

# 적응적 순위 기반 재인덱싱 기법에서의 동일 빈도 값에 대한 우선순위 방법

정희원 유 강 수\*, 유 희 진\*\*, 장 의 선\*\*\*

## Priority Method on Same Co-occurrence Count in Adaptive Rank-based Reindexing Scheme

Kang Soo You\*, Hee Jin Yoo\*\*, Euee S. Jang\*\*\* *Regular Members*

### 요 약

본 논문은 인덱스 영상의 무손실 압축을 위한 적응적 순위 기반 재인덱싱 기법에서 동일 빈도 값에 대한 우선 순위 결정 방법을 제안한다. 발생빈도행렬에서 동일 빈도 값에 대한 우선순위 결정은 발생빈도행렬의 임의의 행에서 물리적으로 처음 위치한 빈도 값, 주대각선 주위에 위치한 빈도 값, 빈도 값이 큰 원소의 주위에 위치한 빈도 값을 사용한다. 실험 결과, 제안 방법은 기존의 Zeng과 Pinho의 방법보다 1.71 비트까지 절감 효율을 보였다.

**Key Words** : rank-based reindexing, index image, lossless compression, priority decision

### ABSTRACT

In this paper, we propose a priority method on same co-occurrence count in adaptive rank-based reindexing scheme for lossless indexed image compression. The priority on same co-occurrence count in co-occurrence count matrix depends on a front count value on each row of co-occurrence count matrix, a count value around diagonal line on each row of the matrix, and a count value around large co-occurrence count on each row of the matrix. Experimental results show that our proposed method can be reduced up to 1.71 bpp comparing with Zeng's and Pinho's method.

### 1. 서 론

인덱스 영상에서의 재인덱싱을 통한 무손실 압축이 효율적이라는 사실은 잘 알려져 있다<sup>[1]</sup>. 이러한 인덱스 영상은 현재 인터넷을 통해 보급되고 있다. 예를 들면, 회사의 로고나 컴퓨터로 작업한 그래픽 이미지 또는 크기를 줄이면서 인터넷상에 빠르게 표시하기 위한 영상 등등이 바로 그것이다. 특히, 의료 영상, 높은 가치가 있는 예술 작품집 그리고 손실 없이 저장할 필요가 있는 영상들을 대상으로 네트워크를 통해 저장하거나 전송하기 위한 무손실

압축이 요구된다. 이러한 조건을 만족하는 영상은 인터넷 온라인상에서 대개 GIF 파일 형태로 이용된다. 그러나 압축 기법으로 LZW 알고리즘을 사용하는 GIF 형태의 영상은 특허권 문제로 사용의 제한이 있었다. 이처럼 특허와 관련된 많은 문제를 안고 있는 LZW 알고리즘이지만 최근에는 거의 모든 특허가 만료됨에 따라 자유로운 연구, 개발을 기대해 볼 수는 있다<sup>[2]</sup>. 하지만 특허 문제로 인해 팔레트 기반의 영상에 대해서 LZW 알고리즘의 변형이나 더 탁월한 방법을 이용한 무손실 압축의 개발 및 연구 대신, 인덱스로 구성된 형태의 영상에서는 인

\* 전주대학교 교양학부 (you.kangsoo@gmail.com),

\*\* 순천제일대학 컴퓨터학과 (hjyoo@suncheon.ac.kr)

\*\*\* 한양대학교 정보통신대학 디지털미디어 연구실 (esjang@hanyang.ac.kr)

논문번호 : KICS2005-07-285, 접수일자 : 2005년 7월 13일

접한 인덱스간의 차이를 최소화하여 어떠한 정보의 손실 없이 압축을 행하는 연구가 2000년도에 시작되었으며 Zeng<sup>[3]</sup>의 알고리즘이나 Phino<sup>[4]</sup>의 알고리즘이 대표적인 것들이다.

본 논문에서는 인덱스간의 차이를 이용하는 참고 문헌 [3],[4]와 같은 알고리즘과는 달리, 무손실 압축 효율을 높이기 위하여 이웃하는 인덱스 쌍을 순차적으로 비교하면서 발생빈도에 따른 순위를 결정하는 방법을 제안한다. 이 방법은 원 영상의 인덱스 정보를 발생빈도수를 이용한 순위 정보로 전환한다. 이때 RIAC<sup>[2]</sup> (Ranked Indexing with Arithmetic Coding) 알고리즘을 적용적으로 처리하면서 순위를 결정할 위한 알고리즘을 제안한다.

따라서 본 논문에서는 인접하는 인덱스 사이의 변화를 측정하고 그 측정된 통계적 발생 빈도의 순위를 기반으로 하는 재인덱싱 알고리즘을 이용함으로써 순위 기반으로 재구성된 영상은 많은 수의 픽셀들이 높은 순위에 집중될 수 있고, 초기 인덱스 영상에서의 색상 수보다 순위 영상에서의 순위가 적게 되는 경향이 있다. 이러한 특징은 산술 부호화의 효율을 높일 수 있는 근거가 된다.

물론, 현재 알려진 무손실 압축으로는 JPEG2000<sup>[5]</sup>, JPEG-LS<sup>[6]</sup>, CALIC<sup>[7]</sup> 등이 있다. 이들 모두는 휘도 성분 기반 영상을 처리한다. 그러나 본 논문에서 제안한 순위 규칙을 적용한 적응적 RIAC 알고리즘에서의 영상 성능 평가는 모두 인덱스 칼라 영상이다. 그러므로 칼라 영상을 휘도 성분의 그레이 레벨을 갖는 영상으로 전환할 필요 없이 인덱스 칼라 영상 자체를 그대로 처리 가능하다는 장점이 있다. 또한 복잡도에 관해서  $O(M_2 \log M)$  연산을 사용하여 제안한 순위 규칙 적용 재인덱싱 방법을 구현하는 것이 가능하다.

## II. 재인덱싱 기법

인덱스 행렬을 적절히 변화시키는 한 인덱스 영상의 어떠한 정보도 잃지 않으면서 그 인덱스 행렬을 재배열할 수 있다는 근거에서 연구가 진행되고 있다. 이러한 연구를 칼라 재인덱싱 알고리즘이라고 한다. 이 연구는 서로 다른  $M$  개의 색상을 가지는 팔레트 기반의 인덱스 영상에서, 압축 효율을 최적화하기 위한 인덱스 배열을 찾는 방법이  $M!$ 의 시간 복잡도(time complexity)를 가지는 N-P(Non Deterministic Polynomial time) complete 문제로 알려져 있다<sup>[1][4]</sup>.

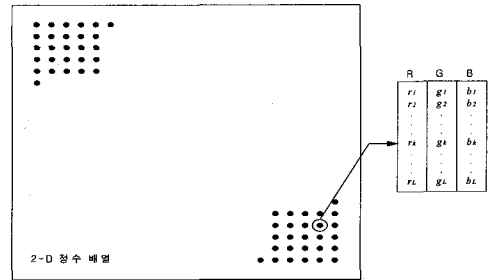


그림 1. 인덱스 영상의 요소 : 정수 행렬과 칼라 맵

그림 1은 인덱스 영상의 요소가 정수 행렬(인덱스 행렬)과 칼라 맵으로 구성되어 있다는 것을 설명한다<sup>[8]</sup>.

### 2.1 Zeng의 알고리즘

칼라 재인덱싱과 관련된 연구가 2000년 Zeng<sup>[3]</sup>에 의해 소개되었으며 Zeng의 알고리즘은 인덱스 영상에서의 인접하는 픽셀에 대한 인덱스 차를 전체적으로 감소시켜 영상의 인덱스 분포를 보다 완만하게 함으로써 압축 효율의 향상을 기대할 수 있다.

이 알고리즘의 핵심은 주어진 영상에서 교차계수(cross-count)  $C(S_i, S_j)$ 를 찾는 것이다. 이것은 색상 심볼  $S_i$  다음에 바로 이어서  $S_j$ 가 나오는 경우가 몇 번이나 발생했나를 측정하는 것이다. 교차계수를 바탕으로  $S_i$  색상에서 다른 색상으로 변화하는 경우가 모두 몇 번이나 발생했는가를 측정할 수 적교차계수를 식 (1)에 의해 계산한다.

$$C_i = \sum_{j=0, j \neq i}^{M-1} C(S_i, S_j) \quad (1)$$

누적교차계수가 가장 큰 색상 심볼  $C_i$ 인  $S_{max}$ 를 찾아  $L_0$ 로 표시하고 공간적으로 정렬된 심볼 묶음  $P$ 에 할당하면  $P=\{L_0\}$ 이 된다.  $P$ 의 크기를  $N$ 이라 하면 현재 상태에서는  $L_0$  하나만 들어 있으므로  $N=1$ 이다. 이후 단계에서 색상 심볼들이 차례로 왼쪽이나 오른쪽에 추가된다. 모든 색상 심볼이  $P$ 에 채워지게 되면 알고리즘은 종료되고  $P$ 에서의 색상 심볼 순서가 새로운 인덱스가 된다.

심볼 묶음  $P$ 의 왼쪽에 들어갈 가장 적당한 색상 심볼을 선택하기 위하여 현재 심볼 묶음  $P$ 의 가장 왼쪽에 위치한 심볼로 변화하는 경향이 가장 큰 심볼을 고른다. 그 이유는 새로운 색상 인덱스에서 서로 인접하게 되어 인덱스의 변화가 줄어들게 되기 때문이다. 심볼 묶음  $P$ 에 들어가지 못하고 남아 있

는 색상 심볼들에 대하여 다음과 같은 전위 함수에 대한 식을 적용하여 심볼 묶음  $P$ 에 들어갈 후보 심볼을 결정한다.

$$D_i = \sum_{j=0}^N w(N, j)C(S_i, L_j) \quad (2)$$

이때 가중치 함수  $w(N, j)$ 가 사용되는데  $L_j$ 가 심볼 묶음  $P$ 의 가장 왼쪽 심볼일 때 가장 크고 오른쪽으로 갈수록 값이 작아지는 함수이다. 여기에서  $L_j$ 와 가장 왼쪽 심볼과의 떨어진 거리가  $d(N, j)$ 라고 하면 가중치 함수는 식 (3)과 같다.

$$w(N, j) = \log_2\left(1 + \frac{1}{d(N, j)}\right) \quad (3)$$

현재 심볼 묶음  $P$ 의 왼쪽에 있는 색상으로 변화하는 경향이 가장 큰 색상 심볼이 계산되면  $D_i$ 에 해당하는  $S_i$ 를  $S_{Lmax}$ 로 정한다. 마찬가지로 방법으로 심볼 묶음  $P$ 의 오른쪽에 있는 색상으로 변화하는 경향이 가장 큰 색상 심볼을 계산하여  $S_{Rmax}$ 로 정한다.

식 (2)의 결과 값  $D_i$ 가 가장 큰  $S_i$ 를  $S_{Lmax}$  또는  $S_{Rmax}$ 로 선택하여  $L_N$ 으로 표시하고 심볼 묶음  $P$ 의 왼쪽 끝이나 오른쪽 끝에  $L_N$ 을 할당한다. 그리고  $N$ 을 1 증가 시킨다. 모든 색상 심볼이 처리될 때까지 위의 과정을 반복한다.

심볼 묶음  $P$ 에 들어 있는 색상 심볼을 왼쪽부터 순서대로  $0, 1, \dots, M-1$ 까지 할당하여 새로운 색상 인덱스로 대체한다. 즉, 재구성된 인덱스 영상은  $S_i$ 에 할당된 새로운 인덱스 값을 가진 초기 인덱스 값  $i$ 를 대체함으로써 생성된다.

### 2.2 Pinho의 알고리즘

2004년 Pinho<sup>[4]</sup>는 앞에서 살펴본 Zeng<sup>[3]</sup>의 알고리즘을 수정하여 엔트로피 효율을 높일 수 있는 알고리즘을 제안하였다. 이 알고리즘은 Zeng이 제안한 것처럼 인덱스 사이의 전체적인 차이를 줄이기 위하여 식 (2)와 식 (3)을 다음과 같이 수정한 것이다.

$$\begin{aligned} D_L(S_i, N) &= \sum_{j=0}^{N-1} \log_2 P(j+1)C(S_i, L_j) \\ &\quad - \sum_{j=0}^{N-1} \log_2 P(j+2)C(S_i, L_j) \quad (4) \\ &= \sum_{j=0}^{N-1} \log_2 \frac{P(j+1)}{P(j+2)} C(S_i, L_j) \end{aligned}$$

$$\begin{aligned} D_R(S_i, N) &= \sum_{j=0}^{N-1} \log_2 P(N-j)C(S_i, L_j) \\ &\quad - \sum_{j=0}^{N-1} \log_2 P(N-j+1)C(S_i, L_j) \\ &= \sum_{j=0}^{N-1} \log_2 \frac{P(N-j)}{P(N-j+1)} C(S_i, L_j) \quad (5) \end{aligned}$$

이제 초기 입력 영상에서 물리적으로 떨어진 거리를 고려하여 식 (3)과 같은 가중치 함수를 다음과 같이 정의한다. 여기에서  $P(k)$ 는 아직도 할당되지 않고 남아 있는 색상 심볼의 발생확률을 의미한다.

$$a_j = \log_2 \frac{P(j+1)}{P(j+2)} \quad (6)$$

$$a_{N-j-1} = \log_2 \frac{P(N-j)}{P(N-j+1)} \quad (7)$$

지금까지 살펴 본, Zeng<sup>[3]</sup>과 Pinho<sup>[4]</sup>에 의해 제안된 재인덱싱 알고리즘은 모두 인덱스 쌍에 대한 발생빈도를 측정하여 인접한 인덱스들의 차이를 줄이기 위한 것이다. 여기에서 초기 인덱스의 배열을 재구성하는 최적의 알고리즘을 구하기 위해 경험적 실험(heuristic strategy)을 통해야만 한다. 또한 인덱스의 재배열로 인해 생성된 영상은 초기 인덱스 영상과 발생빈도가 같으므로 엔트로피 부호화의 효율을 기대할 수 없다.

따라서 본 논문에서는 인접하는 인덱스 사이의 변화를 측정하고 그 측정한 통계적 발생빈도의 순위를 기반으로 산술 부호화의 효율을 높일 수 있도록 재인덱싱 알고리즘을 제안한다.

## III. 제안한 방법

원래의 인덱스 영상을 순위 영상으로 변환한 알고리즘<sup>[2]</sup>에서는 팔레트 정보 이외의 순위에 대한 별도의 정보를 전송해야 한다. 이 전송 정보를 제거하기 위하여 적응적인 순위 기반의 개선된 알고리즘에서의 순위를 적용시키는 몇 가지 규칙을 제안한다.

### 3.1 적응적 순위 기반의 재인덱싱 알고리즘

적응적으로 순위를 적용시키는 과정은 다음과 같다. 먼저, 크기가  $n \times m$ 이고  $M$ 개의 색상을 갖는 초기 인덱스 영상에서의 인덱스 값  $I = \{I_1, I_2, \dots, I_M\}$ , 색상 정보  $S = \{S_1, S_2, \dots, S_M\}$ 와 일대일 대응한다. 이때의 각 픽셀에 해당하는 인덱스 값들

을 모두 1차원  $P = (P_0, P_1, \dots, P_{n \times m})$ 로 표현한다. 이때  $P_0$ 는 색상 팔레트에 정의된 첫 번째 인덱스 값으로 정의한다. 즉,  $P_0 = I_1$ . 각 원소  $P_i$ , ( $1 \leq i < n \times m$ )는 영상의 인덱스 값들을 첫 행부터 왼쪽에서부터 순차적으로 읽어가는 경우  $i$ 번째에 위치한 인덱스 값을 나타낸다.

발생빈도행렬 CCM (Co-occurrence Matrix)의 첫 단계 행렬인  $C^1$ 의 모든 원소들을 0으로 초기화한다. 이제 집합  $P$ 로부터  $P^i = \{P_j | P_j \in P, j \leq i\}$ 를 정의한다. 이는  $P^i$ 가  $P_0$ 부터  $P_i$ 까지의 원소들을 포함한다는 것을 의미한다. 첫 번째  $P^1 = (P_0, P_1)$ 가 1번 발생하였다는 정보를 CCM에 저장하고 이를 행렬  $C^1$ 으로 표현한다. 행렬  $C^1$ 를 토대로 순위를 나타내는 순위행렬 CRM (Co-occurrence Rank Matrix)  $R^1$ 을 작성한다.

두 번째 ( $P_1, P_2$ )에 대한 발생횟수를 CCM ( $C^2$ )에 적용시켜 갱신하고 CRM ( $R^2$ ) 또한 갱신한다. 만약,  $P^1$ 의 인덱스 값이  $P^2$ 와 같다면 발생횟수를 누적한다. 여기에서 CCM과 CRM은 각각 행렬  $C$ ,  $R$ 로 표현한다. 다음과 같은 조건을 만족한다.

$$c^i(j, k) = \{c^i(j, k) | 0 \leq k \leq M\} \quad (8)$$

여기에서  $C^i$ 는 행렬  $C$ 에 대한  $i$ 번째 카운트 행렬을,  $C^i(j, k)$ 는  $C^i$ 의 임의의 원소를 나타낸다.

같은 방법으로,  $P^{n \times m} = (P_{(n-1) \times (m-1)}, P_{n \times m})$ 까지 처리하면 즉, 마지막 순서쌍까지 처리하면 초기 인덱스 영상에 대한 순위 영상이 구해진다. 그림 2는  $M=4$ 로 구성된 즉, 네 가지의 색상으로 구성된 실제 칼라 영상과 그에 대한 인덱스 및 제안한 알고리즘에 의해 변화된 순위 영상을 표시하며, 그림 3과 그림 4는 알고리즘의 두 번째 및 세 번째 단계에서의 각 행렬과 순위를 표현한 것이다.

### 3.2 순위 결정을 위한 방법

발생빈도행렬 CCM에서의 순위를 구하기 위해 앞에서 살펴 본 적응적 순위 기반의 재인덱싱 알고리즘에서는 초기 영상에서의 인덱스 행렬처럼 먼저 위치한 인덱스에 해당하는 원소의 순위를 높게 하였다. 이때 순위를 결정하기 위한, 즉 순위행렬  $R$ 을 구하기 위한 네 가지 방법을 제안한다. 이는 누적 카운트 된 행렬  $C$ 의  $i$ 번째 행에 있는 원소 즉,

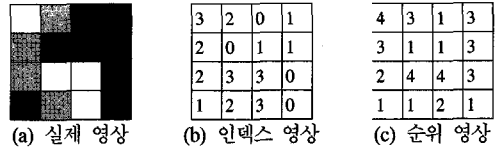


그림 2. 4가지 색상으로 구성된 4x4 크기의 인덱스 영상

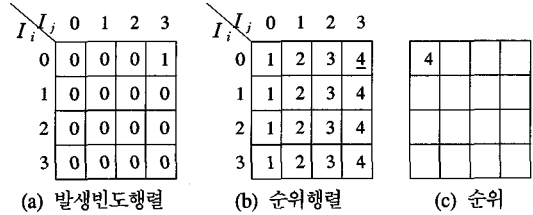


그림 3. 그림 2(b)의 영상을 그림 2(c)로 변환하기 위한 첫 번째 단계

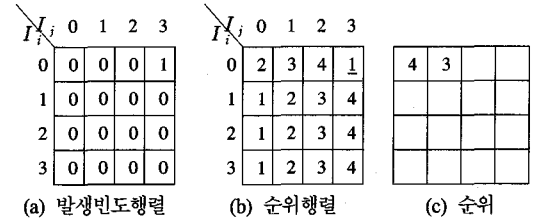


그림 4. 그림 2(b)의 영상을 그림 2(c)로 변환하기 위한 두 번째 단계

인접한 인덱스의 발생빈도를 측정한 값들 중 동일한 값이 있을 경우에 거리에 따른 순위 적용을 고려한 것이다. 순위 영상을 구성하는 순위에 따라 압축 효율이 달라질 수 있기 때문에 순위 결정을 위한 방법을 적용시켜 각각 네 개의 적응적 순위 기반의 재인덱싱 알고리즘을 제안한다. 이때 순위 결정을 위한 네 가지 방법은 다음과 같다.

방법 1. 이 방법은 그림 2(a), 그림 3(a)에서 임의의 행에 있는 원소들을 내림차순 정렬하여 순위를 매긴다. 이때 동일한 값들이 존재하는 경우가 발생하게 되며 이렇게 동일한 원소들의 순위를 결정하기 위하여 그림 2(b), 그림 3(b)와 같이 순차적으로 먼저 위치한 원소의 인덱스를 우선순위가 높게 한 것이다.

방법 2. 방법 1과 마찬가지로 방법 2는 행 단위의 발생빈도수들을 내림차순으로 정렬할 때 동일한 원소 값들이 발생하는 경우, 동일한 값들에 대한 순위를 결정하기 위해 단위행렬  $E$ 에서처럼 주대각

선 상에 존재하는 원소에 해당하는 위치에 우선순위를 두고 주대각선과 가까울수록 높은 순위를 결정하는 방법이다. 이는 실험에서 사용하는 영상의 최대 색상 수가 256이며 또한 일반적인 인덱싱 영상에서의 최대 색상 수가 256이므로 그림 2 및 그림 3에서 사용한 행렬들을 최대 256×256의 정사각형 행렬로 봤을 때, 단위행렬의 성질을 이용한 것이다. 인덱싱 영상에 사용된 색상 수  $M$ 에 따라 발생빈도행렬과 그에 대한 순위행렬의 크기가  $M \times M$ 으로 정해진다. 따라서 크로네커의 델타  $\delta_{ij}$  (단,  $i = j$ )에 해당되는 위치에 존재하는 발생빈도 값이 가장 우선순위가 높으며  $\delta_{ij}$ 의 위치와 가까울수록 순위가 우선한다. 단위행렬에서의 원소 값이 '1'인 곳에서의 위치부터 거리를 계산하는 식은 다음과 같다. 여기에서  $M$ 은 영상에 사용된 색상 수를 나타내며 최대치는 256이다.

$$\widehat{c}_{ij} = g_j + \frac{M-1-|i-j|}{M} \quad (9)$$

방법 3. 발생빈도가 높은 원소의 주위에 위치한 원소가 높은 우선순위를 갖는다. 예를 들면, 임의의 행  $i$ 에서 발생빈도가  $j$  번째가 가장 크고 다음으로 큰 발생빈도가  $j+k$  번째에 위치할 때, 순위는  $j$ 가 '1' 이고 그 다음 순위는  $j+k$ 가 '2' 이다. 만약  $j-1$ (또는  $j+1$ )번째와  $j+k-1$ (또는  $j+k+1$ )번째의 위치에 있는 발생빈도가 동일하다면 전체적인 순위는  $j$ ,  $j-1$ (또는  $j+1$ ),  $j+k-1$ (또는  $j+k+1$ )이 된다. 여기에서  $j$ 나  $k$ 는 임의의 열을 나타내며 다음의 조건을 만족한다.

$$\{j, \forall 0 \leq j < M-1, j \in N\} \quad (10)$$

방법 3은 다음 식과 같이 우선순위를 결정한다.

$$\widehat{c}_{ij} = g_j + \frac{\sum_{n=1}^M c_{in} \omega(j-n)}{\sum_{i=1}^N c_{ii}} \quad (11)$$

여기에서 가중치 함수  $\omega(n)$ 은 아래의 식으로 표현된다.

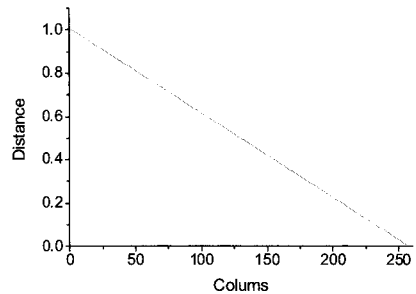
$$\omega(n) = \left(1 - \frac{|n|}{M-1}\right)^2 \quad (12)$$

방법 4. 이 방법은 앞에서 소개한 두 가지 방법 즉, 방법 2와 방법 3을 절충한 것이다. 즉, 식 (11)

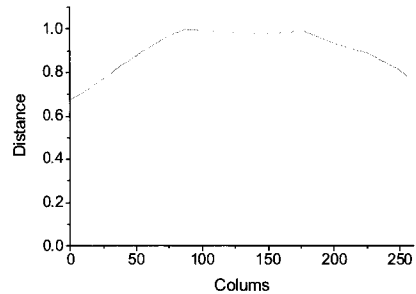
에서의 분모가 어떤 문턱치  $T$ 보다 크면 방법 2를 적용하고  $T$ 보다 작거나 같으면 방법 3을 적용하여 우선순위를 결정한다. 식 (11)의 분모는 임의의 행에서 사용된 색상 발생빈도를 측정할 값이다.

$$M = \sum_{i=1}^N c_{ii} \begin{cases} > T, \text{ 방법 2} \\ \leq T, \text{ 방법 3} \end{cases} \quad (13)$$

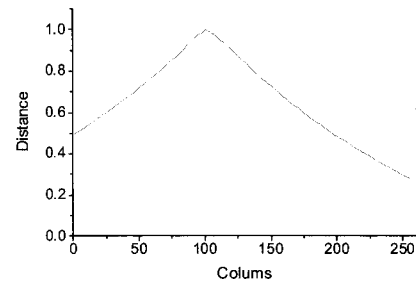
방법 2, 방법 3과 방법 4에 의해 거리를 계산한 결과의 그래프는 그림 5와 같다.



(a)



(b)



(c)

그림 5. 임의의 행에서 거리에 따른 순위; (a) 방법 2, (b) 방법 3, (c) 방법 4

#### IV. 실험 및 결과

제안한 알고리즘의 압축 성능을 평가하기 위하여

모두 16 개의 인덱스 영상을 적용해 보았다. 이들 영상을 두 가지 부류로 나누어 실험한다. 하나는 컴퓨터 그래픽스에 의해 생성된 8 개의 인공적인 팔레트 기반의 인덱스 영상이고 다른 하나는 실제 칼라 영상을 인덱스 모드로 변환한 8 개의 인덱스 영상이다. 실험에 사용된 영상은 12~256 사이의 색상을 가지고 있다. 또한 이들 영상은 모두 기존의 재인덱싱 알고리즘에 관한 실험을 위해 사용된 대표적인 영상이다<sup>9)</sup>.

제안한 알고리즘을 통해 실험한 결과와 Zeng<sup>13)</sup> 및 Pinho<sup>14)</sup>에 의해 제안된 재인덱싱 알고리즘의 결과들과 비교하였다. 무손실 압축 성능 평가는 압축률(CR ; Compression Ration)에 의해 계산된 비트율(bpp ; bits per pixel)에 의해 이루어진다. 표 1과 표 2에 각 영상에 대한 알고리즘 적용 결과의 bpp를 표시하였다. 특히 A-3에서는 식 (11)의 문턱치  $T$ 를 50으로 하여 실험을 하였다. 이는 사용된 색상의 수가 50 정도면 통계적으로 영상의 거의 모든 픽셀들에 대한 순위가 결정되기 때문이다.

표 1. 기존의 알고리즘과 제안한 알고리즘에서의 실험 결과 (인덱스 영상, 단위 : bpp)

영상	Zeng	Pinho	Proposed method			
			A-1	A-2	A-3	AC-4
clegg	5.863	5.456	4.175	4.159	4.166	4.161
cwheel	3.058	2.878	2.857	2.843	2.845	2.845
descent	2.943	2.854	2.816	2.678	2.758	2.755
fractal	6.193	5.828	5.268	5.236	5.245	5.235
frymire	3.619	3.376	2.414	2.408	2.409	2.409
party8	0.318	0.318	0.289	0.287	0.287	0.287
serrano	3.393	3.273	2.379	2.369	2.373	2.373
yahoo	1.798	1.789	1.816	1.753	1.761	1.761
평균 bpp	3.40	3.22	2.75	2.72	2.73	2.73

표 2. 기존의 알고리즘과 제안한 알고리즘에서의 실험 결과 (실제 칼라 영상, 단위 : bpp)

영상	Zeng	Pinho	Proposed method			
			A-1	A-2	A-3	A-4
airplane	5.056	4.445	4.282	4.232	4.234	4.223
anemone	5.806	4.966	4.326	4.296	4.301	4.296
baboon	7.097	6.496	6.101	6.058	6.075	6.052
boat	6.048	5.823	5.321	5.273	5.278	5.269
house	5.180	4.854	4.801	4.716	4.706	4.704
lena	5.710	5.049	4.055	4.036	4.042	4.041
monarch	4.325	3.917	3.233	3.217	3.222	3.221
peppers	5.544	5.019	4.016	3.989	3.995	3.994
평균 bpp	5.596	5.07	4.52	4.48	4.48	4.47

표 1에서 알 수 있듯이, Clegg 영상을 각 방법에서 실험한 결과, Pinho의 알고리즘보다 각각 1.29 bpp, 1.30 bpp, 1.29 bpp, 1.30 bpp의 비트 절감 효율을 보였다. 마찬가지로 표 2에서, Lena 영상은 1.00 bpp, 1.02 bpp, 1.01 bpp, 1.01 bpp의 비트 절감 효율이 있음을 알 수 있다. 표 1과 표 2에서 A-1, A-2, A-3, A-4는 적용적 RIAC 알고리즘에 순위 결정을 위한 방법 1, 방법 2, 방법 3, 방법 4를 각각 적용한 것을 의미한다. 표 1과 표 2에서 알 수 있듯이, 순위 결정을 위한 네 가지의 방법을 적용한 결과, 영상의 전반적인 특성을 반영한 인덱스 모드의 실제 칼라 영상에서 팔레트 기반의 인덱스 영상보다 좋은 압축 효율을 보였다.

이와 같이 실험에 사용된 초기 인덱스 영상과 그에 따른 히스토그램 그리고 제안한 알고리즘에 의해 생성된 순위 영상에 대한 히스토그램을 그림 6에 표시하여 비교하였다. 초기 인덱스 영상에서의 인덱스 분포와 비교할 때 순위 영상에서의 순위 분포는 한쪽으로 치우쳐 있음을 알 수 있고 같은 순위에 있는 픽셀들이 더 많으므로 압축 효율을 기대할 수 있다.

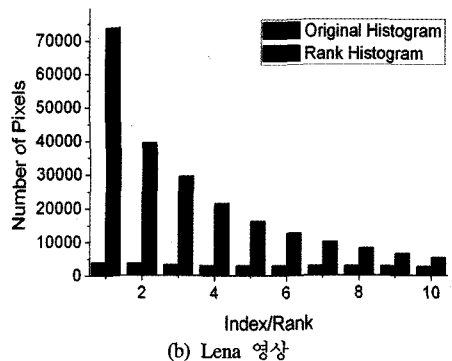
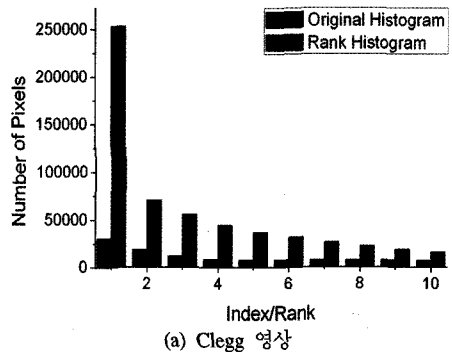


그림 6. 원 영상과 순위 영상의 상위 10 개 bins의 히스토그램 비교

## V. 결론

본 논문에서는 팔레트 정보의 인덱스를 재배열해도 손실 없이 압축되어질 수 있다는 특성을 이용하여 인덱스 영상을 인접한 인덱스 사이의 발생빈도를 측정한 후, 그에 대한 순위를 결정하여 초기 인덱스 영상을 순위 영상으로 바꾸는 알고리즘을 제시하였다. 이렇게 바뀐 순위 영상은 다음과 같은 이유에 의해 산술 부호화를 적용하였다.

특히 발생빈도에 따른 순위를 결정하기 위하여 네 가지의 경우를 정의하여 적용적 순위 기반 재인덱싱 알고리즘에 적용하였다. 이는 같은 색상의 인덱스가 발생한 주위의 픽셀에서의 다른 색상의 인덱스가 빈번히 발생할 수 있다는 특성을 고려한 것이다.

본 논문에서 제안한 알고리즘을 Zeng<sup>[3]</sup>과 Pinho<sup>[4]</sup>에 의해 제안된 알고리즘과 비교·실험하였다. 제안한 A-1, A-2, A-3, A-4 알고리즘에서는 픽셀 당 비트가 Pinho의 알고리즘에 비해 인덱스 영상에서 평균적으로 0.47 bpp, 0.50 bpp, 0.49 bpp, 0.49 bpp 까지 절약할 수 있는 결과를 얻었다. 또한 인덱스 모드의 실제 칼라 영상에서는 기존의 알고리즘에 비해 평균적으로 0.56 bpp, 0.60 bpp, 0.59 bpp, 0.60 bpp만큼의 절감 효율을 보였다.

따라서 영상의 손실을 허용하지 않는 의학 분야, 군대 및 우주에서 사용되는 위성 영상 및 원격 탐사를 통한 고정밀 영상 해석 등 무손실 압축을 요구하는 응용 분야에 적용가능하다 할 수 있다.

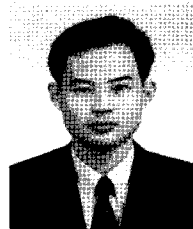
## 참고 문헌

- [1] S. Battiato, G. Gallo, G. Impoco and F. Stanco, "A Color Reindexing Algorithm for Lossless Compression of Digital Images," Proc. of IEEE Int. Computer Graphics, pp. 104-108, 2001.
- [2] 유강수, 이한정, 장의선,곽훈성, "산술부호화를 이용한 인덱스 칼라 이미지에서의 효율적인 무손실 압축 방법," 한국통신학회논문지, vol. 30, no. 1C, pp. 35-43, 2005.
- [3] W. Zeng, J. Li and S. Lei, "An Efficient Color Re-Indexing Scheme for Palette-based Compression," Proc. IEEE int. Image Processing, ICIP-2000., vol. III, pp. 476-479, 2000.

- [4] A.J. Pinho and A.J.R. Neves, "A Survey on Palette Reordering Methods for Improving the Compression of Color-Indexed Images," *IEEE Trans. on Image Processing*, vol. 13, issue 11, pp. 1411-1418, 2004.
- [5] M. Rabbani, R. Joshi, "An Overview of the JPEG 2000 Still Image Compression Standard," *Signal Processing: Image Communication*, vol 17, no. 1, pp. 3-48, 2002.
- [6] M.J. Weinberger, G. Seroussi and G. Shapiro, "The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS," *IEEE Trans. on Image Processing*, vol. 9, no. 8, pp. 1309-1324, 2000.
- [7] Xiaolin Wu, "Context-Based, Adaptive, Lossless Image Coding", *IEEE Trans. on Communications*, vol. 45, no. 4, 1977.
- [8] Ze-Nian Li, Mark S. Drew, *Fundamentals of Multimedia*, New Jersey: Prentice Hall, 2004.
- [9] A.J. Pinho, <ftp://ftp.ieeta.pt/%7Eap/images>

유 강 수 (Kang Soo You)

정회원



1991년 2월 전북대학교 컴퓨터공학과(공학사)  
 1994년 2월 전북대학교 컴퓨터공학과(공학석사)  
 2005년 2월 전북대학교 영상공학과(공학박사)  
 1996년~현재 전주대학교 교양

학부 객원교수

<관심분야> 영상처리, JPEG, MPEG, 영상검색, 멀티미디어

유 희 진 (Hee Jin Yoo)

정회원



1990년 2월 원광대학교 전자계산공학과(공학사)  
 1992년 2월 홍익대학교 전자계산학과(이학석사)  
 2000년 8월 홍익대학교 전자계산학과(이학박사)  
 2002년~현재 순천제일대학 컴

퓨터과학과 조교수

<관심분야> 영상처리, 설계자동화

장 의 선 (Euee S. Jang)

정회원



1991년 2월 전북대학교 컴퓨터  
공학과(공학사)

1994년 6월 미국 버팔로 뉴욕  
주립대학(공학석사)

1996년 2월 미국 버팔로 뉴욕  
주립대학(공학박사)

1996년~2002년 MPEG SNHC

Chair, 삼성종합기술원 책임연구원

1997년~2000년 MPEG-4 Project Editor

2000년~현재 IEEE ICME member, MPEG Forum  
전문위원, 한국방송공학회 편집위원, 한양대학교  
정보통신대학 부교수

<관심분야> 컴퓨터 그래픽스, 고화질 미디어, 3DAV,  
무손실 영상 압축, MPEG/JPEG