:

$$\theta_x = \sin^{-1} \frac{d(x, y_i)}{\|x\|}. \tag{7}$$

For a given input vector *x* with its projection angle α_x to the reference line *l* and the closest codeword y_j with its projection angle α_{y_j}, the following inequalities should be satisfied:

$$\alpha_{y_j} < \alpha_x + \theta_x, \tag{8}$$

and

$$\alpha_{y_j} > \alpha_x - \theta_x. \tag{9}$$

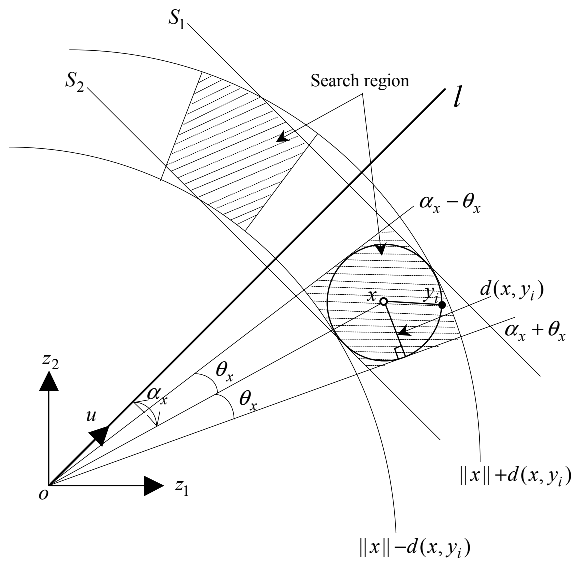

Fig. 3. Geometrical interpretation of the AMC method for 2-dimensional case.

In the mean constraint, for a given input vector x with mean value m_x, and a current best codeword y_i with distance $d_{\min} = d(x, y_i)$, any closer codeword y_j to x than y_i will satisfy the following relationships:

$$m_{y_j} < m_x + d_{\min} / \sqrt{k}\ , \quad (10)$$

and

$$m_{y_j} > m_x - d_{\min} / \sqrt{k}\ , \quad (11)$$

where m_{y_j} is the mean value of the codeword y_j. Thus, as we mentioned before, by (10) and (11) only the codewords that are bounded by the two hyperplanes $S_1 : u z^T = \sqrt{k} m_x + d_{\min}$ and $S_2 : u z^T = \sqrt{k} m_x - d_{\min}$ will be searched.

The inequalities (4), (5), (8), (9), (10) and (11) constrain the distortion calculation to the codewords that are contained in the search region shown in Fig. 3. To speed up the search process, the codebook is sorted according to increasing order of the codeword norms and the search is performed up and down iteratively as shown in Fig. 2.

3.2. Angular and mean constraints with hyperplane decision rule (AMCH)

Most nearest-neighbor search techniques employ searching the best codeword in the same search region for all input vectors. In this section, we introduce a technique using a hyperplane H to divide the signal space into two half-spaces according to the vector feature. This method has been tried with success in [7] for both Cardinal method and the double annulus method.

The chosen hyperplane H contains the centroid of the input vectors $x_c = (x_{c1}, x_{c2}, ..., x_{ck})$ and its projection point on the central line $x_p = (m_{x_c}, m_{x_c}, ..., m_{x_c})$, where m_{x_c} is the mean value of x_c. It is perpendicular to the central line as shown in Fig. 4 and can be expressed as:

$$H : u z^T = u x_c^T = \frac{1}{\sqrt{k}} \sum_{i=1}^{k} x_{ci} = \sqrt{k}\ m_{x_c} = M_c\ . \quad (12)$$

This hyperplane H is used as a decision function that discriminates to which half-space a given vector x belongs by the following conditions:

- If $\quad u x^T = \frac{1}{\sqrt{k}} \sum_{i=1}^{k} x_i < M_c,$ (13)

  then x belongs to the lower half-space.

- If $\quad u x^T = \frac{1}{\sqrt{k}} \sum_{i=1}^{k} x_i \geq M_c,$ (14)

  then x belongs to the upper half-space.

Now we depict the proposed method that uses the hyperplane H to separate both the input vectors and the codewords. The proposed method divides the input vectors into two sub-groups T_{lw} and T_{up}, and each sub-group contains the vectors satisfying (13) or (14), respectively. Also, it divides the codebook into two sub-codebooks Y_{lw}, and Y_{up} by the same equations. Searching for the input vectors in the sub-group T_{lw} is carried out in the sub-codebook Y_{lw} and for the input vectors in the sub-group T_{up} in the sub-codebook Y_{up}, by using the constraints of inequalities (4), (5), (8), (9), (10) and (11). Hence, the proposed method can reduce the search area and speed up the search process.

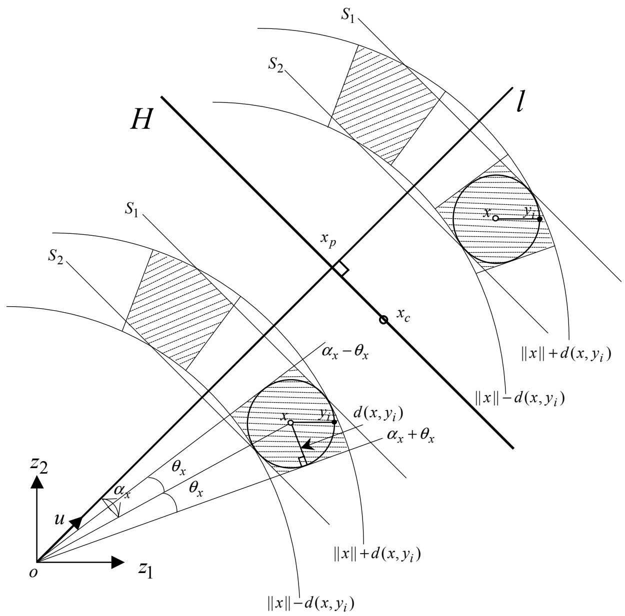

Fig. 4. Geometrical interpretation of the AMCH method for 2-dimensional case.

The proposed method may be easily understood with the geometrical interpretation for 2-dimensional case in Fig. 4. This figure includes the proposed hyperplane H. The hyperplane H divides the signal space into two half-spaces, and each half-space includes its own input vectors and codewords.

4. EXPERIMENTAL RESULTS

Experiments were carried on vectors taken from the USC grayscale image set. We used two images, Lena and Baboon with size 512×512 and 256 gray levels. Each image was divided into 4×4 blocks. Codebooks with different codebook sizes (N = 128, …, 2048) were designed using the full search algorithm for each image. The tested methods are full search (FS), the equal-average equal-variance nearest neighbor search (EENNS), angular and mean constraints (AMC) and angular and mean constraints with hyperplane decision rule (AMCH). Fig. 5 shows the PSNR comparison between the FS method and the AMCH method for different codebook sizes with Lena and Baboon images. The AMC method is equivalent to the FS method as mentioned before, and has the same performance as that of the FS method. Although the AMCH method is a lossy coding method, it has almost the same performance as the FS method at

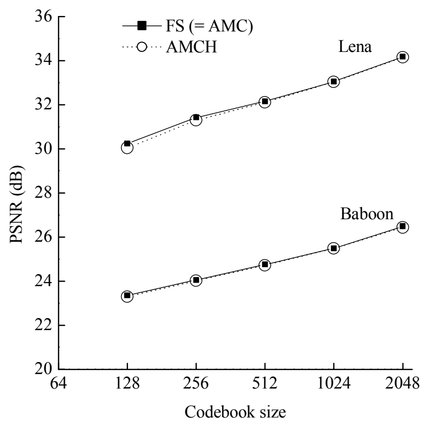

Fig. 5. The PSNR Comparison.

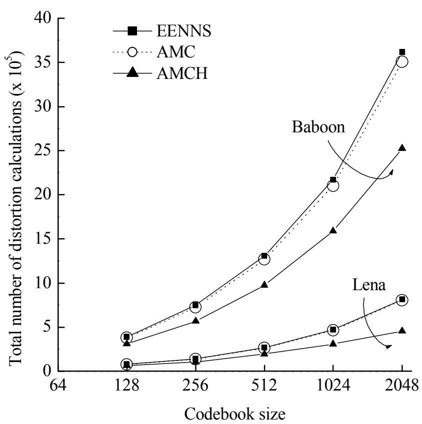

Fig. 6. The total number of distortion calculations comparison.

Table I. Comparison of the execution time (in seconds) at codebook size 512.

	FS	EENNS	AMC	AMCH
Lena	10.218	0.976	0.968	0.718
Baboon	10.219	1.796	1.758	1.355

larger codebook size. For example, the performance of the AMCH method is only 0.192 dB less than the FS method at codebook size 128 for Lena image and 0.049 dB for Baboon image. These values decrease by increasing the codebook size; for example, the AMCH method has 0.011 dB for Lena image and 0.002 dB for Baboon image less than the FS method at codebook size 1024. The reason of performance degradation is that the best codeword happens to be in the other half-space and is missed to be searched out. However, there may be a small failure possibility in the case of large codebook and smooth codewords distribution. Fig. 6 presents a comparison of the total number of distortion calculations, which is a dominant figure of the computational complexity. In the case of Lena image, the total number of distortion calculations of the AMC and the AMCH methods are about 98% and 69% of that of the EENNS. In the case of Baboon image, the total number of distortion calculations of the AMC and the AMCH methods are about 96% and 74% of that of the EENNS. From these results, smaller distortion calculations are carried out in the AMCH method. Table I presents a comparison of the execution time (in seconds) at codebook size 512 for Lena and Baboon images. The timings were made on Pentium III (866 MHZ). The AMC method has almost the same execution time as the EENNS method. However, the execution time of the AMCH method is about 74% of that of the EENNS.

5. CONCLUSIONS

In this paper, we have proposed two new algorithms for encoding of vector quantization. The first algorithm (AMC) uses three constraints; the annular, the angular and the mean. It employs the projection angles of the vectors to a reference line in the signal space and achieves equivalent performance to full search VQ. It has almost the same efficiency as the EENNS method. The second algorithm uses a hyperplane decision technique for separating the input vectors and the codebook into two sub-groups, and carries on searching within one sub-group according to the vector features. By applying this algorithm to the AMC method, the AMCH method is developed. The obtained results show that the AMCH method is more efficient than the EENNS method and its performance is quite close to that of the FS method. Although the AMCH method is a lossy one theoretically, it is comparable to the FS method and has not been found among previously existing methods.

11. REFERENCES

[1] A. Gersho and R. M. Gray, "Vector quantization and signal compression," Bostone, Kluwer, 1991.

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

[3] M. D. Orchard, "A fast nearest-neighbor search algorithm," in Proceedings of ICASSP 91:1991 International Conference on Acoust., Speech and Signal Process., vol. 4, pp. 2297-2300, April 1991.

[4] L. Guan and M. Kamel, "Equal-average hyperplane partitioning method for vector quantization of image data," Pattern Recognition Lett., vol. 13, no. 10, pp. 693-699, Oct. 1992.

[5] C. H. Lee and L. H. Chen, "Fast closest codeword search algorithm for vector quantization," IEE Proc. Vis. Image Signal Process., vol. 141, no. 3, pp. 143-148, June 1994.

[6] K. S. Wu and J. C. Lin, "Fast VQ encoding by an efficient kick-out condition," IEEE Trans. Circuits syst. Video Technol., vol. 10, no. 1, pp. 59-62, Feb. 2000.

[7] K. Imamura, A. Swilem, and H. Hashimoto, "Fast codeword search algorithm for ECVQ using hyperplane decision rule," Proc. ISCAS2003, vol. 2, pp. 476-479, May 2003.