

論文2000-37SP-5-9

복원순서 재구성에 의한 개선된 고속 프랙탈 영상복원

(An Improved Fast Fractal Image Decoding by recomposition of the Decoding Order)

鄭泰一*, 文光碩*

(Tae-Il Jeong and Kwang-Seok Moon)

요약

기존의 프랙탈 복원방법은 모든 치역영역에 대해 반복축소변환을 수행하였다. 그러나 일부영역은 반복축소변환 없이 복원 가능하고, 데이터 의존영역이 존재한다. $R \times R$ 치역을 복원하기 위해서 $2R \times 2R$ 정의역이 필요하다. 이러한 복원과정은 의존그래프로 해석이 가능하다. 치역은 정점에 해당하고, 정점은 치역정점과 정의역 정점으로 구분한다. 에지는 정의역 정점이 다른 치역정점에 참조됨을 나타낸다. 치역정점으로 들어오는 에지 수를 입력수, 치역정점에서 나가는 에지 수를 출력수로 정의한다. 제한한 방법은 프랙탈 코드를 의존그래프로 해석하고, 복원되는 정점의 순서를 재구성하여 출력수의 정보를 이용한다. 즉 정점의 출력수가 영이면, 그 정점은 다른 정점에 참조되지 않는 정점으로서 데이터 의존영역이 되는 정점이다. 이와 같이 복원되는 정점의 순서를 재구성함으로써, 데이터 의존영역을 확장하여 반복축소변환이 필요한 영역을 최소화한다. 그 결과 화질에는 거의 영향을 미치지 않고, 복원과정에서의 불필요한 계산량을 제거하여 고속 복원이 가능하다.

Abstract

The conventional fractal decoding was implemented to IFS(iterated function system) for every range regions. But a part of the range regions can be decoded without the iteration and there is a data dependence regions. In order to decode $R \times R$ range blocks, it needs $2R \times 2R$ domain blocks. This decoding can be analyzed to the dependence graph. The vertex of the graph represents the range blocks, and the vertex is classified into the vertex of the range and domain. The edge indicates that the vertex is referred to the other vertices. The in-degree and the out-degree are defined to the number of the edge that is entered and exited, respectively. The proposed method is analyzed by a dependence graph to the fractal code, and the decoding order is recomposed by the information of the out-degree. That is, if the out-degree of the vertex is zero, then this vertex can be used to the vertex with data dependence. Thus, the proposed method can extend the data dependence regions by the recomposition of the decoding order. As a result, the iterated regions are minimized without loss of the image quality or PSNR(peak signal-to-noise ratio). Therefore, it can be a fast decoding by the reducing to the computational complexity for IFS in the fractal image decoding.

I. 서론

* 正會員, 釜慶大學校 電子工學科
(Dept. of Electronic Eng., Pukyong National University)
接受日:2000年1月28日, 수정완료일:2000年6月26日

영상, 음성, 문자, 등의 총합체인 멀티미디어에서 영상정보는 정보전달 효과가 크다. 이러한 영상을 디지털 형태로 표현하기 위해서는 많은 양의 데이터가 필요하

다. 그러므로 디지털 영상을 저장하거나 전송하기 위해서 데이터 압축은 필수적이다. 영상 데이터의 압축방법은 각 화소간의 상관 관계를 줄이는 예측 부호화 방법과 공간 영역에서의 높은 상관 관계를 주파수 영역으로 변환하여 압축하는 변환 부호화 방법들이 있다^[1].

프랙탈 기하학은 자연 물체에 존재하는 자기 유사성 (self similarity)을 이용하여 자연 물체를 표현하기에 적합하다^[2]. 자기 유사성이란 물체의 임의의 한 부분이 다른 한 부분과 서로 유사하게 생긴 특성을 의미한다.

Barnsley^[3]는 자기 유사성에 기인하는 간단한 규칙에 의하여 복잡한 프랙탈 구조를 생성한다. 또한 영상 데이터를 반복축소변환으로 표현함으로써 데이터 양을 줄이는 프랙탈 영상 압축의 이론적 기반을 처음으로 제시하였다. 이를 이용하면 높은 압축율을 구현할 수 있으며, 임의의 영상에 대한 반복축소변환을 찾는 일은 직관적으로 수행되었다. Barnsley의 이론을 기반으로 하여 Jacquin^[4]은 입력영상을 에지(edge), 중간(middle), 평탄(shade) 영역별로 구분하여, 정의역 탐색영역을 해당영역으로 제한함으로써 부호화 과정을 개선하였고, Fisher^[5]는 영상의 화질에 따라 지역 블록을 분할하는 quad-tree 방식을 제안하였다.

화상전화 서비스, VOD(video on demand), CD-ROM 등과 같이 주로 영상 데이터의 재생만이 요구되는 경우에는 수신측에서 빠른 영상복원이 요구된다. 그래서 Oien 등^[6]과 Lepsøy 등^[7]은 내적 공간 접근 방법을 이용한 반복 적용이 필요 없는 복원 알고리즘을 제안하여 복원 시간을 단축하였다. 그러나 이 방법들은 지역 블록의 크기를 8×8 이하로 제한하므로, 화소값이 부드럽게 변하는 영역을 효율적으로 부호화하지 못하는 단점을 갖는다. 빠른 수렴속도를 갖는 프랙탈 복원 알고리즘^[8]은 정의역과 치역을 1개의 영역으로 생각하여 수렴 속도를 개선하였고, 초기영상추정방법^[9]은 변환 정보로부터 적합한 초기영상을 추정한 후 복원하여 더욱 빠른 복원을 수행하였다. 그러나 이러한 복원방법들은 수렴 속도는 향상되나, 전체 지역블록에 대해 반복축소변환을 적용하므로 한번 반복축소변환에 소요되는 계산량은 감소되지 않는다. 이러한^[8,9] 방법들을 병행하면서 화질저하 없이 반복축소변환시 소요되는 계산량을 감소시킬 수 있는 데이터 의존성을 이용한 고속 복원방법이 제안되었다^[10,11]. Domaszewicz 등은 부호화 과정에서 정의역과 치역과의 관계를 의존그래프로 구성하여 고속 복원이 가능한 알고리즘을 제안하였

다.^[12] 이 방법은 복원순서에 필요한 부가정보가 필요 하다.

본 논문에서는 Domaszewicz등의 의존그래프를 실제로 구성하지 않고, 순차적 데이터 의존영역 확장기법으로 부호화된 프랙탈 코드에 의존그래프 개념을 도입하였다. 즉, 프랙탈 코드 중 출력수의 정보로부터 복원순서를 결정하고, 복원되는 치역정점의 순서를 재구성함으로써 기존의 데이터 의존성 알고리즘을 개선한 고속 프랙탈 영상복원을 제안한다. 정점(vertex)은 치역에 해당하고, 치역정점과 정의역 정점으로 구분한다. 에지(edge)는 정의역 정점이 다른 치역정점에 참조됨을 나타낸다. 임의의 한 정점을 기준으로 들어오는 에지 수를 입력수(in-degree), 정점에서 나가는 에지 수를 출력수(out-degree)라 정의한다. 제안한 방법은 프랙탈 코드 정보 중 임의의 정점의 출력수가 영이면, 그 정점은 자신의 변화가 다른 정점에 참조되지 않는 정점으로서 데이터 의존영역이 되는 정점이다. 이러한 데이터 의존영역에 해당하는 정점들을 확장하기 위해, 의존그래프로 해석하고, 복원순서를 재구성한다. 그 결과 화질에는 거의 영향을 미치지 않고 복원과정에서의 불필요한 계산량을 제거하여 고속 복원이 가능하고, 부호화와 복원 과정에서 부가정보 없이 프랙탈 영상압축을 수행할 수 있다.

II. 프랙탈 영상 압축

1. 일반적인 프랙탈 영상 부호화

프랙탈 영상의 부호화 과정은 부호화 하고자 하는 영상과 동일한 영상을 지역 및 정의역으로 각각 설정한 후 전체와 부분과의 유사성을 잘 반영할 수 있도록 먼저 영상을 분할하여야 한다. 일반적으로 지역 블록은 $R \times R$ 의 크기로 서로 중첩되지 않게 분할하며, 정의역 블록은 $2R \times 2R$ 의 크기로 중첩이 가능하게 분할한다. 이 때 정의역 블록의 크기가 지역 블록의 4배이므로 공간 영역에서의 축소율은 $1/4$ 로 결정된다. 하나의 지역 블록에 대한 변환 계수를 구하는 과정은 임의의 정의역 블록에 대해서 계수도 변화량 s 및 밝기 조절값 o 를 구한 후, 각각의 화소들의 오차 E 가 가장 작은 정의역 블록을 찾아, 그 때의 정의역 블록에 대한 인덱스 및 s, o 값을 변환 계수로 기록한다. 여기서 계수도 변화량 s , 밝기 조절값 o 를 구하는 과정은 식

(1)과 같이 평균자승오차 E 가 주어지며, $\frac{\partial E}{\partial s} = 0$ 과 $\frac{\partial E}{\partial o} = 0$ 를 만족하는 s, o 를 구한다. 이 식을 정리하면 식 (2),(3)과 같다^[5].

$$E = \sum_{k=1}^n (s \cdot d_k + o - r_k)^2 \quad (1)$$

$$s = \frac{\left[n \sum_{k=1}^n d_k r_k - \sum_{k=1}^n d_k \sum_{k=1}^n r_k \right]}{\left[n \sum_{k=1}^n d_k^2 - \left(\sum_{k=1}^n d_k \right)^2 \right]} \quad (2)$$

$$o = \frac{1}{n} \left[\sum_{k=1}^n r_k - s \sum_{k=1}^n d_k \right] \quad (3)$$

여기서 r_k 및 d_k 는 치역 및 정의역 블럭 내의 화소값이며, n 은 블럭내의 화소들 수이다. 이러한 과정을 매 치역 블럭마다 모든 정의역 블럭에 대하여 수행한 다음, 오차가 가장 작은 정의역 블럭의 인덱스와 그 때의 변환 계수들을 구함으로써 부호화가 이루어진다

2. 일반적인 프랙탈 영상복원

프랙탈 영상의 복원은 복원 영상의 반복축소변환 W 가 주어진 상태에서 임의의 초기 영상을 설정한 후 모든 치역 블럭에 대해 주어진 횟수만큼 반복축소변환을 적용함으로써 복원 영상을 얻을 수 있다^[5]. 즉 i 번째 치역 블럭 R_i 를 복원하는 과정은 반복축소변환 계수에 따라 대응되는 j 번째 정의역 블럭 D_j 를 선택하여 공간 축소한 후, 그 정의역 블럭의 각 화소값에 계조도 변화량 s 를 곱하고, 밝기 조절값 o 를 더하여 치역 블럭의 화소값을 구하는 과정으로 수행된다. 이와 같은 과정을 모든 치역 블럭에 대해 수행한 후 주어진 반복 횟수만큼 반복하면 복원된 영상을 얻을 수 있다.

3. 빠른 수렴속도를 갖는 복원방법과 초기영상 추정법

복원시 많은 계산량이 소요되는 문제점을 개선하기 위해, 기존의 빠른 수렴속도를 갖는 복원방법은 복원시 치역 및 정의역을 같이 사용하여 한번 변환된 치역블럭을 다음 치역블럭의 변환에 참조함으로써 복원 영상의 수렴 속도를 개선하였다^[8]. 초기영상 추정법은 반복축소변환을 수행하기 전에 고정점에 근사한 초기영상을 추정하여 반복횟수를 줄였다^[9]. 그러나 이 복원방법들은 수렴 속도는 향상되나 전체 치역블럭에 대해 반복축소변환을 적용하므로 한번 반복축소변환에 소요되는 계산량은 감소되지 않는다.

4. 데이터 의존성과 순차적인 데이터 의존영역 확장 기법을 이용한 고속 복원 알고리즘

프랙탈 부호화시 만들어진 프랙탈 코드에는 정의역과 치역에 대한 정보들이 저장된다. 여기서 복원하고자 하는 치역블럭에 참조된 영역과 참조되지 않은 영역으로 구분할 수 있다. 그래서 참조된 영역은 기존의 Fisher 방법으로 우선적으로 복원하고, 참조되지 않은 영역의 정의역은 이미 복원된 영역에 존재하므로, 참조되지 않은 영역은 단 한번만에 복원이 가능하다. 그래서 참조되지 않은 영역을 데이터 의존영역으로 정의하였다^[10,11]. 데이터 의존영역은 중간단계의 반복축소변환 과정에서 제외시키고, 마지막 반복축소변환시 한번만 적용되므로, 한번 반복축소변환 하는데 소요되는 계산량을 감소시켜 고속 복원이 가능하다. 이러한 데이터 의존성 알고리즘에 의존그래프 개념을 도입하여, 정점에서 나가는 에지 수인 출력수 정보를 의존그래프로 해석하여 기존의 데이터 의존영역을 확장하였다^[13]. 이 방법은 복원과정만 고려되었고, 부호화 과정에서 데이터 의존영역이 더 많이 생기도록 부호화 할 수 있다.

프랙탈 코드정보들은 부호화시 결정된다. 순차적인 데이터 의존영역 확장기법을 이용한 복원방법은 부호화 과정에서 모든 치역영역들이 순차적인 데이터 의존성을 가지도록 정의역과 치역의 대응관계를 조절하여 데이터 의존영역을 확장하였다^[11]. 이 방법은 복원시 순서를 가지며, 복원에 필요한 부가정보가 추가되어야 하는 단점이 있다. 즉 데이터 의존영역이 아닌 영역을 반복축소변환하여 먼저 복원한 후, 이 복원된 영역을 이용하여 반복축소변환 없이 상위의 데이터 의존영역을 먼저 복원한다. 그때까지 복원된 영역을 이용하여 또 다른 하위의 데이터 의존영역을 복원한다.

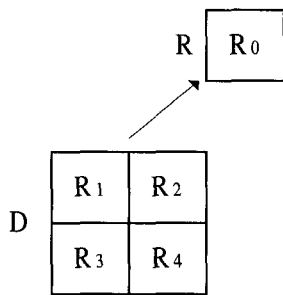
III. 제안한 고속 프랙탈 영상복원 알고리즘

프랙탈 영상 복원시 모든 치역정점에 대해 반복축소변환을 할 필요가 없다. 일부 정점들은 반복축소변환없이 복원할 수 있다. 이러한 정점들이 데이터 의존영역에 해당하는 정점이다. 만약 부호화 과정에서 정의역 정점의 탐색영역을 제한하여 정의역 정점들이 더 많이 중복된다면, 데이터 의존영역에 해당하는 정점은 더 확장될 수 있다. 이러한 순차적 데이터 의존영역 확장기법으로 부호화된 프랙탈 코드를 이용하여 복원할 경우, 복원시 정해진 순서에 따라 복원과정이 진행되어야 하

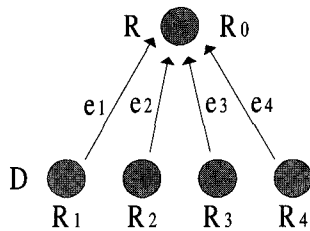
며, 복원순서에 관한 부가정보가 필요하다. 이러한 단점을 보완하기 위해, 본 논문에서는 순차적 데이터 의존영역 확장기법으로 부호화된 프랙탈 코드에 의존그래프 개념을 도입한다. 즉, 정점 R_0 에 들어오는 에지 수인 입력 수와 나가는 에지 수인 출력 수의 정보를 이용하여 복원순서를 재구성한다. 그래서 부호화와 복원과정에서 부가정보 없이, 데이터 의존영역에 해당하는 정점을 확장하여 고속 복원이 가능함을 제안한다.

1. 의존그래프 해석을 이용한 프랙탈 영상복원

프랙탈 영상복원은 부호화시 만들어진 프랙탈 코드로부터 정의역 정점에서 치역정점을 복원하는 것을 의미한다. 프랙탈 코드에는 모든 치역정점마다 유사성을 갖는 정의역 정점에 대한 정보를 가지고 있기 때문에 방향성을 갖는 그래프로 표현된다. $R \times R$ 치역정점을 복원하기 위해서는 $2R \times 2R$ 의 정의역 정점이 필요하다. 그림 1에 임의의 치역 블록(R)과 그 치역 블록에 의해 선택된 4개의 다른 치역 블록들로 구성된 정의역 블록(D)을 나타내었다. 이와 같은 프랙탈 복원과정은



(a)



(b)

그림 1. 의존그래프 (a) 반복축소변환 (b) 에지와 정점
Fig. 1. Dependence graph (a) contractive transformation (b) edges and vertices.

그림 1과 같이 의존그래프로 표현이 가능하다^[13]. 그림 1의 (b)에서 정점은 R_0, R_1, R_2, R_3, R_4 로서 각각 치역

을 나타내고, 4개의 정점 R_1, R_2, R_3, R_4 는 정의역정점이고, 정점 R_0 는 근사화된 치역정점이다. 에지는 e_1, e_2, e_3, e_4 로서 에지 e_1 은 정의역 정점 중의 하나인 R_1 이 치역정점 R_0 에 참조됨을 나타낸다. 나머지 에지들도 같은 방법으로 치역정점 R_0 에 참조된다. 그리고 치역정점 R_0 로 들어오는 에지의 수가 네 개이므로 입력수는 4로서 일반적인 경우이다. 여기서 치역정점 R_0 는 다른 정의역 정점에 참조될 수 있으며, 정점에서 나가는 에지 수인 출력수는 부호화시 만들어지는 프랙탈 코드에 좌우된다. 이와 같이 제안한 방법에서는 정점 R_0 에 들어오는 입력 수와 나가는 출력 수의 정보를 이용하므로 부가정보가 필요 없다.

임의의 정점이 다른 정점에 참조되지 않는 즉, 출력수가 영인 정점은 데이터 의존성을 갖는 정점으로서 정의역을 구성하는 정점들이 수렴될 때까지 반복축소변환 과정이 필요 없다.

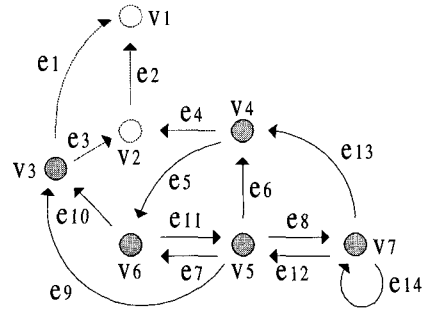
2. 제안한 알고리즘

프랙탈 복원과정에서 에지와 정점으로 구성된 반복축소변환 정보를 의존그래프로 구현하여 데이터 의존영역을 찾을 경우 복잡하게 된다. 따라서 본 논문에서는 각 정점이 다른 정점에 참조되는 수인 출력수로서 정의역 정점에 대한 정보를 의존그래프로 해석하고, 복원되는 치역정점의 복원순서를 재구성하였다.

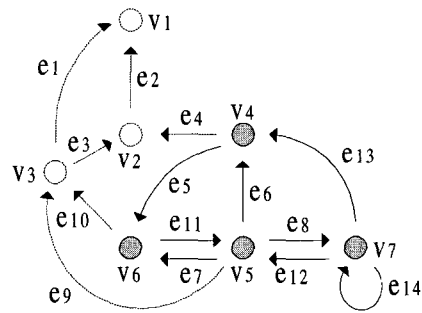
이들을 구하는 과정은 정점과 입력수에 대한 정보는 부호화시 만들어진 프랙탈 코드정보로부터 얻을 수 있고, 출력수에 대한 정보는 복원시 임의의 치역정점에서 정의역 정점으로 참조되어 나가는 수(에지 수)를 조사하여 구한다.

그림 2의 (a)는 7개의 치역정점으로 구성된 프랙탈 영상이 있을 경우, 정의역 정점과 치역정점과의 관계를 그래프로 그린 것이다. 7개의 작은 원(●)은 개별적인 정점들이며, 정의역 정점에서 치역정점으로의 변환 과정은 에지로 표시된다. 여기서 입력수는 2로 가정하였다. 그림 2의 (a)에서 정점 1(v_1)의 출력수가 0이기 때문에 정점 1은 다른 정점들에게 참조되지 않으므로, 정점 2(v_2)와 3(v_3)에 데이터 의존성을 갖는 정점이다. 즉 정점 1의 정의역은 정점 2와 3이므로, 정점 2와 3이 복원된다면 정점 1은 중간 반복축소변환 없이 마지막 한번만에 복원이 가능하고 복원 순서는 제일 마지막이 된다. 일단 복원 순서가 정해지면, 그림 2의 (b)와

같은 그래프에서 정점 1과 이와 관련된 에지 1(e_1), 에지 2(e_2)를 제거한 후, 다시 출력수가 0인 정점을 검색한다. 검색 후 정점 2의 출력수가 0이므로 다른 정점들에게 참조되지 않는 정점이 된다. 따라서 정점 2도 역시 정점 3과 4에 데이터 의존성이 있는 정점이며, 복원 순서는 정점 1보다 앞서는 순차적 구조를 갖게 된다. 이렇게 복원 순서가 정해진 정점 2와 에지 3(e_3), 에지 4(e_4)를 제거하면, 그림 2의 (c)와 같이 정점 3이 다른 정점들에게 참조되지 않는 정점이 된다. 결국 제일 먼저 반복축소변환으로 복원하여야 하는 정점은 그림 2의 (d)와 같이 정점 4(v_4), 5(v_5), 6(v_6), 7(v_7)들이다. 이 정점들이 우선적으로 복원된다면, 정점 5와 6에 데이터 의존성을 갖는 정점 3이 복원되고, 정점 3이 복원되면 정점 3과 4에 데이터 의존성을 갖는 정점 2가 복원된다. 마지막으로 정점 2가 복원되면 정점 2와 3에 데이터 의존성을 갖는 정점 1이 복원된다. 그 결과 그림 2의 7개의 정점을 복원하기 위해서 필요한 반복축소변환은 4개의 정점으로 한정하여 수행하여도 모든 정점이 수렴한 복원 영상을 얻을 수 있다.

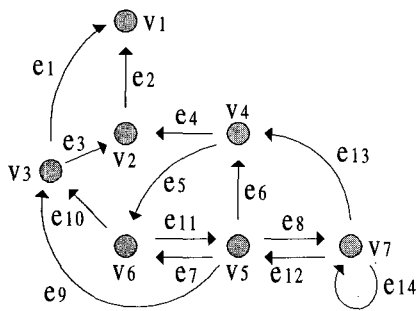


(c)

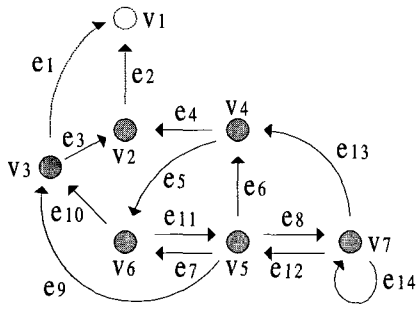


(d)

그림 2. 프랙탈 코드정보에 의한 의존그래프 해석
Fig. 2. Dependence Graph by the fractal code information.



(a)



(b)

표 1은 프랙탈 코드 정보를 의존그래프로 해석하여, 복원되는 정점의 복원순서를 재구성하기 위해 필요한 정보만 나타낸 것으로 그림 2와 연관지어 설명한 것이다. 정의역 정점(domain vertex) 옆에 있는 정점들은 그에 해당하는 치역정점(range vertex)을 복원되기 위해 필요한 정의역에 해당되는 정점들이다. 예를 들면, 치역정점 v_1 을 복원되기 위해서는 v_1 의 정의역 정점인 v_2, v_3 이 복원되고, 이 정의역 정점 v_2, v_3 정보를 이용하여 치역정점 v_1 이 복원된다. 출력수(out-degree) 옆이 나타내는 것은 임의의 정점에 의해 영향을 받는 정점들의 개수(에지 수)이다. 예를 들어, 출력수가 2-1-0인 세 번째 경우, 그림 2의 (a)에서는 정점 3(v_3)에 대해 출력수는 e_1, e_3 로서 2이고, 그림 2의 (b)에서는 e_3 으로서 1이고, 그림 2의 (c)에서는 0이다. 출력수가 0인 정점은 자신의 변화가 다른 정점에 영향을 주지 않는 데이터 의존영역에 해당하는 정점이다.

표 1. 의존그래프 해석에 의한 복원순서의 재구성

Table 1. Recomposition of the decoding order by the dependence graph.

decoding order	range vertex	domain vertex	out-degree	decoding Level
④	v_1	v_2, v_3	0	1
③	v_2	v_3, v_4	1 0	2
②	v_3	v_5, v_6	2 1 0	3
① IFS	v_4	v_5, v_7	2 1	0
	v_5	v_6, v_7	4	3 0
	v_6	v_4, v_5	2	1 0
	v_7	v_5, v_7	3	0

그림 3은 최소의 반복축소변환을 구하기 위해서 표 1과 같은 정보를 가지고 복원순서를 재구성하는 알고리즘을 나타낸 것이다. 여기서 Level은 표 1의 decoding Level이다. Level은 복원 순서를 결정하는 변수로서 값이 클수록 먼저 복원되어야 한다. ZeroCount는 출력수가 0인 정점이 더 이상 존재하는지 여부를 판단하기 위한 변수이다. 만약 ZeroCount가 0이면, 출력수가 0인 정점이 더 이상 존재하지 않는 경우이다.

임의의 치역정점(Ri)의 출력수가 0인 경우, RegisterLevel() 함수를 이용하여 치역 정점의 복원순서를 나타내는 Level 값을 등록하고, OutDegree() 함수를 이용하여 그 치역정점에 해당하는 정의역 정점의 출력수를 1 감소한다(표 1의 출력수(1-0)). 이는 그림 2에서처럼 그 정점을 제거하는 효과와 같다. 그 다음 ZeroCount 변수 값을 증가시킨다. 이 같은 과정을 모든 정점에 대해 수행하고 난 후, ZeroCount 값이 0이 아닌 정점에 대해 위의 과정을 반복한다. 위의 같이 출력수가 0인 정점이 존재하지 않는 동안 수행하면, 모든 정점들은 Level 값이 0인 정점과 아닌 정점으로 나뉘어진다. 이러한 분류과정에서 소요되는 계산량은 비교연산만이 필요하므로, 곱셈의 계산량보다 작다. Level 값이 0인 경우는 반복축소변환이 필요한 정점이며, 0이 아닌 경우는 데이터 의존영역으로 높은 Level 값부터 복원함으로써 한번만에 복원이 가능한 정점이다. 따라서 그림 3과 같이 복원되는 정점의 복원순서를 재구성하여 복원을 수행하면, 최소의 반복축소변환으로 프랙탈 영상을 복원할 수 있다.

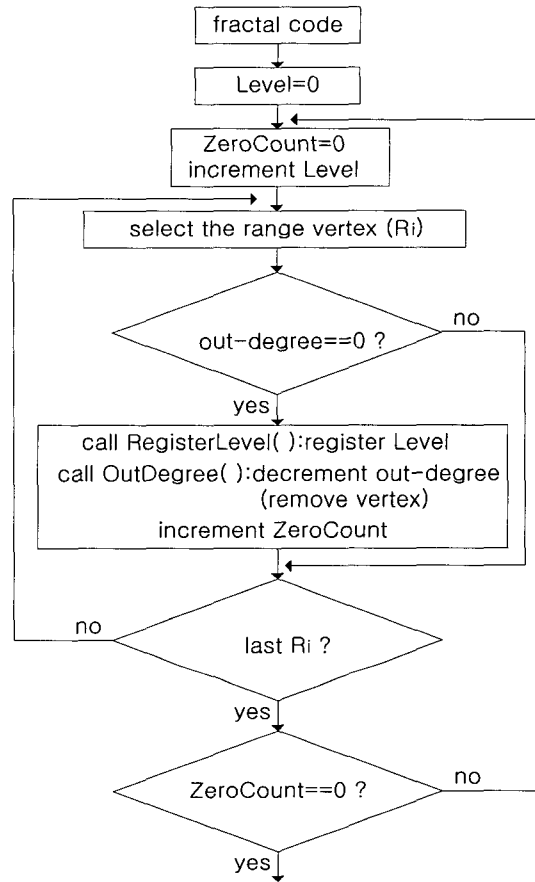


그림 3. 복원순서 재구성을 위한 흐름도
Fig. 3. Flow chart for the recomposition of the decoding order.

IV. 실험 결과 및 고찰

본 논문에서는 512×512 크기의 Lenna영상을 IBM PC상에서 Visual C++를 이용하여 실험하였다. 제안한 방법의 효능을 알아보기 위해 기존의 Fisher의 방법을 이용한 부호화 방법과 순차적 데이터 의존영역 확장기법을 이용한 부호화 방법에 의해 생성된 프랙탈 코드정보를 이용하여 실험하였다. 치역 정점은 16×16, 8×8 크기로 가변 하였으며, 그때의 분할 임계값(partition threshold)은 8로 설정하였다. 그리고 복원영상의 화질을 나타내기 위한 PSNR(peak signal-to-noise ratio)은 식 (4)에 의해 구하였다. 식 (4)에서 $f(x, y)$ 및 $\hat{f}(x, y)$ 는 각각 원 영상의 화소 및 복원 영상의 화소를 나타내며, N 는 각 영상 내의 총 화소수이다.

$$PSNR = 10 \log_{10} \left[\frac{255^2}{MSE} \right] \quad (4)$$

$$MSE = \frac{1}{N^2} \sum_{y=1}^N \sum_{x=1}^N [f(x, y) - \hat{f}(x, y)]^2.$$

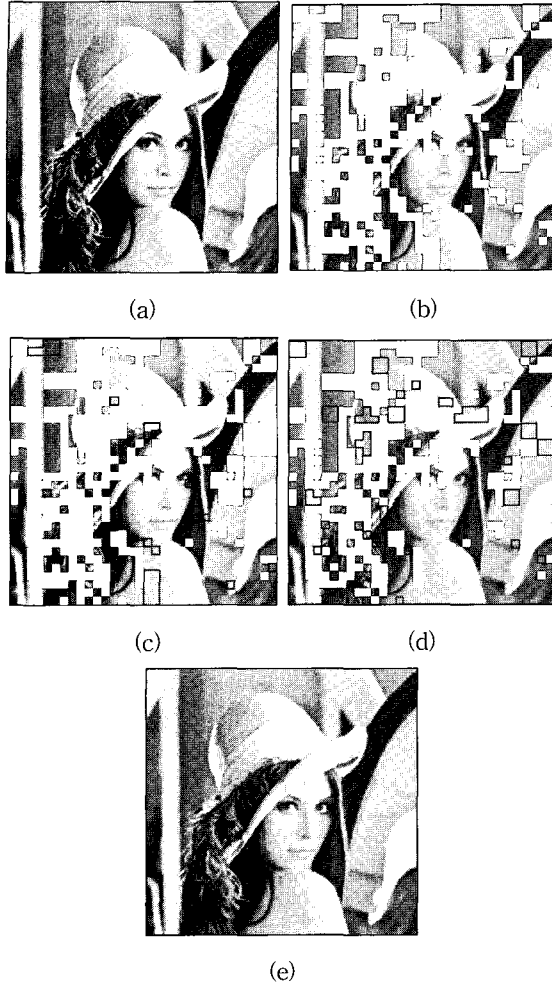


그림 4. 제안한 방법에 의한 영상 복원 과정 (a) 원 영상 (b) 데이터 의존영역 (c) 정점 3(v3) 복원 (d) 정점 2(v2) 복원 (e) 정점 1(v1) 복원

Fig. 4. Decoding process by the proposed method (a) original image (b) Data dependence region (c) decoding vertex 3(v3) (d) decoding vertex 2(v2) (e) decoding vertex 1(v1).

그림 4는 표 1의 프랙탈 코드정보에 의해 치역정점들이 복원되는 과정을 단계별로 설명하였다. 그림 4의 (b)에서 복원되지 않은 하얀 블록들이 데이터 의존영역에 해당하는 정점이고, 그 외의 영역들은 우선적으로 복원된 영역이다. (b)에서 (d)까지의 데이터 의존영역

정점들은 한번만에 복원이 가능하다^[10,11]. 그림 4의 (c)는 정점 3(v_3)이 복원되는 과정을 개략적으로 나타내었다. 그림 4의 (c)에서 굵은 표시로 복원된 블록이 표 1에서 복원순서 두 번째인 정점 3이다. 마찬가지로 그림 4의 (d)에서 굵은 표시로 복원된 블록이 표 1에서 복원순서가 세 번째인 정점 2(v_2)가 복원되는 과정을 나타내었다. 그림 4의 (d)에서 복원되지 않은 데이터 의존영역이 정점 1(v_1)이고, 그림 4의 (e)는 정점 1까지 완전 복원된 이미지를 나타낸다.

표 2는 분할 임계값을 8로 설정한 후 Fisher의 방법과 순차적 데이터 의존영역 확장기법으로 부호화한 후 PSNR 약 29dB에서 기존의 방법들과 제안한 방법으로 복원한 경우, 그때의 반복 횟수와 전체영상에 대해 데이터 의존영역의 비율, PSNR 그리고 누적 곱셈 계산량을 비교하였다. 여기서 기존의 방법 1은 Fisher의 quad-tree 기법을 사용하여 부호화 및 복원한 일반적인 방법이고, 기존의 방법 2는 Fisher의 quad-tree 기법으로 부호화 한 상태에서, 초기 영상을 추정 한 후 빠른 수렴속도를 갖는 복원 방법을 사용하고 여기에 데이터 의존성 알고리즘을 추가한 방법이다. 반복횟수가 4회일 때 각 방법들의 복원영상들은 수렴하였다. Fisher의 부호화 방법을 사용한 후, 수렴할 때까지 소요된 계산량은 기존의 방법 1의 경우 곱셈 계산이 2,097,152회가 소요되었으나, 제안한 방법은 1,518,080회가 소요되어

표 2. 계산량 비교

Table 2. Comparison of the computational complexity.

Encoding method \ Decoding method		Proposed method	Conventional method 1	Conventional method 2
		Iteration Number	4 회	4 회
Fisher's method	Dependence region rate	37 %	0 %	22 %
	PSNR	28.84 dB	28.84 dB	28.84 dB
	Multiplication No.	1,518,080회	2,097,152회	1,743,872회
Extension scheme of sequential data dependence region	Iteration Number	4회	4회	4회
	Dependence region rate	68 %	0 %	33 %
	PSNR	28.62 dB	28.61 dB	28.62 dB
	Multiplication No.	1,029,630회	2,097,152회	1,571,840회

약 28%의 곱셈 계산량이 감소하였다. 이는 데이터 의존영역의 증가로 한 번 반복축소변환 하는데 필요한 계산량이 감소했음을 의미한다. 순차적 데이터 의존영역 확장기법을 사용하여 부호화한 후 복원했을 경우, 제안한 방법은 1,029,630회로서 기존의 방법 1보다 51%, 기존의 방법 2보다 35% 곱셈 계산량이 감소하였다.

그림 5는 Fisher의 방법으로 부호화 된 반복축소변환 정보를 이용하여, 기존의 방법 2와 제안한 방법으로 복원했을 경우, 생성된 데이터 의존영역을 흰 블록으로 나타내었다. 기존의 방법 2에 의해 생성된 데이터 의존영역은 전체영상에서 약 22%를 차지하였으나, 제안한 방법에 의해 생성된 데이터 의존영역은 37%로서 데이터 의존영역이 15% 개선되었다.

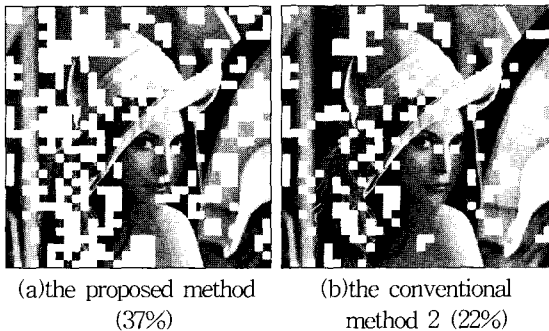


그림 5. 데이터 의존영역 비교(Fisher 방법) (a) 제안한 방법 (b) 기존의 방법 2

Fig. 5. Comparison of data dependence region(using Fisher's method)
(a) the proposed method (b) the conventional method 2.

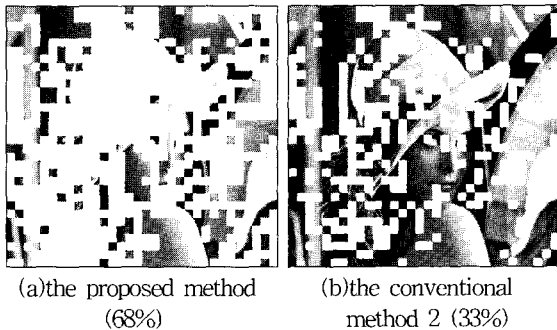


그림 6. 데이터 의존영역 비교(순차적 데이터 의존영역 기법) (a) 제안한 방법 (b) 기존의 방법 2

Fig. 6. Comparison of data dependence region(using extension scheme of sequential data dependence).
(a) the proposed method (b) the conventional method 2.

그림 6은 부호화시 순차적 데이터 의존영역 확장기법을 이용하여 만들어진 프랙탈 코드정보를 이용하여, 기존의 방법 2와 제안한 방법에 의해 생성된 데이터 의존영역 비교를 나타낸 것이다. 제안한 방법으로 복원할 경우, 기존의 방법 2로 복원할 경우보다 데이터 의존영역이 33%에서 68%로 확장되었다.

그림 5와 그림 6을 비교해 보면, 순차적 데이터 의존영역 확장기법을 이용하여 부호화한 후 제안한 방법으로 복원할 경우와 Fisher의 방법으로 부호화한 후 데이터 의존성만을 이용하여 복원한 경우와 비교하면, 특별한 화질저하 없이 제안한 방법의 데이터 의존영역이 22%에서 68%로 약 3배 더 확장되었음을 알 수 있었다.

표 3에는 가변 블록의 분할 임계값을 변경하면서 기존의 방법 2와 제안한 방법에 의해 생성되는 데이터 의존영역을 비교하여 나타내었다. 표 3에서처럼 부호화시 설정하는 가변 블록의 분할 임계값에 따라 데이터 의존영역이 달라진다. 제안한 방법의 데이터 의존영역이 Lenna 영상의 경우 분할 임계값이 4일 때 약 14%, 6일 때 11%, 8일 때 14%가 증가하였고, Boat 영상의 경우 분할 임계값이 4일 때 8%, 6일 때 8%, 8일 때 7% 증가하였으며, Bridge 영상일 때는 분할 임계값이 4일 때 17%, 6일 때 16%, 8일 때 13% 증가하였다. 영상의 특성에 따라 차이가 있지만, 제안한 방법을 이용하여 생성된 데이터 의존영역이 데이터 의존성만을 이용하여 생성된 데이터 의존영역보다 평균적으로 12% 더 많이 생성됨을 알았다. 만약 부호화시 순차적 데이터 의존영역 확장기법으로 부호화 한다면, 그림 6에서처럼 데이터 의존영역이 더 많이 확장될 것이다.

표 3. 분할 임계값에 따른 데이터 의존영역 비교

Table 3. Comparison of data dependence regions by the partition thresholds.

Decoding method		Image		
		Lenna	Boat	Bridge
Proposed method	Th=4	31.74 %	27.93 %	41.99 %
	Th=6	32.23 %	26.07 %	40.92 %
	Th=8	36.82 %	24.22 %	37.70 %
Conventional method 2	Th=4	17.97 %	19.92 %	25.39 %
	Th=6	21.48 %	18.36 %	24.81 %
	Th=8	22.46 %	17.29 %	24.41 %

그림 7은 Fisher의 방법과 순차적 데이터 의존영역 확장기법을 이용한 부호화 방법에 의해 생성된 프랙탈 코드정보를 이용하여 제안한 방법과 기존의 방법들로 복원하였을 경우, 반복 횟수에 따른 PSNR과 누적 곱셈 계산량 비교를 나타내었다. 여기서 각 점들은 반복 횟수를 의미한다. 실선은 순차적 데이터 의존영역 확장기법을 이용한 부호화한 방법이고, 점선은 Fisher의 방법으로 부호화한 방법이다. 표 3에서 설명한 바와 같이 PSNR이 28.8dB일 때, 제안한 방법의 누적 곱셈 계산량이 가장 작음을 알 수 있다.

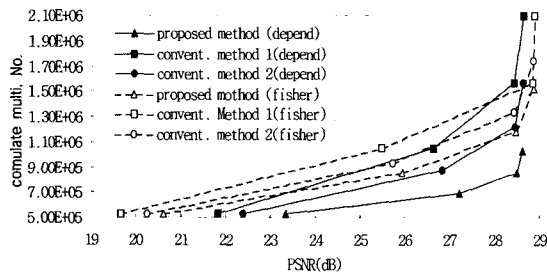


그림 7. 반복 횟수에 따른 PSNR 과 누적 곱셈 계산량 비교

Fig. 7. Comparison of PSNR and the number of calculation according to the number of iteration.

V. 결론

기존의 프랙탈 영상복원은 모든 치역정점에 대해 반복축소변환을 수행하였다. 그러나 일부정점들은 반복축소변환 없이 복원이 가능한 데이터 의존 정점들이 존재하므로, 본 논문에서는 프랙탈 영상복원시 치역과 정점의 관계를 의존그래프로 해석하여, 복원되는 치역 정점의 복원순서를 재구성하였다. 그래서 데이터 의존영역을 확장하여 고속 프랙탈 영상복원 방법을 제안하였다. 데이터 의존영역의 증가는 1회 반복축소변환 하는데 소요되는 계산량을 감소시켜 고속 복원이 가능하게 하였다. 제안한 방법을 순차적인 데이터 의존성 기법에 의해 부호화된 프랙탈 코드정보를 이용하여 복원하는 경우, 복원순서를 위한 추가 정보 없이 복원할 수 있다.

Fisher의 방법으로 부호화된 프랙탈 코드정보를 이용하여 복원할 경우, Lenna, Boat, Bridge 영상에 대하여 제안한 방법에 의해 생성된 데이터 의존영역이 평균적

으로 12% 증가하였다. 그리고 Fisher의 부호화 방법을 사용하여 비슷한 PSNR이 될 때까지 소요된 제안한 방법의 곱셈 계산량은 기존의 방법 1보다 28%, 기존의 방법 2보다 13% 감소하였다. 또 순차적 데이터 의존영역 확장기법에 의한 부호화 방법을 이용했을 경우, 제안한 방법의 곱셈 계산량은 기존의 방법 1보다 51%, 기존의 방법 2보다 35% 감소하였다.

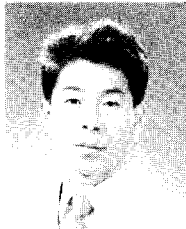
향후 연구과제로는 의존성 영역을 최대한 확장하기 위해서, 치역에 대한 정의역이 더 많이 중복되도록 하는 프랙탈 부호화 방법에 대한 연구가 있어야 할 것 같다.

참고 문헌

- [1] R. C. Gonzalez and R. E. Woods, "Digital Image Processing," Addison-Wesley Publishing Company, Sep. 1993.
- [2] M. F. Barnsley, "Fractal Everywhere," Academic Press, San Diego, 1988.
- [3] M. F. Barnsley, "Application of recurrent iterated function system to images," SPIE Visual Communications and Image Processing, pp. 122-131, 1988.
- [4] A. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," IEEE Transactions on Image Processing, vol. 1, no. 1, pp. 18-30, Jan. 1992.
- [5] Y. Fisher, Fractal image compression: theory and application, Springer-Verlag, New York, pp. 1995.
- [6] G. E. Øien, S. Lepšø, and T. A. Ramstad, "An inner product space approach to image coding by contractive transformations," Proc. of ICASSP '91, pp. 2773-2776, 1991.
- [7] S. Lepšø, G. E. Øien, and T. A. Ramstad, "Attractor image compression with a fast non-iterative algorithm," Proc. of ICASSP '93, vol. 5, pp. 337-340, 1993.
- [8] H. S. Kang and S. D. Kim, "Fractal decoding algorithm for fast convergence," Opt. Eng., vol. 35, no. 11, pp. 3191-3198, Nov. 1996
- [9] 문용호, 김형순, 손경식, 김윤수, 김재호, "고속 프

- 랙탈 복호화 알고리즘,” 한국통신학회 논문지, vol. 21, no. 3, pp. 564-575, Mar. 1996
- [10] 류권열, 정태일, 강경원, 권기룡, 문광석, “프랙탈 영상에서 데이터 의존성을 이용한 고속 복호화 알고리즘,” 한국통신학회 논문지, vol. 22, no. 10, pp. 2091-2101, Oct. 1997
- [11] 정태일, 강경원, 류권열, 권기룡, 문광석, “고속 복호화를 위한 순차적 데이터 의존영역 확장기법을 갖는 프랙탈 영상 압축,” 한국통신학회 하계종합 학술대회 논문집, vol. 16, no. 1, pp. 329-332, July 1997
- [12] J. Domaszewicz and V. A. Vaishampayan, “Graph-theoretical analysis of the fractal transform,” Proc. of ICASSP '95, vol. 4, pp. 2559-2562, 1995
- [13] 강경원, 허영대, 김태수, 권기룡, 문광석, “데이터 의존그래프를 이용한 프랙탈 영상 복호화 알고리즘,” 한국통신학회 하계종합학술대회 논문집, vol. 16, no. 1, pp. 321-324, July 1997

저 자 소 개



鄭 泰 一(正會員)
 1965년 4월 18일생. 1994년 2월 부경대학교 전자공학과 졸업(공학사), 1997년 2월 부경대학교 대학원 전자공학과 졸업(공학석사), 1999년 8월 부경대학교 대학원 전자공학과 박사과정 수료 2000년 3월~현재

동명정보대학교 정보통신과 전임강사(기간제). 주 관심 분야는 영상신호처리, 멀티미디어 통신

文 光 碩(正會員) 第 34卷 S編 第 8號 參照