論文96-33B-6-9

이산 프랙탈 보간법을 이용한 영상 부호화 알고리듬

(An Image Coding Algorithm using Fractal Interpolation Method)

李 昇 玄 *, 鄭 賢 民 **, 尹 鐸 鉉 *, 崔 一 相 *, 朴 圭 泰 *

(Seung Hyeon Rhee, Hyun Meen Jung, Taek Hyun Yun, Illsang Choi and Kyu Tae Park)

요 약

블럭을 기반으로 한 기존의 영상 부호화기들은 일반적으로 하나의 블럭을 부호화할 때 그 블럭 내부의 데이터만을 이용하기 때문에, 블럭의 경계면에서 이질성이 나타나게 된다. 이러한 블럭 현상은 특히 낮은 비트율에서 화질에 큰 영향을 미친다. 이는 DCT를 기반으로 하는 변환 부호화는 물론, 프랙탈 이론에 의한 종래의 방법들에서도 마찬가지이다. 제안한 방법에서는 이를 개선하기 위하여, 블럭의 꼭지점들을 보간점으로 설정하고 이를 기준으로 블럭 내부의 화소값을 반복 함수계에 의해 합성하는 프랙탈 보간법을 이용한다. 이때, 인접한 블럭들은 보간점을 공유하기 때문에, 블럭 경계면에서의 화소값의 변화를 줄임으로써 종래의 블럭 부호화기가 가지는 문제점을 보완할 수 있다. 또한 일반적으로 쓰이는 정방형 블럭 대신 제안한 방법에 적합한 직각 이등변 삼각형의 블럭을 사용하며, 계층적 분할을 수행할 때는 현재 블럭의 1/2 크기를 갖는 두개의 삼각형으로 나눈다. 여기에, 각 데이터들이 가지는 통계적 특성을 이용하여 산술 부호화(arithmetic coding)를 수행한다. 이와 같은 방법을 통하여 블럭 현상의 감소는 물론 복호화 과정에서 수렴 속도가 증가하는 효과도 얻을 수 있다.

Abstract

In this paper, an image coding technique using fractal interpolation is proposed. Similar to the conventional methods, an image is partitioned into blocks and each block is coded independently. However, an interpolation point is shared by its neighboring blocks. This means that each block can use all its interpolation points with minimal increase of new data. For a simple implementation, triangular blocks are used instead of square blocks and new coefficients are defined. Data obtained in the encoding process have statistical characteristics suitable for entropy coding, and thus arithmetic coding is performed for improving the compression efficiency. The results of the proposed coder in comparison with those of a conventional coder show that the interpolation method reduces block effect caused by a memoryless block coder, especially at low bit rates. This improvement is due to sharing of information between adjacent blocks. Moreover, the number of iteration required in decoding process is reduced since more information is used to decode each block.

* 正會員, 延世大學校 電子工學科

(Dept. of Electronic Eng., Yonsei Univ.)

** 正會員, 韓國通信 無線通信研究所 無線通信網研究室

(Network Technology Section, Wireless Communication Research Lab., KT)

接受日字:1995年4月22日, 수정완료일:1996年5月18日

I. 서 론

영상 정보의 저장 및 전송에 대한 요구가 커지면서 이를 효율적으로 압축하는 기술에 대한 연구가 많이 이루어지고 있다. 중복성(redundancy)을 제거함으로써 영상을 부호화하는 방법으로는 DCT를 바탕으로 한 변환 부호화가 주류를 이루고 있으나 최근 프랙탈 이론을 응용하려는 시도가 활발히 이루어지고 있다.

Barnsley가 제안한 반복 함수계(iterated function system, IFS)는 자기 유사성을 가지는 영상을 표현하는 데 있어서 적은 양의 정보로도 매우 복잡한 시각적 특성을 묘사할 수 있음을 보여 주었다[1]. Sierpinski triangle이나 fern의 예에서 결정 프랙탈 영상(deterministic fractal image)의 표현에 IFS가 매우 효율적임을 알 수 있다.

영상을 IFS로 표현하는 것은 주어진 IFS로부터 프랙탈 영상을 얻어내는 것과는 반대의 문제이다. 이는 주어진 영상과 IFS가 수렴하는 고정점의 차가 최소가 되게 하는 IFS를 구하는 것이다. 그러나 IFS를 알아야 고정점을 구할 수 있으므로 고정점과 주어진 영상의 비교를 통하여 IFS를 설계하는 것은 불가능하다. 콜라지 이론(collage theorem)은 고정점 대신 주어진 영상을 한 번 변환한 결과와 영상을 비교함으로써 IFS를 구할 수 있음을 보인 것이다[2].

Jacquin은 자기 유사성이 없는 일반적인 영상을 부호화함에 있어서, 영상을 블럭으로 분할함으로써 얻어지는 블럭 간의 유사성을 이용하였다[3]. 이는 영상의 중복성을 이용해 원영상을 자기 유사 영상으로 근사화한 것으로, 일반 영상에 대하여 프랙탈 부호화 이론을 적용할 수 있음을 보인 것이다. Jacquin은 전체 영상을 균일한 크기의 정방형 블럭인 치역(range) 블럭과 넓이가 이의 네 배인 정의역(domain) 블럭으로 나누고 정의역 블럭의 집합으로부터 각 치역 블럭에 가장 유사하게 변환될 수 있는 블럭을 찾아 그 블럭의 위치와 변환을 부호화하였다. 수축 변환 과정을, 정의역 블럭을 치역블럭에 대응시키는 부분(geometric part)과 블럭 내부의 화소값을 변화시키는 부분(massic part)의 두 단계로 나누었으며, 특히 후자의 과정에서 정의역의 탐색 대상을 다양화함으로써 화질을 향상시켰다. 또한, 각 블럭을 속성에 따라 분류함으로써 블럭 탐색 과정의 효율성을 높이고 근사도에 따라 블럭을 네 개의 부블럭(sub-block)으로 분할하여 다시 부호화함으로써 복잡한 블럭의 화질을 향상시키고자 하였다.

영상의 유사성은 IFS 부호화기의 성능을 결정하는 데 중요한 요소이다. 유사성을 향상시키기 위해서는 영상의 분할이 효율적으로 이루어질 필요가 있으며 Fisher는 그 방안으로 quadtree 분할, HV 분할, 삼각형 분할 등을 제안하였다[4]. 이와 같이 계층적 분할을 수행할 경우, 분할 방법이 복잡하고 분할 횟수가 증가할수록 화질은 향상되지만 부호화 비용도 역시 증가하게 된다.

본 논문에서는 블럭 기반 프랙탈 부호화를 수행하면서, 각 블럭의 부호화에 있어서는 프랙탈 보간법을 이용한다. 2장에서는 일반적인 데이터열에 대한 기존의 프랙탈 보간법을 소개하고, 3장에서는 이를 영상 데이터에 적용하기 위한 방법을 제안한다. 여기에는 삼각형 및 이진 트리 분할에 의한 반복 부호화가 도입되며, IFS의 각 계수가 가지는 통계적 특성을 이용해 엔트로피 부호화를 수행한다. 4장에서는 실험을 통하여 이러한 방법이 영상을 블럭 단위로 부호화함으로써 야기되는 블럭 현상을 줄이는 데 효과적임을 보이고, 5장에서 결론을 맺는다.

Ⅱ. 프랙탈 보간법

프랙탈 보간법은 보간점(interpolation point) 사이의 데이터를 생성하는 데 IFS를 이용하는 것이다. 이러한 방법으로는 Mazel이 제안한, 자기 유사 프랙탈 모델(self-affine fractal model), 즉 선형 프랙탈 보간법(linear fractal interpolation)과 이를 일반 데이터에 적용하기 위한 구간 자기 유사 프랙탈 모델(piecewise self-affine fractal model)이 있다[5].

선형 프랙탈 보간법에서는 실수 값을 갖는 데이터의 집합을 $\{(u_n, v_n): n = 0, \cdots, N: u_n < u_{n+1}\}$으로 나타내고 보간점을 $\{(x_i, y_i): i = 0, \cdots, M: M \le N\}$으로 나타낸다. 여기서 u_n은 데이터의 인덱스(index)이고 v_n은 인덱스 u_n에서의 데이터 값이다. $(x_0, y_0) = (u_0, v_0)$이고 $(x_M, y_M) = (u_N, v_N)$이며 보간점은 데이터 값 중에서 선택된다. xy 평면에서의 보간 함수(interpolation function)는 다음과 같은 M개의 유사 변환으로 정의한다.

$$w_i\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a_i & 0 \\ c_i & d_i \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e_i \\ f_i \end{pmatrix} \quad i = 1, 2, \cdots, M \tag{1}$$

여기서, d_i는 변환 w_i의 수축 인자로서 구간 $(-1, 1)$에 있는 실수 값을 가져야 하며, 각 유사 변환은 다음과 같이 전체 데이터의 양끝 점을 w_i가 생성하는 구간의 양끝 점에 대응시켜야 한다.

$$w_i\begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \begin{pmatrix} x_{i-1} \\ y_{i-1} \end{pmatrix}\text{이고, } w_i\begin{pmatrix} x_M \\ y_M \end{pmatrix} = \begin{pmatrix} x_i \\ y_i \end{pmatrix} \quad i = 1, 2, \cdots, M$$

이와 같이, 함수가 자기 유사성을 가지는 경우에는, 함수를 여러 구간으로 나눈 후 전 영역을 각 구간에 대응시키는 선형 프랙탈 보간법을 이용할 수 있다. 그러나, 함수가 자기 유사성을 가지지 않는 경우에는 구간 자기 유사성을 이용하여 함수의 일부 구간을 다른 구간에 대응시킴으로써 선형 프랙탈 보간법을 일반화할 수 있다.

유사변환 w가 (I, y_I)와 (F, y_F)를 양끝 점으로 하는 데이터의 집합 D를 보간점 (p, y_p)와 (q, y_q)를 양끝 점으로 하는 데이터의 집합 R에 대응시킨다면, w는 식 (1)로부터 다음과 같이 나타낼 수 있다.

$$w\begin{pmatrix} n \\ y_n \end{pmatrix} = \begin{pmatrix} a & 0 \\ c & d \end{pmatrix}\begin{pmatrix} n \\ y_n \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} = \begin{pmatrix} m \\ w_m \end{pmatrix} \quad \begin{matrix} n = I, \cdots, F \\ m = p, \cdots, q \end{matrix} \tag{2}$$

변환에 의해 얻어진 데이터의 집합 R'의 양끝 점은 다음과 같이 보간점 (p, y_p), (q, y_q)와 일치해야 한다.

$$w\begin{pmatrix} I \\ y_I \end{pmatrix} = \begin{pmatrix} p \\ y_p \end{pmatrix}, \quad w\begin{pmatrix} F \\ y_F \end{pmatrix} = \begin{pmatrix} q \\ y_q \end{pmatrix} \tag{3}$$

또한 수축 변환을 보장하기 위하여, 각 구간의 크기에 대해서는 다음의 조건이 성립해야 한다.

$$F - I > q - p$$

식 (2)와 (3)으로부터, 각각의 R에 대하여 변환 w의 각 계수를 다음과 같이 나타낼 수 있다.

$$a = \frac{q-p}{F-I}$$

$$e = \frac{Fp - Iq}{F-I}$$

$$c = \frac{y_q - y_p}{F-I} - d\,\frac{y_F - y_I}{F-I} \tag{4}$$

$$f = \frac{Fy_p - Iy_q}{F-I} - d\,\frac{Fy_I - Iy_F}{F-I} \tag{5}$$

여기서 a와 e는 D의 위치에 의해 결정되며 c, d, f는 D를 R로 가장 유사하게 변환하도록 결정해야 한다. c, d, f를 구하기 위하여, R'의 w_m과 이에 대응되는 R의 y_m 사이의 오차 함수 E를 다음과 같이 정의한다.

$$E = \sum_{n=p}^{q} [\, w_m - y_m]^2 \tag{6}$$

식 (2)에서 $w_m = cn + dy_n + f$이므로 식 (4)와 (5)에 의해 w_m을 다시 쓰면 다음을 얻는다.

$$w_m = \left(\frac{y_q - y_p}{F-I}\right)n + \frac{Fy_p - Iy_q}{F-I} - d\left(\frac{y_F - y_I}{F-I}\,n + \frac{Fy_I - Iy_F}{F-I} - y_n\right)$$

$\frac{F-n}{F-I}$을 ξ_n로 나타내면

$$w_m = y_q(1-\xi_n) + y_p\xi_n - d(\xi_n y_I + (1-\xi_n)y_F - y_n) \tag{7}$$

이 되고, 다시

$$A_n = y_n - [\,\xi_n y_I + (1-\xi_n)y_F] \tag{8}$$

$$B_n = y_m - [\,\xi_n y_p + (1-\xi_n)y_q] \tag{9}$$

로 정의하면, E는 식 (6), (7), (8), (9)로부터 다음과 같이 쓸 수 있다.

$$E = \sum_{n=I}^{F} (A_n d - B_n)^2$$

이때, E가 최소가 되게 하는 수축 인자 d는

$$d = \frac{\sum_{n=I}^{F} B_n A_n}{\sum_{n=I}^{F} A_n^2}$$

로 구할 수 있다.

Ⅲ. 제안한 방법

제안한 방법에서는 프랙탈 보간법의 개념을 확장하여 계조도를 갖는 영상 데이터에 적용한다. 즉, 치역 블럭의 꼭지점을 보간점으로 하고 정의역 블럭의 화소값을 수축 변환하여 치역 블럭 내부의 화소값을 복원한다.

전체 변환 과정은 정의역 블럭의 화소 위치를 변환하여 치역 블럭의 화소에 대응시키는 과정과 정의역 블럭의 화소값을 대응되는 치역 블럭의 화소값에 가깝게 변환하는 과정의 2단계로 나누어 수행한다. 화소값의 변환을 위하여 블럭 내부에 각 화소의 수평 위치, 수직 위치, 계조도를 각각 x축, y축, z축에 대응시키는 좌표계를 설정하고 다음과 같이 화소 (x, y)의 값 z를 새로운 값 $\hat{z}$로 변환한다.

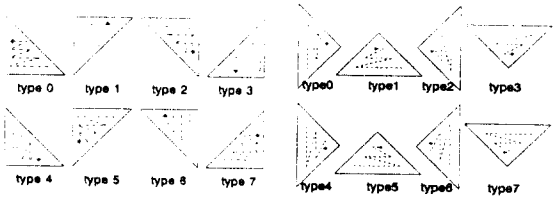
$$\hat{z} = Ix + Jy + Kz + L \tag{10}$$

여기서 K는 식 (2)의 d에 해당하는 수축 인자이다.

Mazel의 방법에서는 식 (4)와 (5)에 보인 세 개의 계수 c, d, f를 결정하기 위해서 보간점에 대한 두 개의 조건과 수축 인자 d에 대한 최소 오차 기준을 이용하였다. 제안한 방법에서는 네 개의 계수 I, J, K, L을 결정하기 위해서 세 개의 조건과 수축 인자 K에 대한 최소 오차 기준을 이용한다. 따라서, 하나의 블럭을 처리하는 데 세 개의 보간점이 필요하다. 보간점은 각 블럭의 꼭지점이므로 이에는 삼각형 블럭을 이용하는 것이 적절하고, 따라서 기존의 방법과는 다른 새로운 형태의 정의역 및 치역 블럭을 설정한다.

1. 블럭의 설정과 화소의 위치 변환

삼각형 블럭은 그림 1에서 보인 바와 같이 두 가지 방법으로 설정한다. 원영상은 일단 정방형 블럭으로 분할되고, 각각의 블럭은 다시 대각선에 의해 두 개(단계 0) 또는 네 개(단계 1)의 직각 이등변 삼각형 블럭들로 재구성된다.



(a) (b)

그림 1. (a) 단계 0의 삼각형 블럭, (b) 단계 1의 삼각형 블럭

Fig. 1. (a) Triangular blocks of level 0, (b) triangular blocks of level 1.

각 블럭 안에 표시된 방향은 각 블럭을 처리할 때 데이터를 읽거나 쓰는 순서를 나타낸 것이다. 여기서 각 단계의 4번부터 7번까지의 형태는 0번부터 3번까지의 형태와 데이터를 취하는 순서만 다르다. 이는 각 이등변삼각형의 밑변에 대한 법선을 중심으로 데이터를 반전(reflect)시킨 형태로서 정의역 블럭을 다양화하여 영상의 유사성을 높이기 위한 것이며 정의역 블럭에만 쓰인다.

그림 2는 보간점을 선택하는 방법을 나타낸 것이다. 그림 2와 같이 4×4 크기의 블럭을 처리할 경우, 치역 블럭을 4×4 크기가 아닌 5×5 크기로 선택하여 인접한 치역 블럭이 한 개의 화소만큼 겹치게 설정하면 보간점을 공유하는 치역 블럭을 만들 수 있다. 이 때 x, y 각 방향으로 네 화소마다 하나의 정의역 블럭을 취하면서 8×8이 아닌 9×9의 크기로 하여 역시 한 화소만큼 겹치게 하면 치역 블럭의 보간점들의 집합이 정의역 블럭의 꼭지점들을 모두 포함하므로 정의역 블럭의 꼭지점에 대한 정보는 따로 전송할 필요가 없다. 그림 2에서 점선은 각 화소의 경계를 나타내고 가는 실선은 일반적인 치역 블럭을, 굵은 실선은 확장된 치역 블럭을 나타낸다. 또한 진하게 표시된 화소는 보간점을 표시한 것이다.

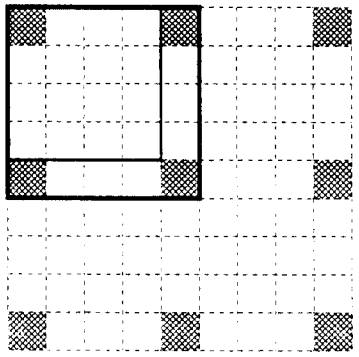


그림 2. 보간점의 선택

Fig. 2. Selection of interpolation points.

일반적으로는 8×8 크기의 정의역 블럭을 4×4 크기의 치역 블럭으로 변환하는 방법을 쓰지만, 제안한 방법에서는 9×9 크기의 정의역 블럭을 5×5 크기의 치역 블럭으로 변환하였다. 또한, 기존의 방법에서는 수축 변환시 정의역 블럭의 4개의 화소값의 평균을 취하였으나 여기서는 그림 3과 같이 x, y 각 방향으로 두 화소마다 하나를 취하는 간단한 방법을 썼다.

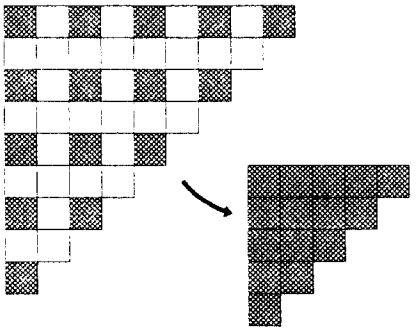


그림 3. 삼각형 블럭의 수축변환

Fig. 3. Contractive transformation of a triangular block.

2. 화소 값의 변환과 수축 인자의 결정

여기서는 화소의 위치 변환을 제외한 화소값의 변환

만을 고려하기로 한다. 즉, 이미 평행 이동 및 1/2 수축 변환과 반전 및 회전 등, 화소의 위치 변환이 이루어진 결과에 대하여 화소값의 변환만 수행한다.

삼각형 블럭의 세 꼭지점 $(x_0, y_0), (x_1, y_1), (x_2, y_2)$의 화소값을 각각 z_0, z_1, z_2라 하고 여기에 식 (10)의 변환을 수행하면 다음과 같다.

$$\hat{z}_0 = Ix_0 + Jy_0 + Kz_0 + L$$

$$\hat{z}_1 = Ix_1 + Jy_1 + Kz_1 + L$$

$$\hat{z}_2 = Ix_2 + Jy_2 + Kz_2 + L$$

$\hat{z}_0$, $\hat{z}_1$, $\hat{z}_2$에 대응되는 치역 블럭의 화소를 각각 $\hat{z}_0$, $\hat{z}_1$, $\hat{z}_2$라 하고 이들을 서로 일치시키면 다음과 같이 계수 I, J, L을 화소의 수축 인자인 K에 대하여 나타낼 수 있다.

$$I = -\frac{\hat{z}_0(y_1 - y_2) + \hat{z}_1(y_2 - y_0) + \hat{z}_2(y_0 - y_1)}{y_0(x_1 - x_2) + y_1(x_2 - x_0) + y_2(x_0 - x_1)} + K\frac{z_0(y_1 - y_2) + z_1(y_2 - y_0) + z_2(y_0 - y_1)}{y_0(x_1 - x_2) + y_1(x_2 - x_0) + y_2(x_0 - x_1)} \tag{11}$$

$$J = \frac{\hat{z}_0(x_1 - x_2) + \hat{z}_1(x_2 - x_0) + \hat{z}_2(x_0 - x_1)}{y_0(x_1 - x_2) + y_1(x_2 - x_0) + y_2(x_0 - x_1)} - K\frac{z_0(x_1 - x_2) + z_1(x_2 - x_0) + z_2(x_0 - x_1)}{y_0(x_1 - x_2) + y_1(x_2 - x_0) + y_2(x_0 - x_1)} \tag{12}$$

$$L = -\frac{\hat{z}_0(x_2y_1 - x_1y_2) + \hat{z}_1(x_0y_2 - x_2y_0) + \hat{z}_2(x_1y_0 - x_0y_1)}{y_0(x_1 - x_2) + y_1(x_2 - x_0) + y_2(x_0 - x_1)} - K\frac{z_0(x_2y_1 - x_1y_2) + z_1(x_0y_2 - x_2y_0) + z_2(x_1y_0 - x_0y_1)}{y_0(x_1 - x_2) + y_1(x_2 - x_0) + y_2(x_0 - x_1)} \tag{13}$$

s_I, s_J, s_L 및 t_I, t_J, t_L을 써서 식 (11), (12), (13)을 간단히 하면 다음과 같다.

$$I = s_I - Kt_I \tag{14}$$

$$J = s_J - Kt_J \tag{15}$$

$$L = s_L - Kt_L \tag{16}$$

(x, y)에서의 화소값을 $z_{x,y}$로 나타내고 치역 블럭을 구성하는 화소의 집합을 B로 나타내면 변환에 의해 얻어진 영상 블럭과 이에 대응되는 원영상 블럭간의 오차는 다음과 같이 나타낼 수 있다.

$$E = \sum_{(x,y)\in B}(\hat{z}_{x,y} - \hat{z}_{x,y})^2 = \sum_{(x,y)\in B}(Ix + Jy + Kz_{x,y} + L - \hat{z}_{x,y})^2$$

이를 식 (14), (15), (16)을 이용하여 전개하면

$$E = \sum_{(x,y)\in B}\{(s_Ix + s_Jy + s_L - \hat{z}_{x,y}) - K(t_Ix + t_Jy + t_L - z_{x,y})\}^2$$

을 얻는다. 여기에 다시

$$S = s_Ix + s_Jy + s_L - \hat{z}_{x,y}$$

$$T = t_Ix + t_Jy + t_L - z_{x,y}$$

를 정의하여 간단히 하면

$$E = \sum_{(x,y)\in B}(S - KT)^2$$

이 되며, 다음과 같이 오차 함수 E를 최소화하는 K를 구할 수 있다.

$$\frac{\partial E}{\partial K} = 2\sum_{(x,y)\in B}(KT^2 - ST) = 0$$

$$K = \frac{\sum_{(x,y)\in B} ST}{\sum_{(x,y)\in B} T^2}$$

이렇게 구한 K는 복호화단으로 전송되며 이 값과 보간점 및 정의역 블럭의 꼭지점의 화소값으로부터 식 (11), (12), (13)에 의해 나머지 계수를 구한 후 이 값들이 정의하는 IFS에 의해 영상을 복원해 낸다. 이때 블럭 경계면의 화소들은 인접 블럭들이 공유하게 되는데 블럭 현상을 최소화하기 위하여 중복되는 값들의 평균을 취한다.

3. 가변 비트율 부호화를 위한 영상의 분할

영상을 부호화하는 과정에서 원영상과의 오차에 따라 영상의 분할 여부를 결정한다. 즉, 최적의 계수를 구한 상황에서도 복원 영상의 오차가 만족스러운 범위내에 있지 않으면 블럭의 크기를 더 작게 하여 다시 부호화를 시도한다. 정방형 블럭을 이용한 부호화의 경우 일반적으로, 어떤 크기의 블럭에 대해 부호화에 실패하면 다음 단계에서는 현재의 크기의 1/4에 해당하는 블럭을 설정하여 다시 부호화한다. 이를 통하여 화질의 향상을 보장하지는 않지만 블럭의 수가 증가함으로써 탐색 대상이 많아지고, 블럭의 크기가 작아질수록 속성이 단순해지므로 유사한 블럭이 얻어질 가능성이 커진다. 본 논문에서는 초기 단계의 블럭에 대하여 부호화를 한 결과, 복원 영상의 화질이 주어진 범위를 만족하지 않을 경우 분할하여 다시 부호화를 시도하였다. 이때 블럭의 분할은 일반적인 정방형 블럭과는 다른

방식으로 이루어진다. 그림 1에서 제시한 두 단계의 블럭 형태는 서로 번갈아 나타나며 한 단계의 블럭이 반으로 나뉘어진 결과는 다른 단계의 블럭 형태에 속함을 알 수 있다. 예를 들면 그림 4와 같이 단계 0의 0번 형태를 분할하면 단계 1의 0번 및 1번 형태가 된다.

그림 1에서 알 수 있듯이 단계 1의 형태는 경계면을 제외하면 겹치지 않는 4개의 블럭이 하나의 정방형 블럭을 구성하는 데 반하여 단계 0의 블럭은 0번과 2번 또는 1번과 3번의 블럭 쌍중 하나가 정방형 블럭을 구성한다. 이 점을 이용하여, 단계 0에서부터 처음 부호화를 시작할 때에는 0번과 2번, 1번과 3번의 블럭 쌍에 의한 부호화를 모두 수행한 후 그 결과에 따라 원영상과 더 유사하게 변환 가능한 것을 선택한다.

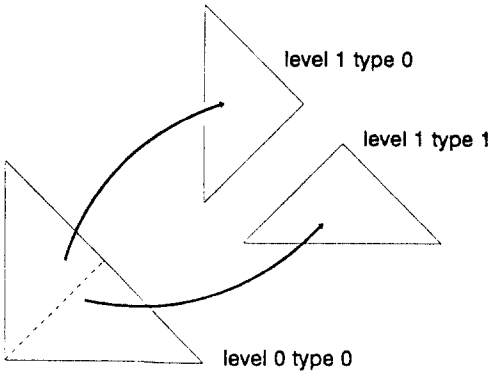


그림 4. 삼각형 블럭의 분할
Fig. 4. Splitting of a triangular block.

4. 압축 부호화 방법

원영상을 제안한 방법에 의해 부호화한 후 각 데이터의 통계적 특성을 이용하여 산술 부호화함으로써 압축 효과를 얻는다[6]. 제안한 방법에서 영상의 복원시 필요한 정보는 보간점 및 정의역 블럭의 꼭지점의 화소값, 정의역 블럭의 위치, 그 블럭을 치역 블럭으로 변환하기 위한 규칙을 정의하는 계수 I, J, K, L, 초기 치역 블럭의 분할 방법이다. 복호화단에서 요구되는 계수 I, J, L는 다른 데이터들로부터 구해 낼 수 있으므로 전송할 필요가 없으며, 보간점이 겹하게 되어 있는 정의역 블럭의 꼭지점 정보도 따로 전송할 필요가 없다.

그림 5에서 보인 바와 같이 보간점의 화소값은 중복성을 가지므로 DPCM을 수행하며 정의역 블럭의 위치는 대응되는 치역 블럭의 주변에 나타날 확률이 크므로 치역 블럭의 위치와의 차를 부호화 한다[7]. 초기 치역 블럭의 형태와 각 치역 블럭에 대응되는 정의역 블럭의 형태는 압축부호화에 이용할 수 있는 통계적 특성이 약하므로 각각 3비트, 1비트씩 할당하여 단순 비트열로 부호화한다. 계층적 분할을 행할 경우에는 분할 여부를 결정하기 위한 1비트의 정보를 추가한다. 끝으로 이렇게 처리한 값들을 산술 부호화하여 비트열로 만든다. 실계수인 K의 경우는 수축성을 보장하기 위해 그 범위를 (-1, 1)로 제한하고 이를 다시 적절한 구간으로 분할하여 균일 양자화한 후 산술 부호화한다.

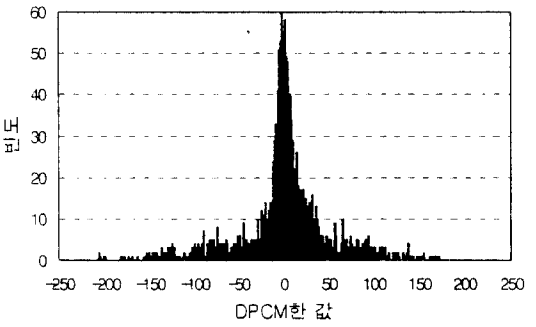


그림 5. DPCM한 보간점의 화소값의 분포
Fig. 5. Distribution of DPCMed values of interpolation points.

Ⅳ. 실험 및 결과

본 실험은 256단계의 계조도를 갖는 256×256 크기의 LENA 영상을 대상으로 하였으며, 압축률은 bpp (bits per pixel), 화질은 PSNR(peak-to-peak signal to noise ratio)에 의해 평가하였다.

그림 6은 제안한 방법에 의하여 1.0 bpp로 부호화한 영상을 다시 복원하는 과정을 보인 것이다. 16×16 크기의 정방형 블럭을 두 개의 삼각형 블럭으로 분할한 형태에서 시작하여 8×8 크기의 정방형 블럭을 네 개의 삼각형 블럭으로 분할한 형태에 이를 때까지 4단계로 분할 부호화하였다. 표 1은 몇가지 비트율에 대하여 부호화를 수행할 때, 각 단계에서 부호화된 블럭의 수를 나타낸 것이다. 각 단계에서 다음 단계로의 분할은 부호화 결과 얻어진 블럭과 원영상 블럭 간의 MSE (mean square error)가 주어진 허용 오차를 초과할 때 이루어진다. 비트율, 양자화 구간의 수, 허용 오차는 상호 영향을 주는 값이므로 조건에 따라 적절한 값을 설정해야 한다.

(a) (b)

그림 6. 제안한 방법에 의해 부호화된 영상의 복원 과정 (1.0 bpp)
(a) 1회 적용 후 (18.66 dB),
(b) 3회 적용 후 (31.14 dB).

Fig. 6. Decoding process of an image encoded by the proposed method (1.0 bpp)
(a) after 1st iteration (18.66 dB),
(b) after 3rd iteration (31.14 dB).

표 1. 각 단계에서 처리된 블럭의 수

Table 1. Number of blocks coded successfully at each stage.

비트율 (bpp)	허용 오차	양자화 구간의 수	부호화된 블럭의 수				전체 블럭의 수	PSNR
			단계 1	단계 2	단계 3	단계 4		
0.7	136.5	40	303	109	230	776	1418	29.25
0.8	83.7	47	237	143	305	1018	1703	30.23
1.0	28.6	50	144	143	371	1630	2288	31.17

K를 양자화할 때 구간의 크기를 작게 할수록 손실은 줄어드는 반면, 구간의 수가 증가하므로 정보량도 늘어 난다. 특히, 일정한 비트율을 유지할 경우 그림 7에서 보는 바와 같이, 양자화 구간의 수가 어느 정도 이상이면 오히려 압축 효율이 감소함을 알 수 있다. 따라서 양자화 구간의 수와 허용 오차의 설정에 따라 부호화 효율이 달라지며 최적의 값은 실험적으로 구한다.

본 실험에서 비교를 위한 종래의 방법으로 Jacquin이 제안한 방법을 개선하여 DCT 분류기를 사용한 것을 택하였다[8]. 그림 8은 이 방법과 제안한 방법에 의해 부호화한 영상의 복원 영상이며 그림 9는 그림 8에서 블럭 현상이 비교적 크게 나타나는 부분인 LENA의 어깨 부분을 확대한 것이다. 그림 10은 그림 9에서 실선으로 표시된 위치의 화소값들을 비교하여 보인 것이다. 이들로부터 제안한 방법에서 블럭 현상이 덜 나타남을 알 수 있다.

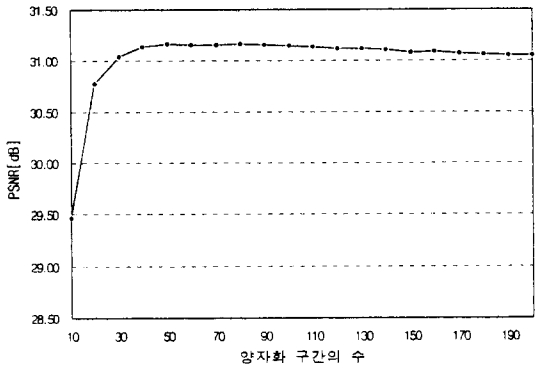
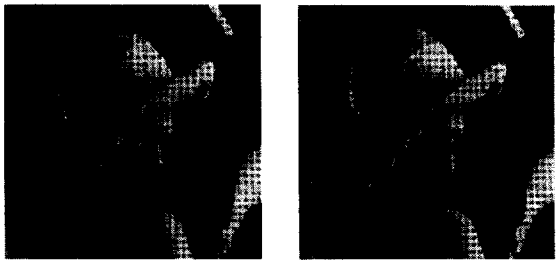


그림 7. 양자화 구간의 수와 화질의 관계 (1.0 bpp)

Fig. 7. Relationship between number of quantization levels and image quality (1.0 bpp).



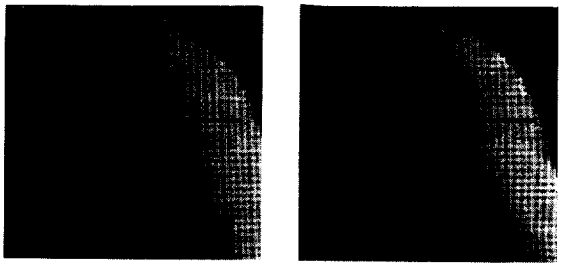
(a) (b)

그림 8. 복원 영상의 비교 (0.7 bpp)
(a) 종래의 방법 (27.27 dB),
(b) 제안한 방법 (29.25 dB).

Fig. 8. Comparison of decoded images (0.7 bpp)
(a) conventional method (27.27 dB),
(b) proposed method (29.25 dB).

Jacquin의 방법은 평탄 블럭으로 분류된 블럭에 대해서는 블럭의 평균값만을 전송한다. 이는 압축 효율을 높이는 데는 효과적이지만 낮은 비트율일수록 블럭 현상을 심하게 하는 주요 원인이 된다. 제안한 방법에서는 수축 인자와 휘도 천이 성분을 구하는 데 블럭 내부의 데이터만을 이용하지 않고 블럭의 꼭지점을 모두 사용한다. 하나의 블럭을 묘사하기 위하여 더 많은 데이터가 사용되는 반면 이들 데이터는 인접한 모든 블럭이 공유하기 때문에 전체적으로 데이터량의 증가는 기의 없다. 또한 제안한 방법에서는 계수 I와 J에 의해, 블럭 내에서의 화소의 상대적인 위치가 그 화소에 적용되는 변환을 결정하는 데 이용됨으로써 블럭 내에서의 화소값들은 보간점을 포함하는 곡면을 형성하게 된다. 따라서 블럭 경계면에서 급격한 화소값의 변화를

줄일 수 있다.



(a) (b)

그림 9. 복원영상의 비교
(a) 그림 8(a)의 LENA의 어깨 부분 (30.17 dB), (b) 그림 8(b)의 LENA의 어깨 부분 (31.69 dB).

Fig. 9. Comparison of decoded images
(a) shoulder of LENA in Fig. 8(a) (30.17 dB), (b) shoulder of LENA in Fig. 8(b) (31.69 dB).

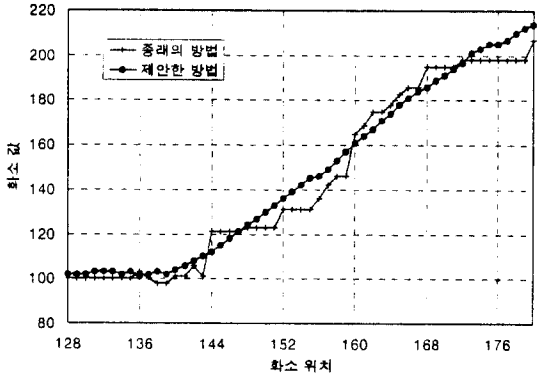


그림 10. 블럭 현상
Fig. 10. Block effect.

그림 11은 전체 영상을 복호화하는 과정에서 종래의 방법과 제안한 방법에 의해 반복 함수계를 적용할 때 각 반복 회수에 따른 수렴성을 측정한 것이다. 제안한 방법의 수렴속도가 빠름을 알 수 있다.

그림 12와 그림 13은 각각 종래의 방법과 제안한 방법에 의해 1.0 bpp로 부호화한 LENA의 어깨 부분을 복호화하는 과정을 보인 것이며 이로부터 두 가지 방법의 특성을 알 수 있다.

V. 결 론

본 논문에서는 프랙탈 보간법을 영상 데이터에 확장 적용하여 블럭 현상을 개선할 수 있음을 보였다. 종래의 방법은 블럭을 부호화하는 과정이 주변 블럭의 데이터와 무관하게 이루어지고 특히 평탄 블럭에서 블럭의 평균값만을 이용하였기 때문에 낮은 비트율에서 블럭 현상이 심하게 나타나는 문제점이 있었다.

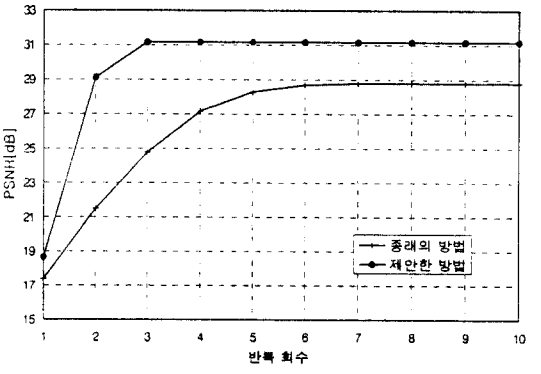


그림 11. 수렴 특성 (1.0 bpp)
Fig. 11. Convergency characteristic (1.0 bpp).

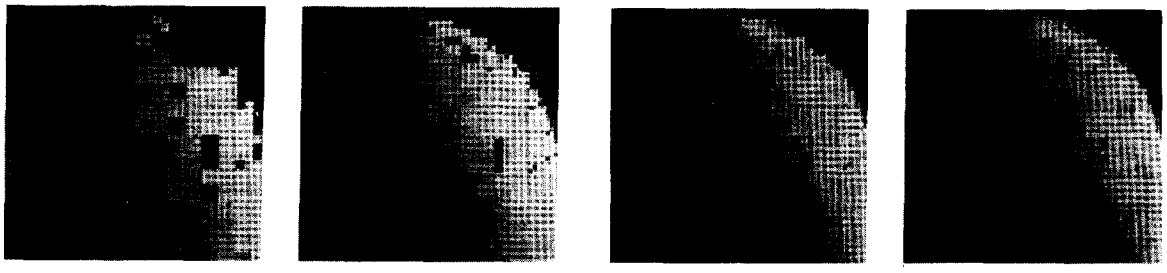
제안한 방법에 의한 복원 영상에서 블럭 현상이 적은 점과 복호화단에서의 수렴 속도가 빠른 것은 복호화단에 이용되는 보간점의 영향 때문이다. 제안한 방법에서 복호화 과정은 보간점을 가지는 상태에서 시작되며 각 치역 블럭은 그 보간점을 포함하는 곡면으로 빠르게 수렴한다. 또한 보간점을 인접 블럭들이 공유함으로써 블럭간의 이질성을 줄이기 때문에 특히 낮은 비트율에서 심하게 나타나던 블럭 현상을 완화할 수 있다.

본 연구에서 개선해야 할 점은 다음과 같다.

첫째, 제안한 방법의 경우, 계수 K가 실수이며 또한 다른 계수를 구하는 데 쓰이기 때문에 양자화에 의한 오차의 전파 문제를 고려해야 한다. 이것이 다른 계수를 구하기 위해 곱해지는 성분이므로 적은 단계로의 양자화시 화질에 미치는 영향이 커진다. 따라서 다른 계수를 K와 분리시키는 것이 바람직하나 이때는 비트율의 증가도 고려해야 한다.

둘째, 부호화에 소요되는 시간을 단축시켜야 한다. 부호책을 탐색하는 대부분의 방법이 안고 있는 공통의 문제라고 할 수 있는데 이미 연구된 바 있는, 정의역 블럭의 분류 방법을 도입하여 해결할 수 있을 것이다. 이에는 정의역 블럭들의 성격을 효과적으로 규정하고 분류, 적용할 수 있는 알고리듬의 개발이 필요하다. 또한 블럭간에 독립적으로 부호화가 이루어지므로 병렬처리 등의 하드웨어적 측면에서도 고려해 볼 수 있다.

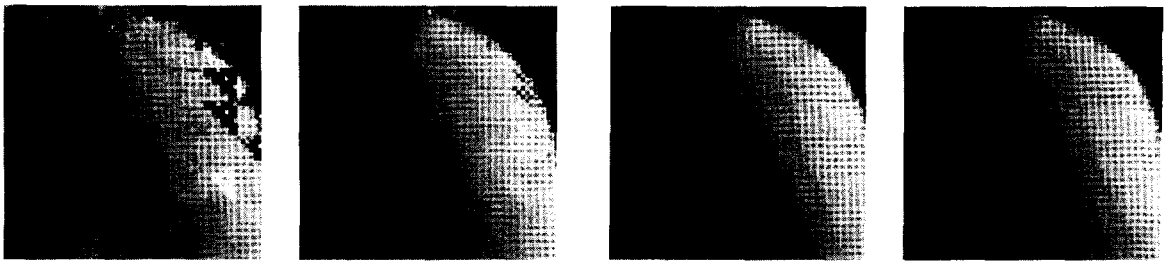
셋째, 부호화 과정에서 적응적으로 분할을 행할 경우



(a) (b) (c) (d)

그림 12. 종래의 방법에 의한 복원 과정
(a) 1회 적용 후 (19.70 dB), (b) 2회 적용 후 (23.67 dB), (c) 3회 적용 후 (27.50 dB), (d) 4회 적용 후 (31.02 dB).

Fig. 12. Decoding process by conventional method
(a) after 1st iteration (19.70 dB), (b) after 2nd iteration (23.67 dB), (c) after 3rd iteration (27.50 dB), (d) after 4th iteration (31.02 dB).



(a) (b) (c) (d)

그림 13. 제안한 방법에 의한 복원 과정
(a) 1회 적용 후 (18.50 dB), (b) 2회 적용 후 (29.31 dB), (c) 3회 적용 후 (35.02 dB), (d) 4회 적용 후 (35.02 dB).

Fig. 13. Decoding process by proposed method
(a) after 1st iteration (18.50 dB), (b) after 2nd iteration (29.31 dB), (c) after 3rd iteration (35.02 dB), (d) after 4th iteration (35.02 dB).

블럭의 복잡한 정도에 따라 효과적인 결과를 얻을 수 있다는 이점이 있으나 분할 여부를 결정하는 기준의 설정도 중요하다.

실제로 분할하여 더 작은 블럭으로 처리할 경우 정보량이 증가하는 것은 틀림없지만 화질의 향상을 보장하지는 않는다.

오히려 정보량의 증가와 함께 화질의 열화를 가져오는 역효과를 얻을 수도 있으므로 합리적인 기준의 설정이 요구된다.

이상과 같은 문제점들이 개선된다면 프랙탈 이론은 실용적인 면에서도 응용 가능성이 매우 큰 방법이다. 특히 본 논문에서 제안한 방법에 의한 블럭 현상의 개선은 인간의 시각 특성을 고려할 때 주관적 화질 향상에도 기여할 것이다.

참 고 문 헌

[1] M. F. Barnsley, *Fractals Everywhere*, Academic Press, San Diego, 1988.

[2] H. Peitgen and D. Saupe, *The Science of Fractal Images*, Springer-Verlag, New York, 1988.

[3] A. E. Jacquin, "Image Coding Based on a Fractal Theory of Iterated Contractive Image Transformations", *IEEE Trans. Image Process.*, vol. IP-1, pp. 18-30, January 1992.

[4] Y. Fisher, "Fractal Image Compression", *SIGGRAPH '92 course notes*, 1992.

[5] D. S. Mazel, "Using Iterated Function Systems to Model Discrete Sequences", *IEEE Trans. Signal Process.*, vol. 40, no. 7, pp. 1724-1734,

July 1992.
[6] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic Coding for Data Compression", *Communications of the ACM*, vol. 30, no. 6, pp. 520-540, June 1987.
[7] G. Vines and M. H. Hayes, III, "Adaptive IFS Image Coding with Proximity Maps", *Proc. ICASSP'93*, Minneapolis, vol. 5, pp. 349-352, April 1993.
[8] 윤택현, "프랙탈 이론에 근거한 영상 부호화기의 성능개선에 관한 연구", 연세대학교 대학원 석사 학위 논문, 1993년 6월

저 자 소 개

李 昇 玄(正會員)
1969년 11월 9일생. 1993년 2월 연세대학교 전자공학과 졸업(공학사). 1995년 2월 연세대학교 대학원 전자공학과 졸업(공학석사). 1995년 3월 ~ 현재 연세대학교 대학원 전자공학과 박사과정 재학. 주관심분야는 영상 신호 처리, 영상 부호화 등임

尹 鐸 鉉(正會員)
1963년 6월 3일생. 1986년 연세대학교 전자공학과 졸업(공학사). 1993년 연세대학교 대학원 전자공학과 졸업(공학석사). 1985년 ~ 1991년 금성사 연구원. 1993년 ~ 현재 연세대학교 대학원 전자공학과 박사과정 재학. 주관심분야는 영상압축, 영상처리임

朴 圭 泰(正會員) 第 31卷 B編 第 2號 參照
현재 연세대학교 전자공학과 교수

鄭 賢 民(正會員) 第 32卷 B編 第 11號 參照
현재 한국통신 무선통신연구소 무선통신망연구실 근무

崔 一 相(正會員)
1966년 4월 5일생. 1988년 2월 연세대학교 전자공학과 졸업(공학사). 1991년 2월 연세대학교 대학원 전자공학과 졸업(공학석사). 1991년 3월 ~ 현재 동대학원 박사과정 재학. 주관심분야는 신호 처리, 영상 부호화 등임