

웨이블릿 변환과 개선된 SOM 알고리즘을 이용한 칼라 이미지 벡터 양자화

준회원 남 미 영*, 정회원 김 광 백**

Color Image Vector Quantization by Using Wavelet Transform and Enhanced SOM Algorithm

Mi-Young Nam* Associate Member, Kwang-Baek Kim** Regular Members

요 약

본 논문에서는 웨이블릿 변환과 개선된 SOM 알고리즘을 이용한 칼라 이미지 벡터 양자화 방법을 제안하였다. SOM 알고리즘의 단점인 코드 벡터의 저활용 문제점을 개선하기 위해 각 블록에 대해서 승자 노드의 빈도수가 유사하도록 조정하고 가중치의 이전 변화량까지 반영하는 개선된 SOM 알고리즘을 제안하였다. 그리고 개선된 SOM 알고리즘에 대해서 경계 현상을 줄이고 재생의 정도를 높이기 위하여 웨이블릿 변환을 적용하여 벡터를 구성하고 이 벡터를 개선된 SOM 알고리즘에 적용하였다. 실험 결과, 제안된 방법이 기존의 SOM 알고리즘보다 압축율과 재생율이 개선되었다.

ABSTRACT

This paper proposes a vector quantization using wavelet transform and SOM algorithm for color image. We proposes the enhanced self-organizing algorithm to improve disadvantages of SOM algorithm. At first, the error between winner node and input vector are reflected to the weight adaptation using the frequency of winner node. At second, the weight is adapted in proportion to the present weight change and the previous weight change as well. For reducing blocking effect and improving resolution, we construct vector by using wavelet transform and we applies it to the enhanced SOM algorithm. Simulation results show that the proposed method is robust to compression ratio and decompression ratio.

1. 서 론

최근에 멀티미디어를 저장하고 전달하기 위한 여러 방법들이 제안되었는데, 그 중 손실 압축 방법의 대표적인 것이 벡터 양자화(vector quantization) 기법이다.

벡터 양자화를 사용한 대부분의 방법들은 린드(Linde), 부조(Buzo), 그레이(Gray)에 의해 개발된 LBG 알고리즘을 사용하고 있다^[1]. 그러나 LBG 알고리즘은 전체 이미지의 벡터들을 여러번 반복해야 최적에 가까운 코드 벡터(codevector)를 생성하는

반복 알고리즘으로 많은 시간이 요구되는 단점이 있다^[2].

인공 신경망을 이용한 방법은 적용되는 데이터로부터 적응적으로 학습되기 때문에 시간의 흐름에 따라 원 데이터의 통계 분포가 달라지는 응용에 적합하다^[3-5]. 신경망 구조는 거대한 병렬구조를 가지므로 고속의 처리기에 적합하고, 이 신경망 구조를 H/W로 구현시 색인에 의한 코드북의 탐색이 O(1)을 제공하게 때문에 특별히 코드북의 설계를 위해 신경 쓸 필요가 없다. 그리고 칼라 이미지의 벡터 양자화는 코드북을 결정하기 위해 이미지에서 화소

* 인하대학교 전자계산공학과(tera@im.inha.ac.kr)

** 신라대학교 컴퓨터공학과 (gbkim@silla.ac.kr)

논문번호: 020106-0311, 접수일자: 2002년 3월 11일

들의 분석이 필요하고, 그 코드북은 미리 알려져 있지 않기 때문에 자율 학습모델인 SOM 알고리즘이 널리 사용되고 있다. 그러나 신경망의 자율 학습 모델인 SOM(Self Organizing Map)을 이용한 벡터 양자화는 일부 부호어의 빈도가 낮게 사용되고 있는 반면에 다른 부호어는 빈번히 사용되는 현상이 있다^[6,7]. 실제로 이러한 현상은 데이터의 분포를 알기 어려우므로 데이터 중심에 대한 정확한 초기 예측이 어렵다. 그리고 압축되어진 이미지가 복원되면 원 이미지에 비해 약간의 왜곡이 발생할 수 있다. 따라서 이미지 데이터의 압축은 이미지의 질을 어느 정도 유지하면서 전송이나 저장 장치 영역에 대한 비트율을 감소시키는 데 그 목표가 있다. 이를 위해 DPCM, DCT, Hybrid Coding 등과 같은 대역폭 압축 기법이 제시되었다^[8]. DPCM 기법은 데이터의 Redundancy를 이용한 Predictive Coding 방식 중의 하나이며, DCT 방식은 데이터의 최대 정보를 최소의 샘플로 묶어 다른 계수로 변환하는 Transform Coding 방식에 속한다. 그리고 Hybrid Coding 방식은 DPCM과 DCT를 결합한 압축 방식이다. 이들 기법의 압축은 이미지 화소의 개별적인 값들 사이에서 진행된다. 그러나 연속되는 화소들은 어느 정도 종속되어 있고 상관 관계가 있기 때문에 이 방법들은 적합하지 않다.

따라서 본 논문에서는 웨이블릿 변환과 개선된 SOM을 이용하여 벡터 양자화의 단점인 경계 현상과 복잡한 계산에 의한 시간 및 기억 공간의 낭비를 줄이고 이미지의 복원 정도를 높인다.

II. 개요

1. 벡터 양자화

벡터 양자화에서는 원 이미지가 n차원 이미지 벡터(image vector)로 분할된다. 분할된 이미지 벡터 X를 미리 생성된 코드북의 대표 벡터 \bar{X} 와 비교하여 최소의 오차를 가지는 대표벡터로 사상하는 과정이 벡터 양자화이다. 즉, 벡터 양자화란 k차원 유클리디안 공간(euclidean space) R^k 에서 R^k 의 유한 부분 집합 Y로의 사상 Q로 정의된다. 이 과정을 수식으로 나타내면 다음과 같다.

$$Q: R^k \rightarrow Y, Y = (X_i; i = 1, \dots, N_c) \quad (1)$$

식 (1)에서 Y는 코드북에 해당되고, N_c 는 Y의 벡터 수, 즉 코드북의 크기이다. 여기서 최소의 오

차를 가지는 대표 벡터를 선택하기 위해서 일반적으로 널리 사용되는 척도가 입력 벡터와 대표 벡터 간의 유클리디안 거리 $d(X, \bar{X})$ 로 정의되며 식 (2)와 같다.

$$d(X, \bar{X}) = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x}_i)^2 \quad (2)$$

생성된 코드 북을 사용하여 벡터 양자화는 인코더(encoder), 디코더(decoder)로 이미지를 압축(compression), 재생(decompression)하게 된다. 인코더는 입력 벡터 X를 받아 대표 벡터 $Q(X)$ 로의 주소를 생성한다. 디코더는 이 주소를 이용하여 재생 벡터를 생성한다.

2. LBG(Linde-Buzo-Gray) 알고리즘

코드북 생성 알고리즘 중에서 대표적인 것이 Linde, Buzo, Gray에 의해 제안된 LBG 알고리즘이다^[3]. 이 알고리즘은 Lloyd가 스칼라 양자화기(scalar quantizer)의 설계 방법을 일반화한 것으로 GLA(Generalized Lloyd Algorithm)^[3]이라고도 한다. LBG 알고리즘은 모든 입력 벡터들을 최소의 왜곡(distortion)을 가지는 대표 벡터로 사상한 후, 평균 왜곡 정도를 계산하여 주어진 왜곡 척도를 만족하지 않으면 그 대표 벡터들의 중심을 다시 계산하여 최적의 코드북을 생성시키는 방법이다. LBG 알고리즘에서 최적의 코드북을 생성하기 위해서는 초기 코드북의 선택이 중요하다. 초기 코드북을 선택하는 방법으로는 임의로 설정하는 임의 코드 방식과 전체 학습 벡터를 이분하는 분할 기법이 있다. 그리고 N개의 클러스터에서 시작해서 두 개의 근접한 벡터를 묶어 점차 N개의 코드북으로 만드는 방법 등이 있다^[3]. 그러나 어떠한 초기화 기법을 사용하더라도 LBG 알고리즘은 전체 이미지의 벡터들을 여러 번 반복해야 최적에 가까운 코드북을 생성하므로 큰 이미지를 처리하는데는 많은 시간이 소요되는 단점이 있다.

3. 신경망을 이용한 벡터 양자화 기법

신경망 구조와 학습 방법은 벡터 양자화가 가능하도록 구성되어 있다. 신경망 알고리즘 중에서 자율학습 방법인 SOM 알고리즘이 벡터 양자화에 널리 적용되고 있다^[6,7]. SOM 알고리즘은 주어진 입력 패턴들에 대해 유사성을 가지는 특징들을 클러스터링 하여 분류하는 방법으로 하나의 입력 패턴에 대한 대표 클래스를 나타내는 승자 노드를 선택

하여 그 승자 노드에 입력 패턴의 정보를 저장하는 방식이다.

SOM은 입력 패턴과 저장 패턴 차이의 정도를 저장 패턴에 반영하여 저장 패턴이 입력 패턴에 유사하도록 조정된다. 따라서 여러 개의 입력 패턴이 하나의 클래스에 해당될 경우 그 저장 패턴은 자신에게 포함되는 모든 입력 패턴의 정보를 포함하면서 적응적으로 조정된다. 이러한 저장 패턴인 가중치를 조정해 나가는 과정이 벡터 양자화에서는 제시되는 입력 패턴들에 대해서 코드북을 생성해 나가는 과정이다. 따라서 SOM 알고리즘을 이용한 벡터 양자화는 칼라 이미지에 대해 코드북을 동적으로 생성한다. 그리고 SOM 알고리즘을 이용한 벡터 양자화 기법은 적용되어지는 학습 벡터들의 분포를 알지 못한 상태에서 초기의 코드 벡터들을 임의로 설정하여 적응적 방식으로 학습을 진행한다. 따라서, 이 적응적 알고리즘은 초기 코드북 결정 후 선택되지 않는 코드 벡터들이 생기므로 코드 벡터의 저활용(underutilization)문제가 발생한다.

III. 웨이블릿과 개선된 SOM 알고리즘을 이용한 칼라 이미지 벡터 양자화

본 논문에서는 그림 1과 같이 이미지에 대해 웨이블릿 변환을 수행한 다음 개선된 SOM 알고리즘을 이용하여 코드북을 생성하고 복원한다. 기존의 SOM 알고리즘을 이용한 벡터 양자화는 코드 벡터의 저활용 문제가 발생한다. 따라서 이를 개선하기 위해 각 블록에 대해서 승자 노드의 빈도수가 유사하도록 조정하고 이전의 변화량까지 반영하여 가중치를 조정하는 개선된 SOM 알고리즘을 제안한다. 그리고 웨이블릿 변환이 적용된 이미지에 개선

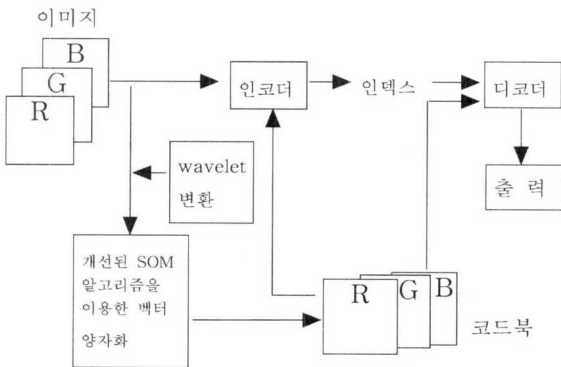


그림 1. 제안된 벡터 양자화의 처리 구조

된 SOM 알고리즘을 적용하여 많은 계산량과 코드 벡터의 저장을 위한 메모리를 줄인다. 또한 웨이블릿과 개선된 SOM 알고리즘을 적용하여 벡터 양자화에서 코드 북의 크기가 충분히 크지 않을 때 발생하는 블록 효과(block effect)도 줄인다.

1. 개선된 SOM 알고리즘

본 논문에서는 실시간 처리로 코드북 생성을 가능하게 하고 원 이미지에 근접한 재생 성능을 갖는 개선된 SOM 알고리즘을 제안한다. 그림 2와 그림 3은 개선된 ISOM 알고리즘을 이용한 코드북의 생성 과정을 나타내었다.

그림 2, 그림 3과 같이 개선된 SOM을 이용한 벡터 양자화는 실시간으로 처리하여 코드북을 생성

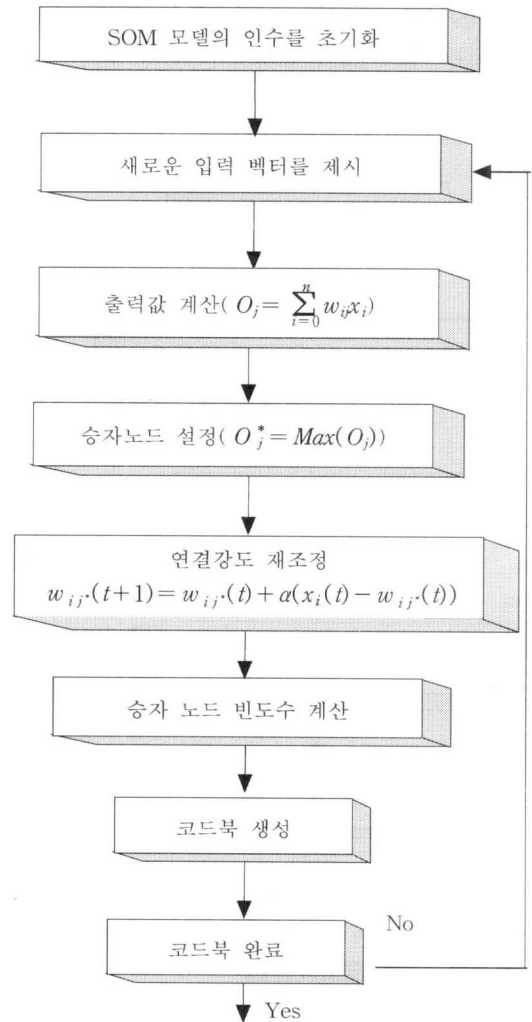


그림 2. 초기 코드북 생성에 적용된 개선된 SOM 알고리즘

하고 LBG 알고리즘보다 향상된 재생 능력을 갖는 경쟁 학습 알고리즘이다. 개선된 SOM 알고리즘은 빈도 수를 이용한 왜곡 척도와 가중치 수정 방법을 적용하여 승자 노드의 대표 벡터와 입력 벡터간의 오류 값에 따라 가중치를 조정하고 경계 현상(블록 효과, block effect)을 개선하기 위해 초기 코드북 생성시 입력 벡터의 인접한 화소에도 승자 노드의 영향을 받도록 한다. 제안된 SOM 알고리즘의 학습은 두 단계로 구성되며 각 단계에서는 전체 학습 벡터들을 한번씩만 적용한다. 첫 번째 단계에서는 주어진 학습 벡터들의 분포를 잘 나타낼 수 있도록 초기 코드북을 생성하고 두 번째 단계에서는 초기 코드북을 이용하여 현재의 각 결정 영역 내의 중심으로 옮겨 다시 코드북을 생성한다.

일반적으로 경쟁학습 신경망에서 가중치 조정에 사용되는 식은 다음과 같다.

$$w_{ij,t} = w_{ij,t-1} + \alpha(x_i - w_{ij,t-1}) \quad (3)$$

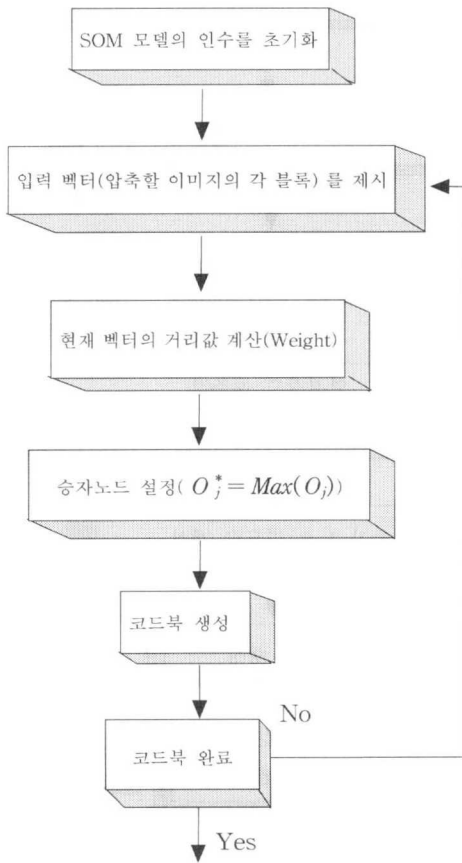


그림 3. 각 블록에 대한 인덱스 결정과 코드북 생성 과정

여기서 α 는 0과 1사이의 값을 가지는 학습 상수이다.

$(x_i - w_{ij,t})$ 는 오류 값으로 입력 벡터와 코드 벡터간의 차이를 의미한다. 식 (3)은 코드북 생성을 위한 가중치가 입력 벡터와 코드 벡터간의 차이만큼 조정되고 코드 벡터에 입력 벡터의 정보를 반영하게 된다. 여기서 그 정보를 반영할 때 입력 벡터와 코드 벡터간에 차이가 크면 코드 벡터에 입력 벡터를 기억시키기 위해 입력 벡터의 정보를 더 많이 반영해야 한다. 이를 위해 제안된 방법에서는 승자 노드에서의 오류 값을 0과 1사이의 값으로 바꾼 정규화 된 값인 출력 오류를 학습 상수로 사용한다. 제안된 출력 오류에 의한 가중치 조정으로 오류의 크기에 따라서 그 비율만큼의 조정이 가능하다. 또한 코드 벡터들의 저활용 문제는 배치 모드(batch mode)로 코드북을 수정하는 LBG 알고리즘과는 달리 한번에 하나의 벡터를 처리하는 적응적 알고리즘인 SOM 알고리즘에서 코드 벡터의 저활용 문제가 발생한다. 따라서 코드 벡터의 저활용 문제를 개선하기 위해 학습 중에 각 코드 벡터들이 수정되고 선택되어지는 횟수가 가능한 동등해지도록 각 노드의 승자 빈도수를 가중치 조정에 적용한다. 즉, 학습이 끝날 때 각각의 노드들이 승자 노드로 선정된 빈도수가 유사하도록 한다. 이를 위해서는 코드 벡터들과 학습 벡터간의 실제적인 왜곡 정도도 충분히 고려되어야 한다. 따라서 경쟁학습 알고리즘에서 승자노드를 선정하기 위한 왜곡의 척도를 승자노드로 선정된 빈도 수와 유클리디안 거리의 곱으로 수정한다. 제시된 방법에 의한 가중치 조정 식은 다음과 같다.

$$w_{ij,t} = w_{ij,t-1} + \alpha(x_i - w_{ij,t-1}) \quad (4)$$

$$\alpha = f(e_{j,t}) + \frac{1}{f_{j,t}} \quad (5)$$

여기서 $f(e_{j,t})$ 는 $e_{j,t}$ 의 값을 [0,1] 사이의 값으로 정규화 하는 함수이고, $e_{j,t}$ 은 j 번째 승자 노드의 오류 값, $f_{j,t}$ 는 j 번째 노드가 승자가 된 빈도수이다.

식 5의 오류 값을 이용한 방법은 현재의 가중치 변화량만 고려하고 그 전의 변화량은 고려되지 않는다. 따라서 비록 적응적으로 가중치 조정이 되지만, 현재의 변화량에 의해 가중치의 값이 많이 좌우된다. 이러한 문제점을 해결하기 위해 가중치 조정

시 현재의 가중치 변화량뿐만 아니라 그 이전의 가중치 변화량에 비례하여 가중치를 조정하도록 한다. 즉 다층 퍼셉트론의 모멘트 개념을 적용한 것으로 이에 따른 가중치 조정 식은 다음과 같다.

$$w_{ij}(n+1) = w_{ij}(n) + \delta(n+1) \quad (6)$$

$$\delta(n+1) = \alpha(n+1)(x_i(n) - w_{ij}(n)) + \alpha(n+1)\delta(n) \quad (7)$$

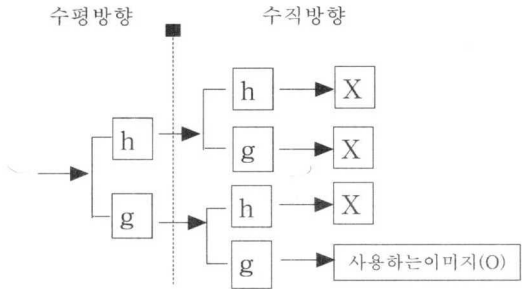
식 (7)에서 첫번째 항은 현재의 가중치 변화량이고 두번째 항은 모멘트를 적용한 이전의 가중치 변화량이다.

2. 웨이블릿 변환 적용

웨이블릿 변환 영역에서 분류 벡터 양자화를 수행하면 분류기(classifier)를 사용하지 않고도 데이터를 분류할 수 있다. DCT 변환에서는 에지의 굵기를 구분하지 못하는 단점이 있으나 웨이블릿 변환 영역에서는 에지의 굵기 구분이 가능하다. 그리고 웨이블릿 변환 영역에서는 경계 현상(블록 효과)이 발생하지 않는 장점이 있다⁹⁾.

따라서 제안된 SOM 알고리즘 대해서 경계 현상을 줄이고 재생의 정도를 높이기 위해 웨이블릿 변환을 적용한다. 웨이블릿의 주파수를 적용한 후 각각의 RGB값을 분리하여 개선된 SOM 알고리즘에 적용하여 칼라 이미지를 벡터 양자화 한다. 웨이블릿의 저주파수를 이용하면 이미지의 특징들이 잘 나타나므로 이미지를 구성하는 특징들이 세분화되어 벡터 양자화 된다. 웨이블릿 변환을 적용시킨 값으로 벡터를 구성하고 이 벡터를 개선된 SOM 알고리즘에 적용한다. 승자 노드에 해당하는 인덱스가 구해지면 그 인덱스에 해당하는 웨이블릿 변환을 적용한 벡터의 값이 아닌 원래 이미지의 값을 코드북에 저장하고 재생한다¹⁰⁾. 웨이블릿 변환은 코드북을 생성하기 전에 원 이미지에 적용하며 이미지를 저주파 대역의 수직 방향과 수평 방향으로 적용한다. 특히 기존의 웨이블릿 변환시 적용하는 다운로드 샘플링 과정을 수행하지 않고 이미지의 정보를 그대로 유지하도록 한다. 웨이블릿의 저주파 대역 필터를 이용하여 이미지의 강한 부분은 부각시키고 약한 부분은 감소시켜서 이미지를 평활화하는 효과를 얻도록 하고 잡음을 제거하여 코드북을 생성하고 각 블록에 대해서 코드북을 기준으로 인덱스를 생성한다. 웨이블릿을 적용한 이미지에서 각 벡터를 기준으로 각 블록간의 인덱스를 파악하여

코드북의 해당 블록(벡터)에 대한 RGB값을 저장하여 코드북의 크기를 축소하고 재생 시간도 단축한다. 즉 이미지의 크기 축소로 인한 압축이 아니라 이미지의 고유한 성질을 이용하여 개선된 SOM 알고리즘에 적용하여 벡터 양자화 한다. 본 논문에서 적용한 웨이블릿 변환 구조는 그림 4이고 저주파 대역과 고주파 대역 필터의 예는 그림 5와 같다.



h : 고주파 대역 통과, g : 저주파 대역 통과

그림 4. 적용한 웨이블릿 변환 구조

$$\begin{bmatrix} 1 + \sqrt{3} & 3 + \sqrt{3} \\ 3 - \sqrt{3} & 1 - \sqrt{3} \end{bmatrix} \times 4\sqrt{2}$$

(a) 저주파 대역 필터

$$\begin{bmatrix} 1 - \sqrt{3} & \sqrt{3} - 3 \\ 3 + \sqrt{3} & -1 - \sqrt{3} \end{bmatrix} \times 4\sqrt{2}$$

(b) 고주파 대역 필터

그림 5. 웨이블릿 변환에 사용된 필터

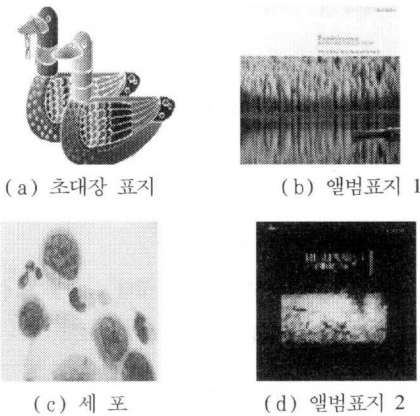
IV. 실험 결과

실험 환경은 IBM 586 펜티엄 III에서 C++ 빌더로 구현하였으며, 실험에서 사용된 이미지는 128 × 128 크기의 칼라 비트맵 이미지이다. 이미지를 4 × 4 크기의 블록으로 나누어 16byte 크기의 벡터로 코드북을 생성하여 성능을 비교하였다.

개선된 SOM 알고리즘을 이용하여 코드북을 생성하고 이미지를 압축했을 경우에는 시간이 LBG 알고리즘보다 5배 향상되었으며 코드북도 두 단계를 거쳐 바로 생성되기 때문에 기억 장소의 공간도 줄어들었다. 그리고 웨이블릿 변환을 적용하여 이미지의 경계 현상을 감소시켰고 이미지의 재생도 향상시켰다.

그림 6은 실험에 사용된 칼라 이미지이다. 제안된 방법은 그레이 이미지에도 적용할 수 있다. 그레이 이미지는 각 화소가 같은 RGB 값을 갖기 때문에 칼라 이미지와 같이 빨강, 녹색, 파랑에 대한 각각의 코드북을 생성하지 않고 그레이 이미지의 그레이 값에 대해 코드북을 생성하므로 속도가 칼라 이미지에 대한 벡터 양자화보다 빠르다. 실험에서는 칼라 이미지에 대한 벡터 양자화를 위한 것이므로 다양한 칼라가 포함되어 있는 이미지를 선택하였다. 그림 6의 이미지는 실험에 사용된 이미지로서 (a)는 초대장 표지이며 (b)는 앨범 표지이다. 그리고 (c)는 세포이미지이고 (d)는 칼라의 분포가 적은 앨범 이미지이다.

그림 6과 같은 이미지에 대해서 LBG 알고리즘의 재생율은 다른 방법에 비해 좋지 않았다. 따라서 기존의 SOM 알고리즘, 개선된 SOM 알고리즘 그리고 웨이블릿과 개선된 SOM 알고리즘간의 압축 정도를 비교하여 표 1로 나타내었다. 표 1은 동적으로 생성된 코드북의 크기를 나타낸 것이다.



(a) 초대장 표지 (b) 앨범표지 1
(c) 세포 (d) 앨범표지 2

그림 6. 실험 이미지

표 1에서 웨이블릿과 개선된 SOM 알고리즘을 이용한 방법이 다른 방법들보다 압축율이 향상되었고 기존의 SOM 알고리즘보다 개선된 SOM 알고리즘에서 생성된 코드북의 크기가 축소된 것을 알 수 있다. 색상의 분포가 다양한 세포 이미지는 다른 이미지에 비해 압축율이 낮았다.

재생된 이미지의 손실 정도를 측정하기 위해서 일반적으로 MSE가 많이 사용되고 있다. 즉 MSE가 크면 손실 정도가 크게 된다. 이것은 $n \times n$ 크기의 블록을 이미지 벡터로 사용했을 때, 칼라 이미지에 대해서 다음과 같이 정의된다.

표 1. 압축율 비교 - 코드북 크기 (단위 :byte)

이미지 \ 알고리즘	SOM	개선된 SOM	Wavelet과
	알고리즘	알고리즘	개선된 SOM
			알고리즘
a	32,208	31,968	22,064
b	48,816	33,672	26,528
c	52,080	51,648	27,376
d	22,656	21,456	18,224

$$MSE = \frac{1}{n^2} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \{ (r_{ij} - \overline{r_{ij}})^2 + (g_{ij} - \overline{g_{ij}})^2 + (b_{ij} - \overline{b_{ij}})^2 \}$$

여기서 r_{ij}, g_{ij}, b_{ij} 는 원 이미지의 칼라 요소 입력 값을, $\overline{r_{ij}}, \overline{g_{ij}}, \overline{b_{ij}}$ 는 원 이미지의 각 요소에 대한 코드 벡터 값을 의미한다. 따라서 표 2에서는 각 이미지에 대한 MSE를 나타내었다.

표 2. MSE(Mean Square Error) 비교

이미지 \ 알고리즘	SOM	개선된 SOM	Wavelet과
	알고리즘	알고리즘	개선된 SOM
			알고리즘
a	13.1	11.2	8.3
b	14.2	13.1	11.3
c	11.3	10.8	9.1
d	9.2	8.4	7.2

표 2의 실험에서 웨이블릿과 개선된 SOM 알고리즘을 적용했을 때 MSE가 가장 낮게 나타났다. 그리고 기존의 SOM 알고리즘보다 제안된 SOM 알고리즘이 이미지 손실 정도가 낮았다. SOM 알고리즘은 Red, Green, Blue에 대해 각각 코드북이 생성되었다. LBG 알고리즘은 10개의 코드북이 생성되었고 코드북을 생성하는 시간도 많이 소요되었다. 그러나 개선된 SOM 알고리즘과 웨이블릿 변환 및 SOM을 이용한 방법은 LBG 알고리즘에 비해 기억 공간이 줄었으며 전송시간도 단축되었다. 그리고 기존의 SOM 알고리즘 보다 개선된 SOM 알고리즘이 코드북을 동적으로 생성하는데 걸리는 시간이 적게 소요되었다. 웨이블릿 변환과 SOM 알고리즘을 이용했을 경우에는 코드북이 하나 생성되어 각 칼라 값에 대한 인덱스를 생성하게 된다. 따라서 다른 방법들에 비해 코드북의 생성시간과 기억 장소의

공간이 줄어들기 때문에 계산상의 복잡성이 감소되었다. 코드북의 생성에 필요한 시간을 비교하여 그림 7로 나타내었다.

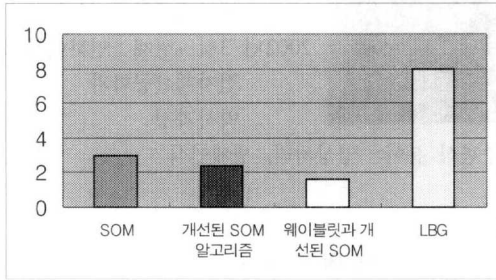


그림 7. 코드북 생성 시간 비교

각 이미지에 대해 동적으로 코드북을 생성하여 이미지를 재생한 결과는 그림8, 그림9, 그림10, 그리고 그림 11과 같다. 실험 결과를 분석하면 기존의 SOM 알고리즘을 이용했을 때 보다 개선된 SOM 알고리즘을 이용하여 코드북을 동적으로 생성했을 경우가 압축율이 향상되었다. 그리고 웨이블릿 변환의 저주파 대역을 적용함으로써 이미지의 특징들이 명확히 구분되었고 이 값을 개선된 SOM 알고리즘의 입력 벡터로 사용하여 승자노드를 결정하고 코드북을 생성했을 때 벡터 양자화의 성능이 향상되었다.



그림 8. 초대장 표지 이미지의 복원 결과

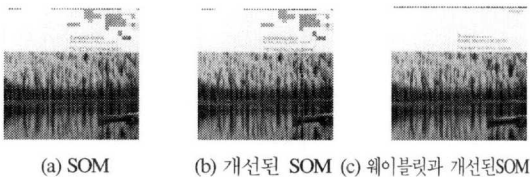


그림 9. 앨범 표지1 이미지의 복원 결과

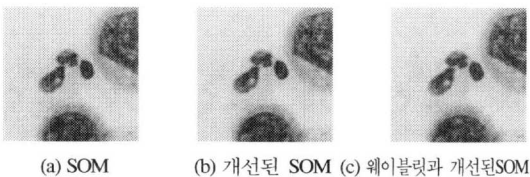


그림 10. 세포 이미지의 복원 결과



그림 11. 앨범 표지2 이미지의 복원 결과

IV. 결론

최근에는 신경망을 이용하여 코드북을 생성시키는 방법들이 연구되고 있으며, 본 논문에서는 이러한 신경망 기법을 적용하여 코드북을 생성하는데 있어 개선된 SOM 알고리즘을 제안하였다. 제안된 SOM은 기존의 SOM 알고리즘을 이용하여 벡터 양자화 할 때 발생하는 코드 벡터의 저활용 문제점을 개선하기 위해 각 블록에 대해서 승자 노드의 빈도수가 유사하도록 조정하고 가중치의 이전 변화량까지 반영하도록 하였다. 개선된 SOM 알고리즘의 학습은 두 단계로 구성되고 각 단계에서 전체 학습 벡터들을 한번씩만 적용하였다. 첫 번째 단계에서는 주어진 학습 벡터들의 분포를 잘 나타낼 수 있도록 초기 코드북을 생성하였으며 두 번째 단계에서는 초기 코드북을 이용하여 현재의 각 결정 영역 내의 중심으로 옮겨 다시 코드북을 생성하였다. 그리고 웨이블릿 변환이 적용된 이미지에 개선된 SOM 알고리즘을 적용하여 많은 계산량과 코드 벡터의 저장량을 위한 메모리를 줄였다.

본 논문의 실험 결과, 웨이블릿 변환과 개선된 SOM 알고리즘을 적용하여 벡터 양자화의 문제점인 시간과 기억 공간 문제로 인한 계산 복잡성, 코드북의 생성 및 경계 효과를 개선하였다. 그리고 본 논문에서 제안한 가중치 조정 기법을 적용하여 기존의 SOM 알고리즘에서 나타나는 일부 부호어의 빈도수 문제 등을 개선하였다.

향후 연구 방향은 입력 벡터와 코드 벡터의 차이를 더 잘 반영해 줄 수 있는 퍼지 기법을 연구하여 제안된 방법을 개선하고 다양한 이미지에 대해서 실시간으로 적용할 수 있도록 할 것이다.

참 고 문 헌

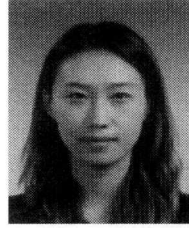
[1] M.Rabbani and P.W.Jones, "Digital Image Compression Technique," Spier Optical Engineering Press, pp.144-169, 1991.
 [2] R.M.Gray, Vector Quantization, IEEE ASSP

Magazine, Apr., pp.4-29, 1984.

- [3] Y.Linde, A.Buzo, and R.M.Gray, "An Algorithm for Vector Quantizer Design," IEEE Trans. On Communications, Vol.28, No.1, pp.84-95, Jan., 1980.
- [4] K.R.L.Godfrey and Y.Attiuzel, "Self-Organized Color Image Quantization for Color Image Data Compression," Proceedings of IJCNN, Vol.3, pp. 162-1626, 1993.
- [5] Bruce E. Watkins and Murali Tummala, "Classification Vector Quantization of Image Data Using Competitive Learning," Proceedings of ICIP, 1984.
- [6] K. B. Kim, E. Y. Cha, "A Fuzzy Self Organizing Vector Quantization for Image," Proceedings of Soft Computing, Vol.2, pp.757-760, 1997.
- [7] Liza Jo and M.R.Kaimal, "A Self Organizing Neural Network for Second Generation Image C-coding," Proceedings of TAINN, pp.130-1371, 1999.
- [8] S.P.Lloyd, "Least square quantization in PCM's," Bell Lab., Paper, Murray Hill, NJ. 1987.
- [9] R .K. Young, Wavelet Theory and Its Applications, Kluwer Academic Publishers, 1993.
- [10] 김진아, 정성환, "Wavelet 기반의 질감 특성을 이용한 영상검색," 한국정보과학회 가을학술발표논문집, Vol.24. No2, pp.379-382, 1997.

남 미 영(Mi-Young Nam)

준회원



1999년 2월 : 신라대학교
전자계산학과 졸업
2001년 2월 : 신라대학교
전자계산학과 석사
2002년 3월~현재 : 인하대학교
전자계산공학과
박사과정

<주관심 분야> 영상처리, 생체인식

김 광 백(Kwang-Baek Kim)

정회원



1993년 2월 : 부산대학교
전자계산학과(이학석사)
1999년 2월 : 부산대학교
전자계산학과(이학박사)
1996년 3월~1997년 2월 :
동의공업대학사무자동화과
전임강사

1997년 3월~현재 : 신라대학교 컴퓨터공학과 조교수
<주관심 분야> 영상처리, 신경망, 컴퓨터비전, 퍼지 시스템, 생체신호처리