

論文98-35T-12-3

가변 블록 기반 프랙탈 영상 부호화

(Fractal Image Coding Based On Variable Block)

盧根洙*, 趙成桓**

(Keun Soo Roh and Seong Hwan Cho)

요 약

본 논문에서는 Barnsley가 제안한 반복 함수 체계(IFS)에 근거한 새로운 영상 압축 기술인 프랙탈 영상 부호화의 성능 개선 방법을 제시한다. 기존의 완전 탐색을 이용한 프랙탈 부호화 방법은 영상의 복원 화질이 다른 프랙탈 기법에 비해 좋은 결과를 얻을 수 있지만 탐색시간이 지나치게 오래 걸리고, 선형 유사 변환을 통하여 블록의 근사화를 수행하므로 복잡한 블록의 근사화가 어렵다는 단점이 있다. 따라서 본 논문에서는 쿼드 트리 분할을 사용하여 코딩 블록의 크기를 가변적으로 설정하도록 하여 복잡한 블록은 크기가 작은 블록으로 단순한 블록은 큰 블록으로 분할되도록 하였다. 이렇게 함으로써 블록의 근사화 정도도 높이고 변환의 수도 줄여 압축률의 개선을 가져왔다. 또한, 탐색 영역을 제한하여 부호화 시간이 짧으면서도 기존의 완전 탐색(full searching) 방법에 비하여 탐색 시간을 줄였으며 복원 영상의 주관적 평가의 화질은 더 우수했다. 그리고 매칭 블록의 탐색이 필요 없는 Monro 방법에 비해 부호화 시간이 다소 느리기는 하지만 화질 면에서 월등한 성능을 보였다.

Abstract

In this paper, we present new method of fractal image coding based on iterated function system(IFS) suggested by Barnsley. In previous Fractal coding using full searching algorithm, the quality of reconstructed image was better than other fractal coding method's, but it took a long time in that algorithm for searching domain blocks matched. And it is performed through linear affine transform, therefore it is difficult to approximate the complex range blocks. In this paper, using quadtree partitioning, complex blocks are divided into more smaller blocks, and simple blocks are merged to more larger blocks. So, we can get more precisely approximated range blocks and reduce the number of transformations. Hence, we have improved the compression ratio. In addition, we restrict the region of searching domains in order to reduce the searching time and coding time. Compared with full searching algorithm, we reduced coding time drastically, and quality of reconstructed image was better in terms subjective criteria. And compared with Monro's, our method is slower, but we could obtain a reconstructed image with better quality.

* 正會員, (株)韓火/情報通信研究所

(Hanwha Corp./Telecom.)

** 正會員, 동서울大學 電子計算機科

(Dept. of Computer Engineering, Dong Seoul College)

接受日字: 1998年8月31日, 수정완료일: 1998年12月10日

I. 서 론

기존의 영상 부호화 기법들은 대개 공간상이나 주파수상의 정보의 중복성을 줄이는 방법으로 압축을 수행해 왔다^[1-9]. 최근 프랙탈을 이용한 영상 부호화 방법들이 많은 관심 속에 활발히 연구되고 있다^[10-25]. 하지만, 프랙탈 영상 부호화 방법은 기존의 방법들과

는 다른 알고리즘들을 사용하고 있다. 즉, 임의의 초기 영상에 일정한 축소 변환을 반복적으로 적용하면 부호화하고자 하는 원영상으로 수렴하는 프랙탈 영상을 얻을 수 있다는 가정에서 출발하고 있다. 다시 말하면, 부호화할 영상을 생성하는 변환식을 찾아내어, 변환식의 파라미터만을 사용하여 부호화할 영상을 표현할 수 있다는 것이다. 그러므로 프랙탈 부호화란 부호화할 영상을 생성해 낼 수 있는 변환식을 찾아내는 것이다.

현재까지 발표된 프랙탈 부호화 방법들은 크게 두 가지 모델로 나눌 수 있다. 하나는 전체 변환 시스템(STS: Self Transformation System)이고 다른 하나는 부분 변환 시스템(PTS: Piecewise Transformation System)이다. STS는 전체 영상을 정의역으로 하고 자신의 부분 영역들을 치역으로 하는 변환을 통해 부호화하고자 하는 영상으로 근사화시키는 방법이다^[10-12, 22-24]. PTS는 영상을 여러 개의 중첩된 정의역 부분으로 나눈 후 중첩되지 않은 개개의 치역 부분들로 변환을 적용하여 서로를 근사시키는 방법을 말한다^[13-21, 25]. Barnsley가 처음 제안한 Iterated Function System(IFS)은 STS이다. IFS는 주로 컴퓨터 그래픽에서 구름이나 산, 지형과 같은 자연 경관을 실물처럼 표현하는 데 널리 쓰이며, 자연 영상 압축에 사용하려면 부호화 파라미터를 결정하는데 있어 인공 지능적인 요소를 필요로 하므로 좋은 성능의 부호화기를 구현하기가 어렵다. 1988년에 이르러 Jacquin이 PTS 모델을 사용하여 기존 부호화의 틀에서 크게 벗어나지 않는 알고리즘을 개발해 내었고, 이후 Fisher, Beaumont등이 Jacquin의 모델을 기반으로 여러 가지 좋은 결과들을 얻어내었다^[16, 17-21]. 그러나 이들 부호화 방법의 단점은 많은 블록을 찾기 위해 동일 분류에 속한 블록들의 완전 탐색이 필요하다는 것이다. 또 블록을 근사시키기 위해 선형 유사 변환(Liner Affine Transform)을 사용하는데 이와 같은 간단한 변환식만으로는 크기가 크고 복잡한 블록을 근사화시키기 어렵다^[13-21]. 이처럼 근사화 오차가 큰 블록을 다시 분할하는 방법을 택할 경우 변환식의 증가로 압축률 저하의 원인이 된다.

이러한 단점을 개선하기 위해 영상 블록의 복잡도에 적응적으로 부호화가 가능하도록 영상을 분할할 수 있는데, 일반적으로 영상 분할 방법에는 쿼드 트리 분할법, HV(Horizontal Vertical) 분할법, 삼각 분할법 등이 있다^[21]. 본 논문에서는 구현이 용이하면서도

비교적 영역 분할 특성이 뛰어난 쿼드 트리 분할을 이용하였다. 영상의 분할 정보는 쿼드 트리 코드를 사용하여 부가 정보의 양을 최소화하였다. 그리고 블록 분류화 기법을 사용하여 정의역 블록의 탐색 개수를 줄였고, 대개의 영상에서 치역 블록으로 축소 맵핑되는 정의역 블록이 치역 블록주위에 존재할 가능성이 높다고 가정하여 탐색 영역 제한을 통해 탐색 및 부호화 시간을 줄였다. 역변환을 통해 구해진 축소 맵핑 함수의 변환계수들은 양자화기를 통해 양자화하여 비트율을 낮추었다.

II. 반복 함수 체계

이 절에서는 모든 프랙탈 부호화기의 기반이 되는 IFS(Iterated Function System) 이론에 대하여 기술한다.

1. 정의 공간

공간 $I = [0,1]$ 과 그 확장 공간(Cartesian product) I^2 상에서 정의되는 모든 르베그 가측 실 함수(real Lebesgue measurable functions) $z = f(x, y)$ ($x, y, f(x, y) \in I^2$)로 이루어진 집합 F 와, 척도(metric, distortion measure) d 로 이루어진 척도 공간(metric space) (F, d) 를 사용하여 영상을 모델링한다. 여기서 f 는 (x, y) 인 점에서의 $[0,1]$ 로 정규화된 빛의 밝기를 나타내고, d 는 공간 F 상의 두 점 x, y 사이의 거리를 나타낼 수 있는 다음의 성질을 가진 함수이며, 척도(metric)라고 한다.

- $d(x, y) = d(y, x), \quad \forall x, y \in F$
- $0 < d(x, y) < \infty, Z \quad \forall x, y \in F \quad x \neq y$
- $d(x, y) = 0, \quad \forall x, y \in F \quad x = y$
- $d(x, y) \leq d(x, z) + d(x, y), \quad \forall x, y, z \in F$

2. 축소 변환(Contractive mapping)

이 절에서는 프랙탈 부호화에 가장 기본이 되는 고정점에 대한 정의, 축소 변환 매핑과 고정점 정리에 대해 서술한다^[12].

[정의 2.1] 고정점(Fixed Point)

$f: X \rightarrow X$ 인 함수가 있을 때, 한 점 $x_f \in X$ 가 $f(x_f) = x_f$ 이면, 이점 x_f 를 고정점이라 한다.

[정의 2.2] 축소 매핑 함수와 축소비

만약 $d(f(x), f(y)) \leq S \cdot d(x, y), \forall x, y \in X$ 을 만족하는 상수 S 가 존재(단, $0 \leq S < 1$)한다면, 함수 $f: X \rightarrow X$ 는 축소 매핑 함수라고 한다. 이때 S 를 축소비(contractivity factor)라고 한다. 그림 1에 축소 매핑 함수를 이용한 축소변환을 나타내었다.

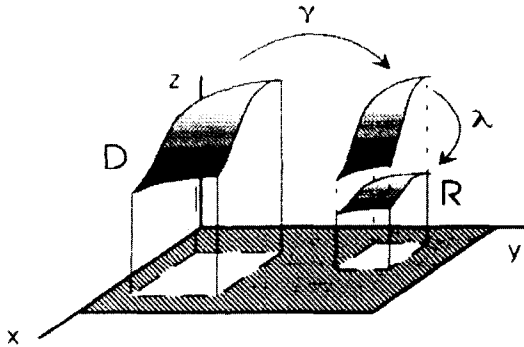


그림 1. 축소 변환
Fig. 1. Contraction transform.

[정리 2.1] 축소 매핑 정리

함수 $f: X \rightarrow X$ 가 완전 척도 공간 (X, d) 의 축소 매핑이라고 가정하면, 함수는 한 개의 고정점 $x_f \in X$ 를 갖고, 임의의 점 $x \in X$ 에서의 f^n 은 x_f 에 수렴한다.

$$\lim_{n \rightarrow \infty} f^n = x_f \quad (\text{각 } x \in X) \quad (1)$$

3. 역변환 문제(Inverse Problem)

위와 같은 공간에서 프랙탈 영상 부호화 작업은 부호화하고자 하는 영상을 고정점으로 갖는 축소 변환 W 를 찾아내는 작업이 된다. 변환 W 의 고정점을 $|W|$ 라고 할 때, 이 변환은 다음 정리를 만족한다.

[정리 2.2] 콜라주(Collage) 정리

축소비 S 인 축소변환 $W: F \rightarrow F$ 와 $f \in F$, 척도(metric) δ 에 대해

$$\delta(|W|, f) \leq \frac{1}{(1-S)} \delta(W(f), f) \quad (2)$$

가 성립한다.

위 정리는 주어진 영상 f 와 그 자신의 변환 $W(f)$ 와의 차이가 작으면 작을수록 고정점 $|W|$ 는 영상 f 에 근접함을 말해준다. 만약 f 가 W 의 고정점이라면 $W(f)$ 와 f 는 같다. 여기서 축소비 S 는 수렴의 속도

를 결정하며, 값이 작을수록 빠르게 수렴한다. 따라서 식 (2)와 같은 기준을 사용하여 IFS를 구성하고, 임의의 영상에 대해 IFS의 변환 W 를 반복적으로 적용해 가면 [정의 2.1]의 고정점 정의에 의해 부호화된 영상을 복원해낼 수 있다.

마지막으로 일반적인 형태의 변환이 갖는 성질에 대해 알아본다. 앞서 살펴본 바에 의하면 변환 W 를 이루는 개개의 부분 변환 w_i 들이 축소변환이면 W 는 축소(contraction)이며 유일한 고정점을 갖는다고 했다. 그러나 조건을 약하게 주어 개개의 모든 w_i 들이 반드시 축소변환이 아닐 때에도 이 정리는 성립한다. 이를 궁극적 축소(eventually contraction)라 하며 다음과 같다.

[정리 2.3] 궁극적 축소

$W: F \rightarrow F$ 에서 W 를 n 번 적용한 W^n 이 축소변환이 되는 양의 정수 n 이 존재하면 변환 W 는 궁극적 축소라 한다. 모든 축소변환은 궁극적 축소이지만 그 역은 성립하지 않는다.

이와 같이 확장된 개념의 축소변환에 대해서도, 다음 정리 2.4와 같은 일반화된 콜라주 정리가 성립한다.

[정리 2.4] 일반화된 콜라주 정리

$w_1 \dots w_n$ 이 I^2 을 채우고(tiling), $S_{\max} = \max\{S: S \text{는 변환 } w_i \text{의 축소비}\}$ 일 때 아래의 식 (3)이 성립한다. (여기서 σ 는 W^{0n} 의 축소비이다)

$$\delta(|W|, f) \leq \frac{1}{(1-\sigma)} \frac{1-S_{\max}^m}{1-S_{\max}} \delta(W(f), f) \quad (3)$$

이 정리가 의미하는 바는 다음과 같다. 개개의 변환 w_i 들의 모음 W 를 반복 적용한 합성변환 W^n 은 다음과 같은 변환들의 조합으로 이루어진다.

$$W^n = w_{i_1} \circ w_{i_2} \circ w_{i_3} \dots \circ w_{i_m} \quad (4)$$

그런데 $\cup w_i$ 는 I^2 을 채우므로(tiling) w_i 의 정의역은 또 다른 변환 w_j 의 치역이 된다. 그러므로 이 영역은 w_j 의 영향을 받게 된다. 따라서 비록 하나의 변환 W_i 가 축소변환이 아니라고 하더라도, 합성 변환 W^{0n} 은 많은 다른 부분 변환들이 섞여 형성되므로 다른 변환들이 충분히 작은 축소비를 가지고 있다면,

상호작용에 의해 전체적으로 축소변환이 된다는 것이다.

Ⅲ. 가변 블록 기반 프랙탈 부호화기

이 절에서는 가변 블록 기반 프랙탈 부호화기와 복호화기의 구현방법에 대해 설명한다.

1. 쿼드 트리 분할

가변 블록 분할 기법은 영역의 활동도에 따라 블록의 크기가 결정되므로 다양한 영상 영역에 대해 융통성 있게 적용할 수 있다^[7, 8].

쿼드 트리 분할법에는 Top-Down 방식과 Bottom-Up 방식이 있는데, 본 논문에서는 Top-Down에 비해 영상 분할이 빠른 Bottom-Up 방식을 사용하였다.

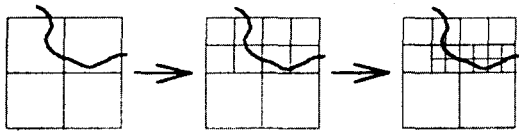


그림 2. Top-Down 방식
Fig. 2. Top-Down method.

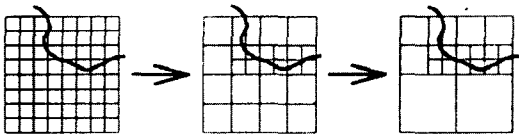


그림 3. Bottom-Up 방식
Fig. 3. Bottom-Up method.

그림 2와 그림 3에서 Bottom-Up 방식과 Top-Down 방식의 블록 병합과 분할의 차이를 보였다. 지금까지 일반적인 Bottom-Up 쿼드 트리 알고리즘은 개개의 화소들을 모두 단말 노드(terminal node)로 하여 병합여부를 결정하고 상위 노드로 올라가는 방식이었다. 하지만, 본 논문에서는 계산량을 줄이기 위해 최소 블록의 크기를 4×4, 최대 블록의 크기를 32×32로 제한하였다. 또한, 블록의 병합으로 발생할 수 있는 오차를 줄이기 위하여 각 레벨마다 블록 병합의 여부결정에 사용되는 임계값을 달리 설정하였다.

영상의 분할은 식 (5)를 이용한 동질성 검사를 통해 레벨 i 에서의 블록간의 오류가 임계값 T_i 보다 작으면 상위 블록으로 병합을 한다.

$$\bigcap_{i,m=0}^1 |x_i(k, l) - x_{i-1}(2k+j, 2l+m)| \leq T_i = \text{True} \quad (5)$$

$$x_i(k, l) = \frac{1}{4} \sum_{j=0}^1 \sum_{m=0}^1 x_{i-1}(2k+j, 2l+m) \quad (6)$$

$$x_0(k, l) = \frac{1}{16} \sum_{j=0}^3 \sum_{m=0}^3 x(4k+j, 4l+m) \quad (7)$$

여기서, $i=1, 2, 3$

$$k, l=0, \dots, 2^{6-i}-1$$

이고, T_i 는 레벨 i 에서의 임계값을 의미한다. 식 (6)에서 레벨 i 의 상위 블록의 값은 레벨 $i-1$ 인 하위 블록들의 평균값을 의미한다. 식 (7)에서 $x_0(k, l)$ 은 각 4×4블록들의 평균값을 나타낸다.

본 논문에서는 사용한 각 레벨간의 임계값 T_i 는 다음의 관계가 있다.

$$T_i = 2^{1-i} T_1 \quad (8)$$

식 (8)은 각 레벨의 임계값이 레벨이 증가할 때마다 이전 레벨의 임계값의 (1/2)값을 임계값으로 사용한다는 것을 의미한다.

T_1 값을 크게 설정하면 병합되는 블록의 수가 증가하여 압축율은 낮출 수 있지만 복원 영상의 화질이 열화된다. 그렇다고 T_1 값을 낮게 설정할 경우 병합 블록의 수가 증가하여 복원 영상의 화질은 개선되지만 압축률이 증가한다. 즉, T_1 값은 화질과 압축률의 Trade off를 통해 설정해야한다.

기존의 프랙탈 부호화기들은 대개 고정된 크기의 지역과 정의역 블록들을 사용하여 왔다. 하지만 고정된 크기의 블록들을 부호화에 사용할 경우 고주파 성분이 많이 포함된 복잡한 블록은 단순한 선형 유사 변환으로 근사화하기 어렵다. 그래서 기존의 부호화기들은 저주파 성분이 많은 블록에서의 근사화는 좋았지만 고주파 성분이 많은 블록에 대해서는 근사화 오차가 많이 발생하였다. 이러한 점을 개선하기 위해 영상의 복잡도에 따라 적응적으로 영상을 분할하여 저주파 성분이 많은 블록들은 더 큰 블록으로 병합하여 부호화하고 고주파 성분이 많은 블록들은 더 작은 블록으로 나누어 작은 블록에 대해 각기 근사화를 하기 위해 임력 영상을 중첩되지 않도록 분할한다.

이렇게 분할된 영상 정보는 그림 4와 같은 쿼드 트

리 구조 코드로서 표현할 수 있다. 이러한 형식으로 영상의 분할을 나타낼 경우 압축률의 효과를 증대시킬 수 있다. 그림 4에서 트리 구조 코드 1이 의미하는 바는 병합되지 않은 자 노드(child node)가 존재한다는 것을 말한다. 0은 병합이 더 이상 되지 않는 단말 노드를 의미한다.

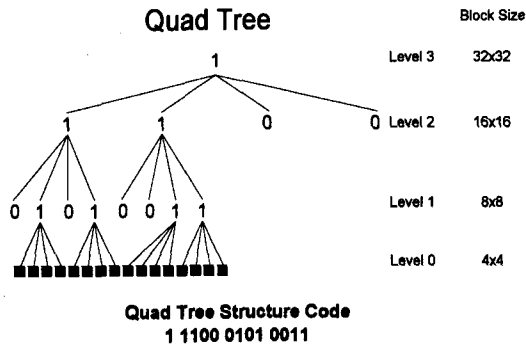


그림 4. 쿼드 트리 구조 코드를 이용한 분할 정보의 표현
Fig. 4. Partition information using quadtree structure code.

그림 4에 32×32 블록의 분할 예를 트리 구조 코드로 예를 들었다. 이 방법을 사용할 경우 기존의 변환 블록마다 분할 정보를 할당하는 방식에 비해 10:1 이상으로 비트를 줄일 수 있다.

2. 탐색 영역의 제한

본 논문에서는 블록 탐색에 앞서 부호화할 치역 블록과 매칭되는 정의역 블록이 근접한 위치에 존재한다고 가정하였다.^[24]

매칭 블록을 찾기 위해 본 논문에서는 쿼드 트리로 영상을 분할한 후 분할된 블록을 중심으로 해서 치역 블록의 크기를 R×R이라고 하고, 정의역 블록의 크기를 D×D라 할때, 탐색 영역의 범위는 2D×2D영역을 완전 탐색하여 매칭 블록을 찾는 방법을 사용하였다. 이때 2R=D의 관계가 성립한다. 그림 5에 치역과 매칭되는 도메인 블록을 탐색하는 범위에 대해 나타내었다. 실제 블록 매칭의 수행에 사용한 척도는 다음의 식 (9)와 같은 자승 평균(mean-square)을 사용하였다.

$$d(f, g) = \sum_{0 \leq x < N} (f(x) - g(x))^2 \quad (9)$$

식 (9)에서 f는 정의역 블록을, g는 치역 블록을

의미하며, N은 블록내의 화소의 개수, d는 척도를 나타낸다.

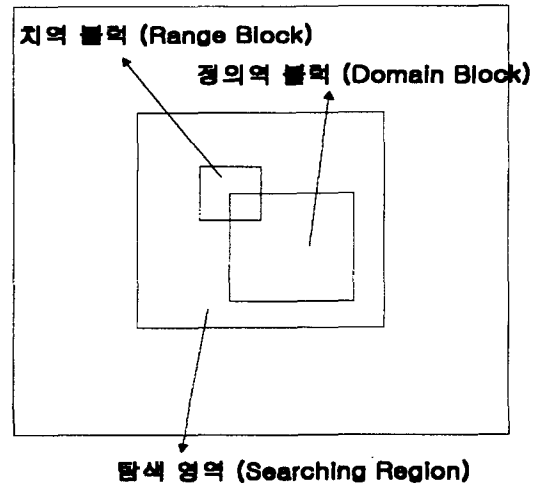


그림 5. 정의역의 제한된 범위에서의 탐색
Fig. 5. Searching domain block within restricted region.

3. 역변환 문제의 해결

앞의 영상 분할 과정을 통하여 영상들이 1차의 선형 변환으로 근사화를 하더라도 오차가 적게 발생한다. 본 논문에도 식 (10)과 같은 선형 변환식을 사용하였다.

$$w_i \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_i & b_i & 0 \\ c_i & d_i & 0 \\ 0 & 0 & s_i \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \\ o_i \end{bmatrix} \quad (10)$$

식 (10)에서 복호화기로 전송되는 파라미터는 e_i, f_i, s_i, o_i 로서 e_i, f_i 는 각각 정의역 블록의 x, y 축상의 위치 정보를 s_i, o_i 는 정의역 블록을 치역 블록으로의 근사화시 필요한 블록의 대조(contrast)와 밝기(brightness)정보이다. 이 값들은 다시 양자화를 거쳐 전송 채널을 통해 전송되어 복호화기에서 [정리 2.4]의 일반화된 콜라주 정리를 이용하여 각 치역 블록들의 변환식 w_i 들의 합집합인 W^{-1} 를 사용하여 주어진 영상을 복원해 낼 수 있다. 지금까지의 모든 부호화과정을 그림 6에 나타내었다.

제안된 가변 블록 기반 프랙탈 부호화기는 다음 순서로 영상 부호화를 수행한다.

단계 1 : 입력된 영상을 쿼드 트리를 이용하여 분할하여 가변 크기의 치역들로 분할하고 영상 분

할 정보를 쿼드 트리 구조 코드로 표현하여 전송한다.

단계 2 : 에지가 없는 부분으로 판명된 최상위 블록은 블록의 평균값을 이용하여 부호화한다. 에지 블록은 분할된 지역을 범위가 정의역의 4배인 영역에서 지역 블록으로 최소의 오류를 가지는 축소변환이 수행되는 정의역 블록을 완전 탐색한다. 이때 얻어진 축소변환 계수값들을 양자화하여 복호화기로 전송한다.

단계 3 : 단계 2를 모든 블록에 대하여 적용한다.

수신단에서의 영상의 복원은 다음과 같은 순서로 이루어진다.

단계 1 : 수신단에서는 영상의 분할 정보로 복원하고자 하는 크기의 초기 영상을 송신단에서 전달된 쿼드 트리 구조 코드를 사용하여 영상의 분할 정보를 얻는다.

단계 2 : 각 분할된 영역에 대해 송신단에서 전송된 프랙탈 파라미터를 사용하여 [정리 2.2]에서 축소비가 2인 축소 변환을 통해 정의역 블록을 축소하여 지역 블록에 복사한다.

단계 3 : 모든 지역에 관해 변환이 수행되었으면 복원 영상과 이전 복원 영상과의 PSNR을 구한다. 이때의 PSNR이 0.001dB이상이면 단계 2로 가서 다시 변환들을 반복 적용한다. PSNR이 0.001dB이하이면 두 영상은 거의 동일 영상이라 볼 수 있고 복호화 과정을 끝마친다.

본 논문에서 제안한 알고리즘을 적용한 결과 대개의 영상들이 복원시 7~8회 정도의 반복 변환 후에는 수렴하였다.

IV. 실험 결과

본 논문에서 제안한 방법과 기존의 방법들에 대한 결과를 그림 7과 그림 8에 비교하였다. 본 실험은 고주파 성분이 상대적으로 적은 LENA 영상과 고주파 성분이 많은 BRIDGE 영상에 적용하였는데, Monro 방법은 지역 블록의 크기를 8x8 블록으로 고정하고 정의역 블록을 16x16으로 설정하여 식 (11)과 같은 4항 근사식을 통해 근사화하여 얻어진 결과이다.

$$z'(x, y) = a_1x + b_1y + sg(x, y) + o \quad (11)$$

식 (11)에서 $g(x, y)$ 는 축소된 정의역을 의미하며, $z'(x, y)$ 는 근사화된 지역을 의미한다. a_1, b_1, s, o 는 근사화 오차를 최소화하는 계수들이다. 또한 전 탐색 방법도 지역 블록과 정의역 블록의 크기는 Monro 방법과 동일하며 DCT(Discrete Cosine Transform)를 사용하여 블록을 분류하고 매칭 블록을 완전 탐색하였다. 비교를 위하여 본 논문에서는 아래와 같이 식 (12)와 같은 PSNR을 사용하였다.

$$PSNR = \frac{255}{\frac{1}{2^n \times 2^n} \sum_{i=0}^{2^n-1} \sum_{j=0}^{2^n-1} [x(i, j) - y(i, j)]^2} \quad (12)$$

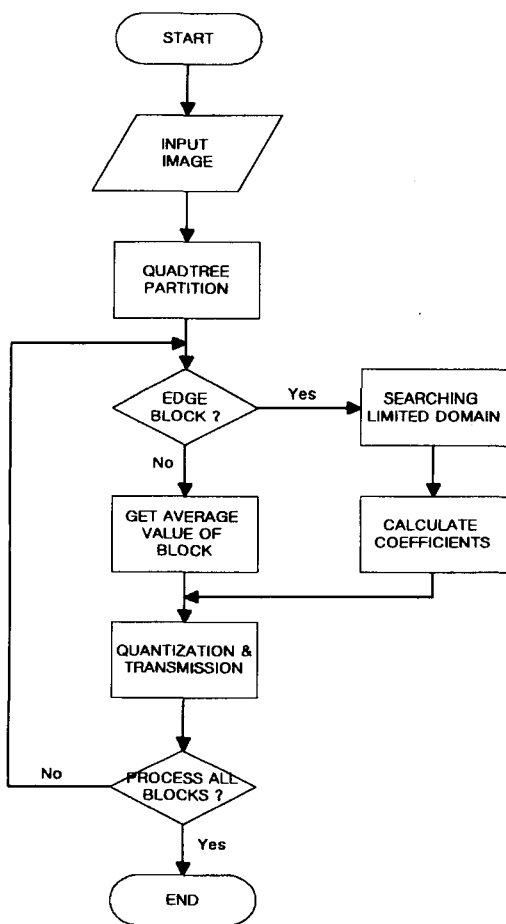


그림 6. 제안한 가변 블록 기반 프랙탈 부호화기
Fig. 6. Fractal image coding flowchart.

4. 복호화

복호화는 부호화와 달리 간단하다. 부호화 과정에서 알아낸 축소 함수들의 계수들을 쿼드 트리 분할한 영역에 반복적으로 적용하면 쉽게 복원 영상을 얻을 수 있다.

식 (12)에서 $n=8$ 이고, $x(i, j)$ 는 원 영상을 의미하며, $y(i, j)$ 는 복원 영상을 나타낸다.

고주파 성분이 많은 영상에 본 알고리즘을 적용시 가져올 수 있는 화질의 열화를 줄이기 위해 T_1 값의 설정시 고주파 성분이 많은 BRIDGE에 T_1 값을 변경하여 적용한 결과 표 1에 나타난 결과를 얻을 수 있었다. 표 1에서 알 수 있듯이 T_1 값이 17.25를 기점으로 하여 BPP가 큰 쪽으로 증가하지만 PSNR의 증가는 적었다. 앞으로의 실험 결과치는 T_1 값을 17.25로 적용하였다.

표 1. 임계값 T_1 에 대한 PSNR과 BPP
Table.1. PSNR and BPP for threshold T_1

T_1	PSNR(dB)	BPP
5	36.3	1.98
10	32.5	1.40
15	27.1	1.03
17.25	25.4	0.66
20	23.3	0.57
30	22.0	0.54
40	22.0	0.51
50	22.0	0.47
60	21.5	0.43
70	21.5	0.41
80	21.5	0.38
90	20.2	0.37
100	20.2	0.37

표 2에서 LENA, BRIDGE 영상에 대한 PSNR과 압축률, 총 부호화 시간을 비교하였다. 저주파 영상인 LENA에 대해서 본 논문의 알고리즘과 완전 탐색(full search)을 사용한 알고리즘과 비교해보면 부호화 시간 면에서 본 알고리즘이 약 60배 정도 빠른 부호화 시간이 소요됨을 알 수 있다. PSNR측면에서 보면 제안한 알고리즘으로 부호화한 영상이 완전 탐색 알고리즘에 비해 0.02dB정도로 약간 높지만 복원 영상에 대한 주관적인 평가는 제안한 알고리즘이 더 우수하다는 것을 결과 영상 그림 7의 (b)와 (d)를 비교함으로써 알 수 있고 압축률은 0.46bpp로 거의 같았다. 본 논문의 알고리즘과 Monro의 알고리즘을 비교해 보면, 표 2에서 보듯이 부호화 시간은 Monro의 알고리즘이 본 제안 알고리즘에 비해 5~6배 정도 빠른 것을 알 수 있다. 하지만, Monro 알고리즘은 정의역 불

록을 항상 치역 블록의 상위 블록으로 가정하여 근사화를 진행하므로 복원 영상의 화질이 본 제안 알고리즘에 비해 3dB이상 떨어지고 압축률도 4배 이상이었다. 그림 7에서 (b)와 (c)를 비교해 보면 Monro 알고리즘에 비해 본 제안 알고리즘의 복원 영상이 우수함을 알 수 있다.

표 2. 256×256 영상에 적용한 결과
Table.2. The results of reconstructed image

실험영상	PSNR(dB)		압축률(bpp)		부호화시간(초)	
	LENA	BRIDGE	LENA	BRIDGE	LENA	BRIDGE
완전탐색 알고리즘	27.36	24.73	0.47	0.62	1861	2407
Monro 알고리즘	24.15	20.78	2	2	6	7
제안 알고리즘	27.38	25.40	0.46	0.66	29	39

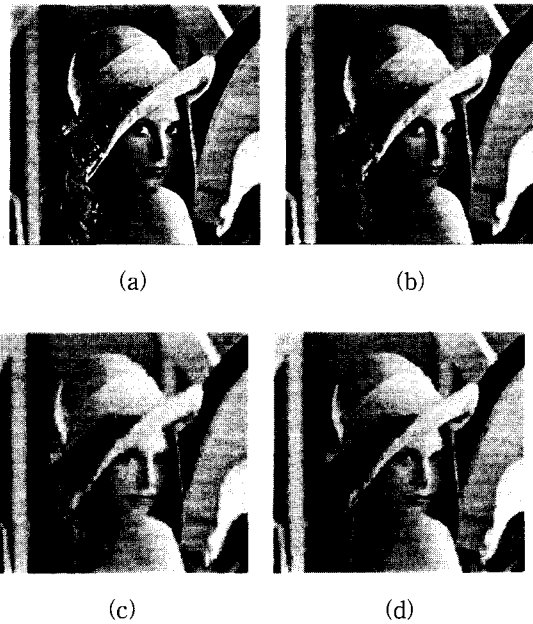


그림 7. LENA 영상을 사용한 결과 비교
(a) LENA 원영상 (b) 제안 알고리즘을 사용한 복원 영상 (c) Monro 알고리즘을 사용한 복원 영상 (d) 완전 탐색 알고리즘을 사용한 복원 영상

Fig. 7. Comparison results of LENA image.
(a) Original LENA image
(b) Reconstructed image using proposed algorithm
(c) Reconstructed image using Monro's algorithm
(d) Reconstructed image using full searching algorithm

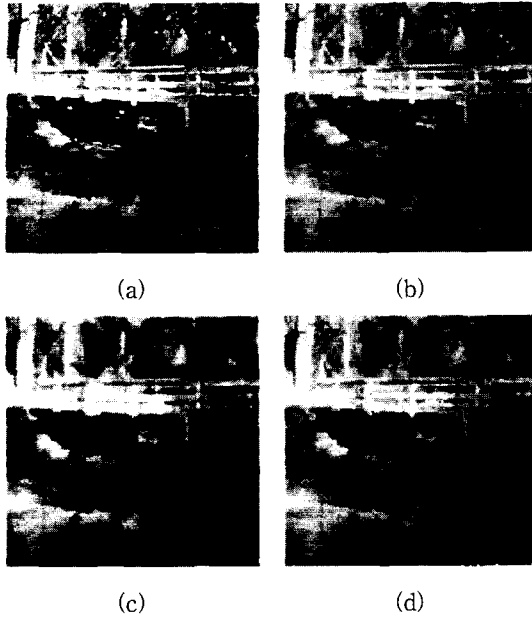


그림 8. BRIDGE 영상을 사용한 결과 비교

- (a) BRIDGE 원영상
- (b) 제안 알고리즘을 사용한 복원 영상
- (c) Monro 알고리즘을 사용한 복원 영상
- (d) 완전 탐색 알고리즘을 사용한 복원 영상

Fig. 8. Comparison results of BRIDGE image.

- (a) Original BRIDGE image
- (b) Reconstructed image using proposed algorithm
- (c) Reconstructed image using Monro's algorithm
- (d) Reconstructed image using full searching

고주파 성분이 많은 BRIDGE 영상에 대해 위 방법들의 비교에서, 완전 탐색 알고리즘을 이용할 경우와 제안 알고리즘을 이용한 부호화 시간을 비교할 경우 제안 알고리즘이 약 60배 정도 부호화시간이 빠르다. 또한 복원 영상을 비교해 보면 본 제안 알고리즘이 완전 탐색 알고리즘에 비해 약 0.7dB정도 좋은 결과를 가짐을 알 수 있다. 또한 Monro 알고리즘과 본 제안 알고리즘을 비교해 보면 5~6배 정도 부호화 시간이 빠르지만 복원 영상을 비교해 보면 본 알고리즘이 약 4dB이상 우수한 결과를 보였다. 압축률은 완전 탐색 알고리즘에 비해 0.04bpp로 약간 높았으나, Monro의 알고리즘에 비해 약 3배의 압축효과를 보였다.

이 결과에서 알 수 있듯이 본 논문에서는 완전 탐색 알고리즘에 비해서 탐색 시간에서의 급격한 감소에도 불구하고 치역 블록으로의 근사화 특성은 상당히

시각적으로 좋아졌고, Monro 알고리즘에 비해서도 대단한 화질의 개선이 있었다.

V. 결 론

본 논문에서는 반복 함수 체계 이론에 근거한 프랙탈 영상 부호화 방법에 퀴드 트리 분할법을 이용한 가변 블록 기법을 적용하여 영상의 복잡도에 따라 적층적으로 영상을 분할하여 복잡한 부분은 작은 블록으로 근사화를 시도하고 영상의 복잡도가 단순한 블록은 블록의 평균값으로 근사화를 시도하도록 하였다. 또한 유사 블록을 탐색할 때 완전 탐색을 적용하기보다는 탐색 영역을 치역 블록의 부근으로 영역을 제한하여 탐색 시간을 줄였다.

특히 제안한 영상 압축 방법에서는 영상 블록을 복잡도에 따라 변환 블록의 크기가 다르게 수행되므로 압축률과 화질 면에서 중요한 요소가 된다. 특히 본 논문에서 적용한 퀴드 트리 분할 방법은 다른 기존의 방법에 비해 많은 평탄 블록을 발생시키고, 제한된 범위에서의 탐색을 통하여 빠른 부호화를 수행한다.

표 2에서 보였듯이 본 논문에서 제안한 알고리즘은 부호화 시간이 적게 소요되고 영상의 고주파 성분에 대한 근사화 정도가 다른 알고리즘에 비해 월등히 뛰어난 것을 알 수 있다.

앞으로의 연구 방향은 영상의 분할시 퀴드 트리 분할보다 분할 특성이 우수한 것으로 알려진 HV 분할을 통한 영상을 분할할 경우와 일반적으로 윤곽선의 보존 특성이 우수한 삼각 분할의 적용도 검토되어야 할 것이다. 그리고, 실시간으로 프랙탈 부호화가 가능한 하드웨어의 구현을 연구해 보고, 또한 치역 블록의 근사화시 선형 근사식을 사용하지 않고 비 선형 근사식을 사용한 블록의 근사화 정도를 높이는 연구가 수행되어야 할 것이다. 또한 보다 많은 영상에 이 알고리즘을 적용해 보고 영상의 특성에 따라 혹은 사용자가 원하는 압축률이나 영상의 화질을 선택할 경우 자동적으로 T_1 값을 구해주는 자동화 알고리즘에 대한 연구도 지속되어야 할 것이다.

참 고 문 헌

- [1] Maher A. Sid-Ahmed, *Image Processing: Theory, Algorithms, and Architectures*,

- McGraw-Hill, Inc., 1995.
- [2] A. N. Netravali and J. O. Limb, "Picture Coding: A Review," Proc. IEEE, vol. 63, pp. 366-406, Mar. 1980
- [3] A. K. Jain, "Image Data Compression: A Review," Proc. IEEE, vol. 69, pp. 349-389, Mar. 1981
- [4] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*, Prentice-Hall, 1984.
- [5] James A. Storer, *Data Compression Methods and Theory*, Computer Science Press, 1988.
- [6] W. Pennebaker, "JPEG Technical Specification, Revision 8," Working Document no. JTC1/SC2/WG10/JPEG-8-R8, Aug. 1990.
- [7] Peter Strobach, "Image Coding Based On Quadtree Structured Recursive Least-squares Approximation," Proc. Int. Conf. Acoust., Speech, Signal Processing, 1989.
- [8] Eli Shusterman and Meir Feder, "Image Compression Via Improved Quadtree Decompression Algorithms.," IEEE Trans. Image Processing, vol. 3, pp. 207-215, March. 1994.
- [9] B. Ramamurthi, A. Gersho, "Classified Vector Quantization of Images." IEEE Trans. Commun., vol. 34, Nov. 1986.
- [10] M. F. Barnsley and Alan D. Sloan, "A Better Way To Compress Images," Byte 13(1), pp. 215-224, 1988.
- [11] M. F. Barnsley, *Fractals Everywhere*, New York: Academic, 1988.
- [12] M. F. Barnsley and Lyman P. Hurd, *Fractal Image Compression*, AK Peters, ISBN1-56881-000-8, 1993.
- [13] A. E. Jacquin, *A Fractal Theory of Iterated Markov Operators with Applications to Digital Image Coding*, Ph. D. Thesis, Georgia Tech, 1989.
- [14] A. E. Jacquin, "Fractal Image Coding Based on a Theory of Iterated Contractive Image Transformations," SPIE Vol. 1360, pp. 227-239, Visual Communications and Image Processing, 1990.
- [15] A. E. Jacquin, "Image Coding Based on a Fractal Theory of Iterated Contractive Image Transformations," IEEE Trans., vol. 1. Jan. 1992.
- [16] J. M. Beaumont, "Image Data Compression Using Fractal Techniques," British Telecommunication Tech. J. 9(4), Oct. 1992.
- [17] E. W. Jacobs, R. D. Boss, Y. Fisher, "Fractal-Based Image Compression II," NOSC TR-1362, Naval Ocean Systems Center, San Diego, CA., June 1990.
- [18] Y. Fisher, E. W. Jacobs, R. D. Boss, "Iterated Transform Image Compression," NOSC TR-408, Naval Ocean Systems Center, San Diego, CA., April 1991.
- [19] Y. Fisher, "Fractal Image Compression," Siggraph 1992.
- [20] Y. Fisher, "Fractal Image Compression," Siggraph 1992 Course Notes
- [21] Yuval Fisher, *Fractal Image Compression Compression Theory and Application*, Springer-Verlag, 1994.
- [22] D. M. Monro, F. Dudbridge, "Fractal Approximation of Image Blocks," ICASSP, pp. III:485-488, 1992.
- [23] D. M. Monro, "A Hybrid Fractal Transform," ICASSP, pp. 169-172, 1993.
- [24] D. M. Monro, "Generalized Fractal Transforms: Complexity Issues," IEEE Data Compression Conference Proceedings'93, James Storer (editor), pp. 254-261, 1993.
- [25] Haibo Li, Mirek Novak, Robert Forchhimer, "Fractal-based Image Sequence Compression scheme," Optical Engineering, July 1993, vol. 32, no. 7, pp. 1588-1595.

저 자 소 개



盧 根 洙(正會員)

1993년 2월 성균관대학교 전자공학과 졸업. 1995년 8월 성균관대학교 산업과학대학원 전기전자공학과 졸업(공학석사). 1995년 9월 ~ 현재 (주)한화/정보통신 연구소 주임 연구원. 주관심분야는 영상압축, 신호처리, 교환기술

리, 교환기술



趙 成 奎(正會員)

1980년 2월 성균관대학교 전자공학과 졸업. 1982년 2월 성균관대학교 대학원 전자공학과 졸업(공학석사). 1991년 8월 성균관대학교 대학원 전자공학과 졸업(공학박사). 1982년 ~ 1985년 해군사관학교 교수부 전임강사. 1985년 ~ 현재 동서울대학 전자계산기과 부교수. 1997년 Columbia대학 Visiting Scholar. 주관심분야는 영상처리, 신경회로망, 패턴인식

사. 1985년 ~ 현재 동서울대학 전자계산기과 부교수. 1997년 Columbia대학 Visiting Scholar. 주관심분야는 영상처리, 신경회로망, 패턴인식