

論文98-35S-5-10

활동도와 신경망을 이용한 벡터양자화 코드북 설계

(Vector Quantization Codebook Design Using Activity and Neural Network)

李京桓*, 李法基*, 崔正鉉*, 金德奎*

(Kyeong Hwan Lee, Bub Ki Lee, Jung Hyun Choi, and Duk Gyoo Kim)

요 약

기존의 벡터양자화 코드북 설계방식은 에지부분의 열화와 많은 계산량을 가지는 단점이 있었다. 본 논문에서는 학습벡터의 가로 및 세로 방향의 활동도를 구하여 활동도 좌표를 만들어 2차원 코드북에 효과적으로 대응시킨 후, 코드벡터들간에 상호 병렬적으로 학습해 나가는 KSFM 신경망을 이용하는 새로운 코드북 설계방법을 제안하였다. 전 설계과정동안 학습벡터들을 저장할 필요가 없고 학습속도가 빠른 신경망학습의 장점을 이용하면서, 설계과정과 부호화시 전역탐색을 하지않고 대응된 코드벡터 주위에 작은 창을 이용하여 탐색을 행하므로써 계산량을 줄일 수 있었다. 모의실험 결과 제안한 방법은 기존의 방식에 비해 높은 PSNR과 좋은 에지부분의 복원성을 가짐을 알 수 있었고, 같은 의도로 계산량이 많은 LBG 알고리즘을 쓰고 여러개의 코드북을 만드는 CVQ에 비해서도 PSNR 측면에서 더 좋은 결과를 보였다.

Abstract

Conventional vector quantization (VQ) codebook design methods have several drawbacks such as edge degradation and high computational complexity. In this paper, we first made activity coordinates from the horizontal and the vertical activity of the input block. Then it is mapped on the 2-dimensional interconnected codebook, and the codebook is designed using Kohonen self-organizing map (KSFM) learning algorithm after the search of a codevector that has the minimum distance from the input vector in a small window, centered by the mapped point. As the search area is restricted within the window, the computational amount is reduced compared with usual VQ. From the results of computer simulation, proposed method shows a better performance, in the view point of edge reconstruction and PSNR, than previous codebook training methods. And we also obtained a higher PSNR than that of classified vector quantization (CVQ).

I. 서 론

벡터양자화(vector quantization)는 영상을 화소단위로 양자화하여 전송하는 스칼라양자화와는 달리 영상을 일정한 크기의 블럭으로 나누어 그 내부 화소들간의 상관성을 이용하여 벡터 단위로 양자화하는 방식

으로서, 높은 압축률에서도 좋은 성능을 나타내므로 저전송율의 영상통신에 적합한 것으로 알려져 있다.^[1]

벡터양자화를 하기 위해서는 먼저 코드벡터들을 학습시켜 코드북을 만들어야 하는데, 지금까지 일반적으로 널리 쓰이는 방법으로 Linde 등^[3]이 제안한 LBG 알고리즘이 있다. 이 방법은 학습벡터들을 모두 입력시킨 다음 초기화된 대표벡터에서부터 지역적으로 최적화되도록 반복하는 설계방법으로, 복원영상의 에지

* 正會員, 慶北大學校 電子工學科

(Dept. of Elec. Eng., Kyungpook Nat'l Univ)

接受日字: 1997年8月18日, 수정완료일: 1998年4月22日

(edge)부분의 화질저하를 초래하고, 학습 전과정에 대해 모든 학습벡터들을 가지고 있어야하므로 많은 기억용량이 필요하며, 초기화를 잘 해주지 않으면 코드벡터들마다 부호화시 발생확률의 차이가 나는 단점이 있다.

한편, 코드북 학습에 신경망을 이용한 방법들이 소개되었으며,^{[4],[5]} Nasrabadi 등^[6]은 이중 KSFМ (Kohonen self-organizing feature map) 신경망을 영상에 대한 벡터 양자화에 적용하여 복원된 영상의 PSNR (peak signal-to-noise ratio) 측면에서 LBG 알고리즘과 비교하여 비슷한 결과를 얻었다. 이 KSFМ 방법은 하나의 입력벡터가 들어오면 2차원적으로 서로 연결된 코드벡터들이 병렬로 작용하며 차차 입력에 적응하여 나가는 방법인데,^{[6],[9]} 입력벡터들을 저장할 필요가 없고 초기화를 잘 해주지 않아도 코드벡터들이 입력쪽으로 변하여 가므로 각각마다 부호화시 발생확률의 차이가 크지 않다는 장점이 있다. 그러나 이 방법 역시 복원 영상의 에지부분에서 화질저하를 초래한다.

복원 영상에서 에지부분의 열화는 인간의 눈에 많이 거슬리므로 에지영역의 복원특성이 화질에 큰 영향을 미친다.^{[7],[8]} 그러나 LBG 알고리즘에 의한 방법은 최소거리에 의해 나누어진 훈련벡터 집합 (cluster)의 평균으로 코드벡터를 만들므로 코드북의 크기가 크지 않으면 에지부분의 열화가 심하고, KSFМ에 의한 방법 역시 주로 사용되는 영상들의 통계로 볼때 70% 이상이 평탄 (shade) 혹은 중간영역 (midrange)의 블럭이므로^[7] 입력에 적응해가는 과정에서 이들 영역의 벡터들이 코드북의 많은 부분을 차지하게 된다. 이를 해결하고자 Ramamurthi 등^[7]은 공간영역에서 학습 블럭의 가로방향과 세로방향의 활동도 (activity)를 구하여 이를 통해 평탄, 중간영역과 에지영역으로 나누고, 에지영역은 다시 에지의 방향과 블럭내에서의 그 위치별로 분류하여 각각에 대해서 LBG 알고리즘으로 코드북을 따로 만들어 주는 CVQ (classified vector quantization)를 제안하여 에지의 복원성과 PSNR 측면에서 좋은 결과를 보였다. 그리고, 블럭의 영역과 에지방향별로 해당 코드북만 탐색하여 벡터 양자화를 함으로써 부호화 시간을 줄일 수 있었다.

본 논문에서는 CVQ에서와 같이 여러개의 코드북을 만들지 않고 단일 코드북을 사용함에도 불구하고 CVQ의 장점인 좋은 에지 복원특성과 짧은 부호화시

간을 가지며, 코드북 학습시 입력벡터들에 대한 기억용량이 필요없고 학습시간이 짧으며 초기화의 영향을 덜받는 신경망 학습의 장점을 아울러 가지는 새로운 코드북 설계 방식을 제안하였다. 먼저 간단한 Hadamard 변환을 이용하여^[8] 입력 학습벡터의 가로 및 세로방향의 활동도를 구하고, 이를 이용해 활동도 좌표를 만들어 정방형의 코드북에 대응시켜 대응된 코드벡터 주위에서 최소거리 벡터를 탐색한 후, KSFМ 방법으로 코드북 학습을 시행함으로써 코드북 내에 에지 벡터들이 그 활동도의 크기와 방향별로 적절히 분포하도록 하였다. 또한 부호화시 코드북을 전역탐색하지 않고 활동도 좌표를 중심으로 창을 만들어 그 내부의 벡터들만 탐색하므로 계산량 및 그에 따른 부호화 시간을 줄일 수 있었다.

II. 코드북 설계방법

코드북 학습을 위한 영상은 $N \times N$ 블럭 (K 차원의 벡터)들로 분리하여 입력한다. 코드북을 초기화한 다음 이 학습벡터들로 M 개의 코드벡터들을 훈련시켜 최종 코드북을 만든다. 코드북 설계에 사용되는 잘 알려진 학습방법들을 간단히 설명하면 다음과 같다.

1. LBG 알고리즘에 의한 방법

LBG 알고리즘은 대표벡터들을 초기화 시킨 후, 입력벡터들을 모두 받아들인 다음 NN (nearest neighbor) 조건에 따라 최소거리의 대표벡터가 같은 것들끼리 입력벡터의 집합을 만들고 중심 (centroid) 조건에 의해 집합의 평균인 새로운 대표벡터를 취한 후, 다시 NN조건과 중심조건에 따라 학습과정을 반복한다. 이 반복은 전체 입력벡터와 각각 속한 집합의 대표벡터들과의 거리가 미리 정한 수준으로 수렴할 때까지 계속되고 반복이 끝나면 대표벡터들은 코드북으로 사용된다.

그러나 최종 코드북은 초기화에 많은 영향을 받으므로 초기화를 잘못하면 각 집합에 속한 입력벡터의 수가 많은 차이를 보일 수가 있고, 집합에 속한 입력벡터들의 평균으로 코드벡터를 만들므로 코드북의 크기가 작으면 에지부분이 잘 표현되지 않게 된다. 이를 해결하기 위하여 제안된 CVQ는 코드북을 활동도의 크기와 에지 방향에 따라 여러개를 만들어 입력벡터를 이 기준에 의하여 분류한 후 학습시킴으로써

에지 부분이 잘 복원되었고, 부호화과정에서 전체 코드벡터를 탐색하지 않고 분류과정에 따라 해당 코드북만 탐색하므로 계산량과 부호화 시간을 줄였다. 그러나 여러개의 코드북을 각각에 속하는 벡터들의 발생빈도에 따라 크기를 달리하여 따로 학습시켜야하는 번거로움이 따른다.

2. 신경망학습에 의한 방법

코드북 학습에 사용된 신경망들은 두개의 층으로 되어 있는데, 입력층 (input layer)에서 입력벡터의 원소들을 출력층 (winner-take-all layer)의 각 신경단위로 보내어 거리계산을 통해 최소거리의 승리한 신경단위를 정하고, 승리한 신경단위의 대표벡터를 갱신시킨다. 이를 이용한 간단한 방법으로는 CL (competitive learning) 신경망을 이용한 방법이 있다. 이것은 먼저 M 개의 신경단위의 대표벡터들 $W_i(0), i = 1, 2, \dots, M$ 을 초기화 시키고, 입력벡터들에 의해 최소거리의 신경단위의 대표벡터를 입력쪽으로 갱신해 나가는 방법이다. t 시점의 입력벡터 X 에 대해 각 대표벡터들의 거리를 측정했을 때 최소거리의 신경단위가 j 라면, 그 대표벡터는

$$W_j(t+1) = W_j(t) + \epsilon(t)(X - W_j(t)) \quad (1)$$

와 같이 갱신된다.

여기서 $\epsilon(t)$ 는 학습비율로서 학습과정이 경과함에 따라 영으로 감소하는 함수이다. 반복과정이 모두 끝나면 대표벡터들은 코드북의 코드벡터들로 사용한다. 그러나 갱신이 승리한 신경단위에서만 이루어지므로 초기화의 문제가 있는 등 신경망의 장점을 활용할 수 없으므로 잘 쓰이지 않고, 이를 개선하여 거리계산시 해당 신경단위의 선택누계 $u_i(t)$ 를 곱하여

$$d^*(X, W_j) = d(X, W_j(t)) \times u_j(t) \quad (2)$$

와 같이 해줌으로써 의도적으로 적게 선택된 신경단위를 승리할 수 있게 만들어 코드북 전체가 발생빈도가 높은 입력벡터들로 적응해 나가는 FSCL (frequency sensitive competitive learning) 방법이 많이 사용된다. 여기서 $u_i(t)$ 는 i 번째 신경단위가 선택된 횟수이므로, 이 방법으로 학습을 시킬 경우 입력 빈도가 높은 벡터들쪽으로 많은 코드벡터가 만들어짐을 짐작할 수 있다.

또한, 이와 비슷한 의도로 사용되는 KSFM은 구조가 CL 신경망과는 달리 2차원적으로 연결된 출력층의 대표벡터들이 병렬로 상호 작용하여 하나의 입력벡터에 대해 승리한 신경단위 뿐만 아니라 그 이웃의 신경단위까지 갱신되는 방법이다. 그림 1에서는 입력노드들의 층과 신경단위들이 병렬적으로 연결된 출력층을 가지는 전형적인 KSFM 신경망을 나타내고 있다. 입력벡터와 최소거리의 j 번째 신경단위 주위의 신경단위들의 갱신은

$$W_i(t+1) = W_i(t) + \epsilon(t)N(j, t)(X - W_i(t)) \quad (3)$$

와 같이 이루어진다. 여기서 $N(j, t)$ 는 승리한 j 신경단위부근의 갱신되는 이웃 신경단위의 범위를 결정하는 함수이며 학습과정이 경과함에 따라, 또 먼 거리에 있는 신경단위일수록 감소하는 함수이다.

FSCL과 KSFM 방법은 LBG 알고리즘에 비해 코드벡터마다 발생확률이 고르게 분포하게되고, 벡터를 입력시키면서 학습을 해나가므로 학습시간이 줄어드는 장점이 있다. 그러나, 통계상 발생 빈도가 높은 평탄 및 중간영역의 코드벡터가 많이 생김에 반해, 발생 빈도가 적은 에지 영역의 코드벡터가 상대적으로 적어지므로 에지부분의 화질저하가 두드러지는 단점이 있다.

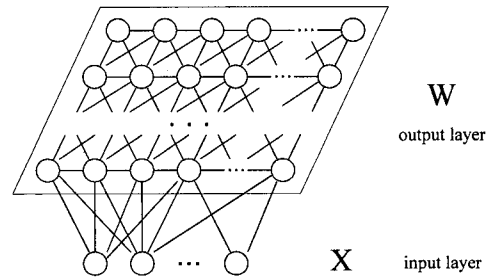


그림 1. KSFM 신경망의 구조
Fig. 1. Network structure of KSFM.

III. 활동도 좌표를 이용한 KSFM 코드북 설계

블럭내의 가로 및 세로방향의 활동도는 공간영역이나 변환영역에서 여러가지 방법으로 구할 수 있다. 본 논문에서는 간단한 4×4 Hadamard 변환을 이용하여 활동도 좌표를 만들어 2차원 코드북에 효과적으로 대응시킨 후, 앞 장에서 설명한 KSFM 신경망 학습방

법을 이용해 코드북을 학습시킴으로써 모든 활동도의 크기와 방향에 대해 충실한 코드북을 만들 수 있었다.

1. Hadamard 변환 계수를 이용한 활동도 좌표 정렬된 Hadamard 행렬 H는

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \quad (4)$$

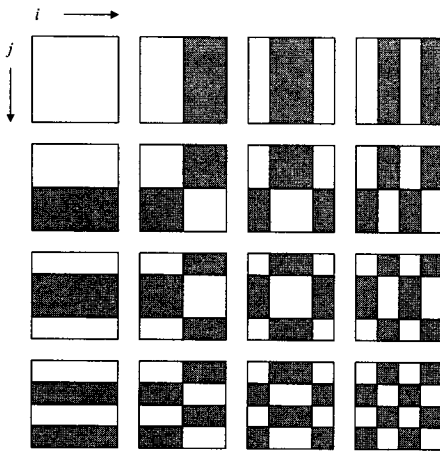
와 같이 1과 -1의 원소로 구성된 직교성을 갖는 4개의 기본 벡터로 되어 있다.

H의 행과 열들은 각각 Hadamard 기본 벡터이며, 이들의 외적으로 형성된 기본 평면들은 그림 2에 나타나 있다. 4×4 블록 X의 Hadamard 변환은

$$F = HXH \quad (5)$$

$$= \begin{bmatrix} f_{00} & f_{01} & f_{02} & f_{03} \\ f_{10} & f_{11} & f_{12} & f_{13} \\ f_{20} & f_{21} & f_{22} & f_{23} \\ f_{30} & f_{31} & f_{32} & f_{33} \end{bmatrix}$$

와 같이 된다.



white: higher intensity
shade: lower intensity

그림 2. 4×4 Hadamard 변환의 기본 평면들
Fig. 2. Basis planes of 4×4 Hadamard transform.

F의 각각의 계수들 f_{ij} 의 절대값은 블록내에서 그림 2의 기본평면과 같은 화소값 천이 성분의 강도를 나타내고, 그 부호는 천이의 방향을 나타낸다.

f_{00} 는 X의 평균 화소값에 대한 정보를 가지고 있으므로 활동도를 구하는 데는 사용하지 않고, 1행과 1열의 계수들로 순수 가로방향과 세로방향의 변화성분을

가지는 두 배열 $F_h = [f_{01}, f_{02}, f_{03}]$, $F_v = [f_{10}, f_{20}, f_{30}]$ 을 만든다. 이 배열들을 이용하여 좌표가

$$e_h = \begin{cases} \text{MAX}[|F_h(j)|], & \text{if } F_h(j) > 0 \\ -\text{MAX}[|F_h(j)|], & \text{if } F_h(j) < 0 \end{cases} \quad (6)$$

$$e_v = \begin{cases} \text{MAX}[|F_v(j)|], & \text{if } F_v(j) > 0 \\ -\text{MAX}[|F_v(j)|], & \text{if } F_v(j) < 0 \end{cases}$$

와 같이 주어지는 2차원 벡터 E를 정의한다. 여기서 $\text{MAX}[| \cdot |]$ 는 $j = 0, 1, 2$ 에 대하여 F_h 와 F_v 의 원소들 중 최대 절대값을 나타내는 연산자이다.

E 벡터는 e_h - e_v 평면에 나타낼 수 있는데, 그 원점으로부터의 거리는 블록의 활동도를 나타내며 30보다 크면 중간영역이나 에지 블록으로 볼 수 있으며,^[7] 방향은 에지의 방향을 나타낸다. 그림 3에서는 평탄영역의 벡터 E_1 과 에지영역의 벡터 E_2 를 e_h - e_v 평면상에서 표현되는 예를 보이고 있다.

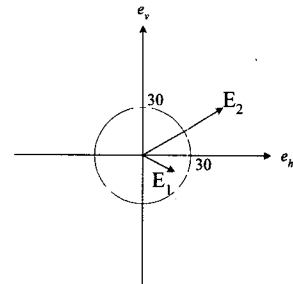


그림 3. E 벡터 표현의 예
Fig. 3. Example of expressing E vector.

2. 좌표의 대응과 신경망을 이용한 코드북 설계

구해진 활동도 벡터 E의 좌표 (e_h , e_v)를 본 논문의 32×32 코드북에 대응시키기 위하여 그림 4와 같은 스케일링 함수를 사용하여 좌표값을 -15.5에서 15.5까지의 값으로 변환하였다. 이렇게 변환된 좌표 (h , v)를 정수화하고 $h=0$, $v=0$ 가 코드북 중심에 오도록 값을 천이 시킨 좌표 (H , V), $H, V: 0, 1, 2, \dots, 31$ 을 얻는다. 이것은 2차원 코드북 배열의 지표로 사용된다.

최소거리 벡터의 탐색은 일반적인 학습방법에서처럼 코드북내의 모든 벡터에 대해서 최소거리 벡터를 탐색하지 않고, 그림 5에서와 같이 구해진 (H , V) 지표를 중심으로 정방형의 창내에 들어있는 벡터들에 대해서만 탐색을 수행하므로 코드북을 학습시킬 때나 완성된 코드북으로 부호화시킬 때 계산량 및 탐색시간을 줄일

수 있다.

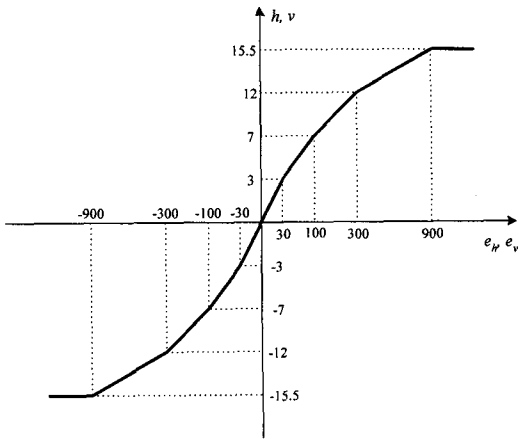


그림 4. 스케일링 함수
Fig. 4. Scaling function.

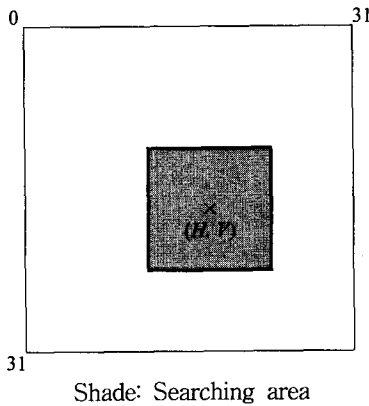


그림 5. 창을 이용한 2차원 코드북내에서의 최소거리 탐색
Fig. 5. Searching the minimum distance vector using a window in 2-dimensional codebook.

본 논문에서는 코드북을 학습시킬 때 (H, V) 를 중심으로 7×7 창내의 벡터에 대해서 탐색하여 최소 거리 벡터를 찾은 후에 KSFM 신경망 학습을 시행하였다. KSFM 학습방법을 선택한 이유는 학습속도가 빠르고, 탐색과정에 의해서 대응된 입력벡터들에 코드북이 잘 적응해 나가며, 신경단위간의 병렬작용으로 활용성이 큰 코드벡터들이 만들어지기 때문이다.

또한, 완성된 코드북으로 입력영상을 부호화할 때는 입력 블럭의 Hadamard 변환 계수가 30이 넘지 않으면 활동도가 낮은 블럭 (평탄 영역)이라고 단정하고

코드북의 중심으로부터 16×16 창내에 있는 벡터에 대해 거리를 측정하여 최소 벡터를 탐색하고, 30이 넘을 경우 활동도가 높은 블럭 (중간, 에지 영역)이라고 단정하고 구해진 (H, V) 를 중심으로 9×9 의 창내에 있는 벡터에 대해 최소거리의 벡터를 찾아서 입력벡터를 부호화하였다.

IV. 모의실험 및 결과

제한한 활동도 좌표와 신경망을 이용한 벡터 양자화 코드북 설계 방법의 성능을 평가하고, 기존의 설계방법들과 비교하기 위하여 컴퓨터 모의실험을 수행하였다. 본 논문에서는 기존의 벡터 양자화에서 보편적으로 사용하는 바와 같이 $N=4, K=16$, 코드북 크기는 $M=1024 (32 \times 32)$ 로 하였다. 그리고, 학습에는 10개의 256×256 영상, 즉 40960개의 벡터를 사용하였다. 벡터들간의 거리계산법으로는 널리 쓰이는 MSE (mean square error) 즉,

$$d(X, Y) = \frac{1}{K} \sum_{k=0}^{K-1} |x_k - y_k|^2 \quad (7)$$

을 사용하였다.

또한, 학습비율 함수 $\varepsilon(t)$ 와 이웃의 범위를 나타내는 함수 $N(j, t)$ 는

$$\begin{aligned} \varepsilon(t) &= 0.9 \left(1 - \frac{t}{T}\right) \\ N(j, t) &= \exp\left[-\frac{\|i-j\|^2}{2 \times 0.1^{t/T}}\right] \end{aligned} \quad (8)$$

와 같은 함수들을 사용하였다.

이때, t 는 현재의 입력 반복수이고 T 는 총 입력 반복수이며, $\|i-j\|$ 은 i 신경단위와 최소거리로 판정된 j 신경단위와의 2차원 평면상의 거리를 나타낸다.

그림 6에서는 기존의 방법들과 제안한 방법으로 학습시킨 코드북을 보여주고 있는데 기존의 방법들에서는 평탄영역의 벡터들이 많고 중복된 코드벡터들이 생기는 반면, 제안한 방법의 코드북에서는 중앙부에 평탄영역의 벡터들이 분포하며 가장자리로 갈수록 각 방향의 벡터들이 활동도 순으로 잘 나열됨을 볼 수 있다.

그림 7에서는 이들을 이용해 부호화하여 복원한 256×256 Pepper 영상의 에지부분을 확대하여 보여주고 있는데, 기존의 방법에 비해 제안한 방법과

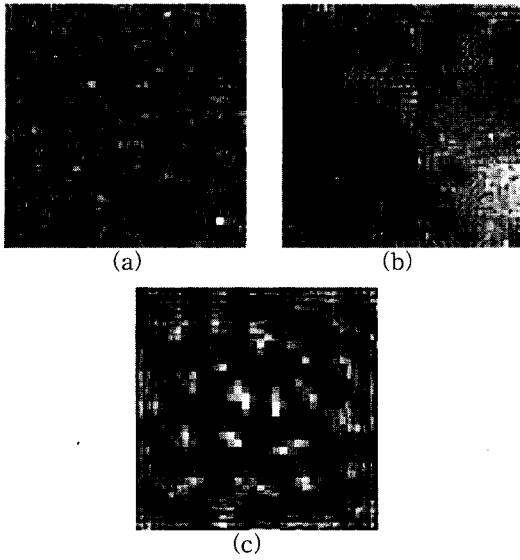


그림 6. 기존의 방법과 제안한 방법으로 만든 코드북
 (a) LBG (b) KSFM (c) 제안한 방법
 Fig. 6. Completed codebooks using (a) LBG algorithm (b) KSFM, and (c) proposed method.

CVQ가 에지부분의 열화가 적음을 볼 수 있다. 그리고, 표 1에서는 각각의 PSNR과 pentium 133MHz PC로 실험했을 때의 부호화 시간을 비교하고 있다. 먼저 PSNR을 볼 때 제안한 방법은 기존의 방법이나 발생빈도가 높은 평탄영역의 코드벡터 개수를 에지에 비해 적게한 CVQ에 비해 좋은 특성을 가짐을 알 수 있었고, 부호화 시간에서는 전역 탐색을 행하는 기존의 방법이나 에지의 위치까지도 분류하여 분류화 과정이 복잡한 CVQ에 비해 현저하게 줄어듦을 볼 수 있다.

표 1. 다른 학습방법과의 성능 비교

Table 1. Performance comparison with other learning methods.

Method	LBG	KSFM	CVQ	Proposed
Test image	256×256 Pepper			
Bit rate[bpp]	0.625			
PSNR[dB]	28.76	28.58	28.91	29.04
Encoding time[sec]	223	183	61	

표 2. CVQ와의 성능 비교

Table 2. Performance comparison with CVQ.

Method	CVQ			Proposed
Test image	512×512 Lena			
Bit rate[bpp]	0.7	1.0	0.625	0.75
PSNR[dB]	29.79	31.55	30.16	31.25

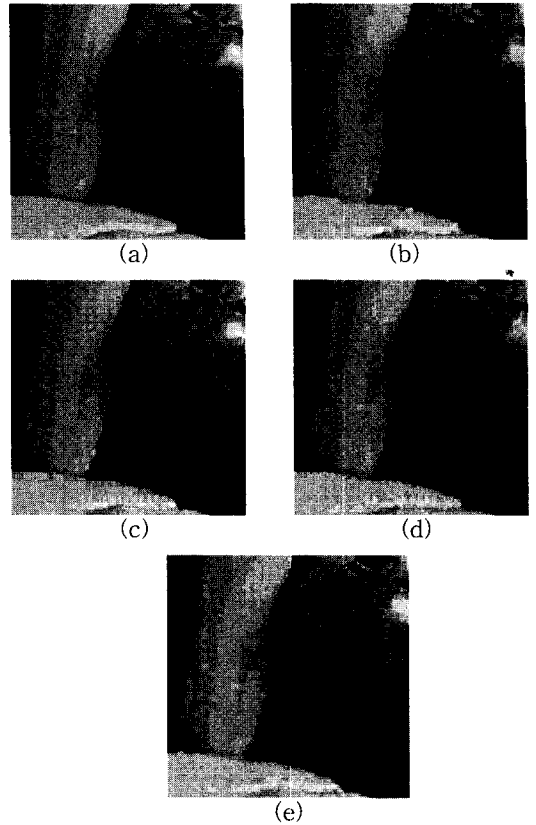


그림 7. 기존의 방법과 제안한 방법으로 복원한 256×256 Pepper의 에지부분영상 (a) 원영상 (b) LBG (c) KSFM (d) CVQ (e) 제안한방법
 Fig. 7. (a) Original partial edge image of 256×256 Pepper and it's reconstructed images using (b) LBG algorithm, (c) KSFM, (d) CVQ, and (e) proposed method.

본 논문에서 제안한 방법의 성능을 더 정확히 파악하기 위하여 코드북의 크기를 변화시키고 512×512 영상을 사용하여 실험하여 Gersho등^[7]의 CVQ 결과와 비교하였다. 그 결과는 표 II에 나타나 있으며, 제안한 방법은 에지부분의 복원성을 높이려는 같은 의도의 복잡한 CVQ보다도 성능이 우수함을 알 수 있다. 원영상과 제안한 방법으로 부호화하여 복원된 512×512 Lena 영상은 그림 8에 나타내었다.

V. 결론

본 논문에서는 블럭내의 가로 및 세로방향의 활동도를 측정하여 좌표로 만들고 정방향의 2차원 코드북에 대응시켜 그 부근의 코드벡터들을 탐색하여 KSFM

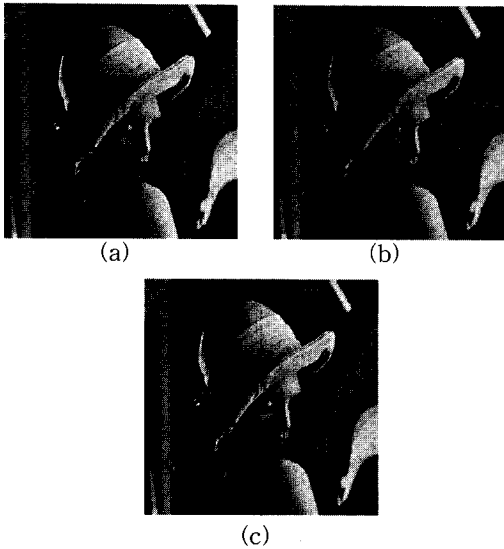


그림 8. 원 영상과 제안한 방법을 이용하여 복원한 512×512 Lena 영상 (a) 원영상 (b) 0.625 bpp (c) 0.75 bpp

Fig. 8. (a) Original image of 512×512 Lena and its reconstructed images using proposed method at (b) 0.625 bpp and (c) 0.75 bpp.

신경망 학습방법으로 코드북을 학습시키므로써, 에지 벡터들이 방향별로 적절히 분포하게 하고 부호화시 전역탐색을 하지 않아도 되므로 탐색시간을 줄일 수 있는 새로운 코드북 설계방법을 제안하였다. 제안한 방식은 학습시간이 빠르고 학습 벡터에 대한 기억용량이 부가적으로 필요없는 신경망 학습의 장점을 효과적으로 이용하고, 전역 탐색을 하지 않고 창을 이용한 최소 거리 벡터 탐색을 하므로 일반적인 코드북 학습 방법들을 사용하였을 때에 비해 부호화 시간을 대폭 줄일 수 있었다. 그리고, 에지 벡터들이 방향과 활동도의 크기에 의해 분류되어 코드북내에 적절히 분포하게 함으로써 에지부분의 복원성을 향상시킬 수 있었으며, 동일한 의도로 코드북을 여러개 만들고 LBG 알고리즘을 써서 복잡하고 학습시간이 많이 걸리는 CVQ 보다 높은 PSNR 특성을 얻을 수 있었다.

참 고 문 헌

- [1] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*: Kluwer Academic Publishers, 1992.
- [2] R. M. Gray, "Vector Quantization," *IEEE ASSP Mag.*, vol. 1, pp. 4-29, Apr. 1984.
- [3] Y. Linde, A. Buzo, and R. M. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Trans. on Comm.*, vol. COM-28, pp. 84-95, Jan. 1980.
- [4] T. Kohonen, "The Self-organizing Map," *Proc. IEEE*, vol. 78, no. 9, pp. 1464-1480, Sep. 1990.
- [5] S. C. Ahart, A. K. Krishnamurthy, P. Chen, and D. E. Melton, "Competitive Learning Algorithms for Vector Quantization," *Neural Networks*, vol. 3, pp. 277-290, 1990.
- [6] N. M. Nasrabadi and Y. Feng, "Vector Quantization of Images Based Upon the Kohonen Self-organizing Feature Maps," *IEEE IJCNN '88*, vol. 1, pp. 101-105, Jul. 1988.
- [7] B. Ramamuthi and A. Gersho, "Classified Vector Quantization of Image," *IEEE Trans. on Communication*, vol. COM-34, no. 11, Nov. 1986.
- [8] K. N. Ngan and H. C. Koh, "Predictive Classified Vector Quantization," *IEEE Trans. on Image Processing*, vol. 1, no. 3, Jul. 1992.
- [9] 김남철, 서정오, 홍원학, "반복 KSFM 신경망에 의한 영상분할," 제 5회 영상처리 및 이해에 관한 워크샵 발표논문집, pp. 79-84, Feb. 1993.

저 자 소 개



李京桓(正會員)

1969년 11월 16일생. 1994년 2월 경북대학교 전자공학과 졸업. (공학사). 1996년 2월 경북대학교 대학원 전자공학과 졸업 (공학석사). 1998

년 2월 경북대학교 전자공학과 박사과정 수료 주관심 분야 영상신호처리 및 압축

李法基(正會員)

1968년 6월 6일생. 1992년 2월 경북대학교 전자공학과 졸업 (공학사). 1994년 2월 경북대학교 대학원 전자공학과 졸업 (공학석사). 1996년 2월 경북대학교 전자공학과 박사과정 수료 주관심 분야 영상신호처리 및 압축.

崔正鉉(正會員)

1966년 3월 10일생. 1991년 2월 경북대학교 전자공학과 졸업 (공학사). 1993년 2월 경북대학교 대학원 전자공학과 졸업 (공학석사). 1995년 2월 경북대학교 전자공학과 박사과정 수료 주관심 분야 영상신호처리 및 압축.



金德奎(正會員)

1951년생. 1973년 2월 경북대학교 전자공학과 졸업 (공학사). 1977년 8월 경북대학교 대학원 전자공학과 졸업 (공학석사). 1989년 3월 일본 동경대학교 전기공학과 박사과정 졸업 (공학박사). 1980년 3월 ~ 현 경북대학교 전자전기공학부 교수. 현 대한전자공학회 총무이사, 한국공학기술학회 학술이사. 주관심 분야 영상 및 음향 시스템, PFC 등.