

분산특징을 이용한 고속 프랙탈 부호화 방법

정회원 이 상 문*, 나 성 웅*

A Fast Fractal Encoding Algorithm using Variances of Image Blocks

Sang Moon Lee*, Sung Woong Ra* *Regular Members*

요 약

프랙탈 부호화의 가장 큰 문제점인 정의역 블록의 탐색 시간을 줄이기 위한 고속탐색 방법을 제안하였다. 주어진 정의역 블록들을 각 블록의 분산값 크기순으로 정렬한 다음 치역블록과 분산값이 가까운 정의역블록 순서로 탐색을 수행하며, 현재까지 찾아진 치역블록과 정의역블록 간의 최소거리를 근거로 치역블록과 분산값의 차가 일정값 이상인 정의역 블록은 탐색에서 제외함으로써 탐색시간을 줄일 수 있었다. 본 알고리즘은 전체 정의역 블록 탐색법(Full Search)과 완전히 동일한 프랙탈 부호코드를 발생시킴을 증명하였으며 이에 따른 실험결과를 제시하였다.

ABSTRACT

To relieve the excessive computational burden of fractal image coding, a fast fractal encoding algorithm based on the variances of image blocks is proposed. The number of domain blocks which were searched to find the best matched one for each range block could be safely reduced by eliminating the ineligible domain blocks using the current minimum distortion and variance difference between the candidate domain block and the range block. For the effective and timely termination of the search process, all of the elements of domain pool are sorted by their variances in advance. It was proved that our algorithm produces a completely identical fractal code with that of the full search in reduced time. The simulation results confirmed the effectiveness of the proposed algorithm.

I. 서 론

3차원 자연영상(natural scene)의 수학적 모델을 찾기 위해 제안된 프랙탈 이론은 영상내의 자기유사성(self-similarity)과 영상의 복잡도를 나타내는 프랙탈 차원(fractal dimension) 들로서 전체 영상의 재현(accurate discription)이 가능함을 확인 하였다[1]. 그 후 M.F. Barnsley는 영상내의 단위 블록 화상간의 자기유사성을 선형변환(affine transform)으로 표현하였으며, 그 affine계수들의 반복적 적용으로 원래의 영상을 복원 할 수 있는 반복함수계(IFS: iterated function system)를 이용한 영상 압축 방법을 제안 하였다[2]. 1990년 A.E. Jacquin은 반복함수계 이론을 바탕으로 밝기 정보를

성(block-wise self-similarity)과 유한개의 반복수축변환(iterated contractive transform)을 적용함으로써, 실제적인 프랙탈 영상 압축 방법의 기틀을 마련하였다[3].

프랙탈 이론에 따른 영상 압축 방법으로 원영상 복원이 가능함에 대한 이론적 근거에도 불구하고, 프랙탈 영상부호화 방법의 실제적인 구현에는 가장 유사한 정의역 블록을 찾기 위해 요구되는 과도한 계산량의 극복이 가장 큰 문제점으로 손꼽히고 있다. Jacquin은 치역 블록(range block)과 정의역 블록(domain block)을 블록의 복잡도에 따라 3가지로 분류(classification)한 다음 각 치역블록에 대해 동일 분류의 정의역블록 집합에 대해서만 탐색을 수행하여 탐색블록 숫자를 줄이는 방법을 택하였다. 그리고 화질 개선을 위하여 왜곡이 어느기준치 이상인 치역블록에 대해서 치역블

* 충남대학교 전자공학과(les3@sunam.kreonet.re.kr, swra@comsun.chungnam.ac.kr) 정회원
논문번호: 98156-0406, 접수일자: 1998년 4월 6일

록과 영역블록의 크기를 반으로 축소하여 재탐색을 실시하는 2단계 분할(two level partition) 방법을 사용하였다. Monro[4]등은 정의역 탐색 범위를 지역의 인근 블록으로만 제한하여 탐색 속도를 높였으나 블록 경계면에서의 화질저하(block effect)가 문제점으로 제기되었다. Fisher[5]의 방법은 Jacquin의 방법과 유사한 변환식을 사용하였으나 정의역 탐색 범위를 줄이기 위해 블록분류를 72가지로 세분화 하였으며, 수학적으로 프랙탈 계수를 유도 하였다. Oien과 Lapsoy[6]는 지역블록의 근사화를 위해 미리 정해진 기저블록(basis block)의 선형조합으로 프랙탈 변환(transform)을 표시하였으며, 내적공간(inner space)에서의 직교성(orthogonality)을 이용하여 최적 변환 계수를 찾아내는 방식을 제안하였다. 이 방식에서는 기저블록의 조합으로 근사화가 잘되는 정의역 블록을 탐색과정에서 제외 시킴으로서 탐색속도를 높일 수 있다.

지금까지 언급된 프랙탈 부호화 방법들은 각 지역블록에 대한 최적(best matched)의 정의역블록을 찾기 위해 전체 영역블록 탐색(full search or exhaustive search)을 가능한 피하면서 탐색 속도를 향상시키려고 노력한 반면, 복구된 화질의 신호대 잡음비(signal-to-noise ratio) 측면에서는 전체영역 탐색에 비해 떨어지는 결과를 보이고 있다. 본 논문에서는 Jacquin의 프랙탈 부호화 방법을 기반으로 전체 영역탐색과 동일한 프랙탈 코드를 만들어 내면서 부호화 시간을 전체영역 탐색에 비해 약 1/4이하로 줄일 수 있는 고속 프랙탈 부호화 방법을 제시하고자 하며, 이론적으로 이러한 알고리즘의 타당성을 증명하였다.

II. 이론적 배경

프랙탈 부호화 및 복호화를 가능하게 하는 이론적 배경으로는 축소변환(contractive mapping)과 반복함수계(iterated function system)의 두가지 정의와 고정점 정리(fixed-point theorem)와 콜라지정리(collage theorem)를 들 수 있다.

거리공간(metric space) (X, d) 가 주어졌을 때 변환 $f: X \rightarrow X$ 가 다음조건

$$d(f(x), f(y)) \leq s \cdot d(x, y) \quad \forall x, y \in X \quad (1)$$

(여기서 $0 \leq s < 1$ 일 때)

을 만족할 때 이 변환 f 를 축소변환이라 한다.

이러한 축소변환 f 에 대해 임의의 점 $x \in X$ 로부터

변환 f 를 반복적용하면 하나의 고정점(fixed-point) x_f 로 수렴됨을 나타내는 정리가 고정점 정리이다. 즉 다음 관계식

$$\lim_{n \rightarrow \infty} f^{(n)}(x) = x_f, \quad \text{각 } x \in X \quad (2)$$

$$f(x_f) = x_f \quad (3)$$

을 만족하는 x_f 를 고정점이라 하며, 여기서 $f^{(n)}(x) = f(f^{(n-1)}(x))$ 을 의미한다.

거리공간 (X, d) 에서 정의된 유한개의 축소변환 집합 $w_n: X \rightarrow X$ 을 다음

$$\{X; w_n, n = 1, 2, \dots, n\}$$

과 같이 표시하고 이를 반복함수계(IFS)라 부른다.

프랙탈 영상을 표시하기위한 프랙탈 공간을 $(H(X), h)$ 라 표시할 때, X 공간에서 형성된 축소반복함수계 w_n 에 대해, 프랙탈 공간상에서

$$W(B) = \bigcup_{n=1}^n w_n(B) \quad \text{for all } B \in H(X) \quad (4)$$

로 정의되는 변환 $W: H(X) \rightarrow H(X)$ 는 축소변환이며 그에따르는 고정점 정리가 적용되고, 프랙탈 공간상에서의 고정점

$$A = W(B) = \lim_{n \rightarrow \infty} W^{(n)}(B) \quad (5)$$

를 끌개(attractor)라고 부른다[2]. 위에서 설명한 이론의 물리적 의미를 살펴보면 식(3),(5)는 축소반복함수계를 임의의 영상에 반복적으로 적용하면 하나의 고정된 영상에 수렴되므로, 부호화하고자 하는 영상을 반복함수계의 attractor로 간주하고서 그 attractor에 이르게 하는 반복함수계의 계수들로 원영상을 표시할 수 있음을 나타낸다. 식(4)는 원영상을 한꺼번에 처리하기 보다는 일정 크기의 블록단위로 분할하여 블록별 축소변환을 찾는 것이 가능함을 설명한다. 주어진 영상에 대한 반복함수계를 찾는과정이 프랙탈 부호화 과정이며, 찾아진 반복함수계를 반복적용하여 원래의 영상을 복원하는 과정을 복호화라 한다.

그러나 주어진 임의의 입력영상으로 부터 입력영상이 attractor가 되는 IFS를 찾는 것은 쉬운일이 아니며

일반해가 없는 것으로 알려져 있다[2]. 따라서 실제적으로 프랙탈 부호화를 위한 IFS를 설계하기 위해서 다음 식으로 표현되는 콜라지 정리를 적용한다.

$$h(L, A) \leq (1-s)^{-1} h(L, \bigcup_{n=1}^n w_n(L))$$

for all $L \in H(X)$ (6)

즉, 주어진영상 L에 대해 L과의 거리가 가장 가까운 콜라지(부분영상의 합)를 형성하는 변환 집합을 찾는다면, 그 변환집합의 attractor는 주어진 영상 L과의 거리가 식(6)에서 주어진 거리범위 이내로 근접됨을 의미한다. 다시말해서 주어진 영상 L과의 거리를 가장작게 만드는 콜라지를 형성하는 변환집합을 구하는 작업을 프랙탈 부호화라 할 수 있다.

일반적으로 프랙탈 변환을 찾기 위해서 주로 사용되는 수식으로는

$$\begin{bmatrix} x' \\ y' \\ I'(x', y') \end{bmatrix} = w_i \begin{bmatrix} x \\ y \\ I(x, y) \end{bmatrix}$$

$$= \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ 0 & 0 & \alpha \end{bmatrix} \begin{bmatrix} x \\ y \\ I(x, y) \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta g \end{bmatrix}$$

(7)

과 같은 affine transform을 들 수 있다. 이 식에서 계수 a, b, c, d에 의해 변환의 공간적 축소율과 shuffle transform(or isometry transform)이 결정되고, Δx, Δy에 의해 공간적 변위가 결정된다. α값은 블록내 화소값의 범위(dynamic range)를 α ∈ [0, 1]만큼 축소하는 스케일링(contrast scaling factor)이고, Δg는 블록내 화소의 평균 밝기를 이동시키는 값(luminance shift)이다. α값과 Δg값을 변환시키는 과정을 수치변환(massic transform)이라하며 이 두값의 선정은 복호화된 영상의 화질에 큰 영향을 미치므로 이 값을 적절히 결정하기 위한 여러 가지 방법들이 제시되고 있다.

III. 제안된 프랙탈 부호화 알고리즘

3.1 개요

실제적인 프랙탈 부호화의 방법은 식(4)의 이론을 근거로 주어진 입력영상을 겹침이 없는 4×4 또는 8×8 pixel 크기의 치역블록으로 나누고 동일영상을 치역블록의 2 내지 4배 크기의 겹침이 있는 정의역 블록으로 분할한 다음, 각 치역블록에 가장 정합이 잘되는 정

의역 블록의 위치와 두 블록의 상관관계식(축소변환식)을 구하는 방식을 취한다[3]. 이때 식(7)에서 주어진 affine transform은 편리하게 다음과

$$\hat{f}(R_i) = I_i[\alpha_i(S(f(D_i)))] + \Delta g$$

(8)

과 같이 다소 변형하여 사용한다. 주어진 정의역 블록 D_i를 정해진 비율로 크기를 치역블록과 같게 축소(함수S(·))시키고 화소의 밝기값(f(·))을 α_i ∈ [0, 1] 배로 축소시킨다. 여기에 화소의 위치를 회전시키고 대칭변환시키는 isometry 변환(I_i(·))을 수행하게되는데 Jacquin의 경우[3] α_i를 4가지 값으로, isometry 변환을 8가지로 제한하여 부호화를 수행하였다. Δg는 정의역블록 D_i의 축소변환 블록과 치역블록 R_i가 가장 잘 근사화되도록 정의역블록의 화소밝기를 평균적으로 이동시켜주는 부분으로 주로 다음의 식

$$\Delta g = \text{quant}(\overline{f(R_i)} - \alpha_i \overline{f(D_i)})$$

(9)

과 같이 두 블록의 평균값의 차를 사용한다.

치역 블록과 변환된 정의역 블록간의 정합정도를 판단하는 왜곡척도(distortion measure)로는 squared Euclidean distance(SED)

$$d^2(X, Y) = \sum_{i=1}^k (x_i - y_i)^2$$

(10)

를 사용한다. 정해진 왜곡척도를 기준으로 치역블록 하나하나에 대해서 위에서 설명된 축소변환의 변수값들을 하나씩 변화시키면서, 모든 정의역블록을 탐색하여 가장 적합한 정의역 블록과 변환 계수값을 찾아내는 데는 막대한 시간이 소모되며, 이 점이 프랙탈 부호화 방법의 실용화에 최대 걸림돌로 지적되고 있다. 부호화 과정의 결과는 원영상의 치역블록 각각에 대해 근사화가 가장 잘되는 정의역블록의 위치, α_i, I_i(·)의 번호, Δg의 값들로서 전송채널의 전송률을 고려하여 양자화되어 전송된다.

프랙탈 부호화의 고속화를 위해서 임의 영상블록의 어떤 특징값을 추출(feature extraction)하고 그 값과 식(10)에서 제시된 왜곡척도와의 관계를 유도할 수 있다면, 그 관계를 이용하여 상당부분의 정의역블록을 탐색에서 제외시키는 부분탐색방법이 가능할 수 있다. 영상블록의 대표적인 특징값으로는 블록의 평

균값과 분산값 등이 사용된다. 프랙탈 부호화에서는 식(9)에서 표시된 바와같이 비교하려는 두 영상블록 화소들의 평균값이 일치되도록 하는 luminance shifting 과정이 포함되므로 평균값을 블록의 특징으로 사용하기에는 부적절하다. 따라서 본 논문에서는 영상 블록의 분산을 특징값으로 이용하였으며 식(10)의 왜곡척도와의 관계를 유도하여 효과적인 부분영역 탐색법을 제안하였다.

3.2 제안된 부호화 알고리즘

영상블록의 분산값과 두 개의 영상블록의 정합정도를 나타내는 왜곡척도와의 관계식을 유도하기 위하여 $N \times N$ 크기의 영상블록 $X(i, j)$, $Y(i, j)$ 에 대해 분산 V_x^2 , SED $d^2(X, Y)$ 는

$$V_x^2 = \sum_{i=1}^N \sum_{j=1}^N (X(i, j) - \bar{X})^2 \quad (11)$$

$$\bar{X} = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N X(i, j) \quad (12)$$

$$d^2(X, Y) = \sum_{i=1}^N \sum_{j=1}^N (X(i, j) - Y(i, j))^2 \quad (13)$$

로 정의한다.

프랙탈 부호화의 경우 luminance shift가 이루어지므로 지역블록과 정의역블록의 평균값이 같아지게 된다. 이 성질을 이용하여 평균값이 같은 두 영상블록의 왜곡척도와 분산과의 관계식을 유도하면

$$d^2(X, Y) \geq (V_x - V_y)^2 \quad (14)$$

과 같은 부등식이 성립한다. 이에대한 증명은 부록에 실었다.

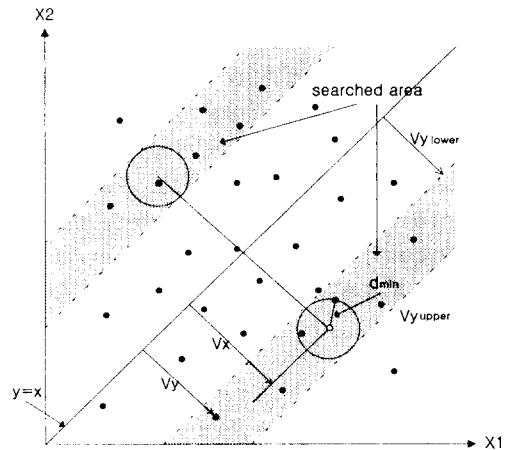
식(14)의 의미는 식(13)의 왜곡척도와 식(11)의 분산정의를 사용할 때, 두 영상블록의 분산값의 차를 이용하면 두 블록간의 거리가 어느 범위에 있는지를 사전에 예측할 수 있음을 나타내고 있다. 이 관계식으로부터 하나의 지역블록에 대해 정의역 블록의 탐색을 진행하는 과정에서 현재까지 구해진 지역과 정의역블록의 최소거리를 d_{min}^2 이라할 때, 다음 조건

$$(V_R - V_D)^2 \geq d_{min}^2 \quad (15)$$

을 만족하는 정의역 블록은 왜곡척도를 계산할 필요가 없이 탐색에서 제외시킬 수 있음을 알 수 있다.

여기서 V_R 및 V_D 는 각각 지역블록과 정의역블록의 분산을 나타낸다.

식(15)의 의미를 개념적으로 설명하기 위하여 2차원 영상벡터 $X = \{x_1, x_2\}$, $Y = \{y_1, y_2\}$ 에 대해 제안된 알고리즘의 탐색영역을 그림 1에 나타내었다.



● 정의역 벡터 ○ 지역벡터
 d_{min} : 현재까지의 탐색된 정의역 벡터의 최소거리
 V_x, V_y : 벡터 X, Y의 분산

그림 1. 제안된 알고리즘의 탐색영역범위 - 2차원 영상벡터의 경우
 Fig. 1 Search range of the proposed algorithm for 2-D image vector

그림 1에서 흰색으로 표시된 X 벡터는 지역 영상벡터를, 검은색으로 표시된 Y 벡터들을 정의역 영상벡터를 표시한다. 식 (15)의 의미는 주어진 지역벡터와 거리가 가장 가까운 정의역 벡터를 찾기위해서 전체 정의역 벡터를 탐색 할 필요가 없이 음영으로 표시된 영역내의 정의역 벡터 탐색만으로 전체 정의역 벡터를 탐색했을때와 동일한 최소거리 정의역이 찾아짐을 의미한다. 음영으로 표시된 부분의 경계선은 식(15)로부터

$$V_{y \text{ upper}} = V_x + d_{min}, V_{y \text{ lower}} = V_x - d_{min} \quad (16)$$

과 같이 표시된다. 그림 1에서 음영부분이 X 벡터 주변에만 나타나는 것이 아니고 $Y = X$ 축에 대칭되는 영역에도 나타나는 이유는 $X' = \{x_2, x_1\}$ 벡터도 평

균값과 분산이 X 벡터와 같기 때문에 탐색영역에 포함되지 지기 때문이다.

위에서 설명된 부분탐색 개념을 효과적으로 구현하기 위해서 주어진 원영상의 정의역 블록집합(domain pool)을 각 블록의 분산값 순서대로 정렬하는 것이 식(15)의 탐색종료 조건을 적용하기에 용이한 방법으로 판단되었다. 탐색 순서는 부호화하고자 하는 지역 블록의 분산값과 분산값이 가장 가까운 정의역블록에서 시작하여 분산값의 차가 커지는 방향으로 탐색을 수행하였다. 왜냐하면 지역블록과 거리가 가까운 정의역블록은 서로 그 분산값이 거의 같을 것이며, 분산값의 차가 적을수록 두영상의 거리가 가까울 확률이 높기 때문이다. 분산값이 가장 가까운 정의역 블록부터 분산값이 점점 커지는 순서로 탐색하면서 식(13)에 주어진 거리를 계산하여 거리값이 가장 작은 정의역블록의 정보를 up-date시킨다. 이 과정은 탐색종료 조건이 만족될 때까지 반복되며, 탐색 종료는 지금까지 계산된 지역블록과 정의역블록의 최소거리 d_{min} 을 기준으로 식(15)의 조건이 만족되는 정의역 블록에서 탐색을 종료하게 된다.

Jacquin의 프랙탈 부호화 방법에서는 3.1절에서 설명한 바와 같이 4가지의 contrast scale factor $\{a_1, a_2, a_3, a_4\}$ ($a_1 > a_2 > a_3 > a_4, a_1 = 1$)를 적용하여, 정의역 블록의 화소값을 축소시켰을 경우에 지역 블록과의 정합이 더 잘 이루어지는지 여부를 탐색하게 된다.

이 과정에 대해서도 위의 부분탐색법을 확장하기 위하여 본 알고리즘에서는 축소 scaling된 정의역 블록들도 하나의 독립된 정의역 블록으로 간주하여 각각의 분산값을 구한 다음 모든 정의역 블록들을 그 분산값에 따라 전체적으로 정렬하여 탐색을 수행하였다. 다시말해서 주어진 영상을 M 개의 정의역 블록으로 분할하였다고 가정할 경우, 전체적으로 정의역 블록 집합은 $4M$ 개의 정의역 블록으로 구성되며 모든 블록들을 각각의 분산값에 따라 내림차순으로 정렬한 다음 위에서 설명된 순서로 탐색을 수행한다. 이와 같은 탐색방법은 주어진 지역블록과 분산값의 차가 가장작은 순서로 정의역 블록을 탐색하게 되고, 조기에 최소 d_{min} 값을 얻을 수 있으므로써, 탐색해야할 정의역 블록 갯수의 범위를 최소화 할 수 있다. 탐색 결과는 최소 d_{min} 값을 제공하는 정의역블록의 번지수와 그때의 contrast scale factor가 동시에 구하여 진다.

축소 scaling된 정의역 블록의 분산값은 식(11)의 계산과정을 반복해서 적용할 필요가 없이

$$V_{i(x_i)}^2 = a_i^2 * V_{i(x_i)}^2 \tag{17}$$

의 관계식을 통하여 간단히 구할 수 있다. 여기서 $V_{i(x_i)}^2$ 은 scaling이 없는($\alpha = 1$) 경우 정의역 블록의 분산값이다.

지금까지 설명한 분산값에 따라 정렬된 정의역 블록의 부분탐색 알고리즘(VPS : Variance-Ordered Partial Search Algorithm)의 예를 그림 2에 나타내었다.

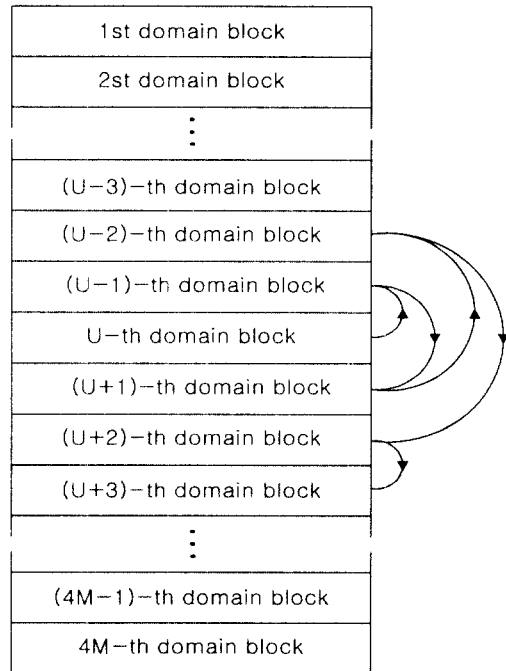


그림 2. 제안된 알고리즘의 탐색과정 예
Fig. 2 An example of the proposed domain block search procedure

그림 2에서 전체 정의역 블록의 집합은 주어진 영상을 M 개 정의역 블록으로 분할하고, 4개의 contrast scale factor를 적용하였다고 가정하여 $4M$ 개의 정의역 블록 $D(i, j, a_n)$ 이 각각의 분산값에 따라 내림차순으로 정렬되어 있다. 탐색순서는 먼저 부호화하려는 지역 블록의 분산값 $V_{R(i, j)}$ 을 구한 후, $V_{R(i, j)}$ 와 가장 가까운 분산값을 가지는 정의역 블록 $D(i, j, a_n)$ 부터 거리 계산을 수행한다. 그림 2에서 D_u 정의역 블록이 최소 분산값 차를 제공하는 정의역 블록임을

나타낸다. 이 계산과정에서 isometry 변환 갯수에 해당하는 횡수만큼의 거리계산이 수행되며, 수행결과는 최소거리값을 제공하는 isometry 번호와 정의역블록의 번지(i, j), α_n , Δg , d_{min} 이다. 이 계산이 끝나면 위에서 구한 D_u 를 기준으로 윗방향과 아래 방향으로 번갈아서, 정의역블록의 분산값과 치역블록의 분산값의 차가 증가하는 방향으로 탐색을 수행한다. 이때 각 정의역블록의 거리계산을 하기전에 식(15)의 탐색 종료조건을 검사하여 이 조건이 만족되면 해당 방향(위 또는 아래방향)으로의 탐색이 종료된다.

그림 2에서는 윗 방향으로서는 D_{u-2} 정의역 블록에서 탐색 종료 조건이 만족 되었고, 아래 방향으로서는 D_{u+3} 정의역 블록에서 탐색 종료조건이 만족 되었음을 나타내고 있다. 따라서 전체 정의역블록 집합에서 D_1 에서 D_{u-2} 까지의 정의역 블록과 D_{u+3} 에서 D_{4M} 까지의 정의역 블록들은 모두 탐색 영역에서 제외 됨으로써 탐색 속도가 상당히 개선됨을 알 수 있다. 그림 2의 탐색과정은 하나의 치역블록에 대해서 최소 거리값을 제공하는 정의역 블록을 찾는 과정을 표현하고 있으며, 모든 치역 블록에 대해 동일한 과정을 반복 수행함으로써 주어진 영상에 대한 최종 프랙탈 코드 집합을 얻을 수 있다.

제안된 VPS 탐색 알고리즘에 대한 의사 C언어 프로그램은 다음과 같다.

```
void main ( )
{
    /* making the domain pool */
    calculate the variances Vd of all domain blocks;
    sort D(i, j,  $\alpha_n$ ) by Vd in descending order;
    /* encoding */
    for(i = 0; i < 64; i++)
        for(j = 0; j < 64; j++) /* for each range block */
        { calculate the variance Vr of the range block;
          find the domain block D[u] such that abs(Vd[u]-Vr) is
          minimized;
          sfr = 0x00; k[0] = u-1; k[1] = u;
          while(sfr != 0x03) /* sfr : search termination flag register */
              for(dir=1; dir>=0; dir-- /* dir=1:up, dir=0:down search */
                  if((sfr & (0x01 << dir)) == 0)
                  { if(powf(Vd[k[dir]]-Vr, 2) >= powf(dmin, 2))
                      sfr |= (0x01 << dir);
                    else
                    { delta_g = mean_range - mean_domain;
                      for(iso=0; iso<no_iso; iso++)
                      { [T_D[k[dir]] = isometry transform of (D[k[dir]] +
                        delta_g);
                        sum_distance =  $\sum$ powf(range_block-T_D[k[dir]], 2.0);
```

```
distance = sqrt(sum_distance);
if(distance < dmin)
    { dmin = distance;
      update fractal code; } /* D(i,j), iso,  $\alpha_n$ , delta_g */
} // for iso
k[dir] = k[dir] + (-1)dir;
if(k[dir]<0 || k[dir]>D_max) sfr |= (0x01 << dir); }
```

IV. 실험결과

본 알고리즘의 성능을 시험하기 위하여 256×256 Lena 영상에 대하여 Jacquin의 프랙탈 알고리즘을 적용하여 성능시험을 수행하였다. Jacquin의 부호화방법은 영상블록의 복잡도에 따른 분류과정 및 2단계 계층적 탐색방법을 사용하고 있으나, 본 실험에서는 탐색해야할 정의역 블록의 갯수가 정해졌을 때, 전체 정의역블록에서 거리계산이 불필요한 블록을 제거하는 부분탐색법의 가능성을 증명하는 것이 목적이므로, 분류방법 및 계층적 탐색은 적용하지 않았다. 또한 isometry 갯수도 Jacquin은 8가지를 사용하였으나 isometry 갯수가 실험목적에 크게 영향을 미치지 않으므로, 본 실험에서는 identity와 180도 회전시키는 경우 두가지만 사용하였다. 치역블록의 크기는 4×4 , 정의역블록의 크기는 가로, 세로로 4 pixel씩 겹침이 있는 8×8 로 하였다.

실험결과를 표 1에 나타내었다. 위 실험조건으로 부터 탐색해야할 정의역 블록의 갯수는 $3,969(63 \times 63)$ 개이며, 기존의 전체영역 탐색방법으로는 치역블록 하나에 대해서 4가지의 contrast scale factor를 고려하여 총 $15,876(3,969 \times 4)$ 개의 정의역블록을 탐색해야 한다. 제안된 VPS 알고리즘으로 탐색하였을 때 경우 하나의 치역블록에 대해 평균적으로 3,774개(23.77%)의 정의역블록만 탐색이 수행 되었음을 알 수 있다. 물론 생성된 프랙탈 부호는 모든 치역블록에 대해서 전체영역 탐색방법과 동일함을 확인 하였다. 이 결과는 본 논문에서 제안한 정의역 탐색방법이 효과적임을 증명하고 있다. 전체적으로 부호화에 소모되는 시간은 본 알고리즘에서 기존의 방법에서는 필요하지 않는, 각 치역블록 및 정의역블록의 분산값 계산시간, 정의역블록의 정렬에 필요한 시간, 가까운 분산값을 찾기위한 분산값 비교시간 등이 추가로 소모되기 때문에, 탐색블록의 갯수가 줄어드는 비율보다는 다소 낮게 전체영역 탐색방법에 비해 25.88%로 줄어들고

있음을 나타내고 있다. VPS 알고리즘으로 생성된 프랙탈 부호를 기반으로 복호화된 영상의 PSNR은 32.3[dB] 있으며, 그림 3에 복호화된 Lena 영상을 나타내었다.

표 1. Lena영상에 대한 제안된 알고리즘 실험 결과
Table 1. Simulation results of the proposed algorithm on Lena image

	Average number of searched domain blocks per range block		Encoding time		PSNR [dB]
	Number of blocks	%	sec	%	
Full search	15,876(3,969 × 4)	100	8,290	100	32.3
VPS	3,774	23.77	2,146	25.9	32.3



그림 3. 복호화된 Lena 영상
Fig. 3 The decoded Lena image

V. 결 론

본 논문에서는 영상블록의 분산특징을 이용하여 부분탐색에 의한 고속 프랙탈 부호화 방법을 제안하였다. 제안된 알고리즘은 이론적 증명을 바탕으로 하여 실험 결과에서 보인 바와 같이 전체영역 탐색방법과 동일한 프랙탈 부호를 생성하면서 부호화 속도면에서 4배정도의 빠른 부호화 성능을 나타내는 것으로 확인되었다. 탐색속도가 빨라지는 정도는 부호화하려는 각 영상의 특성에 따라서 약간씩의 차이가 있을 수 있다. 영상의 분할 방법에 따른 지역블록과 정의역블록의 분산값 분포특성과 초기탐색에서 얼마나 빨리 지역블록과 거리가 가까운 정의역블록이 탐색되는냐에 따라 탐색영역의 폭이 결정된다.

모의실험에서는 영상블록의 분류기법이 사용되지 않

았지만 본 알고리즘에서는 탐색할 정의역 블록이 분산값 순서로 정렬되어있기 때문에, 분산값에 따른 블록분류가 가능할 것이며, 분류된 정의역블록 그룹단위로도 본 알고리즘을 적용하여 탐색속도를 향상시킬 수 있을 것이다.

부 록

부등식 $d^2(X, Y) \geq (V_x - V_y)^2$ 의 증명

크기가 $N * N$ ($k = N^2$)이고 평균값이 같은 두 영상벡터를 X, Y 라 하면 다음과 같이 영상벡터의 분산, 평균 및 거리등을 표시할 수 있다.

$$\text{벡터 } X = \{x_1, x_2, x_3, \dots, x_k\}$$

$$\text{벡터 } Y = \{y_1, y_2, y_3, \dots, y_k\}$$

$$\text{분산 } X: V_x^2 = \sum_{i=1}^k (x_i - m_x)^2$$

$$\text{분산 } Y: V_y^2 = \sum_{i=1}^k (y_i - m_y)^2$$

$$\text{평균 } m_x = \frac{1}{k} \sum_{i=1}^k x_i, \quad m_y = \frac{1}{k} \sum_{i=1}^k y_i$$

$$\text{거리 : } d^2(X, Y) = \sum_{i=1}^k (x_i - y_i)^2$$

위 정의로부터 거리 $d^2(X, Y)$ 는 다음과 같이 전개될 수 있다.

$$\begin{aligned} d^2(X, Y) &= \sum_{i=1}^k (x_i - y_i)^2 \\ &= \sum_{i=1}^k (x_i - m_x + m_x + m_y - m_y - y_i)^2 \\ &= \sum_{i=1}^k [(x_i - m_x)^2 + (m_x - y_i)^2 + (m_x - m_y)^2 \\ &\quad + 2(x_i - m_x)(m_x - y_i) + 2(x_i - m_x)(m_x - m_y) \\ &\quad + 2(m_x - y_i)(m_x - m_y)] \end{aligned} \tag{A.1}$$

여기서 주어진 조건 $m_x = m_y$ 를 대입하여 다시 정리하면

$$\begin{aligned} d^2(X, Y) &= V_x^2 + V_y^2 - 2 \sum_{i=1}^k (x_i - m_x)(y_i - m_y) \\ &= (V_x - V_y)^2 + 2V_x V_y - 2 \sum_{i=1}^k (x_i - m_x)(y_i - m_y) \end{aligned} \tag{A.2}$$

가 되고 식(A.2)의 마지막 두 항의 대소관계는 Cauchy-

Schwarz 부등식으로부터

$$\begin{aligned} (V_x - V_y)^2 &= \sum_{i=1}^k (x_i - m_x)^2 \sum_{i=1}^k (y_i - m_y)^2 \\ &\geq \left[\sum_{i=1}^k (x_i - m_x)(y_i - m_y) \right]^2 \end{aligned} \quad (A.3)$$

식(A.3)이 성립되므로 다음 식(A.4)가 성립된다.

$$d^2(X, Y) \geq (V_x - V_y)^2 \quad (A.4)$$

증명 끝.

참 고 문 헌

1. Alex P. Pentland, "Fractal-based description of natural scenes.", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. PAMI-6. no. 6, pp.661-674, Nov. 1984.
2. M. Barnsley, Fractals Everywhere, Boston : Academic Press Professional, 1988.
3. A.Jacquin, "Image coding based on a fractal theory of iterated contractive image transformation." IEEE Trans. on Image Processing, vol. 1, pp.18-30, Jan. 1992.
4. D. M. Monro and F. Dudbridge, "Fractal approximation of image blocks", Proc. ICASSP '92, vol. 3, pp. 485-488, 1992.
5. Y.Fisher, "Fractal image compression : Theory and Application", Springer-Verlag, New York, 1995
6. G. E. Øien, S. Lepsøy and T. A. Ramatad, "An inner product space approach to image coding using contractive transforms.", Proc. of ICASSP'91, pp.2773-2776, 1991.



이 상 문 (Sang Moon Lee) 정회원
 1982년 2월 : 경북대학교 전자공학과 졸업(공학사)
 1984년 2월 : 경북대학교 전자공학과 졸업(공학석사)
 1984년 3월 ~ 현재 : 국방과학연구소 선임연구원
 1994년 3월 ~ 현재 : 충남대학교 전자공학과 박사과정 재학중

<연구분야> 영상압축, 신호처리, GPS.



나 성 웅 (Sung Woong Ra) 정회원
 1976년 2월 : 서울대학교 전기공학과 졸업(공학사)
 1978년 2월 : 한국과학기술원 전기 및 전자공학과 졸업(공학석사)
 1992년 8월 : 한국과학기술원 전기 및 전자공학과 졸업(공학박사)

1980년 7월 ~ 현재 : 충남대학교 전자공학과 교수
 <연구분야> 영상압축, 신호처리, GPS, MPEG-II, 멀티미디어통신, 화상회의 시스템.