KURAに登録されているコンテンツの著作権は、執筆者、出版社（学協会）などが有します。

KURAに登録されているコンテンツの利用については、著作権法に規定されている私的使用や引用などの範囲内で行ってください。

著作権法に規定されている私的使用や引用などの範囲を超える利用を行う場合には、著作権者の許諾を得てください。ただし、著作権者から著作権等管理事業者（学術著作権協会、日本著作出版権管理システムなど）に権利委託されているコンテンツの利用手続については、各著作権等管理事業者に確認してください。
FAST SEARCH ALGORITHMS FOR ECVQ USING PROJECTION PYRAMIDS AND VARIANCE OF CODEWORDS

A. Swilem, K. Imamura, H. Hashimoto

Kanazawa University, Kanazawa, Japan

ABSTRACT

Vector quantization for image compression requires expensive time to find the closest codeword through the codebook. Codebook design based on empirical data for entropy-constrained vector quantization (ECVQ) involves a time consuming training phase in which a Lagrangian cost measure has to be minimized over the set of codebook vectors. In this paper, we propose two fast codebook generation methods for ECVQ. In the first one, we use an appropriate topological structure of input vectors and codewords to reject many codewords that are impossible to be candidates for the best codeword. In the second method, we use the variance test to increase the ability of the first algorithm to reject more codewords. These algorithms allow significant acceleration in the codebook design process. Experimental results are presented on image block data. These results show that our new algorithms perform better than the previously known methods.

1. INTRODUCTION

Vector quantization (VQ) [1] has played an important role in numerous data compression systems. It is defined as a mapping \( Q \) from a \( k \)-dimensional Euclidean space \( \mathbb{R}^k \) to a finite set \( C = \{ y_0, y_1, ..., y_{N-1} \} \) of vectors in \( \mathbb{R}^k \) called the codebook. A vector quantization process includes three phases: codebook design, encoding and decoding. The objective of codebook design is to construct a codebook \( C \) 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 find the vector \( Q(x) = y_i \in C \) minimizing the distortion \( d(x,y_i) \) defined as the 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 \( x \).

Entropy-constrained vector quantization (ECVQ) [3] employs a modified cost measure using both the effective distortion of the signal and the expected length of the transmitted code. We define the cost function for encoding the vector \( x \) by the codeword \( y_i \) as the Lagrangian function,

\[
J(x,y_i) = d(x,y_i) + \lambda R(y_i),
\]

where \( \lambda \) is a constant called the Lagrange multiplier allowing to control the rate-distortion ratio and \( R(y_i) \) is the length of the codeword \( y_i \).

The computational cost of finding the best suitable codeword in the codebook design and encoding imposes practical limits on the codebook size \( N \) and the vector dimension \( k \). When \( N \) and \( k \) become larger, the computational complexity problem occurs for full codebook search. This has motivated developing many fast nearest neighbor search algorithms [4]-[11]. Many algorithms concentrate on narrowing the area of the candidate codewords for which distortion must be calculated. For example, Lee and Chen [4] introduced one of such algorithms for standard VQ, which uses the mean and the variance of the vector to reduce the search area. Cardinal [6] generalized the technique in [4] to apply it to ECVQ. To constrain the search area, Cardinal method uses two elimination rules; the first rule utilizes the projection of the vector on a central line in the signal space, while the second rule employs the distance between the vector and its projection point on the central line.

Other algorithms exploit the topological structure of the codebook to avoid unnecessary codeword matching procedure. Lee and Chen [4] proposed a fast search algorithm for the codebook generation of standard VQ based on mean pyramids of codewords. Pan et al. [10] improved the encoding search process by adopting variance pyramids in addition to mean pyramids. Another technique using \( L_2 \)-norm pyramids of codewords has been proposed in [11].

This paper proposes two methods to reduce the complexity in the codebook design for ECVQ. The proposed methods are developed by combining the idea of projection pyramid data structure with narrowing the search area by Cardinal method [6]. In the first method, we derive a multilevel inequality based on the projection pyramids of codewords. By employing this inequality, the codebook design process can be speeded up. In the second method, we add a variance test to the first algorithm to reject more codewords and speed up the codebook design process. These methods have the same coding quality as the full search method.

The paper is organized as follows. Section 2 describes the projection pyramid data structure. Section 3 introduces the proposed algorithms in detail. Experimental results are shown in section 4, and concluding remarks are given in section 5.

2. PROJECTION PYRAMID DATA STRUCTURE

Image pyramid data structure was originally developed for image coding by Burt and Adelson [12]. In this data structure, an image is represented hierarchically, with each level corresponding to a reduced-resolution approximation. Given an image \( Z^m \) of size \( 2^m \times 2^m \), its pyramid can be defined as a sequence of matrices \( \{ Z^0, Z^1, ..., Z^{r-1}, Z^r, ..., Z^m \} \), where an image \( Z^0 \) in level \( r-1 \) has a size of \( 2^r \times 2^r \) and is a reduced-resolution version of \( Z^r \). Note that \( Z^0 \) has only one pixel. A pyramid data structure can be formed by successively performing appropriate operations over \( 2 \times 2 \) neighboring pixels in the next lower level. Therefore, the value of a pixel \( z^{r-1}(i,j) \) in level \( r-1 \) can be obtained by

\[
z^{r-1}(i,j) = f( z^r(2i,2j), z^r(2i+1,2j), z^r(2i,2j+1), z^r(2i+1,2j+1)),
\]

where \( f \) is a function depending on the pyramid level.
Lemma 1:

\[ d(x, y_i) \geq \frac{1}{\sqrt{k}} \left( \sum_{j=0}^{k-1} |x_j - y_{ij}| \right) \leq d(x, y_i) \geq u \left( |x - y_i| \right)^2 \geq \frac{1}{\sqrt{k}} \left( \sum_{j=0}^{k-1} |x_j - y_{ij}| \right) \]

Proof: For any real values \( \alpha_0, \alpha_1, ..., \alpha_{k-1} \), we have
\[
\sum_{j=0}^{k-1} (|\alpha_j|)^2 = (|\alpha_0| + |\alpha_1| + ... + |\alpha_{k-1}|)^2 = \sum_{j=0}^{k-1} |\alpha_j| |\alpha_{k-j}|.
\]

Also we have
\[
\sum_{j=0}^{k-1} \sum_{h=0}^{k-1} |\alpha_j| |\alpha_{h}| = 2k \sum_{j=0}^{k-1} |\alpha_j|^2 - 2 \sum_{j=0}^{k-1} \sum_{h=0}^{k-1} |\alpha_j| |\alpha_{h}| + |\alpha_{h}|^2.
\]

This implies
\[
k \sum_{j=0}^{k-1} |\alpha_j|^2 - k \sum_{j=0}^{k-1} |\alpha_j| |\alpha_{h}| = \frac{k \sum_{j=0}^{k-1} |\alpha_j|^2}{2} - k \sum_{j=0}^{k-1} \sum_{h=0}^{k-1} |\alpha_j| |\alpha_{h}| \geq 0,
\]

hence,
\[
k \sum_{j=0}^{k-1} |\alpha_j|^2 \geq (\sum_{j=0}^{k-1} |\alpha_j|)^2.
\]

Let \( \alpha_j = x_j - y_{ij} \) where \( j = 0, 1, ..., k-1 \), then
\[
d(x, y_i) = \frac{1}{\sqrt{k}} \left( \sum_{j=0}^{k-1} |x_j - y_{ij}| \right)^2 \geq \frac{1}{\sqrt{k}} \left( \sum_{j=0}^{k-1} |x_j - y_{ij}| \right),
\]

this completes the proof. From lemma 1 we can obtain the following corollary:

**Corollary 1:**

\[ J(x, y_i) = d(x, y_i) + \lambda R(y_i) \geq u \left( |x - y_i| \right)^2 + \lambda R(y_i) \]

**Corollary 2:**

\[ J(x, y_i) = d(x, y_i) + \lambda R(y_i) \geq u \left( |x - y_i| \right)^2 + \lambda R(y_i) \]

Let us assume that the current best codeword is \( y_n \) with the minimum distortion \( J(x, y_n) = J_{\text{min}} \). If the codeword \( y_j \) satisfies the inequality \( J(x, y_j) \geq J_{\text{min}} \), then \( J(x, y_j) \geq J_{\text{min}} \) is always guaranteed. Hence \( y_j \) cannot be closer to \( x \) than \( y_n \) and can be rejected.

By using the projection pyramid data structure described in section 2, we can extend Eq. (8) to a general case, where tighter decision boundaries for eliminating search operations are obtained in a multilevel. Let us assume that the vector dimension \( k = 2^m \times 2^m \), and two projection pyramids for \( x \) and \( y_i \) are \( \{X^0, X^{1}, ..., X^{r-1}, X^r, ..., X^{m}\} \) and \( \{Y_i^0, Y_i^1, ..., Y_i^{r-1}, Y_i^r, ..., Y_i^m\} \), respectively. Then we can redefine Eq. (8) as,

\[ J^m(x, y_i) = u^m (|X^m - Y_i^m|^2) + \lambda R(y_i) \]

**Corollary 3:**

\[ J^m(x, y_i) = u^m (|X^m - Y_i^m|^2) + \lambda R(y_i) \]

From the last definition in Eq. (9) and corollary 2, we can easily get the following theorem.
Theorem 1:
\[
\tilde{J}(x, y) = \tilde{J}^m(x, y) \geq \tilde{J}^{m-1}(x, y) \geq \tilde{J}^{m-2}(x, y) \geq \ldots \geq \tilde{J}^{0}(x, y).
\]
Proof: Since
\[
\tilde{J}^m(x, y) = \frac{1}{2^m} \sum_{j=0}^{2^{m-1}} \sum_{k=0}^{2^{m-1}} |X^m(j, h) - Y^m_i(j, h)| + \lambda R(y_i),
\]
and from Eq. (3) if \( l = r + 1 \), we get,
\[
\tilde{J}^{m-1}(x, y) = \frac{1}{2^{m-1}} \sum_{j=0}^{2^{m-1}-1} \sum_{k=0}^{2^{m-1}-1} \frac{1}{2} |(X^m(j, 2h) - Y^m_i(2j, 2h)) + (X^m(2j, 2h) - Y^m_i(2j, 2h)) + (X^m(2j + 1, 2h + 1) - Y^m_i(2j + 1, 2h + 1)) + \lambda R(y_i)|
\]
\[
\leq \frac{1}{2^{m-1}} \sum_{j=0}^{2^{m-1}-1} \sum_{h=0}^{2^{m-1}-1} |(X^m(j, 2h) - Y^m_i(2j, 2h)) + (X^m(2j, 2h + 1) - Y^m_i(2j, 2h + 1)) + (X^m(2j + 1, 2h + 1) - Y^m_i(2j + 1, 2h + 1)) + \lambda R(y_i)|
\]
\[
\leq \tilde{J}^{m-2}(x, y).
\]

4. EXPERIMENTAL RESULTS
Experiments were carried out 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 is divided into 4×4 blocks, whereby each block becomes a 16-dimensional vector (\( k = 16 \) and \( m = 2 \)). The tested methods are Cardinal (CARD), which is known as the fastest existing method for ECVQ, the projection pyramid (PP) and the projection pyramid with variance test (PPV).

Figs. 2 and 3 present the execution time (in seconds) of the three methods with various codebook sizes at \( \lambda = 0.5 \) for Lena and Baboon, respectively. The timings were made on Pentium III (866 MHz). We can see that both PP and PPV methods significantly accelerate the codebook design more than the CARD method. In case of Lena, PP and PPV reduce the time by average ratios 37.1% and 40.6%, respectively. But in case of Baboon, they reduce the execution time by average ratios 39.4% and 42.9%, respectively. It can be seen that as the codebook size increases, the efficiency of the proposed algorithms becomes better than the CARD method. This is an important merit of our algorithms, because design of a larger codebook size requires more intensive computation.

Figs. 4 and 5 compare the rejection ratio of the codewords in each iteration compared to the FS method at \( N = 256 \) and \( \lambda = 0.5 \) for both Lena and Baboon, respectively. The average rejection ratios are 95.3% for CARD, 96.5% for PP, and 96.9% for PPV in case of Lena, while they are 79.9%, 86.5% and 88.2% for baboon, respectively. We can see that the rejection ratio at iteration 1 is smaller than the ratios at other iterations. The main reason is that the tested methods employ a random best codeword for each training vector \( x \) at the first iteration, while in any other iterations, they use the best codeword obtained for \( x \) in the previous iteration.

Improvement of rejection ratio in the intermediate level for Baboon image is larger than that of Lena image. This difference can be explained from the statistics of the used test images in the following. For a given input vector \( x \), the first elimination rule of the CARD method, which is also employed in the top level test of both the PP and the PPV methods, constrains the search area between two parallel hyperplanes normal to the central line\[6\]. The width of this search area is determined by the Lagrangian...
distortion $J(x, y_b)$, where $y_b$ is the current best codeword. This search area is affected by the distribution of codewords, which is also related to that of input vectors. For instance, when the codeword distribution is dense like in Lena image, $J(x, y_b)$ will be small and the width of the search area becomes narrow. Thus, most of codewords will be rejected at the top level test, and there is less chance to reject more codewords in the other levels as in Fig. 4. While in Baboon image, the codeword distribution is not so dense as in Lena image, then $J(x, y_b)$ may be larger and more codewords are contained in the search area at the top level. Then, there is a chance to reject more codewords in the other levels of the proposed methods as in Fig. 5. As a result, the proposed projection pyramid algorithm is more effective for difficult images of which coding efficiency is low.

5. CONCLUSIONS

In this paper, we have presented two fast search algorithms for ECVQ. The first algorithm uses the projection pyramids of codewords to reject many unmatched codewords, thus drastically speeding up the search process in ECVQ codebook design. In the second algorithm, the variance test was added to the projection pyramid method to reject more codewords in the search process. Simulation results show that the complexity of the proposed algorithms is significantly reduced while the coding quality remains the same as of the exhaustive search algorithm.

6. REFERENCES


