

벡터量子化의 高速인코딩과 인덱스 無損失 壓縮

정희원 정 일 화*, 최 인 호**, 이 대 영***

A Fast Encoding and Index Lossless Compression of Vector Quantization

Il-hwa Jung*, In-ho Choi**, Dai-young Lee*** *Regular Members*

요 약

벡터 양자화는 압축효율에 비해 수신기의 디코딩 과정이 매우 간단하다는 장점을 지니는 방법이지만, 코드북의 크기를 줄여 압축률을 높이면 복원 영상의 화질, 특히 윤곽 부분의 화질이 저하되고, 화질 저하를 막기 위하여 코드북의 크기를 크게 하면 코드북 구성 시간, 부호화시간과 코드북 저장을 위한 기억장소의 증가 등의 문제점 또한 지니고 있는 방법이라 할 수 있다. 본 논문에서는 코드벡터의 부분 사용으로 검색영역을 축소하는 방법과 검색 중지 조건 등을 도입한 고속 인코딩 방법과, 전송되는 코드북 인덱스에 가변길이 부호화의 개념을 적용하는 인덱스 무손실 압축 방법을 제안하고자 한다.

ABSTRACT

Vector quantization has often been used for image compression because of its simple decoding process, however, this can increase the codebook size or vector dimension in order to increase coding efficiency, and it causes a number of problems such as, increased codebook generation time and encoding time, and large memory. The purpose of this paper is to propose a modified algorithm for fast encoding and index lossless coding method, the former includes algorithms to reduce the search space using partial codebooks, and to introduce exit conditions, the latter uses variable length coding for the transferred VQ indexes.

I. 서 론

21세기 정보화시대의 개막과 함께 우리의 통신환경은 매우 급속한 속도로 변화되고 있는 상황이며, 그 중심에 멀티미디어가 자리하고 있다고 할 수 있으며, 지금까지 멀티미디어 데이터의 처리기술은 디지털 신호처리, 디지털 통신, 반도체와 컴퓨터 등의 기술발전이 힘입어 발전되어 왔는데, 주문형 비디오(VOD), 디지털 라이브러리, 전자상거래와 의료 비디오 검색 시스템 등의 새로운 서비스 요구가 발생됨에 따라 더욱더 중요한 부분을 차지하고 있다고 볼 수 있다.

영상신호는 그 의미 전달 효과가 다른 형식의 데

이터에 비해 매우 크기 때문에 멀티미디어 데이터의 핵심이라고 할 수 있지만, 단위시간당 발생 또는 재생되는 데이터량이 매우 방대하여 저장 또는 전송을 위해 대용량의 저장매체와 넓은 대역폭의 전송채널이 요구되는 특성을 갖는다. 따라서 영상데이터 압축은 멀티미디어 서비스의 원활한 운용을 위해 해결해야 하는 핵심과제 중 하나라고 할 수 있다.

영상 압축 방법은 기본 형식에 의해 분류해 보면 크게 예측부호화(Predictive coding) 방식과 변환부호화(Transform coding) 방식으로 대별할 수 있는데, 예측부호화 방식은 영상 샘플이 갖는 상관관계를 이용하는 방법으로서, 시스템의 구성이 간단하고

* 신성대학 전자·정보통신계열(jungih@shinsung.ac.kr),

** 김포대학 전자·정보계열(inhochoi@kimpo.ac.kr)

*** 경희대학교 전자정보학부(dylee@khu.ac.kr)

논문번호 : 020221-0507, 접수일자 : 2002년 5월 7일

데이터의 무손실(lossless) 압축이 가능하지만 압축률의 제한이 따른다는 단점을 가지며, DPCM(Differential Pulse Code Modulation), RLC(Run Length Coding), VLC(Variable Length Coding) 등의 방법이 제안되어 있다.

변환부호화 방식은 영상 화소(pixel)간의 높은 상관도를 이용하는 방법으로 공간영역에서 주파수영역으로의 변환을 통해 상관도를 낮추고 에너지를 집중시킨 후에 높은 에너지 분포를 갖는 변환계수만을 부호화하는 방식으로 비교적 높은 압축률에서 좋은 성능을 나타내지만 시스템의 구성이 복잡해지는 특히 역변환이 이루어지는 수신측의 디코딩 과정이 다른 방법에 비해 복잡해지는 단점을 가지며, DCT(Discrete Cosine Transform), WT(Wavelet transform) 등을 이용하는 방법들이 제안되어 있다.

예측부호화와 변환부호화 방식에 존재하는 장단점을 고려하여 JPEG 표준안에서도 무손실 모드(lossless mode)에서는 예측부호화 방식만을 사용하고, 손실모드(lossy mode)에서는 DCT와 예측부호화를 동시에 사용하는 방식을 취하고 있다.^[14]

손실허용(lossy) 압축에서는 예측부호화와 변환부호화를 결합시켜 사용하는 방법이 주류를 이루고 있는데, JPEG 표준안 이외에도 웨이브렛 변환(Wavelet Transform)을 이용한 압축 방법, 프랙탈(Fractal) 압축 방법과 벡터양자화를 이용한 압축 방법 등의 연구가 활발하게 이루어지고 있으며, 이 중에서 벡터 양자화는 1980년 Y. Linde, A. Buzo, R. M. Gray 등에 의해 LBG 알고리즘이 제안된 이후로 많은 연구가 수행되었는데, 압축효율에 비해 시스템의 구조가 간단하다는 특히, 수신기의 디코딩 과정이 매우 간단하다는 장점을 갖는 방법이다.

벡터양자화는 코드북 구성(codebook generation), 인코딩(encoding), 디코딩(decoding)의 세 가지 단계로 구성되며, 기존의 다른 영상압축 방식과 마찬가지로 계산량, 압축률, 복원영상의 화질 등에 의해 성능이 평가되는데, 디코딩 과정의 경우 특별한 연산이 필요치 않으므로 코드북 구성과 인코딩에 의해 그 성능이 결정되어진다고 할 수 있다.

따라서 본 논문에서는 양자화 과정에서 복원영상의 화질저하가 심각하게 발생하지 않는다는 전제조건하에 코드북 구성과 인코딩 과정에서의 계산량을 줄이는 방법과, 이 때 저하된 화질을 보완하기 위한 방법, 인코딩 과정에 의해 선택되는 코드북 인덱스의 무손실 압축을 통해 시스템 전체의 압축률을 높일 수 있는 방법 등을 포함하여 벡터양자화 시스템

전반에 걸친 성능향상 기법에 대해 제안하고자 한다.

II. 벡터 양자화 알고리즘

벡터 양자화는 입력영상의 블록을 같은 크기를 갖는 블록들의 집합인 코드북의 블록들과 비교하여 가장 유사하다고 판단되는 블록으로 대체하는 과정에서 압축효과를 얻는 방법으로, 코드북의 크기에 따라 압축률, 계산량, 복원 영상의 화질 등에 변화가 있게 된다.

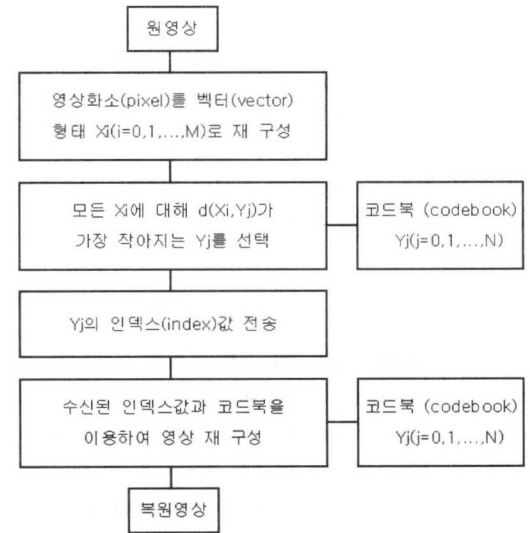


그림 1. 벡터양자화의 블록도

이와 같은 원리에 의해 구성된 벡터 양자화의 블록도가 그림 1인데, 입력벡터와 코드벡터와의 오차를 계산하기 위해서는 식(1)의 Euclidean distance가 사용되고, 복원영상의 화질을 판단하기 위해서는 식(2)의 MSE(Mean Squared Error)와 PSNR(Peak Signal-to-Noise Ratio)가 사용된다.

$$d(X, Y) = \|X - Y\|^2 = \frac{1}{n} \sum_{i=1}^n (X_i - Y_i)^2 \quad (1)$$

$$MSE = \frac{1}{M} \sum_{i=1}^M (x_i - y_i)^2 \quad (2)$$

$$PSNR = 10 \log \left[\frac{255^2}{MSE} \right] [dB]$$

(M: 화소수, x: 입력영상, y: 복원영상)

1. 코드북 구성과 인코딩을 위한 알고리즘

1-1. LBG(Lide-Buzo-Gray) 알고리즘

LBG 알고리즘은 1980년 Y. Linde, A. Buzo, R. M. Gray 등에 의해 처음 제안된 알고리즘으로 벡터양자화에서 가장 일반적으로 사용되는 방법이라 할 수 있는데, 코드북 구성이나 인코딩 과정에서 전역 검색(Full search)을 기본으로 하기 때문에 계산량이 상당히 많은 문제점을 지니게 되고, 이러한 문제점을 해결하기 위한 많은 알고리즘들이 제안되고 있다.

1-2. PDS(Partial Distortion Search) 알고리즘⁽⁵⁾

PDS 알고리즘은 입력벡터와 가장 오차가 작은 코드벡터를 선택하는 과정을 변형하여 계산량을 줄이는 방법으로, 코드벡터를 이루는 n차원(일반적으로 16차원)의 요소에 대해 일괄적으로 유클리드의 거리(Euclidean distance)를 계산하지 않고 이를 각 요소로 분리하여 판단하는 과정을 거치게 된다.

새로운 코드벡터 Yj와 Xi의 오차 d(Xi, Yj)를 일괄적으로 계산하지 않고, n개의 벡터요소에 대해 차례로 오차를 계산하여 누적시켜 진행하다가 그 값이 dMIN을 넘어서면 계산을 중단하고 다음 코드벡터로 이동하는 형식을 취하여 계산량을 줄이게 된다.

n개의 벡터요소 중 먼저 계산되는 요소들의 편차가 큰 경우에는 계산량의 감소가 이루어지지만, 그렇지 않을 경우에는 각 요소마다 판단에 필요한 연산이 추가되므로 계산량 감소의 효과가 급격히 줄어들게 된다.

1-3. MPS(Mean-distance-ordered Partial code-book Search) 알고리즘⁽⁶⁾

MPS 알고리즘은 코드북을 평균값에 의해 정렬한 후 수행되며, 검색단계에서 현재까지 결정된 가장 유사한 코드벡터가 YMIN 이라고 가정하면, 코드벡터 Yi 가 식(3)을 만족할 경우에만 d(X,Yi)를 계산하는 방식을 취한다.

$$d_M(X, Y) \leq m \times d(X, Y) \quad (3)$$

(m : 벡터의차원)

이 때 $\sum_{i=1}^m Y_i$ 는 한정된 수의 코드벡터에 대해 미리 계산할 수 있는 값이므로 코드북 구성, 인코딩시의 계산량을 줄일 수 있게 된다.

1-4. Kick-out condition(proposed by Wu and Lin) 알고리즘⁽⁷⁾

Wu와 Lin에 의해 제안된 Kick-out condition은 다음과 같이 d(X,Yi)를 식(4)에서와 같이 전개하여 사용하는데, 식(4)에서 $\sum_{i=1}^m X_i^2$ 는 공통 포함된 항이므로 전체 계산량에 영향을 주지 않게 되고, $\sum_{i=1}^m Y_i$ 는 한정된 코드북에 관한 수치이고 미리 전체적으로 계산해 놓으면 실제의 인코딩 과정에서는 계산이 불필요하게 되므로 전체적인 계산량을 줄일 수 있다.

$$d(X, Y) = \sum_{i=1}^m (X_i - Y_i)^2 \quad (4)$$

$$= \sum_{i=1}^m X_i^2 + \sum_{i=1}^m Y_i^2 - 2 \sum_{i=1}^m X_i Y_i$$

2. 인덱스의 무손실 압축을 위한 알고리즘

2-1. DPCM/Huffman coding⁽⁸⁾

정지영상 압축의 표준방식인 JPEG에는 베이스라인 모드(baseline mode), 확장 모드(extended mode), 무손실 모드(lossless mode), 계층 모드(hierarchical mode) 등의 여러 가지 동작모드가 제시되어 있는데, 여기서 무손실 모드는 예측 부호화와 가변길이 부호화 등을 이용하여 무손실 압축을 행하는데, 이를 인코딩 과정에서 선택된 코드벡터의 인덱스에 적용시키는 방법을 그림 2에 나타냈는데, 인덱스 X에 대한 예측치 Y를 계산한 후 X와 Y의 오차를 허프만 테이블을 이용하여 무손실 압축을 행하게 된다.⁽⁹⁾

2-2. Search-Order Coding (SOC)⁽¹⁰⁾

벡터양자화에서 영상은 연속되는 블록 단위로 처리가 이루어지고 이 때의 블록의 크기는 전체 영역과 비교해 보면 매우 작은 영역이라 할 수 있으므로, 이들 블록사이에는 높은 상관관계가 존재하게 되고 이들 블록들과 최소오차를 갖는 코드북 인덱스를 결정하였을 때에도 인덱스 사이의 상관도 또한 높다고 생각할 수 있다.

따라서 Search-Order coding에서도 DPCM/Huffman coding에서와 같이 인접영역의 코드북 인덱스간의 상관관계가 높은 것을 이용하여 전송되는 인덱스의 무손실 압축을 행하게 되는데, 그림 3에서와 같은 검색경로에서 같은 인덱스가 발견되면 그 위치만을 전송하고, 발견하지 못하면 원래의 인덱스를 그대로 전송하는 방식을 사용하여 전송되는 데

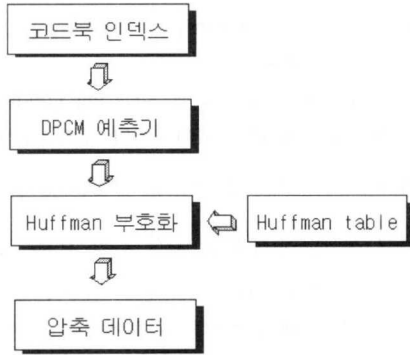


그림 2. JPEG 무손실 모드를 이용한 인덱스의 압축

이터량을 줄일 수 있게 된다.

이 때에 검색경로를 표현하는 비트를 크게 하면, 즉 검색경로를 길게 하면 같은 값을 갖는 인덱스를 검색할 확률은 높아지지만 검색경로 표현을 위한 비트가 늘어나 있는 상태이므로 경로가 짧을 때와 비교하여 데이터량이 반드시 줄어들지는 않는다고 볼 수 있다.

	01	10	11	
	00			

	110	111		
	101	001	010	011
	100	000		

그림 3. SOC에서의 검색 경로

2-3. SOC with Correction Code⁽¹¹⁾

Search-Order Coding에서와 같은 검색경로와 방법을 사용하지만 인덱스가 일치하는 위치만을 표시하지 않고, 정해진 범위 내에서 오차가 발생하는 부분에 대해서도 표현을 하는 방법이다.

예를 들어 오차표현을 위해 2비트를 사용한다고 가정하면 인덱스의 차이가 (-4)~(+4) 내에서 차이가 나는 인덱스를 검색하게 되므로 검색범위는 넓어지지만 인덱스의 일치여부를 표현하기 위해 1비트만이 추가되는 SOC 알고리즘에 비해 인덱스가 일치하는 경우, 기준 인덱스 보다 오차 허용 범위 내에서 작은 경우, 기준 인덱스 보다 오차 허용 범위 내에서 큰 경우, 인덱스가 일치하지 않으며 차이가 오차 허용 범위 밖인 경우까지를 표현해야 하므로 2비트가 필요하게 된다.

III. 제안 알고리즘

1. 코드북 구성과 인코딩을 위한 알고리즘

1-1. 검색영역의 축소

트레이닝 데이터를 이용하여 코드북을 구성하거나 구성된 코드북을 이용한 입력영상의 인코딩(encoding) 과정에서 입력벡터와 최소오차를 갖는 코드벡터를 검색할 때, 입력벡터에 대해 코드북의 전체 영역을 검색(full search)하는 방법에 비해 검색영역을 제한한다면 계산량의 상당 부분을 줄일 수 있게 된다. 따라서 본 논문에서는 코드북 검색의 결과 즉, 복원영상의 화질은 전체영역 검색의 경우에 비교하여 저하되지 않아야 한다는 제한요건을 만족하면서 검색영역을 축소하여 계산량을 줄일 수 있는 방법을 제안하였다.

에너지 크기로 재배열(rearrangement)한 코드북(256개의 코드벡터)에 대해 전역 검색(Full search)을 통해 최소오차를 갖는 코드벡터 인덱스를 구하고, 단지 에너지 크기가 가장 근접한 인덱스를 구해 이들간의 오차 분포 결과를 나타낸 것이 표 1의 결과인데, -64~64의 구간 즉, 256개의 코드벡터를 갖는 코드북의 절반정도의 영역에 전체의 99.98%가 누적될 정도로 오차가 코드북 전체 크기에 비해 상당히 작다는 것을 알 수 있으므로, 이와 같은 결과에 의해 그림 4에서와 같이 인코딩하려는 입력벡터에 대해 검색범위 W를 정해서 최소오차 코드북 검색을 시행하여 계산량을 줄일 수 있게 된다.

1-2. 코드벡터 요소의 부분 사용

불필요하다고 판단되는 영역에 대한 검색 배제에 의해 검색영역을 제한하는 앞서의 제안에 의해 계산량을 감소할 수 있지만, 이를 코드벡터를 구성하는 각 요소에 대해서도 적용한다면 더 많은 계산량

표 1. 인덱스 오차 분포

인덱스 오차	비율	누적비율
0	16.24 %	16.24 %
-4 ~ 4	58.73 %	74.97 %
-8 ~ 8	15.57 %	90.54 %
-16 ~ 16	7.00 %	97.54 %
-32 ~ 32	2.06 %	99.60 %
-64 ~ 64	0.38 %	99.98 %
-255 ~ 255	0.02 %	100.00 %

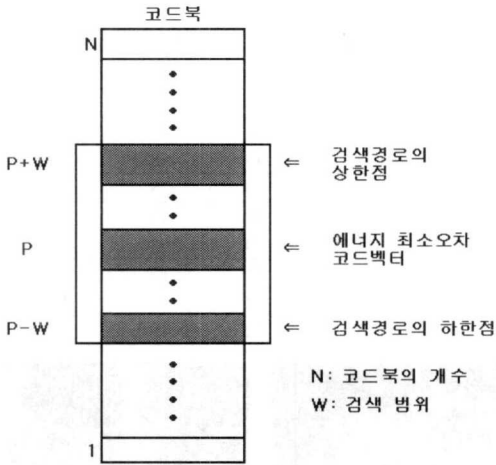


그림 4. 코드북의 검색영역

의 감소가 가능하리라 판단되어 코드벡터 요소의 부분 사용을 통한 계산량 감소 방법에 대해서도 제안하였다. 코드벡터와 입력벡터를 여러 영역으로 분리하여 단계적으로 오차를 비교하는 과정에서 최소 오차를 넘어서면 그 이후의 계산은 생략하는 방법을 제안하였는데, 이 때에 너무 여러 영역으로 분리할 경우 비교연산이 많이 추가되므로 16차원의 코드벡터를 4개의 영역으로 분리하여 적용하였다.

또한, 코드벡터를 4개의 영역으로 분리하는 경우에도 어느 영역을 먼저 계산하느냐에 따라 계산량이 달라질 수도 있으므로 이러한 점을 고려하여 대상영역의 우선 순위를 정하기 위하여 기준을 마련하였는데, 이미 만들어진 코드북에 대해서 코드벡터 요소의 각 위치에서의 분산값을 구하여 그 값에 대한 순위를 얻었으며, 이를 기초로 그림 5에서와 같이 4개의 영역을 설정하여 차례로 검색과정에 활용하는 방법을 제안하였다.

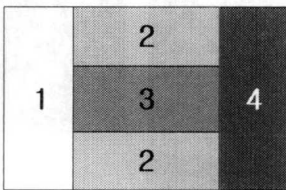


그림 5. 코드벡터의 영역구분

기본적으로 코드북은 여러 입력영상에 대해 연속적으로 사용되므로, 코드북의 분산값 정보를 미리 계산해 놓으면 이를 인코딩 과정에 적용할 때 추가

되는 계산이 필요 없이 활용할 수 있다.

1-3. 검색중지 조건의 도입

검색영역 축소로 계산량을 상당히 줄일 수 있지만, 이때에도 불필요한 계산은 여전히 존재하게 되는데, 표 1에서 오차범위 -16~16 에는 97.54%가 존재하고 오차범위 -32~32 에는 99.6%가 존재하므로 -32~-17, 17~32의 범위에는 2.06% 만이 존재하게 된다.

하지만 이를 무시할 경우 복원영상의 화질저하는 당연한 결과가 되므로 단지 2.06%의 입력벡터에 의해 검색영역은 2배로 늘어나게 되므로, 입력벡터에 따라 검색영역을 가변적으로 적용할 수 있는 검색중지 조건을 제안하였다.

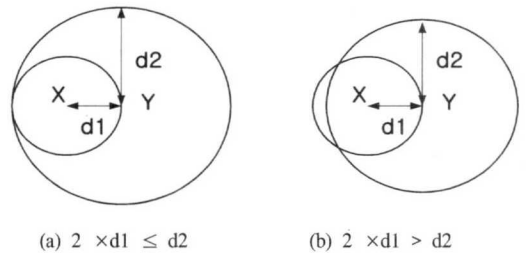


그림 6. 검색 중지 조건

그림 6에서 X를 인코딩 하려는 입력벡터, Y를 코드벡터, d_1 을 X와 Y 사이의 오차, d_2 를 코드벡터 Y와 다른 코드벡터 사이의 오차 최소값이라 가정하면, (a)에서와 같이 $2 \times d_1 \leq d_2$ 의 조건을 만족하면 이는 Y가 X의 최소오차 코드벡터라는 것을 의미하므로 검색을 중단해도 무방하다고 할 수 있는데, 이러한 중지 조건을 활용하면 모든 입력벡터가 같은 크기의 영역을 검색하는 것을 피할 수 있으므로 결과적으로 계산량을 줄일 수 있게 된다.

1-4. 저하된 화질의 보완

앞에서 제안된 검색영역 축소 방법을 사용하면 표 1에서 볼 수 있듯이 검색영역을 벗어나는 약간의 코드벡터에 의해 전역 검색(Full search)에서와는 다른 인덱스를 가지는 입력벡터가 존재하게 되는데, 이로 인해 복원영상에서 미세하나마 화질저하가 있을 수 있으므로 이를 해결하기 위하여 코드북 구성을 위한 트레이닝 데이터와 인코딩하기 위한 입력벡터를 윤곽선(edge)이 포함된 영역과 포함되지 않은 영역으로 구분하여 화질을 보완할 수 있는 방법을 사용하였는데, 영상의 윤곽선을 검출하기 위한 방법으로는

같이 BTC(Block Truncation Coding)^[12] 알고리즘에서 사용하는 영역구분 방법을 사용하여 윤곽선이 존재하는 영역과 존재하지 않는 영역을 구분하였다.

2. 인덱스의 무손실 압축을 위한 알고리즘

입력 영역과 코드북 인덱스가 같거나 오차한계 내의 인덱스를 갖는 인접 블록의 위치를 표현하기 위한 데이터량(비트)을 줄이기 위해 검색위치를 제한한다. 정정 부호(correction code)를 포함하는 SOC 방법에서 볼 수 있었던 바와 같이 검색위치의 인덱스가 기준위치의 인덱스보다 큰 경우와 작은 경우를 따로 표현하여 정정 부호를 사용하면

- ① 기준위치와 일치하는 경우
- ② 기준위치 보다 큰 경우
- ③ 기준위치 보다 작은 경우
- ④ 오차 한계를 벗어나는 경우

의 4가지 경우가 존재할 수 있으므로 이를 표현하는데 각 2비트씩이 필요하게 되지만, ②와 ③의 경우를 합하여 전체를 3가지로 표현한 후 출현빈도에 따라 1, 00, 01을 각각 할당하는 방식으로 가변길이 부호화(variable length coding)를 이용하는 방식으로 표현되는 데이터량을 줄일 수 있도록 구성한 VLIC(Variable Length Index Coding) 알고리즘을 제안하였다.

이 때 ②와 ③의 구분을 위해 1비트의 부호비트를 추가하였는데, ②와 ③이 합쳐진 영역 다시 말해, 오차 범위내의 영역의 확률이 가장 높을 경우 코드 "1"을 할당하고 여기에 부호비트 1비트가 추가되어 원래의 고정비트 방식에서와 같은 결과를 보일 것이라 예상된다.

그리고 경우①이나 경우④가 빈도가 높을 경우에는 오차범위내의 영역 표현에 3비트가 사용되지만, 이보다는 더 큰 빈도를 갖는 영역의 표현에서 1비트가 줄어들게 되므로 데이터량의 감소효과를 기대할 수 있게 된다.

표 2에서 볼 수 있는 바와 같이 코드북, 정정 부호의 크기에 따라 영역 분포가 달라지고 있으므로 가변길이 코드를 적용하면 개선효과를 기대할 수 있게 된다.

IV. 결과 및 고찰

본 논문에서 제안한 알고리즘에 대한 검증을 위한 모의실험은 그림 7에서와 같은 256×256(8bpp) 크기의 9가지 영상에 대해 수행하였다.

표 2. 인덱스 오차 분포

코드북 크기	correction bit	인덱스 오차(%)		
		일치	범위內	범위外
128	3	35	33	32
	4	35	48	17
	5	35	58	7
256	3	30	25	45
	4	30	40	30
	5	30	53	17
512	3	22	17	61
	4	22	32	46
	5	22	48	31



그림 7. 모의실험 영상

1. 인코딩(Encoding)

인코딩에서의 계산량 비교가 표 3에 제시되었는데, 제안 알고리즘Ⅱ는 화질보완을 위한 알고리즘이 추가된 경우이며 이 때의 수치는 LBG(Full search) 알고리즘의 계산량을 기준으로 하였을 때의 상대적 계산량을 나타낸 수치이다. 결과에서 128개의 코드벡터를 갖는 코드북의 경우 본 논문에서 제안한 방식이 LBG 알고리즘에 비해서는 73.1~88.8%, PDS 알고리즘에 비해서는 45.3~63.7%, MPS 알고리즘에 비해서는 14.6~43.4%, Kick-out 알고리즘에 비해서는 36.7~58.1%의 계산량 감소가 있었음을 볼 수 있으며, 256개의 코드벡터를 갖는 코드북의 경우 본 논문에서 제안한 방식이 LBG 알고리즘에 비해서는 88.3~91.1%, PDS 알고리즘에 비해서는 61.1~70.4%, MPS 알고리즘에 비해서는 31.6~47.9%, Kick-out 알고리즘에 비해서는 56.2~66.7%의 계산량 감소가 있었음을 확인할 수 있다.

또한 실험결과에서 복원영상의 PSNR 값이 거의 대등하거나 조금 증가된 것을 볼 수 있는데, 이는 검색영역의 축소에 따라 예상된 화질저하가 영역분류 방법의 도입으로 인해 보완되었음을 보여 준다.

표 3. 인코딩 결과 비교

(a) 코드북 = 128개

알고리즘	계산량(%)	화질(%)	
LBG	100.0	100.0	
PDS	30.9	100.0	
MPS	19.8	99.9	
Kick-out	26.7	100.0	
제안 알고리즘 I	w=20	16.9	99.8
	w=16	14.4	99.6
	w=12	11.7	99.5
제안 알고리즘 II	w=20	16.2	100.4
	w=16	13.8	100.1
	w=12	11.2	99.8

(b) 코드북 = 256개

알고리즘	계산량 (%)	화질(%)	
LBG	100.0	100.0	
PDS	30.1	100.0	
MPS	17.1	100.0	
Kick-out	26.7	100.0	
제안 알고리즘 I	w=28	11.8	99.9
	w=24	10.4	99.7
	w=20	9.1	99.6
제안 알고리즘 II	w=28	11.3	100.5
	w=24	10.1	100.4
	w=20	8.9	100.1



그림 8. 복원영상의 비교

2. 코드북 인덱스의 무손실 압축

전송되는 인덱스의 무손실 압축을 위한 제안 알고리즘의 결과는 표 4에 보여졌는데, 제안 알고리즘의 경우 정정부호(correction code)를 3비트(C=3)에서 5비트(C=5)까지 사용하여 다른 방식과 비교하였다.

표 4에서 제안된 VLIC 알고리즘의 결과를 살펴 보면, 128개의 코드벡터를 갖는 코드북의 경우 DPCM/Huffman 방식에 비해 3.2~8.1%, SOC 방법에 비해서는 2.7~4.5%, 정정부호를 사용한 SOC

방법에서는 5.1~9.9%의 데이터량 감소가 있었다.

256개의 코드벡터를 갖는 코드북의 경우 DPCM/Huffman 방식에 비해 6.2~10.4%, SOC 방법에 비해서는 2.6~6.9%, 정정부호를 사용한 SOC 방법에서는 4.0~8.3%의 데이터량 감소가 있었다.

512개의 코드벡터를 갖는 코드북의 경우 DPCM/Huffman 방식에 비해 2.5~7.8%, SOC 방법에 비해서는 3.0~5.1%, 정정부호를 사용한 SOC 방법에서는 2.6~4.7%의 데이터량 감소가 있었음을 각각 확인할 수 있었다.

이와 같은 결과에 의해 검색위치를 고정시키고 가변길이 코딩을 적용하는 제안 알고리즘의 성능이 어느 정도 확인되었다고 볼 수 있으며, 코드북의 크기가 커질수록 정정부호의 크기가 클 경우에 좋은 결과를 보이는 두 요소 사이의 상관관계 또한 확인할 수 있었다.

표 4. 화소당 비트율 비교

알고리즘	bpp		
	코드북 128	코드북 256	코드북 512
DPCM/Huffman	0.345	0.406	0.485
SOC	0.332	0.391	0.471
SOC with CC	0.352	0.397	0.469
VLIC (C=3)	0.323	0.381	0.473
VLIC (C=4)	0.317	0.368	0.457
VLIC (C=5)	0.334	0.364	0.447

V. 결론

영상데이터 압축을 위한 방법 중 하나인 벡터 양자화는 압축효율에 비해 시스템의 구조가 간단하다는 특히, 수신기의 디코딩 과정이 매우 간단하다는 장점을 지니는 방법으로 알려져 있다.

하지만, 코드북의 크기를 줄여 압축률을 높이면 복원 영상의 화질, 특히 윤곽 부분의 화질이 저하되고, 화질 저하를 막기 위하여 코드북의 크기를 크게 하면 코드북 구성 시간, 부호화시간과 코드북 저장을 위한 기억장소의 증가 등의 문제점 또한 지니고 있는 방법이라 할 수 있다.

따라서 본 논문에서는 복원영상의 화질저하가 심각하게 발생하지 않는다는 전제조건하에 검색영역의 축소, 코드벡터의 부분 사용과 검색 중지 조건 등을 도입하여 코드북 구성과 인코딩 과정에서의 계산량을 줄이는 방법을 제안하였다.

또한 이 과정에서 저하되는 화질을 보완하기 위한 방법과 인코딩 과정에 의해 선택되어 전송되어지는 코드북 인덱스에 가변길이 부호화의 개념을 적용하여 시스템 전체의 압축률을 높일 수 있는 방법 등 벡터양자화 시스템 전반에 걸친 성능향상 기법에 대해 제안하였다.

제안 알고리즘에 대한 모의실험 결과, 인코딩 과정에서는 128개의 코드벡터를 갖는 코드북의 경우 LBG 알고리즘에 비해서는 73.1~88.8%, PDS 알고리즘에 비해서는 45.3~63.7%, MPS 알고리즘에 비해서는 14.6~43.4%, Kick-out 알고리즘에 비해서는 36.7~58.1%의 계산량 감소가 있었다.

256개의 코드벡터를 갖는 코드북의 경우 LBG 알고리즘에 비해서는 88.3~91.1%, PDS 알고리즘에 비해서는 61.1~70.4%, MPS 알고리즘에 비해서는 31.6~47.9%, Kick-out 알고리즘에 비해서는 56.2~66.7%의 계산량 감소가 있었다.

이러한 결과는 코드북 구성에 비해 인코딩에서의 계산량이 상대적으로 더욱 감소하였음을 보여주는 데, 이는 코드북 구성은 코드북 갱신이 여러 차례 이루어지며, 이 때마다 코드북 재배열이 필요하게 되기 때문에 인코딩의 경우에 비해 상대적으로 계산량 감소가 적었던 결과라고 생각된다.

또한 실험결과에서 복원영상의 PSNR 값이 거의 대등하거나 조금 증가된 것을 볼 수 있는데, 이는 검색영역의 축소에 따라 예상된 화질저하가 영역분류 방법의 도입으로 인해 보완되었음을 보여 준다.

코드북 인덱스의 무손실 압축에서는 128개의 코드벡터를 갖는 코드북의 경우 DPCM/Huffman 방식에 비해 3.2~8.1%, SOC 방법에 비해서는 2.7~4.5%, 정정부호를 사용한 SOC 방법에서는 5.1~9.9%의 데이터량 감소가 있었다.

256개의 코드벡터를 갖는 코드북의 경우 DPCM/Huffman 방식에 비해 6.2~10.4%, SOC 방법에 비해서는 2.6~6.9%, 정정부호를 사용한 SOC 방법에서는 4.0~8.3%의 데이터량 감소가 있었다.

512개의 코드벡터를 갖는 코드북의 경우 DPCM/Huffman 방식에 비해 2.5~7.8%, SOC 방법에 비해서는 3.0~5.1%, 정정부호를 사용한 SOC 방법에서는 2.6~4.7%의 데이터량 감소가 있었음을 각각 확인할 수 있었다.

이와 같은 결과에 의해 검색위치를 고정시키고 가변길이 코딩을 적용하는 제안 알고리즘의 성능이 확인되었다고 볼 수 있으며, 코드북의 크기가 커질수록 정정부호의 크기가 클 경우에 좋은 결과를 보

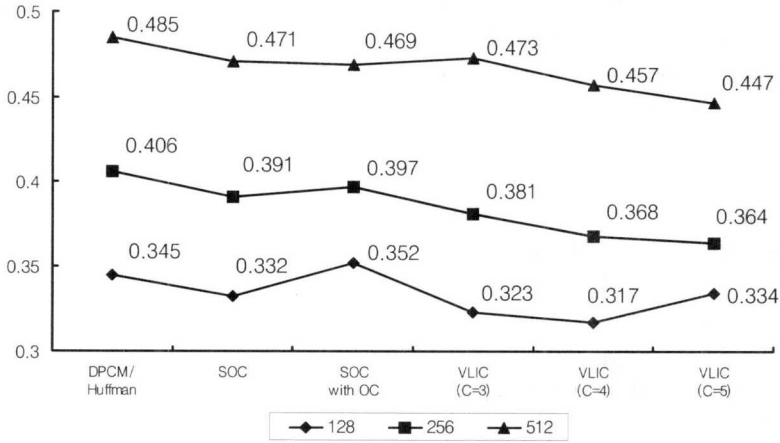


그림 9. 화소당 비트율 비교

이는 두 요소 사이의 상관관계 또한 확인할 수 있었다.

이와 같은 결과를 종합적으로 살펴보면 화질저하 없이 계산량을 줄이고 압축률을 높힐 수 있도록 하기 위한 본래의 목적은 달성되었다고 생각된다.

하지만, 계산량 감소를 위한 코드벡터 요소의 부분 사용에서 최적의 조건을 찾기 위한 일반화 과정이 필요하다고 생각되며, 영상 영역의 특성에 따라 코드북의 종류와 크기를 달리하는 CVQ(Classified VQ)의 개념을 도입하여 보완한다면, 계산량 감소와 압축률 개선에 더욱 효과가 있으리라 예상되므로 이에 대한 추가 연구가 필요하다고 생각된다.

참 고 문 헌

[1] Pamela C. Cosman, Robert M. Gray, Martin Vatteli, "Vector Quantization of Image Subbands: A Survey", IEEE Trans. on Image Processing, VOL 5, No 2, pp.202-225, 1996.
 [2] N. Ahmed, T. Natarajan, K. R. Rao, "Discrete Cosine Transform" IEEE Trans. on Computers, vol.23, pp.90-93, 1974
 [3] W. B. Pennebaker, J. Mitchell, "JPEG still image compression standard", Van Nostrand Reinhold, 1993
 [4] G. K. Wallace, "The JPEG still picture compression standard", Communication ACN, vol.34, pp.31-44, 1991
 [5] C. Bei, R. M. Gray, "An Improvement of the

Minimum Distortion Encoding Algorithm for Vector Quantization" IEEE Trans. on Commun., VOL. 33, pp1132-1133, 1985.

[6] S. W. Ra, J. K. Kim, "A Fast Mean-Distance-Ordered Partial Codebook Search Algorithm for Image Vector Quantization", IEEE Trans. on Circuits and Syatems-II : Analog and Digital Signal Processing, VOL. 40, No.9, pp576-579, 1993.
 [7] Kuang-Shyr Wu, Ja-Chen Lin, "Fast VQ Encoding by an Efficient Kick-Out Condition", IEEE Trans. on Circuits and System for Video Technology, VOL. 10, No. 1, pp59-62, 2000.
 [8] 김백기 "지문영상의 무손실 압축 및 분류 알고리즘에 관한 연구", 박사학위 논문, 경희대학교, 1997
 [9] I. H. Witten, R. M. Neal, J. G. Cleary, "Arithmetic coding for data compression", Communications of the Association for Computing Machinery ACM, vol.30, no.6, pp520-540, 1987.
 [10] C. H. Hsieh, J. C. Tsai, "Lossless Compression of VQ index with Search-Order Coding" IEEE Trans. on Image Processing, VOL 5, No. 11, pp1579-1582, 1996.
 [11] E. C. Liu, T. C. Wang, "An Improvement on Lossless Compression of VQ Indexes" Proceedings of the Globecom '98 - Volume 3, pp. 1699-1704, 1998.

[12] Chou-Chen Wang, Chin-Hsing Chen and I-Hong Chen, "Modified VQ-BTC algorithm for Image compression", Electronics Letters, vol. 34, No. 14, pp. 1390-1392, 1998

정 일 화(Il-hwa Jung)

정회원



1991년 2월 : 경희대학교 전자
공학과 학사
1993년 8월 : 경희대학교 전자
공학과 석사
2001년 2월 : 경희대학교 전자
공학과 박사

최 인 호(In-ho Choi)

정회원



1990년 2월 : 경희대학교 전자
공학과 학사
1992년 2월 : 경희대학교 전자
공학과 석사
2000년 2월 : 경희대학교 전자
공학과 박사

이 대 영(Dai-young Lee)

정회원



1964년 2월 : 서울대학교 문리
대학 물리학과 이학사
1971년 2월 : 캘리포니아 주립
대학원 컴퓨터학과 석사
1979년 2월 : 연세대학교 전자
공학과 박사