논문 2004-41SP-4-8

고속 탐색 방법에 의한 부호책 생성 알고리즘

(Codebook Generation Algorithm Using Fast Searching Method)

김 형 철*, 조 제 황**

(Hyeung-Cheol Kim and Je-Hwang Jo)

요 약

고속 부호책 생성 방법들로서 기존의 대표적인 기법들에는 PDS, FNNS, 그리고 FC가 있다. 본 논문에서는 코드북을 설계하기 위해 기존의 방법들을 통합한 FCNNPDS를 제안한다. 모의실험 결과 FCNNPDS의 계산양이 기존의 방법들보다 약 40-95% 감소되는 것을 보인다. 그러나, 비교 계산은 벡터의 차원 k와 관계가 없으며, 이것은 비교의 계산양이 다른 계산들보다 훨씬 적어지는 이유이다, 그래서 FCNNPDS는 기존의 방법들보다 가장 좋은 방법이다.

Abstract

The conventional typical techniques as fast codebook generation methods are PDS, FNNS, and FC. In this paper, we propose FCNNPDS integrated the conventional methods to generate a codebook. The results of simulations show that the computational magnitude of FCNNPDS is reduced to 40-95% lower than conventional techniques. But comparison computation has no relation with k dimension of vectors, that is, because the computational magnitude of comparison is smaller than others, therefore FCNNPDS may be the best method than the conventional methods.

Keywords : codebook, PDS, FNNS, FC, FNNPDS

I. 서 론

벡터양자화는 우수한 압축비율과 비교적 단순한 구조 때문에 큰 주목을 받아왔다. 벡터양자화는 입력신호의 각 벡터들은 블록으로 나누어진 후 부호벡터들로 구성된 부호책과 비교되고, 입력신호벡터와 가장 유사한 부호벡터의 인덱스를 전송한다. 수신부에는 동일한 부호책이 존재하며, 수신된 인덱스에 해당하는 부호벡터로 입력신호를 복원하는 단순한 구조를 갖는다[1][2]. 샤논의 왜곡률(rate-distortion) 이론에 따르면 스칼라 대신에 벡터들을 부호화함으로써 더 좋은 성능을 얻을 수 있으며, 높은 차원을 갖는 벡터를 양자화함으로써 압축율을 높일 수 있다. 그러나, 실제 알고리즘에서는 벡터 차원과 함께 지수적으로 증가하는 계산양과 메모리 용량에 의해 제한을 받게 된다. 부호화 단계에서는 각각의 입력신호벡터에 가장 유사한 부호책의부호벡터를 찾아야 하는데, 기존의 FS 방법(full search)은 입력신호벡터와 부호책의 모든 부호벡터를 탐색하는 것을 요구하므로, 많은 연산 횟수를 필요로 한다. 이러한 계산양의 문제를 피하기 위해 많은 대안들로 PDS(partial distance search), FNNS(fast nearest neighbor search), ENNS(equal-average nearest neighbor search), ENNSV(equal-average nearest neighbor search with variance) 등이 제시되어 왔으며, 그 결과 계산 양의 감소를 가져왔다[3]-[6].

현재 벡터양자화에서 사용되는 부호책을 구하는 방법 중에서 가장 널리 쓰이는 것은 K-means 알고리즘이다[7]. 이 알고리즘은 소속 클래스의 학습벡터와 부호벡터 사이의 오차를 학습반복을 통해 줄여가는 것으로,

* 정회원, 동신대학교 전기전자공학과
(Dept. of Electrical & Electronic Eng., Dongshin University)
** 정회원, 동신대학교 정보통신공학부
(Dept. of Information & Communication Eng., Dongshin University)
※ 이 논문은 2003학년도 동신대학교 학술연구비에 의하여 연구되었음.
접수일자: 2004년4월16일, 수정완료일: 2004년6월4일

학습의 각 반복단계에서 학습벡터의 소속 클래스를 결정하기 위해 FS 방법을 사용하기 때문에 이에 따른 시간이 많이 소요된다는 것이 단점이다.

본 논문에서는 학습벡터의 소속 클래스를 고속으로 결정하기 위해 학습벡터와 부호벡터 간의 거리 계산 과정에서 차원마다 비교하여 도중에 중단하는 PDS 방법과, 삼각부등식을 적용하여 학습벡터와 부호벡터간의 거리계산을 위해 일부 부호벡터를 선택하여 계산하는 FNNS 방법, 그리고, 전후단계 학습반복에서의 부호벡터와의 거리 비교를 기준으로 나머지 부호벡터와의 계산을 생략하는 FC(fast clustering) 방법[8]을 모두 통합하여 부호책을 구할 때 전체 계산양을 줄일 수 있는 FCNNPDS(fast clustering nearest neighbor partial distance search) 방법을 제안한다.

Ⅱ. 기존의 고속 탐색 방법

1. FS 방법

FS 방법은 부호책을 구하는 학습과정에서 학습벡터의 소속 결정과 부호화 과정에서 입력신호벡터에 가장 유사한 부호벡터를 찾기 위해 사용된다. 학습벡터의 소속 결정에서의 예를 들면, 이 방법은 유클리드 거리에 기반을 둔 기존의 최단거리 이웃조건을 이용하여 임의의 학습벡터에 대해 모든 부호벡터를 전체 탐색하는 방법으로 계산양을 알아보기 위해 다음과 같이 가정한다. 학습벡터의 집합 $X = \{X_O ; 1 \le O \le N\}$ 는 k 차원을 갖고, 동일 차원을 가지며 부호벡터의 크기가 L 인 부호책을 $Y = \{Y_I ; 1 \le I \le L\}$ 이라 하면, 부호벡터는 $Y_I = \{Y_{I1}, Y_{I2}, \cdots, Y_{Ik}\}$ 로 나타낼 수 있다. 이때, 각 학습벡터에 가장 가까운 부호벡터를 찾기 위해 유클리드 거리를 사용하는 경우, 거리 계산을 위한 수식은 다음과 같이 정의한다.

$$d(X_O, Y_I) = \sum_{p=1}^{k} (X_{Op} - Y_{Ip})^2 \qquad (1)$$

식 (1)은 하나의 학습벡터와 하나의 부호벡터간의 거리 계산을 나타낸 것으로 한번의 학습과정에서 필요한 전체 계산양은, kLN 번의 곱셈과 뺄셈, $(k-1)LN$ 번의 덧셈, 그리고 $(L-1)N$ 번의 비교 계산이 요구된다. 본 논문에서 사용되는 영상에 대해 k=16, L=256, N=16,384 이므로 전체 계산양은 각각 67,108,864번의 곱셈과 뺄셈, 62,914,560번의 덧셈, 그리고 4,177,920번의 비교 계산이 요구되며, 부호책을 구하는 과정에서의 전체 계산양은 위의 각 계산양에 학습반복 횟수 m 을 곱한 만큼의 계산이 필요하게 된다.

2. PDS 방법

PDS 방법은 FS 방법과 동일한 과정에서 계산양을 줄이기 위해 사용된다. 임의의 학습벡터와 부호벡터간의 거리를 정해진 순서대로 계산할 때, 첫 번째 구해진 거리의 값을 $d_{\min}$ 이라 하고, 다음 부호벡터와의 거리를 계산할 때, 위의 식 (1)과 같이 k 차원까지의 거리를 전부 계산하지 않고, 차원이 하나씩 증가할 때 마다 $d_{\min}$ 과 비교하여 이 값보다 큰 값이 나오면, 즉시 계산을 멈추고 다음 부호벡터와의 거리 계산으로 넘어가게 된다. 만약 k 차원까지 구해진 거리의 값이 $d_{\min}$ 보다 작게 되면, 이 값을 새로운 $d_{\min}$ 으로 갱신하게 된다. 위와 같은 과정을 반복적으로 수행한 후, 더 이상 새로운 $d_{\min}$ 이 갱신되지 않으면, 그 때의 부호벡터를 현재 학습벡터의 가장 가까운 부호벡터로 결정한다. 이 방법은 FS 방법과 같이 전체 부호벡터를 탐색하지만, 거리 계산은 부호벡터에 따라 부분적으로 하게 되므로 FS 방법보다 계산양을 줄일 수 있고, 이 방법으로 구한 부호책은 FS 방법에 의해 생성된 부호책의 성능과 동일한 결과를 갖는다.

3. FNNS 방법

FNNS 방법은 삼각 부등식을 이용한 대표적인 방법으로 PDS 방법과 달리 부호벡터 간의 거리를 기억해야 하므로 추가적인 메모리가 요구된다. 즉 부호벡터가 L 개인 경우, 각 부호벡터 간의 거리를 나타내는 $L-1$ 개의 거리를 기억해야하는 전처리 과정을 갖는다. FNNS 방법과 본질적으로 동일하지만, 제곱근 계산이 들어가지 않으면서 학습벡터와 부호벡터의 평균을 이용하는 ENNS 방법, 학습벡터와 부호벡터의 평균 이외에 분산을 이용하는 ENNSV 방법 등이 제시되었다. 이러한 방법은 FNNS 방법보다 계산양을 줄일 수 있으나, 추가적인 메모리 부담이 증가하는 단점이 있다.

FNNS 방법의 절차는 다음과 같다. 학습벡터 X_O 에 가장 가까운 부호벡터를 Y_I 라 하고, 학습벡터와 부호벡터 간의 거리 $d(X, Y_I)$ 를 $d_{\min,I}$ 라 하자. 이젠 부호벡터들 간의 거리 $d(Y_I, Y_J)$ 를 구하여 $2d_{\min,I}$ 보다 큰 부호벡터는 학습벡터 X_O 와의 거리 계산에서

제외하고, $2d_{\min,I}$보다 거리계산 값이 작은 부호벡터만을 대상으로 거리를 계산하여 부호벡터 Y_J와의 거리 $d_{\min,J}$가 최소 값을 갖는 경우 학습벡터 X_O는 J번째 클래스 C_J에 소속되는 것으로 결정한다. 부호벡터를 선택하기 위한 기준은 그림 1과 같다.

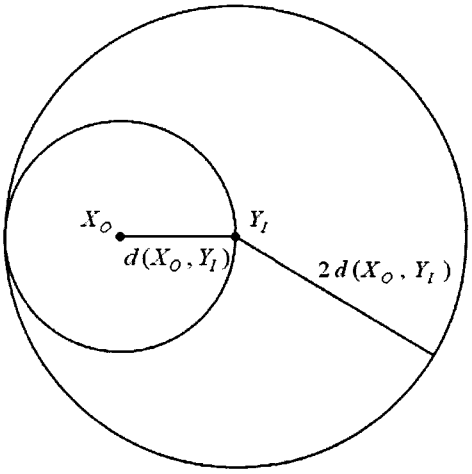

그림 1. 부호벡터를 선택하기 위한 기준
Fig. 1. Criterion to select codevectors.

그림에서 X_O가 중심이고, 거리 $d(X_O, Y_I)$가 반지름인 내부 원내에 있는 부호벡터는 Y_I보다 더 근접한 것으로써 X_O가 소속되는 클래스의 대표벡터가 될 수 있다. 따라서 이 원 내에서 부호벡터를 찾으면 되지만, X_O와 관련된 거리 계산을 해야 하므로 계산 양을 줄이려는 목적에 적합하지 않으므로 Y_I가 중심이고 $2d(X_O, Y_I)$이 반지름인 외부 원의 밖에 위치하는 경우 즉,

$$2d(X_O, Y_I) < d(Y_I, Y_J) \tag{2}$$

식 (2)를 만족하는 부호벡터를 거리 계산에서 제외함으로써 계산양을 줄일 수 있다.

4. FC 방법

FC 방법은 학습반복이 진행될수록 학습벡터가 소속되는 클래스의 이동이 적어진다는 점을 이용한 것으로, 이전 학습반복에서 특정 클래스에 소속된 학습벡터는 다음 학습반복에서 동일한 클래스에 대부분 소속된다는 성질에 기반을 둔 것이다. 식 (3)은 거리계산을 줄이기 위한 고속 소속 결정 조건이다.

$$d(X_O, Y_I^{m-1}) > d(X_O, Y_I^{m}) \tag{3}$$

여기서, $m-1$번째와 m번째 학습반복에서의 I번째 부호벡터는 각각 Y_I^{m-1}와 Y_I^{m}이고, O번째 학습벡터는 X_O이다. 식 (3)을 만족하는 학습벡터 X_O는 $m+1$번째 학습반복에서 소속 결정을 위해 모든 부호벡터와의 거리 계산을 하지 않고, I번째 클래스 C_I^{m+1}에 속하는 것으로 결정한다.

III. 제안된 고속 탐색 방법

벡터양자화의 부호화단계에서 학습벡터의 소속 결정을 고속으로 이루고자하는 대표적인 방법들을 II장에서 살펴보았다. PDS 방법은 모든 부호벡터를 탐색하되 부호벡터의 전체 성분 중에서 일부만을 계산함으로써 계산양을 줄일 수 있고, FNNS 방법은 부호벡터의 일부만을 계산함으로써 계산양의 부담을 줄이는 방법이다. 또한 FC 방법은 학습반복 전후 단계의 부호벡터를 고려하여 식 (3)의 조건을 만족하는 특정 부호벡터와의 비교 계산을 생략함으로써 계산양을 줄이는 방법이다.

본 논문에서는 이 세 가지 방법을 효과적으로 결합하여 부호책 생성시의 전체 계산양을 줄이는 방법을 제안하고자 한다. 여기에서 제안된 방법은 FC와 FNNS, 그리고 PDS를 결합한 방법으로써 FCNNPDS 방법이라 명명한다.

다음은 m번째 학습반복에서의 FCNNPDS 방법에 대해 설명한다.

1. if $d(X_O, Y_I^{m-1}) > d(X_O, Y_I^{m})$,
   then $X_O \in C_I^{m+1}$,
   and go to step 1 for other pair (X, Y)
2. else $d_{\min} = d(X_O, Y_I^{m})$
3. if $d(Y_I^{m}, Y_J^{m}) > 2d_{\min}$, then $J = J+1$
4. if $J \le L$, then go to step 3
5. else $(I = I+1) \neq$ all J in step 3 ,
   go to step 1
6. else if $\sum_{p=1}^{p \le k} (X_{Op} - Y_{Jp}^{m})^2$,
   then stop calculation and go to step 3
7. else $d_{\min} = \sum_{p=1}^{k} (X_{Op} - Y_{Jp}^{m})^2$
   and go to step 3

Ⅳ. 실험 및 결과

본 실험에서는 512×512 픽셀의 256 그레이 레벨을 갖는 Lena 영상을 사용하였고, 부호책을 생성하기 위해 영상을 4×4 블록 단위로 나누어 16,384개의 학습벡터로 학습하였으며, 학습반복 횟수는 30회로 제한을 두었다. 초기 부호책은 splitting 방법으로 구하였으며, 부호책의 크기는 256으로 하여 학습반복에 대한 곱셈, 덧셈, 비교 시의 계산양을 구하였다.

그림 2, 3은 학습반복에 대한 FS, PDS, FNNS, FC, 그리고 제안된 FCNNPDS 방법의 곱셈, 덧셈의 계산양을 나타낸 것이다. 여기서 뺄셈 계산양은 곱셈 계산양과 동일하기 때문에 생략되었다. 그림 2와 3에서 알 수 있듯이 제안된 방법이 곱셈과 덧셈에서 다른 방법에 비해 우수하고 학습반복이 많을수록 더욱 유리함을 알 수 있다.

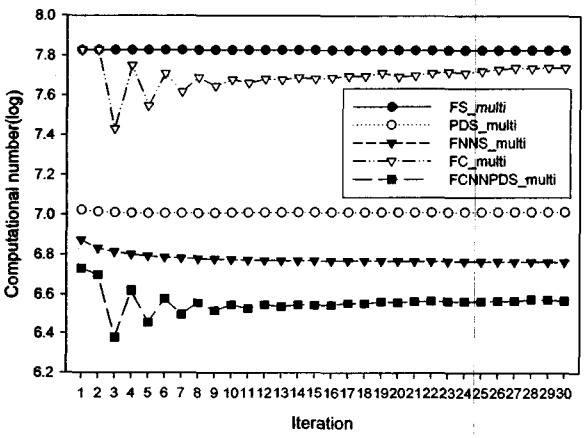

그림 2. 학습반복에 대한 곱셈 수
Fig. 2. The number of multiplications versus training iteration.

그림 4는 학습 반복에 대한 비교 계산량을 나타낸 것으로 FC방법이 제안된 FCNNPDS 방법보다 우수하지만, 곱셈이나 덧셈 연산과는 달리 벡터 성분을 의미하는 k와 무관하기 때문에 계산양이 월등히 적다.

Ⅴ. 결 론

벡터양자화에서 사용되는 부호책을 생성하는 과정에서 학습벡터의 소속 클래스 결정을 위한 방법으로는 FS 방법이 대표적이었으나, 이를 개선하기 위한 고속 탐색 방법인 PDS 방법, FNNS 방법, FC 방법 등이 제안되었다.

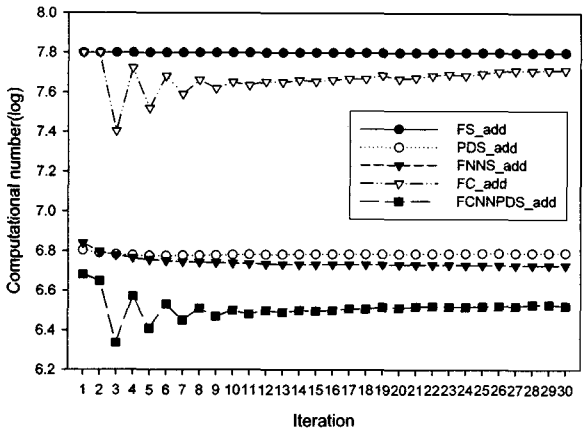

그림 3. 학습반복에 대한 덧셈 수
Fig. 3. The number of additions versus training iteration.

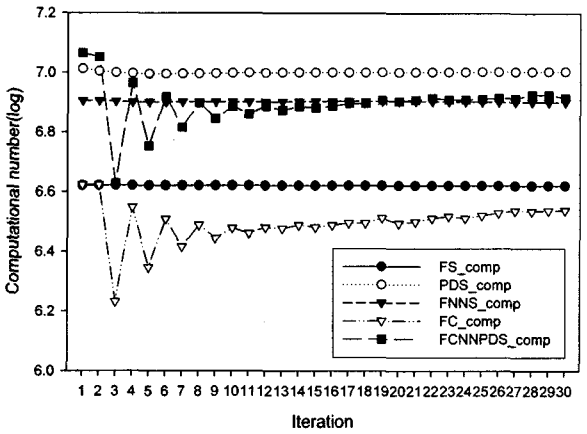

그림 4. 학습반복에 대한 비교 수
Fig. 4. The number of comparisons versus training iteration.

본 실험 결과에 의한 계산양을 분석해보면 곱셈, 뺄셈, 덧셈에서의 계산양은 FCNNPDS < FNNS < PDS < FC < FS 순서로 제안된 방법이 모든 학습반복에서 가장 적은 계산양을 나타냈으며, 비교 계산에서는 FC < FS < FNNS ≈ FCNNPDS < PDS 순서로 FC 방법이 가장 적은 계산양이 필요하였다. k차원으로 구성된 학습벡터 N개와 L개의 부호벡터를 갖는 경우, 곱셈(뺄셈), 덧셈에서는 k, N, L의 크기에 비례적으로 계산양이 증가하므로 영향을 많이 받지만, 비교 계산에서는 k의 크기에 무관하다. 또한, 비교 계산은 곱셈(뺄셈), 덧셈 계산에 비해 연산횟수가 크게 적어 전체 계산양에 미치는 부담이 작다. 따라서 제안된 FCNNPDS의 방법이 부호책을 생성하는 과정에서 학습벡터의 소속 클래스 결정시와 전체 연산횟수의 면에서 다른 방법에 비해 상당히 우수함을 알 수 있다.

참 고 문 헌

[1] R. M. Gray, "Vector quantization," IEEE ASSP Mag., vol. 1, pp. 4-29, April 1984.

[2] A. Gersho and R. M. Gray, Vector Quantization and Signal Compression, KAP, pp. 309-343, 1992.

[3] C. D. Bei and R. M. Gray, "An Improvement of the minimum distortion encoding algorithm for vector quantization," IEEE Trans. Commun., vol. COM-33, pp. 1132-1133, Oct. 1985.

[4] M. Orchard, "A fast nearest neighbor search algorithm," Proc. IEEE ICASSP, pp. 2297- 2300, May 1991.

[5] S. W. Ra and J. K. Kim, "A fast mean-distance-ordered partial codebook search algorithm for image vector quantization," IEEE Trans. Circuits Syst.II: Analog and Digital Signal Process., vol. 1, no. 40, pp. 576-579, Sept. 1993.

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

[7] 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.

[8] S. J. Baek, B. K. Jeon, D. R. Lee and K. M. Sung, "Fast clustering algorithm for vector quantization," Electronics Letters, vol. 34, no. 2, pp. 151-152, Jan. 1998.

저 자 소 개

김 형 철(정회원)
1997년 동신대학교 전자공학과 졸업(학사)
1999년 동신대학교 전기전자공학과 졸업(석사)
1999년~현재 동신대학교 전기전자공학과 박사과정
2004년~현재 (주)디셈 대표이사
<주관심분야: 영상처리, 패턴인식, 임베디드시스템>

조 제 황(정회원)
1984년 광운대학교 전자공학과 졸업(학사)
1986년 광운대학교 전자공학과 졸업(석사)
1990년 광운대학교 전자공학과 졸업(박사)
1989년~현재 동신대학교 정보통신공학부 교수
<주관심분야: 적응신호처리, 영상처리, 패턴인식, SoC>