

論文2002-39SP-1-4

고속 벡터 양자 부호화를 위한 L_2 -평균 피라미드 기반 탐색 기법 (L_2 -Norm Pyramid-Based Search Algorithm for Fast VQ Encoding)

宋秉哲*, 羅鍾範*

(Byung Cheol Song and Jong Beom Ra)

요약

영상 압축을 위한 기법인 벡터 양자 부호화는 입력 벡터와 가장 근사한 부호어를 찾기 위해 일반적으로 상당히 많은 계산량을 요구한다. 본 논문은 벡터 양자 부호화 과정을 고속 처리를 위한 새로운 탐색 기법을 제안한다. 먼저, 탐색 과정에서의 불필요한 정합 연산들을 대폭 줄이기 위해 부호책의 효과적인 기하학적 구조에 기반한 강력한 후보 제거 조건을 유도한다. 그런 다음, 그 후보 제거 조건을 이용한 고속 탐색 기법을 제안한다. 모의 실험 결과는 적은 전처리 연산과 메모리만을 사용하는 제안한 기법이 전역 탐색 기법과 동일한 부호화 성능을 유지하면서 부호화 시간을 대폭 줄일 수 있음을 보인다. 또한 기존 탐색 기법들과 비교할 때, 제안한 기법의 성능이 매우 우수함을 알 수 있다.

Abstract

Vector quantization for image compression needs expensive encoding time to find the closest codeword to the input vector. This paper proposes a search algorithm for fast vector quantization encoding. Firstly, we derive a robust condition based on the efficient topological structure of the codebook to dramatically eliminate unnecessary matching operations from the search procedure. Then, we propose a fast search algorithm using the elimination condition. Simulation results show that with little preprocessing and memory cost, the encoding time of the proposed algorithm is reduced significantly while the encoding quality remains the same with respect to the full search algorithm. It is also found that the proposed algorithm outperforms the existing search algorithms.

1. 서론

벡터 양자화 (VQ)는 비트율-왜곡 (rate-distortion) 측면에서 좋은 성능을 보이기 때문에 효과적으로 신호를 압축할 수 있다. 그래서, 오랫동안 음성 및 영상 부호화 (image coding) 등 여러 부호화 분야에 많이 적용되어

왔다^{1~4}. 영상의 VQ 부호화는 K -차원 유클리디안 (Euclidean) 공간 R^K 에서 R^K 의 유한 집합인 Y 로의 사상 함수 Q 로 정의될 수 있다. 즉, $Q: R^K \rightarrow Y$. 여기서 $Y = \{y_i; i=1, 2, \dots, N\}$ 는 부호책 (codebook)이라 불리는 복원 벡터들의 집합이며, N 은 부호책의 크기를 의미한다. 부호화에 앞서 입력 영상은 여러 겹치지 않는 블록들 (이하 벡터)로 나뉜다. 그런 다음, 각 K -차원 입력 벡터 $\mathbf{x} = (x_1, x_2, \dots, x_K)$ 는 부호책의 모든 부호어 (codeword)들과 비교된 후, $Q(\mathbf{x}) = \mathbf{y}_i = (y_{i1}, y_{i2}, \dots, y_{iK})$ 의 관계에 따라 가장 근사한 부호어로 양자화된다. 제곱 유클리디안 거리 (squared Euclidean distance)를 거리 척도로 사용할 경우 \mathbf{x} 와 $\mathbf{y}_i \in Y$ 사이의 거리는

* 正會員, 韓國科學技術院 電子電算學科

(Dept. of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology)

接受日: 2000년 12월 14일, 수정완료일: 2001년 8월 6일

$$d(\mathbf{x}, \mathbf{y}_i) = \sum_{k=1}^K (x_k - y_{ik})^2 = \|\mathbf{x} - \mathbf{y}_i\|^2 \quad (1)$$

사상 함수 Q는 다음 조건을 만족하는 \mathbf{x} 에서 \mathbf{y}_i 로의 사상으로 정의된다.

$$d(\mathbf{x}, \mathbf{y}_i) = \min_{i=1, \dots, N} d(\mathbf{x}, \mathbf{y}_i) \quad (2)$$

최적 정합 부호어의 색인값을 벡터 값 대신 복호기(decoder)에 전송함으로써 압축이 이루어진다. 이 때, 복호기는 부호기와 동일한 부호책을 가지며, 복호화는 전송된 색인값을 이용해 'table look-up' 과정을 거쳐 간단하게 이루어진다. 이런 복호화의 단순성이 VQ의 가장 큰 장점이다.

최적 정합 부호어를 찾기 위해, VQ 부호기들은 일반적으로 전역 탐색 기법(full search algorithm; FSA)을 사용한다. FSA는 입력 벡터와 부호책에 있는 모든 부호어들 간 유클리디안 거리들을 일일이 계산하고 비교하는 방식이기 때문에 막대한 계산량을 요구한다. 예를 들어, 벡터 당 평균 연산 수를 부호화 복잡도의 척도로 할 경우, FSA의 연산 복잡도는 KN 이다. 따라서, 부호화 연산량이 벡터 차원과 부호책 크기에 따라 증가하게 되고, VQ를 실시간 응용 시스템에 적용하기 힘들게 만든다.

이런 문제점을 해결하기 위해, 몇몇 고속 탐색 기법들이 제안되었다^[5-14]. 이 기법들은 크게 손실 방식과 무손실 방식으로 구분될 수 있다^[12]. 손실 방식들은 계산량을 줄이기 위해 부호책에 구조적 제한을 가한다^[5]. 그러나, 최적이지 않은 탐색 방식이나 최적이지 않은 부호책을 사용하기 때문에 부호화 성능이 FSA에 비해 떨어진다^[1,5]. 한편, 무손실 방식들은 FSA와 동일한 부호화 성능을 보장한다.

여러 무손실 기법들 중에서, Bei와 Gray가 제안한 PDE(partial distance elimination)는 특정 후보 부호어와의 거리 값 계산 중에 그 부분 거리 값이 현재 최소 값보다 커지면, 더 이상의 계산 없이 해당하는 부호어를 후보에서 제거하는 기법이다^[6]. PDE는 부호화 과정의 추가적인 속도 향상을 위해 종종 다른 VQ부호화 기법들과 함께 사용되곤 한다. TIE(triangle inequality elimination) 기반 기법들은 삼각형 부등식 정리를 이용하여 탐색 공간을 줄임으로써 계산량을 줄였다^[7-13].

Huang과 Chen이 제안한 기법의 경우^[7], 모든 부호어들 간 거리들을 미리 계산하여 저장해야 하기 때문에 많은 메모리가 요구된다^[7]. 그래서, Ra와 Kim은 메모리 비용을 현격히 줄이기 위한 방법으로 정렬된 평균값 기반의 고속 탐색 기법을 제안하였다^[9]. 최근에는, 단일 삼각형 부등식이 아닌 다중 삼각형 부등식 기반의 고속 탐색 기법들이 개발되고 있다^[10-12]. 다중 삼각형 부등식 기반의 기법들은 여러 탐색 영역들의 교집합이 최종 탐색 영역이기 때문에, 탐색 영역 자체가 작아져 계산량을 크게 줄일 수 있다. 한편, Wu와 Lin은 새로운 후보 제거 조건을 적용한 고속 탐색 기법을 제안하였다^[14]. 유클리디안 거리 척도에 비해 계산량은 적으면서 동일한 성능을 갖는 새로운 거리 척도를 제안하고, 제안한 거리 척도를 최소화하는 부호어를 찾기 위해 신뢰성이 없는 후보 부호어들을 미리 제거하는 Cauchy-Schwarz부등식 기반 제거 조건을 유도하였다.

본 논문은 VQ부호화 속도를 향상시키기 위해 부호어들의 L_2 -평균 피라미드 기반의 고속 탐색 기법을 제안한다. 탐색 과정에서 불필요한 정합 연산들을 제거하기 위해, L_2 -평균 피라미드 기반의 Cauchy-Schwarz부등식을 제안하고, 이를 이용한 강력한 후보 제거 조건을 유도한다. 제안한 기법은 이 부등식 조건을 적용하여 VQ부호화의 성능 저하 없이 계산량을 크게 줄일 수 있다.

본 논문은 다음과 같이 구성된다. 다음 절에서, L_2 -평균 피라미드 데이터 구조에 대해 설명한다. III절에서 제안한 기법이 자세히 설명된다. IV절에 모의 실험 결과가 주어지며, 마지막 V절에서 결론을 맺는다.

II. L_2 -평균 피라미드 데이터 구조

영상 데이터 피라미드 구조는 원래 영상 부호화를 위해 개발되었다^[15]. 이 데이터 구조에서, 영상은 여러 계층들로 분할되며, 각 계층 영상들은 원 영상에 비해 낮은 해상도를 갖는다. $2^L \times 2^L$ 크기의 영상 U^L 의 피라미드(그림 1)는 $\{U^0, \dots, U^{l-1}, U^l, U^{l+1}, \dots, U^L\}$ 의 행렬 집합으로 정의될 수 있다. 여기서 U^{l-1} 은 $2^{l-1} \times 2^{l-1}$ 의 크기를 가지며, U^l 의 해상도를 1/4로 줄인 영상에 해당한다. 결국 U^0 는 한 화소에 해당한다. 피라미드 데이터 구조는 현재 계층의 각 화소가 상위 계층의 인접한 2×2 화소 블록에 연산을 취하여 얻어지

는 방식으로 구성된다. 말하자면, 계층 $l-1$ 의 한 화소 $u^{l-1}(m, n)$ 은 계층 l 에서 대응하는 2×2 인접 화소들 $u^l(2m-1, 2n-1)$, $u^l(2m-1, 2n)$, $u^l(2m, 2n-1)$, $(2m, u^l 2n)$ 로부터 얻어진다. 즉,

$$u^{l-1}(m, n) = O(u^l(2m-1, 2n-1), u^l(2m-1, 2n), u^l(2m, 2n-1), u^l(2m, 2n)) \quad (3)$$

식 (3)에서 $O(\cdot)$ 는 연산자를 의미한다.

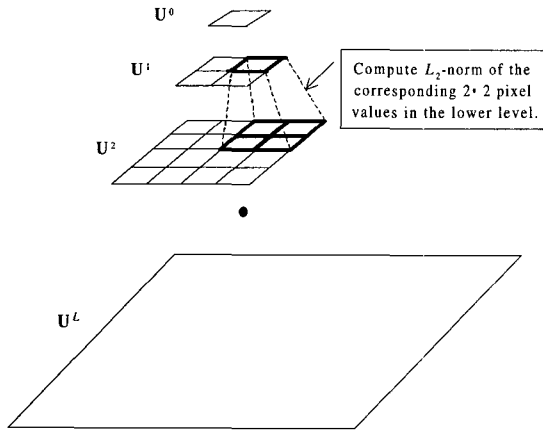


그림 1. L_2 -놈 피라미드 데이터 구조
Fig. 1. L_2 -norm pyramid data structure.

함수 O 에 따라 많은 종류의 영상 피라미드들이 존재한다. 가장 간단한 피라미드 구조가 L_1 -평균 피라미드 구조이며, 인접한 2×2 화소들의 평균을 취함으로써 얻어진다. 그러나, 본 논문에서는 L_2 -평균 피라미드를 사용한다. 가장 간단한 L_1 -평균 피라미드 대신 L_2 -평균 피라미드 데이터 구조를 사용하는 이유는 대부분의 VQ 부호기가 제곱 유클리디안 거리, 즉 L_2 -평균 거리를 양자화를 위한 측정자로 사용하기 때문이다. L_2 -평균 피라미드 데이터 구조에서 맨 아래 계층은 원 영상 자체이며, 최상위 계층은 그 영상 전체의 L_2 -평균에 해당한다. 계층 $l-1$ 에서의 L_2 -평균 화소 $u^{l-1}(m, n)$ 은 다음과 같이 얻어진다.

$$u^{l-1}(m, n) = \sqrt{\sum_{p=0}^1 \sum_{q=0}^1 (u^l(2m-1+p, 2n-1+q))^2} \quad (4)$$

다음 절에서 부호어들의 L_2 -평균 피라미드 기반의 최

적 근사 부호어 탐색 기법을 소개한다.

III. 제안한 기법

이 절에서 최적 근사 부호어 탐색 기법을 제안한다. L_2 -평균 피라미드 기반의 강력한 부등식 조건을 유도하고, 이를 이용하여 VQ 부호화 속도를 향상시키는 새로운 탐색 기법을 제안한다.

1. 전 단계

I절에서 언급했듯이, 입력 벡터 x 의 최적 정합 부호어 y_{bm} 을 찾기 위해서는 부호책 내의 모든 부호어들과 x 를 비교하여야 한다. 이러한 과정은 $y_{bm} = Q(x)$ 로 표현될 수 있다. 대부분의 VQ 부호기가 제곱 유클리디안 거리를 양자화를 위한 측정자로 사용하기 때문에, 본 논문에서도 제곱 유클리디안 거리 즉, L_2 -평균 거리를 측정자로 한다. 이 때, 식 (2)는 다음과 같이 재정리된다.

$$d(x, y_{bm}) = \min_{i=1, \dots, N} d(x, y_i) = \min_{i=1, \dots, N} (\|x\|^2 + \|y_i\|^2 - 2 \sum_{k=1}^K x_k y_{ik}). \quad (5)$$

VQ의 궁극적인 목표는 식 (5)를 최소화하는 부호어를 찾는 것이다. 그러나, 이와 동일한 결과를 가지면서 시간을 절약할 수 있는 방법은 다음 조건을 만족하는 부호어를 찾는 것이다.

$$d'(x, y_{bm}) = \min_{i=1, \dots, N} d'(x, y_i) \quad (6)$$

식 (6)에서, $d'(x, y_i)$ 는

$$d'(x, y_i) = d(x, y_i) - \|x\|^2 = \|y_i\|^2 - 2 \sum_{k=1}^K x_k y_{ik} \quad (7)$$

이다. 식 (7)에서 $\|x\|^2$ 는 모든 부호어들에 대해 공통항이기 때문에 제거해도 무방하며, 따라서, $\{\|y_i\| \mid 1 \leq i \leq N\}$ 이 전처리 과정을 통해 미리 계산되어 있기 때문에, 식 (7)를 계산하는 것이 식 (1)을 계산하는 것보다 빠르다. 따라서, 본 논문은 식 (1)의 유클리디안 거리를 최소화하는 것이 아니라, 식 (7)의 d' 거리를 최소화하는데 초점을 맞춘다.

현재까지 부호책의 부호어들 중 일부가 조사되었다고 할 때, 현재까지의 최소 거리를 d_{\min} 이라고 하자. 식 (7)과 Cauchy-Schwarz 부등식에 의해 다음 성질을 유도할 수 있다.

$$d'(\mathbf{x}, \mathbf{y}_i) = \|\mathbf{y}_i\|^2 - 2 \sum_{k=1}^K x_k y_{ik} \geq \|\mathbf{y}_i\|^2 - 2\|\mathbf{y}_i\| \cdot \|\mathbf{x}\| = \|\mathbf{y}_i\| \cdot (\|\mathbf{y}_i\| - 2\|\mathbf{x}\|). \quad (8)$$

따라서, 부호어 \mathbf{y}_i 가 식 (9)의 조건을 만족하면, 항상 $d'(\mathbf{x}, \mathbf{y}_i) \geq d_{\min}$ 이므로 \mathbf{y}_i 를 후보에서 제거할 수 있다.

$$\|\mathbf{y}_i\| \cdot (\|\mathbf{y}_i\| - 2\|\mathbf{x}\|) \geq d_{\min} \quad (9)$$

다시 말해 y_i 는 현재까지의 최적 근사 부호어보다 x 에 더 근사할 수 없기 때문에 더 이상의 거리 계산 없이 제거될 수 있다.

$\{\|\mathbf{y}_i\| \mid 1 \leq i \leq N\}$ 이 전처리 과정을 통해 미리 계산되어 있기 때문에, 입력 벡터 x 에 대한 $\|\mathbf{y}_i\| \cdot (\|\mathbf{y}_i\| - 2\|\mathbf{x}\|)$ 의 계산은 매우 간단하다. Wu와 Lin은 식 (8)과 (9)를 이용한 고속 VQ 부호화 기법을 제안하였다^[14].

2. L_2 -평균 피라미드 데이터 구조 기반의 부등식 성질

II절에서 도입한 L_2 -평균 피라미드를 이용하여, 본 논문은 식 (8)의 부등식 성질을 일반적인 경우로 확장한다. 벡터 차원 K 가 $2^L \times 2^L$ 이라고 하고, 입력 벡터 \mathbf{x} 와 부호어 \mathbf{y}_i 의 계층 l 에서의 벡터들을 각각 \mathbf{x}^l 와 \mathbf{y}_i^l 로 표현하기로 하자. 이 때, $\mathbf{x}^0 = \|\mathbf{x}\|$ 이다. 그러면, Cauchy-Schwarz 부등식으로부터 다음 부등식을 유도할 수 있다:

$$\sum_{k=1}^K x_k y_{ik} \equiv \sum_{k=1}^{2^L \times 2^L} x_k^l y_{ik}^l \leq \sum_{k=1}^{2^{l-1} \times 2^{l-1}} x_k^{l-1} y_{ik}^{l-1} \leq \dots \leq \sum_{k=1}^{2 \times 2} x_k^1 y_{ik}^1 \leq x^0 y_i^0 \equiv \|\mathbf{x}\| \cdot \|\mathbf{y}_i\| \quad (10)$$

식 (10)에서 x_k^l 은 \mathbf{x}^l 의 k 번째 원소를 가리키며, 계층 $l+1$ 의 바로 인접한 2×2 화소들의 L_2 -평균에 해당한다. 식 (10)에 대한 증명은 부록에서 보인다. 식 (10)으로부터 식 (8)이 다음과 같이 일반화할 수 있다:

$$d'(\mathbf{x}, \mathbf{y}_i) \equiv d^L(\mathbf{x}, \mathbf{y}_i) \geq d^{L-1}(\mathbf{x}, \mathbf{y}_i) \geq \dots \geq d^0(\mathbf{x}, \mathbf{y}_i) \quad (11)$$

여기서,

$$d^{l-1}(\mathbf{x}, \mathbf{y}_i) = \|\mathbf{y}_i^l\|^2 - 2 \sum_{k=1}^{2^{l-1} \times 2^{l-1}} x_k^l y_{ik}^l \quad (12)$$

식 (12)에서, $d^0(\mathbf{x}, \mathbf{y}_i) = \|\mathbf{y}_i\|^2 - 2x^0 y_i^0 = \|\mathbf{y}_i\|^2 - 2\|\mathbf{y}_i\| \cdot \|\mathbf{x}\|$ 이다.

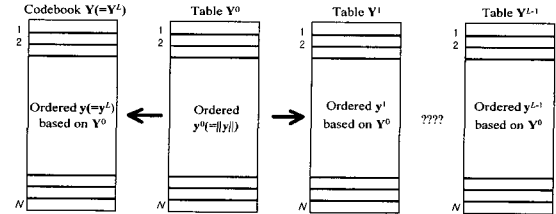


그림 2. $\|\mathbf{y}_i\|$ 의 정렬에 따른 부호어들 및 L_2 -평균 기반 다 단계 부호어들의 정렬. 모든 테이블들은 전처리 단계에서 생성된다

Fig. 2. Ordering of ordinary codewords and L_2 -norm-based multilevel codewords according to the ordering of $\|\mathbf{y}_i\|$. All the tables are prepared in the preprocessing step.

3. 제안한 기법

식 (11)에 기반한 새로운 VQ 부호화 기법을 제안한다. 그림 2에서 알 수 있듯이, 부호화하기 전에 부호책 내의 모든 부호어들의 L_2 -평균 값들의 올림차 순으로 저장한다. 부호책과 테이블의 원소들은 모두 Y^0 와 같은 순으로 정렬된다. 각 입력 벡터 x 에 대한 초기 최적 정합 부호어의 선택은 계산량을 감소시키는데 매우 중요하다. 제안한 기법에서는, 입력 벡터와 가장 가까운 L_2 -평균을 갖는 부호어 y_{ibm} 을 찾아 초기 시작점으로 삼는다. 이 때, Y^0 가 미리 올림차 순으로 저장되어 있으므로, 이진 탐색을 통해 $\log_2 N$ 단계 만에 가장 가까운 L_2 -평균을 갖는 부호어를 찾을 수 있어 전체 계산량을 줄일 수 있다. 따라서 d_{\min} 은 $d(x, y_{ibm})$ 이 된다.

효율적인 탐색을 위해 부호책 내의 부호어들은 그림 3과 같이 y_{ibm} 에서 출발하여 전후방 번갈아가며 탐색된다. 다음 후보인 y_{ibm+1} 이 y_{ibm} 보다 x 에 더 근사한지 아닌지를 알기 위해, L_2 -평균 피라미드의 최상위 계층부터 y_{ibm+1} 이 검사된다. 먼저, (x, y_{ibm+1}) 을 계산하고, d_{\min} 과 비교한다. $(x, y_{ibm+1}) < d_{\min}$ 이면, 식 (11)에 의해 $(x, y_{ibm+1}) < d_{\min}$ 이다. 따라서, y_{ibm+1} 는 최적

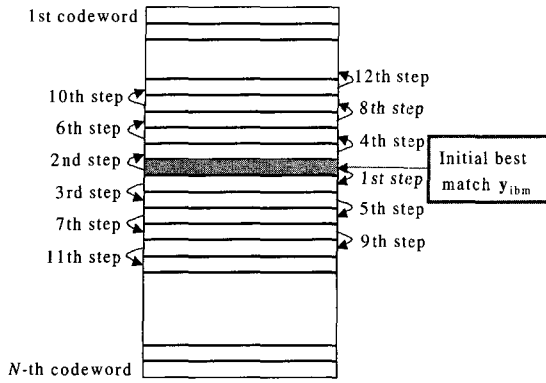


그림 3. 부호책 내 부호어 탐색 순서
Fig. 3. Codeword search order in the codebook.

근사 부호어가 될 수 없기 때문에 후보에서 제거될 수 있다. 반대로 $(\mathbf{x}, \mathbf{y}_{ibm+1}) < d_{min}$ 이면, $(\mathbf{x}, \mathbf{y}_{ibm+1})$ 을 계산한 후 d_{min} 과 다시 비교한다. $(\mathbf{x}, \mathbf{y}_{ibm+1}) \geq d_{min}$ 이면, 위와 비슷한 이유로 \mathbf{y}_{ibm+1} 을 후보에서 제거한다. 그렇지 않으면, 다음 세 번째 계층에서 검사는 계속된다. 이러한 과정은 \mathbf{y}_{ibm+1} 이 제거되거나 맨 아래 계층에 도달할 때까지 반복된다. 바다 계층에 도달하면, $(\mathbf{x}, \mathbf{y}_{ibm+1})$ 이 계산되어, d_{min} 과 비교된다. $(\mathbf{x}, \mathbf{y}_{ibm+1}) \geq d_{min}$ 이면, \mathbf{y}_{ibm+1} 는 후보에서 제거된다. $(\mathbf{x}, \mathbf{y}_{ibm+1}) < d_{min}$ 이면, 현재 최소치 d_{min} 이 $(\mathbf{x}, \mathbf{y}_{ibm+1})$ 로 갱신되고, 현재까지의 최적 정합 부호어는 \mathbf{y}_{ibm+1} 이 된다. 이런 과정이 그림 3의 탐색 순서에 따라 모든 부호어들에 대한 검사가 끝날 때까지 계속된다.

제안한 기법은 다음과 같이 요약될 수 있다.

- 1) 오프라인 처리: 테이블 Y^0 를 부호어들이 L_2 -평균의 올림차 순으로 정렬되도록 즉, $\|\mathbf{y}_1\| \leq \|\mathbf{y}_2\| \leq \dots \leq \|\mathbf{y}_N\|$ 이 되도록 구성한다. Y^0 와 같은 순서를 가지도록 테이블 Y^l 부터 Y^{L-1} 까지 구성한다.
- 2) 온라인 처리: 각 입력 벡터 \mathbf{x} 에 대해,
 1. $\|\mathbf{x}\|$ 과 \mathbf{x}^L 까지를 계산한다.
 2. \mathbf{x} 에 가장 가까운 L -평균을 갖는 초기 최적 부호어 \mathbf{y}_{ibm} 을 찾고, \mathbf{x} 와의 거리를 계산하여 초기 최소치 $d_{min} = d(\mathbf{x}, \mathbf{y}_{ibm})$ 로 한다. 이 과정의 계산량을 줄이기 위해, Y^0 가 올림차 순으로 정렬되어 있는 점을 이용하여, 이전 탐색을 통해 초기 최적 부호어를 $\log_2 N$ 단계 만에 찾는다. 한편, $R = \{\mathbf{y}_i | \mathbf{y}_i \in Y, \mathbf{y}_i \neq \mathbf{y}_{ibm}\}$ 이

라 하자.

3. R 이 공집합이면, 단계 8로 간다.
4. 그림 3의 탐색 순서에 따라 R 에서 하나의 후보 벡터 \mathbf{y}_i 를 선택한다.
5. $d^0(\mathbf{x}, \mathbf{y}_i) \geq d_{min}$ 이면, 단계 A 혹은 단계 B를 취한다. 그렇지 않으면, $l=1$ 로 한다.
 - A. $(\|\mathbf{y}_i\| \leq \|\mathbf{x}\|)$ 이면, R 에서 $i^{\circ}j$ 인 모든 \mathbf{y}_i 를 제거하고 단계 3으로 간다.
 - B. $(\|\mathbf{y}_i\| \leq \|\mathbf{x}\|)$ 이면, R 에서 $i^{\circ}j$ 인 모든 \mathbf{y}_i 를 제거하고 단계 3으로 간다.
6. $d^l(\mathbf{x}, \mathbf{y}_i)$ 를 계산한다. $d^l(\mathbf{x}, \mathbf{y}_i) \geq d_{min}$ 이면, R 에서 \mathbf{y}_i 를 제거하고 단계 3으로 간다. 그렇지 않으면, $l=l+1$ 로 한다. $l=L$ 이면, 단계 7로 간다. 그렇지 않으면, 단계 6을 반복한다.
7. $d^0(\mathbf{x}, \mathbf{y}_i)$ 즉, $d(\mathbf{x}, \mathbf{y}_i)$ 를 계산한다. $d(\mathbf{x}, \mathbf{y}_i) < d_{min}$ 이면, $d_{min} = d(\mathbf{x}, \mathbf{y}_i)$. R 에서 \mathbf{y}_i 를 제거하고 단계 3으로 간다.
8. d_{min} 에 대응하는 부호어를 최적 정합 부호어 \mathbf{y}_{bm} 으로 선택한다.

IV. 모의 실험 결과

1. 실험 조건

제안한 기법의 성능을 최근 개발된 다른 고속 기법들과 비교 평가하기 위해 여러 가지 영상들에 대해 모의 실험을 수행하였다. 실험은 SUN ULTRA1 워크스테이션에서 이루어졌다. 실험에 사용된 실험 영상 (test image) 및 학습 영상 (training image)은 모두 512×512 크기의 밝기 (luminance) 영상이다. 각 영상은 부호화에 앞서 겹치지 않는 4×4 블록들로 나뉜다. 따라서, 각 블록은 16차원 벡터가 된다 ($K=16; L=2$). 잘 알려진 LBG 기법^[2]이 부호책을 설계하기 위해 사용된다. Lena 영상이 부호책 설계를 위한 학습 영상으로 사용되며, 설계된 부호책이 네 개의 실험 영상들인 Peppers, Jet, Baboon, Lena를 부호화하기 위해 사용된다. 제안한 기법은 최근에 개발된 두 고속 기법들과 비교된다. 하나는 MTIE^[12]이고, 다른 하나는 Wu와 Lin이 개발한 기법이다^[14]. MTIE는 지금까지 알려진 삼각형 부등식 기반 기법들^[7~13] 중 가장 성능이 좋은 기법이다. Wu와 Lin의 기법은 식 (8)과 (9)를 이용한 고속 VQ부호화

기법이며, 제안한 기법은 Wu와 Lin이 제안한 기법의 일반화된 형태이다. 제안한 기법은 물론 비교 대상 기법들 모두 FSA와 동일한 부호화 성능을 가짐을 주목하라. 본 논문에서는 두 가지 종류의 실험들이 수행되었다. 첫번째 실험에서는 실험 영상 당 CPU 실행 시간을 비교하였고, 두 번째 실험에서는 화소 당 평균 연산 수를 비교하였다^[11,12]. 이 때, 입력 벡터들의 L_2 -평균 피라미드를 구하기 위한 연산량들도 모든 실험 데이터에 포함된다.

2. 성능 평가

표 I은 부호책 크기 (M)에 따른 네 개의 실험 영상들에 대한 CPU 실행 시간을 보인다. 256, 512, 1024 등 세 개의 N 에 대해 실험하였다. FSA에 대한 상대적인 시간 복잡도를 계산하여 다른 기법들과 비교하였으며, 그 비교 척도는 다음과 같이 정의된다.

$$\frac{T_{COMP}}{T_{FSA}} \times 100 (\%) \tag{13}$$

여기서, T_{FSA} 과 T_{COMP} 는 각각 FSA과 비교되는 기법의 CPU 실행 시간을 가리킨다. 표 I에서 부호책 크기와 상관없이, 제안한 기법은 모든 실험 영상들에 대해 기존의 기법들보다 빠름을 알 수 있다. 최악의 경우인 *Baboon*영상에 대해서도, 제안한 기법은 다른 고속 기법들보다 2배 가량 빠르다. 한편, 부호책 크기가 증가함에 따라, 제안한 기법의 효율성이 다른 기법들보다 향

상됨을 알 수 있다. 예를 들어 N 이 1024일 때 *Pepper*와 *Jet* 영상들에서, 제안한 기법은 Wu와 Lin의 기법보다 세 배 이상 빠르다. 이것은 부호책의 크기가 커짐에 따라 일반적으로 연산량은 훨씬 많이 요구되기 때문에 이런 성향은 제안한 기법의 큰 장점이 된다.

표 2. N 이 256일때 화소 당 연산량 비교.

Table 2. Comparison of computation complexity per pixel when codebook size N is equal to 256.

Test images	Algorithms	The average number of operations				
		×	±	CMP	SQRT	Total
Lena	Wu and Lin	19.8	21.5	4.9	0.063	46.3
	MTIE	12.6	26.0	5.68	0.34	44.6
	Proposed	9.4	12.2	4.99	0.31	26.9
Pepper	Wu and Lin	22.1	23.9	5.28	0.063	51.3
	MTIE	13.3	27.1	6.07	0.33	46.8
	Proposed	9.3	12.3	5.36	0.31	29.3
Jet	Wu and Lin	16.5	17.9	4.20	0.063	38.7
	MTIE	11.5	23.6	4.83	0.31	40.2
	Proposed	7.93	10.2	4.23	0.31	22.7
Baboon	Wu and Lin	58.8	62.8	12.2	0.063	133.9
	MTIE	43.7	86.8	14.8	0.40	145.7
	Proposed	32.0	37.4	13.0	0.31	82.7
FSA		256	510	16	.	782

표 1. 상대적 CPU 수행시간 비교.

Table 1. Comparison of the relative CPU running time.

Test images	Codebook size	Algorithms		
		Wu and Lin	MTIE	Proposed
Lena	N=256	6.4 %	6.3 %	3.3 %
	N=512	5.5 %	4.3 %	2.1 %
	N=1024	4.5 %	3.1 %	1.5 %
Pepper	N=256	7.3 %	6.7 %	3.3 %
	N=512	6.5 %	4.9 %	2.7 %
	N=1024	6.0 %	3.9 %	1.9 %
Jet	N=256	5.1 %	5.1 %	2.6 %
	N=512	4.2 %	3.8 %	1.7 %
	N=1024	4.0 %	3.4 %	1.3 %
Baboon	N=256	17.3 %	18.7 %	8.9 %
	N=512	16.2 %	16.5 %	7.2 %
	N=1024	16.1 %	16.1 %	6.6 %

표 II는 화소 당 평균 연산 수 관점에서의 여러 고속 기법들의 연산량을 보여 준다. 이 실험은 가장 보편적인 부호책 크기 $N=256$ 인 경우에 대해서만 수행하였다. 표 II는 제안한 기법이 두 비교 대상 기법들보다 역시 우수하다는 사실을 증명해준다. 예를 들어 *Jet*영상에 대해, 제안한 기법의 연산량은 MTIE의 58.7%, Wu와 Lin의 기법의 56.5%에 불과하다.

표 I과 II에서 알 수 있듯이, 제안한 기법은 성능저하 없이 VQ부호화의 속도를 상당히 향상시킬 수 있을 뿐만 아니라, 기존의 탐색 기법들보다 우수한 성능을 가진다. 위 실험들에서는 거리 계산 시 연산량을 줄이기 위한 방식으로 널리 알려진 PDE를 사용하지 않았다. PDE를 사용할 경우 추가적인 성능 향상을 얻을 수 있으므로, 위 실험 결과들은 상한 수행 시간 혹은 상한 연산량으로 볼 수 있다.

V. 결론

본 논문은 VQ 부호화를 위한 새로운 고속 탐색 기법을 제안하였다. 먼저, L_2 -평균 피라미드 데이터 구조 기반 Cauchy-Schwarz 부등식을 이용한 강력한 후보 제거 조건을 유도한다. 모든 부호어들의 L_2 -평균 피라미드를 미리 구하여 저장한 다음, 제안한 제거 조건을 이용하여 불필요한 후보 부호어들을 탐색 과정에서 제거한다. 따라서, VQ부호화 과정의 연산량을 대폭 줄일 수 있다. 모의 실험 결과를 통해 제안한 기법이 FSA와 동일한 부호화 성능을 보이면서 속도를 크게 향상시킴을 보였다. 또한 제안한 기법이 기존의 탐색 기법들보다도 2~3배 빠른 속도를 가짐을 보였다.

부 록

식 (10)의 증명은 다음과 같다.

계층 l 에서 두 벡터 x, y 의 내적은

$$\mathbf{x}^l \cdot \mathbf{y}^l = \sum_{k=1}^{2^l \times 2^l} x_k^l y_k^l = \sum_{k=1}^{2^{l-1} \times 2^{l-1}} (x_{4k}^l y_{4k}^l + x_{4k-1}^l y_{4k-1}^l + x_{4k-2}^l y_{4k-2}^l + x_{4k-3}^l y_{4k-3}^l) \quad (\text{A.1})$$

이다. 한편, 1차원 관점에서 $x_k^{l-1} = \sqrt{\sum_{p=0}^3 x_{4k-p}^l}$ 이기 때문에, 계층 $l-1$ 에서 두 벡터의 내적은

$$\mathbf{x}^{l-1} \cdot \mathbf{y}^{l-1} = \sum_{k=1}^{2^{l-1} \times 2^{l-1}} x_k^{l-1} y_k^{l-1} = \sum_{k=1}^{2^{l-1} \times 2^{l-1}} \left\{ \sqrt{(x_{4k}^l)^2 + (x_{4k-1}^l)^2 + (x_{4k-2}^l)^2 + (x_{4k-3}^l)^2} \times \sqrt{(y_{4k}^l)^2 + (y_{4k-1}^l)^2 + (y_{4k-2}^l)^2 + (y_{4k-3}^l)^2} \right\}. \quad (\text{A.2})$$

이다. 식 (A.1)과 (A.2), 그리고 Cauchy-Schwarz 부등식으로부터,

$$\sum_{k=1}^{2^{l-1} \times 2^{l-1}} (x_{4k}^l y_{4k}^l + x_{4k-1}^l y_{4k-1}^l + x_{4k-2}^l y_{4k-2}^l + x_{4k-3}^l y_{4k-3}^l) \leq \sqrt{\sum_{k=1}^{2^{l-1} \times 2^{l-1}} \left\{ (x_{4k}^l)^2 + (x_{4k-1}^l)^2 + (x_{4k-2}^l)^2 + (x_{4k-3}^l)^2 \right\}} \cdot \sqrt{\sum_{k=1}^{2^{l-1} \times 2^{l-1}} \left\{ (y_{4k}^l)^2 + (y_{4k-1}^l)^2 + (y_{4k-2}^l)^2 + (y_{4k-3}^l)^2 \right\}}. \quad (\text{A.3})$$

즉,

$$\sum_{k=1}^{2^l \times 2^l} x_k^l y_k^l \leq \sum_{k=1}^{2^{l-1} \times 2^{l-1}} x_k^{l-1} y_k^{l-1} \quad (\text{A.4})$$

을 의미한다. 따라서, 식 (A.4)로부터 다음 관계를 유도할 수 있다:

$$\sum_{k=1}^K x_k y_k \equiv \sum_{k=1}^{2^l \times 2^l} x_k^l y_k^l \leq \sum_{k=1}^{2^{l-1} \times 2^{l-1}} x_k^{l-1} y_k^{l-1} \leq \dots \leq \sum_{k=1}^{2 \times 2} x_k^1 y_k^1 \leq x^0 y^0 \equiv \|\mathbf{x}\| \cdot \|\mathbf{y}\| \quad (\text{A.5})$$

참 고 문 헌

- [1] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Norwell, MA: Kluwer Academic Press, 1992.
- [2] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, Vol. COM-28, No. 1, pp. 84-95, Jan. 1980.
- [3] R. M. Gray, "Vector quantization," *IEEE Acoust., Speech, Signal Process. Mag.*, pp. 4-29, April 1984.
- [4] W. H. Equitz, "A new vector quantization clustering algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. 37, No. 10, pp. 1568-1575, Oct. 1989.
- [5] H. Q. Cao and W. Li, "A fast search algorithm for vector quantization using a directed graph," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 10, No. 4, pp. 585-593, June 2000.
- [6] C. D. Bei and R. M. Gray, "An improvement of the minimum distortion encoding algorithm for vector quantization," *IEEE Trans. Commun.*, Vol. COM-33, No. 10, pp. 1132-1133, Oct. 1985.
- [7] M. T. Orchard, "A fast nearest neighbor search algorithm," in *Proc. IEEE ICASSP*, pp. 2297-2300, 1991.
- [8] C.-M. Huang, Q. Bi, G. S. Stiles, and R. W. Harris, "Fast full search equivalent encoding algorithms for image compression using vector quantization," *IEEE Trans. Image Processing*, Vol. 1, No. 3, pp. 413-416, July 1992.
- [9] S. W. Ra and J. K. Kim, "A fast mean-distance-oriented partial codebook search algorithm for image vector quantization," *IEEE*

- Trans Circuits Syst. II: Analog and Digital Signal Processing*, Vol. 40, No. 9, pp. 576-579, Sept. 1993.
- [10] W. Li and E. Salari, "A fast vector quantization encoding method for image compression," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 5, No. 2, pp. 119-123, April 1995.
- [11] T. -S. Chen and C. -C. Chang, "Diagonal axes method (DAM): a fast search algorithm for vector quantization," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 7, No. 3, pp. 555-559, June 1997.
- [12] C. -H. Hsieh and Y. -J. Liu, "Fast search algorithms for vector quantization of images using multiple triangle inequalities and wavelet transform," *IEEE Trans. Image Processing*, Vol. 9, No. 3, pp. 321-328, March 2000.
- [13] M. H. Johnson, R. E. Ladner, and E. A. Riskin, "Fast nearest neighbor search of entropy-constrained vector quantization," *IEEE Trans. Image Processing*, Vol. 9, No. 8, pp. 1435-1437, August 2000.
- [14] 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.
- [15] P. J. Burt and E. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. Commun.*, Vol. COM-31, No. 4, pp. 532-540, April 1983.

 저 자 소 개

宋 秉 哲(正會員)

1972년 11월 8일생. 1994년 2월 한국과학기술원 전기 및 전자공학과 공학사. 1996년 2월 한국과학기술원 전기 및 전자공학과 공학석사. 2001년 2월 한국과학기술원 전기 및 전자공학과 공학박사. 2001년 3월~현재 삼성전자 디지털미디어연구소 책임연구원. <주관심분야> 영상 압축, 영상 표현, 영상 탐색.

羅 鍾 範(正會員)

1953년 5월 23일생. 1975년 2월 서울대학교 전자공학과 공학사. 1977년 2월 한국과학기술원 전기 및 전자공학과 공학석사. 1983년 2월 한국과학기술원 전기 및 전자공학과 공학박사. 1983년 7월~1987년 6월 미국 Columbia 대학교 연구 조교수. 1987년 7월~현재 한국과학기술원 전기 및 전자공학과 교수. <주관심분야> 디지털 영상처리, 동영상 신호처리, 3차원 시각화, 의료 영상 시스템 등.