

프랙탈 근사화와 벡터 양자화를 이용한 가변 블록 부호화 알고리즘

正會員 金仁權*, 朴來弘

Variable Block Size Coding Algorithm Using Fractal Approximation and Vector Quantization

In Kwon Kim*, Rae-Hong Park Regular Members

要 約

본 논문에서는 프랙탈 근사화와 벡터 양자화를 이용한 정지 영상 부호화 기법을 제안한다. 기존의 프랙탈(fractal) 부호화 방법이 축소 매핑(contraction mapping)을 이용하여 원영상의 밝기 형태를 간접적으로 이용하는 반면, 제안한 방법에서는 미리 근사화된 영상을 domain pool로 사용하며 근사화 영상의 밝기 형태를 이용한다. 따라서, 축소 매핑의 제한조건에 관계없이 원영상의 자기 유사성(self-similarity)을 이용하는 프랙탈 근사화를 사용한다. 영상을 근사화하기 위해서 기저함수로 다항식을 이용하는 직교 변환을 사용하며, 변환계수를 부호화하기 위해서 벡터 양자화를 사용한다. 또한 적합한 블록 크기를 결정하기 위하여 블록내의 영역들에 대한 유사성과 밝기 표면의 거칠기를 나타내는 프랙탈 차원을 사용하였다. 컴퓨터 모의 실험을 통해 제안한 방법이 기존의 프랙탈 부호화 기법보다 정지영상 부호화에 더 좋은 성능을 나타냄을 보였다.

ABSTRACT

In this paper, we propose a coding algorithm for still images using fractal approximation and vector quantization. The conventional fractal coding algorithms indirectly used the gray pattern of an original image with contraction mapping, whereas the proposed method uses a previously approximated image without the constraint of contraction mapping. To approximate an original image, as domain pool and its gray pattern. Thus, the proposed algorithm employs fractal approximation using the self-similarity of an original image, we use an orthogonal transform using polynomials as basis functions, and employ vector quantization to encode the transform coefficients. Also, to determine the appropriate block size, we use the fractal dimension that represents the similarity of regions in a block and the roughness of gray surface of a region. Computer simulations with several test images show that the proposed method shows better performance than the conventional fractal coding methods for encoding still pictures.

*西江大學校 電子工學科
Dept. of Elec. Eng., Sogang Univ.
論文番號 : 94158-0616
接受日字 : 1994年 6月 16日

I. 서론

프랙탈(fractal)⁽¹⁻³⁾을 이용한 영상 압축 기법⁽⁴⁻¹³⁾은 공간적 중복성을 제거하는 정지 영상의 손실 부호화로서 제안되었다. 프랙탈 부호화 방법은 변환 시스템이 갖는 끌개(attractor)⁽¹⁻³⁾로 영상을 모델링하며 이를 이용하여 영상을 부호화한다. 끌개 개념은 복잡하게 보이는 자연 현상들을 간단한 규칙으로 표현하여 이들을 이해하기 쉽게 한다. 프랙탈 영상 부호화 방법은 영상내에서의 자기유사성(self-similarity)⁽⁴⁻¹³⁾을 기술함으로써 영상을 끌개로 표현하는 방법이다. 영상에서의 자기유사성은 밝기값의 형태가 유사한 두 영역간의 상관 관계를 말한다. 프랙탈을 이용한 영상 부호화 방법은 영상의 자기유사성 표현에 따라 복원 영상의 특성이 달라지기 때문에 자기유사성을 효과적으로 기술하기 위한 여러 방법⁽⁹⁻¹⁴⁾들이 연구되어 왔다. 자기유사성을 기술하기 위해서는 오차함수를 이용하여 영상내의 유사한 두 영역을 결정하여야 하며 설정된 두 영역에 대해 변환식을 이용하여 관계를 기술하여야 한다. 변환식에 의해 변환될 영역을 domain 블럭이라고 하며 변환된 domain 블럭으로 표현되는 영역을 range 블럭이라고 한다. 프랙탈 부호화 기법의 가장 큰 매력은 원래 영상 정보인 domain 블럭을 이용하여 다시 원영상, 즉 range 블럭을 복원한다는 것이다. 이는 반복 변환 시스템(IFS:Iterated Function System)과 콜라주(collage)이론에 의해 가능하며 Barnsley⁽¹⁾에 의해 제안되었다.

Jacquin^(4,5)은 원영상을 일정한 형태의 range 블럭으로 나눈 후 각 image 블럭에 대하여 변환에 이용할 domain 블럭을 영상 전체에서 찾는 조각 변환(piece-wise transform) 기법을 이용하였다. 반면 Monro와 Dudbrige⁽⁶⁾는 range 블럭대신 domain 블럭으로 영상을 나눈 후 각 domain 블럭을 다시 여러개의 range 블럭으로 분할하는 자기 변환(self transform) 기법을 이용하였다. Jacquin의 방법의 경우 변환 계수와 domain 블럭을 결정하기 위한 탐색 시간이 많이 소요되며, Monro와 Dudbrige의 방법에서는 탐색 영역이 작기 때문에 빠르게 수행된다. 그러나 후자의 방법^(6,7)에서는 제한된 탐색 영역때문에 최적의 domain을 선택하지 못하며 이 때문에 복원영상의 성능이 저하된다.

제안한 방법에서는 원래 프랙탈 기법이 사용하는 domain 블럭 즉, 원영상의 정보를 미리 다항식을 이용

한 직교 변환으로 부호화하여 이를 domain 블럭으로 이용하였다. 따라서, domain 블럭에 대한 직접적인 정보없이 이를 이용하는 기존의 방법과는 달리 제안한 방법에서는 domain 블럭에 대한 직접적인 정보를 이용한다. 이는 기존의 방법에서 필요한 변환식의 축소성(contractivity)에 대한 제한 조건과 복원시 필요한 반복적인 수렴과정을 없애준다. 또한 기존의 변환식에 사용되는 다항 근사식을 직교 다항 변환으로 확장하였으며, 각 변환 계수를 차수별로 분할하여 벡터 양자화하는 대역 분할 방법을 사용하여 domain 영상을 부호화하였다.

II 장에서는 영상부호화에 이용되는 기존의 반복 변환 시스템 이론을 살펴보고, III 장에서는 기존의 반복 변환 시스템이 어떻게 영상부호화에 적용되는 지를 설명한다. IV 장에서는 프랙탈 근사화와 벡터 양자화를 이용한 제안한 부호화 방법을, V 장에서는 실험 결과 및 토의를 보였으며, 마지막으로 VI 장에서는 결론을 맺었다.

II. 반복 변환 시스템

프랙탈 영상 부호화는 영상의 자기유사성을 이용하여 영상을 부호화한다. 이는 Barnsley⁽¹⁾가 제안한 반복 변환 시스템에 의해 원영상의 정보를 효과적으로 이용할 수 있기 때문이다. 본 장에서는 프랙탈 부호화에 사용되는 반복 변환 시스템과 변환에 사용되는 변환식에 대하여 살펴본다.

1. 수학적 배경⁽¹⁻³⁾

최도 공간(metric space)에서 정의되는 변환 $f: X \rightarrow X$ 가 다음의 조건

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

을 만족하는 상수 $0 \leq s < 1$ 를 가질 때 변환 f 를 축소적(contractive) 또는 축소 매핑(contraction mapping)이라고 한다.⁽¹⁾ 여기서 d 는 공간 X 의 점들의 쌍 x 와 y 사이의 거리를 나타내는 척도(metric)를 말하며, s 는 변환 f 의 축소율(contractivity factor)이라고 한다. 변환 f 가 축소 매핑이라고 하면, f 는 $f(x_f) = x_f$ 를 만족하는 단 하나의 부동점(fixed point) $x_f \in X$ 를 가진다. 또한, 임의의 점 $x \in X$ 에 대하여 반복 변환 $\{f^n(x) : n=0, 1, 2, \dots\}$ 는 x_f 로 수렴한다. 즉,

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

을 만족한다. 여기서 반복 변환 $f^{\circ n}: X \rightarrow X$ 는 $f^{\circ 0}(x)=x, f^{\circ 1}(x)=f(x), f^{\circ(n+1)}(x)=f^{\circ}f^{\circ n}(x)=f(f^{\circ n}(x)), n=0, 1, 2, \dots$ 로 정의되는 변환들이다. 식 (1)은 두 입력 영상이 변환에 의해 거리 d 가 즉, 오차가 줄어드는 것을 말하며, 반복 변환에 의해 식 (2)로 표현되는 똑같은 영상인 부동점으로 수렴함을 뜻한다.

프랙탈 부호화에 이용되는 반복 변환 시스템⁽¹⁻³⁾은 척도 공간 (X, d) 에서 정의되는 축소율이 $s_n, n=1, 2, \dots, N$ 인 축소 매핑 $w_n: X \rightarrow X$ 의 유한 집합이다. 반복 변환 시스템은 $\{x: w_n, n=1, 2, \dots, N\}$ 로 표기하며 축소율은 $s = \text{Max}\{s_n; n=1, 2, \dots, N\}$ 로 주어진다. 반복 변환 시스템 $\{x: w_n, n=1, 2, \dots, N\}$ 의 축소율이 s 라고 하면, X 에 포함되는 유한한 공간 $H(X)$ 에 대하여

$$W(B) = \bigcup_{n=1}^N w_n(B), \quad \forall B \in H(X) \quad (3)$$

와 같이 정의되는 변환 $W: H(X) \rightarrow H(X)$ 는

$$h(W(B), W(C)) \leq s \cdot h(B, C), \quad \forall B \in H(X) \quad (4)$$

를 만족하는 축소 매핑이다. 여기서, B 와 C 는 공간 $H(X)$ 에 포함되는 점들의 집합이며, h 는 두 점사이의 거리 d 에 의해 정의되는 두 집합 사이의 Housdorff 거리⁽¹⁾를 나타낸다. 식 (3)과 (4)는 영상을 블럭 단위로 표현하여 각 블럭에 대한 변환 w_n 을 결정함으로써 전체 영상을 반복 변환 시스템 W 로 표현할 수 있음을 나타내며 N 은 분할된 블럭의 갯수를 의미한다. 영상 공간은 3차원 벡터 공간 X 에 대한 집합 $H(X)$ 로 표현되며, 크기가 $I \times J$ 이며 K 개의 밝기값으로 양자화된 영상의 경우 2차원의 영상 평면 $\{(i, j): 0 \leq i < I, 0 \leq j < J\}$ 과 평면좌표 (i, j) 에서의 밝기값 $f(i, j), 0 \leq f(i, j) < K$ 로 이루어진 3차원 벡터공간의 부분집합 $H(X)$ 로 생각할 수 있다. 그리고, 척도 h 의 경우에는 두 영상 영역간의 왜곡을 나타내는 오차함수로 나타낼 수 있다. 변환 W 의 유일한 부동점 $A \in H(X)$ 는

$$A = \lim_{n \rightarrow \infty} W^{\circ n}(B), \quad \forall B \in H(X) \quad (5)$$

로 주어지며, 부동점 A 를 반복 변환 시스템의 끝개라고 한다. 반복 변환 시스템을 이용하는 프랙탈 부호화 기법은 변환 시스템의 부동점 즉 영상 끝개로 영상을 복원한다.

만약, 척도 공간 (X, d) 에서 $B \in H(X)$ 과 $\epsilon \geq 0$ 가 주어질 때,

$$h\left(B, \bigcup_{n=1}^N w_n(B)\right) \leq \epsilon \quad (6)$$

를 만족하는 축소율이 $0 \leq s < 1$ 인 반복 변환 시스템 $\{X: w_1, w_2, \dots, w_n\}$ 을 결정하면, 그 변환 시스템의 끝개 A 에 대하여

$$h(A, B) \leq (1 - s)^{-1} h\left(B, \bigcup_{n=1}^N w_n(B)\right), \quad (7)$$

$$\forall B \in H(X)$$

를 만족한다(collage theorem).⁽¹⁻³⁾ 식 (7)은 주어진 영상 B 를 반복 변환 시스템 $W(B) = \bigcup w_n(B)$ 로 표현할 경우 이 시스템의 끝개 A 가 척도 h 에 대하여 실제 영상 B 에 가까워짐을 나타낸다. 즉, 주어진 영상에 대하여 이 영상에 가까운 변환 $W(B)$ 를 찾는다면 우리는 실제 영상의 정보없이 변환 시스템의 끝개로서 영상을 복원할 수 있음을 뜻한다.

2. 축소 매핑

일반적으로 관계 변환식(affine transform)^(1-4,5)을 축소 매핑으로 이용하였으나, Li 등⁽¹¹⁾은 두 영역의 유사성을 보다 효과적으로 기술하기 위하여 관계 변환식을 일반적인 비선형 변환식 $w_n: Q^{n+1} \rightarrow Q^{n+1}$ 으로 확장하였다. Q^{n+1} 은 실수를 원소로 하는 $n+1$ 차원의 벡터 공간 R^{n+1} 의 부분집합이며, 척도 d^{n+1} 은 K^{n+1} 에서 정의된다. 변환 $W_n: Q^{n+1} \rightarrow Q^{n+1}$ 은

$$w_n \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} G_n(a) \\ T_n(a) + C_n(b) \end{bmatrix} \quad (8)$$

와 같이 정의되며, 여기서 $a \in K^n, b \in K, T_n: K^n \rightarrow K, G_n: K^n \rightarrow K^n$, 그리고 $C_n: K \rightarrow K$ 이다. G_n 과 C_n 이 각각 d^n 과 d 에 대하여 축소 매핑이며, T_n 이 Lipschitz 연속

일 경우 w_n 은 d^n 에 대하여 축소 매핑이다.⁽¹¹⁾ 3차원 영상 공간의 경우 n 은 2이며, a 는 영상 평면의 좌표 (i, j) 에 b 는 a 에서의 밝기값 $f(i, j)$ 에 해당한다. 변환 w_n 에서 G_n 은 domain블럭을 range블럭으로 매핑시키는 기하학적인 형태 변환이며, T_n 과 C_n 은 매핑되는 블럭의 화소값을 결정짓는 수치 변환이다. 기존의 프랙탈 부호화에서 변환 T_n 으로는 영상좌표 (i, j) 를 이용한 다항 근사식이 사용되며, C_n 은

$$C_n(f(i, j)) = \alpha_n f(i, j) \quad (9)$$

의 변환 형태가 주로 사용된다. 여기서 α_n 을 scaling factor라고 하며 축소율을 만족하기 위해서는 절대값이 1을 넘지 않아야 한다. 변환 T_n 은 단순한 영상 근사화 부분이며 영상의 자기 유사성을 이용하는 프랙탈 개념은 변환 C_n 과 G_n 에 의해서 구현된다.

III. 기존의 프랙탈 영상 부호화

1. Jacquin의 방법

Jacquin^(4,5)의 경우 부호화하고자 하는 원영상을 $M \times M$ 크기의 range 블럭으로 분할하며, 각각의 range 블럭에 대하여 탐색하게 될 $D \times D$ ($D=2M$) 크기의 domain 블럭을 영상내에서 결정한다. Domain 블럭은 영상을 수직·수평으로 $M/2$ 씩 이동하면서 겹쳐서 선택한다. 각각의 domain 블럭과 range 블럭은 속성에 따라 shade 블럭, midrange 블럭, 그리고 edge 블럭으로 분류되며, shade 블럭인 domain 블럭은 domain pool에서 제외된다. 먼저, 각 domain 블럭을 range 블럭의 크기로 축소시키기 위하여 네화소의 평균값을 하나의 화소로 대응시킨다.

Range 블럭이 shade 블럭인 경우 수치 변환은 range 블럭의 평균 밝기값 β_n 만으로 이루어지며, range 블럭이 midrange 블럭인 경우에는 scaling factor α_n 와 평균 밝기값 β_n 을 변화시키는 수치 변환을 이용한다. Scaling factor는 $\{0.7, 0.8, 0.9, 1.0\}$ 의 값 중에서 선택되며, 선택된 scaling factor로 변환된 domain 블럭과 range 블럭과의 평균 빛의 밝기값이 같도록 β_n 값을 더해준다. 마지막으로 range 블럭이 edge 블럭인 경우 scaling factor와 평균 밝기값을 변

화시키는 수치 변환과 두 블럭을 매핑시키는 8개의 서플 변환^(4,5)을 사용한다. 이 때, α_n 과 β_n 을 계산하기 위해 형태 변환을 거친 domain 블럭과 range 블럭을 각각 어둡고 밝은 두 영역으로 영역화하며, domain 블럭에서의 밝은 영역과 어두운 영역의 밝기값 차가 range 블럭에서의 차와 같아지도록 scaling factor α_n 을 계산한다. 계산된 값은 미리 결정된 네개의 값으로 양자화되며, C_n 변환후의 domain 블럭에서의 어두운 영역 또는 밝은 영역의 밝기값이 range 블럭의 어두운 영역 또는 밝은 영역의 밝기값과 같아지도록 β_n 을 계산한다.

2. Monro와 Dudbridge의 방법

Monro와 Dudbridge⁽⁶⁾의 방법에서는 Jacquin의 방법과는 달리 range 블럭대신 $D \times D$ 크기의 domain 블럭으로 영상을 분할한다. 분할된 각 domain 블럭들은 하나의 분할점 (splitting point)을 갖도록 다시 네개의 range 블럭으로 나눈다. 즉, 하나의 domain 블럭이 네개의 range 블럭으로 대응되도록 하였다. 또한, Monro⁽⁷⁾는, 보다 나은 domain 블럭의 선택을 위하여 정해진 범위만큼 주위의 블럭을 탐색하도록 하였다.

Monro와 Dudbridge⁽⁶⁾는 8개의 서플 변환중에서 회전 변환만을 이용하였으며, Monro⁽⁷⁾는 서플 변환을 사용하지 않았다. 따라서 회전 변환에 대한 탐색 과정과 탐색 영역에서의 domain 블럭에 대한 탐색 과정만이 필요하다. 그러나 이 경우에도 제한된 domain 블럭에서 탐색이 이루어지므로 영상내의 전체 블럭에 대하여 domain을 탐색하는 Jacquin의 방법보다 수행시간이 훨씬 빠르다. 그러나 실제 영상에서 range 블럭에 대한 최적의 domain 블럭을 선택할 수 없기 때문에 성능은 조각 변환 방법에서보다 떨어진다. 이를 해결하기 위하여 Monro와 Dudbridge는 최적의 domain 블럭을 선택하는 대신 다항 근사식 T_n 의 차수를 높여 근사화 정도를 높였다. 다항식의 계수값들은 range 블럭의 화소값과 변환된 domain 블럭의 화소값 차를 최소화하는 최소자승법으로 계산된다.

3. Lepsoy 등의 방법

Lepsoy 등⁽⁸⁾의 방법에서는 다른 프랙탈 부호화 기법

과는 달리 반복 변환을 이용하여 영상을 복원하지 않으며,

$$A = W(A) = \sum_{n=1}^N w_n(A) \quad (10)$$

를 만족하는 부동점 A 를 직접적으로 구한다. 이를 위해서는 변환에 이용되는 domain 블럭이 반복 변환 시스템에 의해 영향을 받지 않아야 한다. 따라서 Lepspy 등은 range 블럭을 $M \times M$ 으로 설정할 때 domain 블럭을 $M^2 \times M^2$ 의 크기로 설정하였다. 즉, domain 블럭을 range 블럭으로 매핑할 경우 domain 블럭내의 $M \times M$ 블럭이 range 블럭의 한 화소로 대응되도록 하여 반복 변환에 의한 영상의 변화를 흡수하도록 하였다. 그러나 부동점을 생성하기 위해서는 반복 변환을 하거나 domain 블럭을 직접 이용하여야 한다. Lepspy 등의 방법에서는 다른 방법과는 달리 변환 C_n 을

$$\begin{aligned} C_n[f(i,j)] &= a_n O_n[f(i,j)] \\ &= a_n (f(i,j) - E_n[f(i,j)]) \end{aligned} \quad (11)$$

와 같이 정의하였다. 여기서 변환 E_n 은 $f(i,j)$ 블럭을 변환 T_n 으로 근사화함을 뜻한다. 즉 O_n 은 $f(i,j)$ 블럭에서 근사화 다항식의 기저 함수로 근사화가 가능한 부분을 제거하는 함수이다. 따라서, 근사화 다항식의 기저 함수로 상수항을 사용할 경우 블럭의 평균값으로 근사화되며 수신단에서는 전송되는 평균값을 이용하여 domain pool을 복원한다. O_n 변환은 공간상에서 축소된 domain 영상의 밝기 패턴을 range 영상에 대한 밝기 패턴으로 변환시키는 orthogonalization 함수이다. 따라서 T_n 의 기저 함수와 변환 O_n 을 거친 domain 영상은 직교성을 가지며 각각 range 영상을 근사화하는 직교 기저 함수로 사용된다.

IV. 프랙탈 근사화와 벡터 양자화를 이용한 제안한 영상 부호화 방법

제안한 프랙탈 부호화⁽¹⁵⁾에서는 다항식을 이용하는 근사화와 scaling factor를 이용하는 프랙탈 근사화로 나누어 부호화한다. 다항 근사화에서는 직교 다항 변환을 이용하여 변환계수를 부호화하는 변환 벡터 양자화 기법을 사용하며, 프랙탈 근사화에서는 다항식으로 근사화가 이루어지지 않는 교차 영역을 영상내의 유사블

럭으로 근사화한다. 프랙탈 근사화에서는 실제 원영상 대신 다항 함수로 근사화된 영상을 domain pool로 사용하며, 프랙탈 근사화를 위한 변환에서 Lepspy 등이 사용한 변환 O_n 을 사용함으로써 근사화 부분을 제거한 교차 영역에서의 유사 형태를 이용한다.

1. Domain pool의 생성

프랙탈 부호화의 벡터 양자화 부호화는 특정 영상 블럭들로부터 원래의 블럭을 복원한다. 프랙탈 부호화의 경우에는 원영상이, 벡터 양자화의 경우에는 여러 영상으로부터 생성된 부호책(codebook)이 각각 사용된다. 벡터 양자화와는 달리 프랙탈 부호화는 원영상을 부호책으로 이용함으로써 영상이 갖는 유사성을 이용할 수 있다는 장점이 있다. 여기서 유사성이란 영상이 갖는 밝기값 표면의 분포 특성을 말한다. 벡터 양자화의 경우에는 여러 영상에 대한 최적의 부호책을 생성하기 위하여 학습과정이 필요하며, 각각 다른 특성이 있는 영상들에 대해 똑같은 부호책을 사용함으로써 각 영상의 전체적인 특성을 이용할 수 없다는 단점이 있다. 반면, 프랙탈 부호화는 입력 영상을 부호책으로 사용하기 위해서 수축도를 만족하는 변환을 이용해야 하며, 원영상에서 부호벡터(codevector)를 찾기 위한 탐색 시간이 많이 요구되는 단점이 있다. 따라서 제안한 방법에서는 원영상의 밝기 특성을 부호책으로 사용하는 프랙탈 기법과 영상의 직접적인 부호벡터 정보를 이용할 수 있도록 하는 벡터 양자화를 결합한 방법을 제안하였다.

그림 1에 제안한 방법의 전체 블럭도를 나타내었다. 먼저 영상이 주어지면 quad-tree를 이용하여 여러 크기의 블럭들로 분할한 후 이를 다항 근사식을 통하여 근사화한다. 근사화된 블럭은 프랙탈 근사화를 위한 domain pool로 사용되며, 원영상과 근사화영상과의 차영상을 range 영상으로 사용하여 프랙탈 근사화를 수행한다. Range 영상에 대한 프랙탈 근사화에서는 quad-tree분할에서 사용된 블럭정보를 이용하여 각각 다른 양자화기를 사용하도록 하였다. 즉, 제안한 방법에서는 영상 복원에 사용되는 domain 정보로 근사화된 원영상을 사용함으로써 원영상의 밝기 특성을 이용하는 프랙탈 부호화 특성을 이용하도록 하였다. 다항 근사화와 블럭 분할에 대한 자세한 내용은 각각 2절과 3절에서 설명할 것이다.

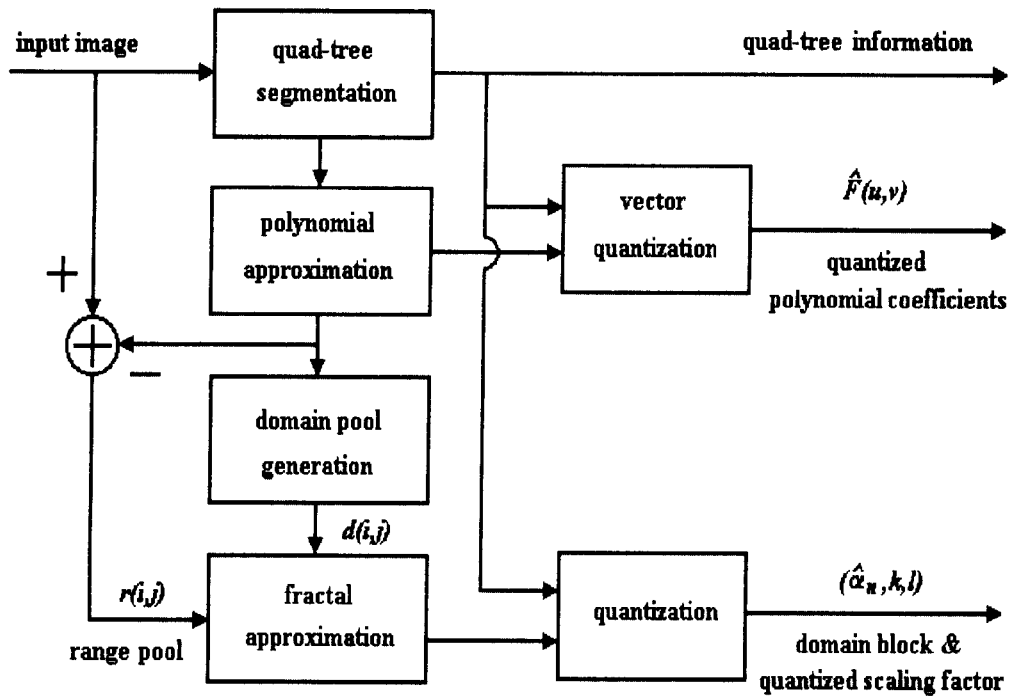


그림 1. 제안한 방법의 블록도
Fig. 1. Block diagram of the proposed method

제안한 방법은 반복적인 변환없이 영상을 복원한다는 점에서 Lepsoy 등의 방법과 비슷하나 Lepsoy 등의 방법에서는 부동점으로 영상을 표현하기 위해서 domain 블록의 크기를 크게 설정한다. 따라서 domain pool의 크기가 제한되고 실제 영상이 갖는 다양한 형태의 밝기 값 분포를 사용할 수 없는 단점이 있다. 또한, 평균 영상을 domain 블록으로 사용하지만, 반복 변환 시스템을 이용함으로써 결과적으로 축소 매핑의 제한 조건을 만족해야 한다. 반면, 제안한 방법에서는 벡터 양자화 결과를 domain 블록으로 이용함으로써 domain 블록의 크기에 대한 제약이 없으며, 단순한 평균 영상만을 이용하는 방법보다 원영상의 정보를 충분히 이용할 수 있는 장점이 있다. 이는 제안한 방법에서는 직접적인 domain 정보를 이용함으로써 반복 변환 시스템의 제한 조건없이 프랙탈 근사화가 이루어지기 때문이다.

2. 변환식

부호적으로 사용하는 근사화 영상은 프랙탈 부호화에서 사용되는 다항 근사식을 이용하여 생성하였다. 기존의 방법에서 사용된 다항식은 영상의 위치 즉, (i, j) 의 0내지 3차 정도의 근사식⁽⁴⁻¹³⁾을 이용하며, 근사화 오차를 domain 블록으로 보상하는 형태를 취한다. 제안한 방법에서는 다항 근사식을 블록 전체에 대한 직교 기저 함수를 이용하여 정의되는 변환식

$$F(u, v) = \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} f(i, j) P_u(i) P_v(j) \quad (12)$$

$$f(i, j) = \sum_{u=0}^{M-1} \sum_{v=0}^{M-1} F(u, v) P_u(i) P_v(j) \quad (13)$$

을 이용하였다. 여기서 $f(i, j)$ 는 영상 평면 (i, j) 에서의

밝기값을, $F(u, \nu)$ 는 변환 평면에서의 기저 함수에 대한 계수값을 나타내며, M 은 블럭의 크기를 나타낸다. P_n 는 직교 정규 기저 함수로서 n 차 Legendre 다항식^(9,16)으로 표현되는 함수이다. 따라서, $P_u(i)P_\nu(j)$ 는 영상 위치 (i, j) 에 대한 다항 영역 (u, ν) 에서의 기저 함수값을 나타내며, $f(i, j)$ 와의 내적 연산을 통하여 $F(u, \nu)$ 값을 계산할 수 있다. 역변환은 식 (13)과 같이 표현되며, 원영상의 밝기값 $f(i, j)$ 를 복원할 수 있다. 식 (12)은 기존의 프랙탈 부호화에서 이용되는 다항식을 $M \times M$ 블럭에 대한 일반화된 형태로 나타낸 것이다.

제안한 방법에서는 원영상을 domain pool로 사용하는 대신 다항식으로 근사화한 결과를 이용하여 복호화 과정에서 이를 이용할 수 있도록 하였다. 다항 근사식으로 근사화된 영상을 이용함으로써 실제 원영상을 이용하지는 않지만 원영상의 전체적인 밝기 특성을 이용할 수 있다. 근사화된 영상의 경우 필요한 비트수를 줄일 수 있도록 변환 벡터 양자화를 이용하여 부호화하였으며, 변환 계수값들은 대역별로 나누어 각각 벡터 양자화하는 벡터 분할 기법을 사용하였다.^(17,18) Quad-tree로 분할된 영상은 각 블럭 크기에 따라 근사화 차수를 달리 하여 식 (12)로 근사화하며, 근사화에 사용된 계수값들은 벡터 양자화를 통하여 부호화한다.

3. 블럭 크기 결정

제안한 방법에서는 여러 크기의 블럭을 사용함으로써 영상 정보에 적응적인 부호화 시스템을 이용하였다. 가변 크기의 블럭 근사화는 프랙탈 근사화의 화질 개선과 비트를 감소를 위하여 사용되었던 방법이다. 제안한 방법에서도 가변 블럭을 이용하여 부호화하였으며, 기존의 분산^(3,18)이나 PSNR(Peak Signal to Noise Ratio)^(4,8)을 문턱값으로 하는 방법과는 달리 프랙탈 차원을 이용하여 블럭의 크기를 결정하였다. 프랙탈 차원은 영상 표면의 거친 정도를 나타내는 값으로서 프랙탈 영상의 특성을 표현한다.^(1-3,19,20)

Domain 블럭과 range 블럭간의 자기 유사성을 높이기 위한 영역 분할 방법들에는 블럭들을 같은 크기의 네블럭으로 분할하는 quad-tree 분할 방법, 하나의 블럭을 수평이나 수직으로 나누어 두개의 블럭으로 분할하는 HV 분할 방법, 삼각형에 기반을 두어 사각형이 아닌 삼각형의 블럭들로 나누는 방법^(8,10) 등이 있다.

Quad-tree 분할 방법이 영역내의 화소값에 상관없이 똑같은 크기로 분할하는 반면, HV 분할 방법이나 삼각형 분할 방법은 분할되는 블럭의 크기나 위치가 다양하다. 따라서 HV 분할 방법이나 삼각형 분할 방법이 보다 나은 자기 유사성을 표현할 수 있지만 부호화의 과정이 복잡하게 되며 많은 비트가 필요한 단점이 있다. 따라서, 제안한 방법에서는 블럭 분할에 대한 최소한의 정보와 구현의 용이함 때문에 quad-tree 분할 방법을 사용하였다. 가변 블럭은 16×16 크기를 최대로 하여 4×4 크기까지 분할하였으며, 각 블럭의 프랙탈 차원을 구하기 위하여 자기 변환 모델을 이용하였다. 즉, 블럭을 똑같은 크기의 네 블럭으로 나누어 전체 크기의 블럭이 네개의 블럭에 매핑되는 반복 변환 시스템을 결합함으로써 프랙탈 차원을 구한다. 프랙탈 차원을 구하는 방법으로 fBm(fractional Brownian motion) 모델⁽¹⁹⁾을 사용하는 경우도 있으나 통계적 모델에 의한 fBm으로 일반적인 영상을 표현하는 것은 힘들며, 프랙탈 차원을 fBm 모델을 이용하여 구하는 경우는 대부분이 텍스처(texture) 영상에 대한 경우이다. 이는 텍스처 영상이 통계적 특성을 갖기 때문이다. 따라서 제안한 방법에서는 일반적인 영상에 대한 밝기값 패턴의 유사성을 새로운 형태의 구조적 유사성으로 모델화하여 영상을 부호화하는 반복 변환 시스템을 적용하였다. 따라서 부호화와 복호화 그리고 가변 블럭 분할 모두가 반복 변환 시스템으로 해석될 수 있다.

반복 변환 시스템 $\{X; w_n, n=1, 2, \dots, N\}$ 가 주어질 때 끌개 A 의 프랙탈 차원 $D(A)$ 는

$$\sum_{n=1}^N |s_n|^{D(A)} = 1, \quad D(A) \in [0, m] \quad (14)$$

의 유일한 해로 주어지며, m 은 양의 실수값을, s_n 은 변환 w_n 의 축소율을 나타낸다.⁽¹⁻³⁾ 따라서 네개의 축소 매핑에 대한 계수값을 구한 후 식 (14)를 이용하여 프랙탈 차원값을 추정한다. 제안한 방법의 블럭 분할 흐름을 그림 2에 나타내었다. 먼저 입력 블럭 B 에 대한 반복 변환 시스템 $\{H(X); w_n, n=1, 2, 3, 4\}$ 을 결정하며, 축소율 s_n 을 이용하여 $D(A)$ 가 1일 때의 식 (14)값과 문턱값 s_{th} 을 비교하여 블럭 분할 여부를 결정한다. 축소 매핑을 위한 변환으로는 식 (9)의 C_n 변환식만을 사용하였으며, 축소율 s_n 은 식 (1)을 이용하여 α_n 의 제곱

으로 근사화하였다. 추정된 프랙탈 차원은 실제 영상이 아닌 변환의 끝개에 대한 차원으로 주어진다. 그러나, 원영상과 유사한 끝개를 결정하도록 하는 콜라즈 이론은 끝개에 대한 특성이 충분히 원영상의 특성을 표현하도록 하며, 이 프랙탈 차원값을 구분하여 블럭을 분할할 수 있도록 한다.

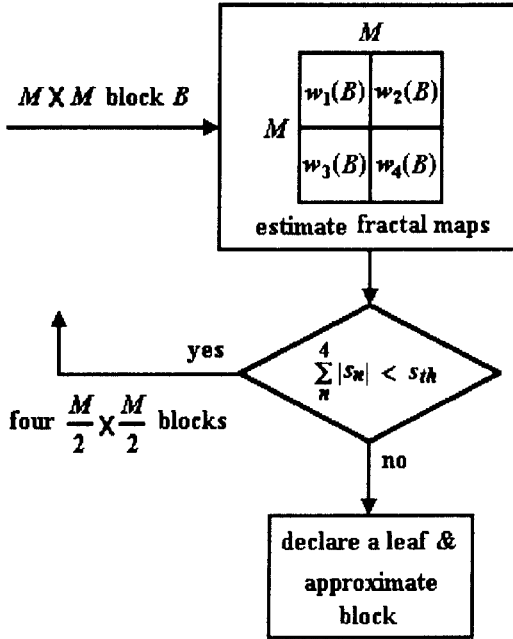


그림 2. 프랙탈 차원을 이용한 제안한 가변 블럭 분할 흐름도
Fig. 2. Flowchart of the proposed variable block size segmentation using fractal dimension

4. 프랙탈 근사화

근사화 영상이 생성되면 이 영상을 domain pool로 사용하여 근사화 오차를 부호화한다. 즉 근사화된 영상과 원영상과의 오차를 프랙탈 근사화를 통하여 부호화한다. 변환식은 사용하지 않으며, shape-gain 벡터 양자화^(17,21)에서와 같이, 사용할 블럭에 대한 정보, 즉 shape정보와 C_n 즉, gain만을 이용한다. 이는 domain 블럭 정보를 직접적으로 사용함으로써 수축도를 만족시킬 필요가 없기 때문이다. Jacquini가 사용한 8가지 변환은 탐색 영역이 학습과정을 거치지 않은 블럭들이기 때문에 다양한 형태의 밝기값 표면을 생성하기

위해 사용된 변환이다. 그러나, 그중 사용빈도가 적은 변환들이 있으며, 탐색 시간과 정보에 비해 성능 향상이 작기 때문에 선택되는 빈도수에 따라 4개만을 이용하도록 하였다.⁽¹⁵⁾

실제 영상 블럭으로부터 근사화 영상 즉, 직교 성분을 제거하고 나머지 성분만을 프랙탈 근사화하는 과정은 영상에서 블럭의 평균값을 제거한 뒤 나머지 차영상을 양자화하는 mean/residual 벡터 양자화나, 샘플링한 영상을 보간하여 그 residual 영상을 양자화하는 interpolative/residual 벡터 양자화^(17,21)와 비슷하다. 그러나, 제안한 방법은 입력 영상마다 각각의 근사화 영상을 부호책 즉, domain 정보로 사용함으로써 입력 영상의 밝기 정보를 참조하는 동적 형태의 부호책을 사용하는 점이 다르다. Domain 블럭을 선택하기 위해 사용되는 오차 함수 h 는

$$h(r, C_n(d)) = \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} (r(i, j) - \alpha_n O_n[d(i+k, j+l)])^2 \quad (15)$$

와 같이 주어지며, O_n 은 Lepsey 등이 사용한 근사화 성분을 제거하는 함수를 나타낸다. Range 블럭과 변환된 domain 블럭과의 차이를 나타내는 오차함수 h 는 domain 블럭의 위치 (k, l) 과 scaling factor α_n 에 의해 결정된다. O_n 변환은 다항 근사화된 domain pool의 밝기 특성을 range 영상인 근사화 오차와 같아지도록 하기 위한 것이다. 따라서 최종적인 복원 영상 $R(i, j)$ 는

$$R(i, j) = T_n + \alpha_n [d(i+k, j+l) - T_n] \quad (16)$$

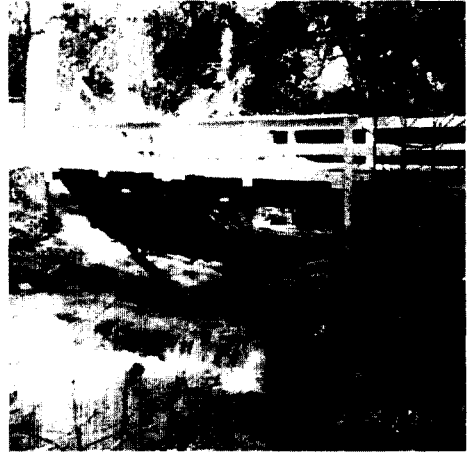
으로 복원된다. 여기서 T_n 은 입력 영상을 기저 함수로 근사화하는 부분이며 T_n' 은 domain 영상을 기저 함수 T_n 과 만족하도록 하는 orthogonalization 부분이다.

V. 실험 결과 및 토의

실험 영상으로는 8비트로 양자화된 512x512 Lenna 영상과 Bridge 영상을 사용하였으며, range 블럭의 크기는 16x16에서 4x4까지의 가변 블럭을 사용하였다. 제안한 방법의 경우 근사화에서 사용되는 벡



(a) Lenna 영상



(b) Bridge 영상

그림 3. 실험 영상
Fig. 3. Test images

표 1. 각 블럭 크기에 대한 블럭의 갯수
Table. 1. Number of blocks for each block size

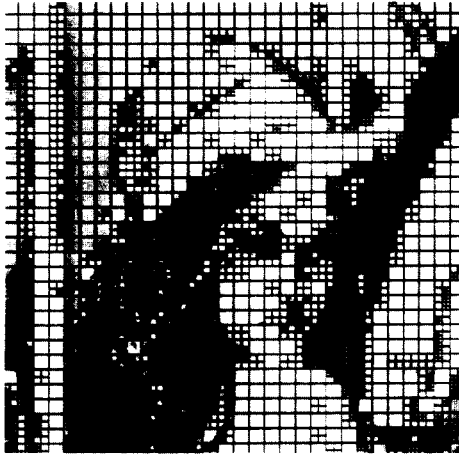
블럭 크기	Lenna 영상			Bridge 영상		
	Vaisey(45,16)	Vaisey(60,8)	제안한 방법	Vaisey(45,14)	Vaisey(60,6)	제안한 방법
16×16	551	432	564	138	34	141
8×8	1158	1119	715	1837	1075	670
4×4	2936	4996	4500	6828	11540	11448

터 양자화를 위하여 512×512 크기의 9개 영상 (Bank, Girl, Crowd, Baboon, Man, Cable Car, Jaguar, Einstein, Ringnecked)을 이용하였다. 그리고, 복원한 영상의 성능비교를 위해서

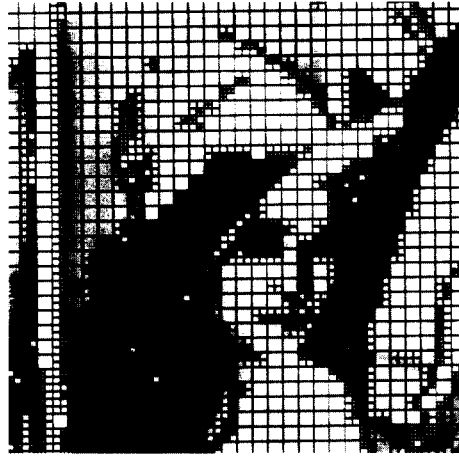
$$PSNR = 10 \log_{10} \left(\frac{255^2}{\frac{1}{I \cdot J} \sum_{i=0}^{I-1} \sum_{j=0}^{J-1} (f(i,j) - R(i,j))^2} \right) \quad (17)$$

와 같이 정의되는 PSNR을 이용하였다. 여기서 $f(i,j)$ 는 원영상을, $R(i,j)$ 는 부호화/복호화한 영상을 나타낸다. 영상 크기는 $I \times J$ 이며 최대 밝기값은 255이다. 그림 3은 실험 영상 Lenna와 Bridge 영상을 나타낸다.

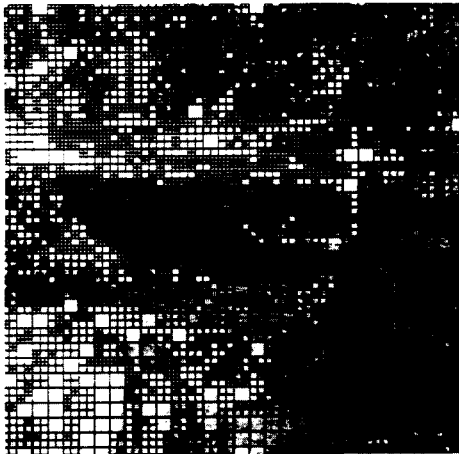
먼저 영상 분할 결과를 비교하기 위하여 분산과 평균을 이용하는 Vaisey⁽¹⁸⁾의 결과와 제안된 프랙탈 차원을 이용한 결과를 표 1에 나타내었다. Vaisey 방법의 경우 분산과 평균에 대한 문턱값을 Lenna 영상의 경우에는 각각 45, 16인 경우와 60, 8인 경우를 보였으며, Bridge 영상의 경우에는 45, 14 그리고 60, 6인 결과를 나타내었다. 이는 제안한 방법과의 비교를 위하여 각각 최소 블럭의 갯수가 비슷한 경우와 최대 블럭의 갯수가 비슷한 두 경우를 보이기 위한 것이다. 그림 4는 블럭 분할한 결과를 나타낸 것으로 제안한 방법의 경우 에지에서 4×4 블럭 빈도수가 Vaisey의 방법보다 높은 것을 볼 수 있다. 또한 Lenna 영상의 어깨 부분에서와 같이, Vaisey의 방법과는 달리 네개의 부분



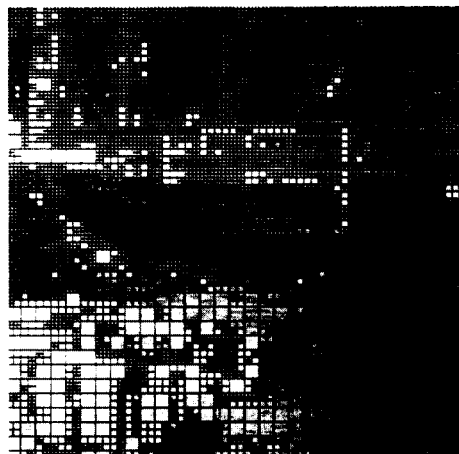
(a) Vaisey(Lenna 영상, 45, 16)



(b) 제안한 방법(Lenna 영상)



(c) Vaisey(Bridge 영상, 45, 14)



(d) 제안한 방법(Bridge 영상)

그림 4. 블럭 분할 결과

Fig. 4. Block segmentation results

력에 대한 밝기값 변화가 상위 블럭의 밝기값 변화와 유사할 경우 이를 하나의 블럭으로 처리하는 것을 볼 수 있으며, 이는 유사성을 척도로 하는 프랙탈 차원값을 이용하기 때문이다. 프랙탈 근사화의 경우 블럭의 크기가 커짐에 따라 원영상내의 유사블럭이 존재할 확률이 줄어들어 근사화 결과가 나빠지기 때문에 본 연구에서는 16×16 을 블럭 최대 크기로 하였다. 이는 프랙탈 부호화에서 원영상 크기를 줄여 domain 블럭으로 사용하기

때문이며, 특히 제안한 방법의 경우에는 근사화 영상을 행과 열에 대하여 각각 $1/4$ 크기로 줄여 사용하므로, 원영상에서 32×32 블럭으로 분할되는 블럭의 경우 domain pool에서 이와 유사한 32×32 블럭을 찾아내기 어렵기 때문이다. 프랙탈 차원을 이용한 경우, Vaisey의 분산과 평균을 이용하는 방법보다 최소 블럭과 최대 블럭으로 분할되는 블럭의 갯수가 상대적으로 많은 것을 알 수 있다. 프랙탈 차원을 이용한 방법은 단

순한 블럭 전체의 특성보다는 블럭내의 영역에 대한 상관 관계를 이용한다. 또한, 프랙탈 차원은 표면의 거칠기에 대한 정량적인 값으로서 Vaisey가 블럭의 크기마다 분산과 평균에 대한 문턱값을 다르게 설정한 것과는 달리 하나의 문턱값을 사용하여 영상을 분할할 수 있는 장점이 있다. 그림 4는 제안한 방법의 프랙탈 차원 추정으로 영상을 분할할 수 있음을 보여준다. 그러나 제안한 방법의 최종 결과에는 두 방법의 경우 큰 변화가 없었다.

영상 근사화를 위하여 각 블럭 크기마다 각기 다른 차수의 다항 근사식을 사용하였으며, 직교 다항 변환의 각 대역별로 즉, 다항식의 차수마다 벡터 분할하여 그 계수값을 벡터 양자화하였다. 프랙탈 근사화는 원영상의 밝기값을 그대로 이용하기 보다는 그 형태를 이용하

는 방법이다. 따라서 domain pool을 생성하기 위한 근사화에서 그 성능보다는 근사화에 필요한 비트수를 줄일 수 있도록 변환 벡터 양자화를 이용하였다. 표 2는 블럭 크기에 따른 벡터 분할과 부호책의 크기를, 그림 5는 변환 평면에서의 변환 계수들의 스캐닝 순서를 나타내고 있다. 16×16 블럭은 (i,j)의 2차 다항식까지, 즉 스캐닝 순서 1번에서 6번까지의 계수로 근사화하며, 1, 2-3, 4-6의 계수들로 벡터 분할하여 각각을 벡터 양자화하였다. 8×8은 (i,j)의 1차 다항식까지 근사화하며 1, 2-3의 계수들로 벡터 분할한다. 그리고 4×4는 평균에 해당하는 1번의 dc 계수값만을 이용하여 영상을 근사화하였다. 분할된 벡터들은 랜덤 초기화를 이용한 LBG(Linde-Buzo-Gray)⁽²²⁾알고리즘으로 양자화하며 그 결과들은 Huffman code⁽²¹⁾로 부호화하였다.

표 2. 각 블럭 크기에 대한 벡터 분할과 부호책 크기
Table. 2. Vector partition and codebook size for each block size

벡터#	블럭 크기					
	16×16		8×8		4×4	
	벡터 분할	부호책 크기	벡터 분할	부호책 크기	벡터 분할	부호책 크기
0	1	64	1	32	1	16
1	2-3	32	2-3	16		
2	4-6	16				

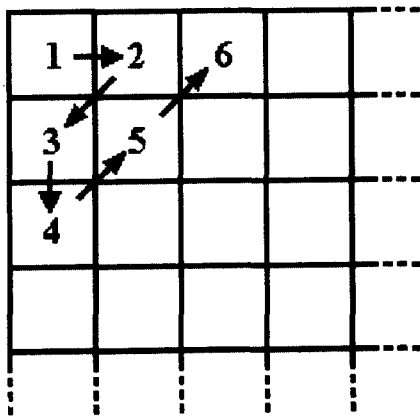


그림 5. 다항 변환 계수에 대한 스캐닝 순서
Fig. 5. Scanning order for polynomial transform coefficients

근사화된 영상을 domain pool로 사용하기 위해서 128×128 영상으로 축소하였다. 이는 영상의 최소 블럭 크기가 4×4로서 하나의 평균값으로 근사화되기 때문이며, 블럭간의 경계효과를 줄이기 위한 것이다. 근사화된 영상의 PSNR과 축소된 원영상과 근사화 영상의 PSNR을 표 3에 나타내었으며, Lenna와 Bridge 영상에 대한 비트율은 각각 0.13과 0.22bpp이다. 표에서 보듯이 축소된 영상의 PSNR이 높은 것을 알 수 있으며, 그만큼 원영상의 정보를 충분히 이용할 수 있다. 낮은 비트수와 에지 등으로 인해 4×4 블럭의 성능이 가장 낮으나, domain pool로 사용하기 위해 축소되고 프랙탈 근사화에서 밝기값 형태를 이용하기 때문에 그 영향은 거의 없다. 다만 근사화 오차인 range 영상을 효과적으로 부호화할 수 있어야 하는데 프랙탈 근사화는 블럭의 크기가 작을수록 좋은 성능을 나타내기 때문에 이를 보상할 수

표 3. 벡터 양자화에 의한 근사화 결과(Lenna:0.13bpp, Bridge:0.22bpp)
Table. 3. Approximation result by vector quantization (Lenna:0.13bpp,Bridge:0.22bpp)

영상 \ 블럭크기	Lenna	Bridge
16×16	34.71	25.12
8×8	32.06	27.15
4×4	22.24	20.51
전체 영상	26.57(36.91)*	21.88(32.36)*

*128×128로 축소된 영상의 PSNR

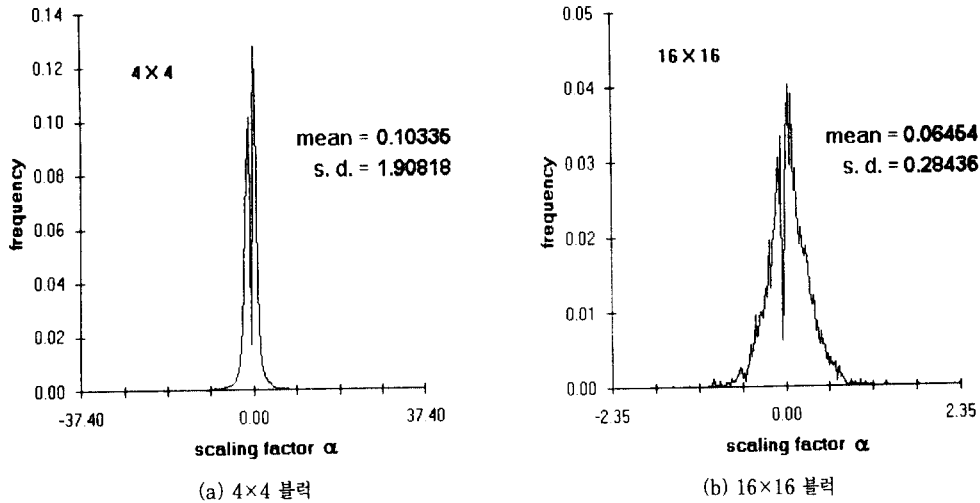


그림 6. Scaling factor의 분포
Fig. 6. Histogram of a scaling factor

있다. Lenna 영상의 경우 16×16 블록이, Bridge 영상의 경우 8×8 블록의 성능이 가장 좋은 것을 알 수 있다. 이는 Lenna 영상의 경우 16×16 블록에서 거의 일정한 밝기값을 갖기 때문이며 Bridge 영상의 경우 예지나 불규칙적인 밝기값으로 인해 성능이 떨어지기 때문이다.

제한한 프랙탈 근사화에서는 Lepsoy 등이 사용한 C_n 변환을 이용하였으며, Jacquin이 사용한 8가지의 서플 변환중에서 영상좌표 (i, j) 의 색인 뒤바뀌지 않는 4개의 변환만을 이용하였다. 이는 (j, i) 와 같이 i 와 j 의 두 좌표값이 뒤바뀌는 변환의 발생빈도가 상대적으로 낮기 때문이며,⁽⁵⁾ 필요한 비트수를 줄이기 위해서는 빈도수가 가장 높은 0° 와 180° 회전 변환만을 이용할 수 있다.

또한, 제한한 방법에서는 탐색 영역을 줄이기 위하여 -15에서 +15 화소까지로 탐색 영역을 제한하였다. 이는 현위치와 선택되는 domain 블록간의 위치차이가 0을 주위로 하여 선형적으로 감소하는 분포를 보이기 때문이다.

프랙탈 근사화는 shape-gain 벡터 양자화와 같이 원영상의 밝기 형태를 shape 정보로 이용한다. 즉, 벡터 양자화에서 필요로 하는 학습과정없이 원영상 자체의 밝기 형태를 그 영상을 부호화하는 과정에 효과적으로 사용할 수 있다. 따라서 밝기값 형태뿐만 아니라 gain에

해당하는 scaling factor 정보가 필요하다. 그림 6에 16×16 과 4×4 블록에 대하여 scaling factor 분포를 나타내었다. 실험에서 사용한 9개의 영상으로부터 구한 scaling factor α_n 은 0을 주위로 그 값이 몰려 있는 분포를 보이며, 블록의 크기가 작아질수록 α_n 의 분포가 넓게 퍼져있는 것을 관찰할 수 있다. 특히 음의 값으로 0에 가까운 scaling factor의 빈도가 주위의 값보다 상당히 작은 것을 볼 수 있다. 제한한 방법에서는 이러한 분포 특성을 이용하여 블록의 크기에 따라 α_n 의 양자화 비트수를 달리 하였다. 양자화를 위해서 분포 특성에 따른 1차원 LBG를 이용하였으며, Huffman code를 이용하여 부호화하였다.

표 4와 5에 프랙탈 근사화로 복원된 range 영상의 PSNR과 최종적으로 복원된 영상의 PSNR을 나타내었으며, 각 블록들에 대한 PSNR도 함께 나타내었다. 표 5의 최종적인 결과는 표 3의 벡터 양자화 결과에 표 4의 프랙탈 근사화 결과를 더해 근사화 오차를 보성한 것이다. Bridge 영상은 Lenna 영상과는 달리 높은 주파수 즉, 높은 차수에서의 정보량이 많은 영상으로서 다항 근사화 정도가 Lenna 영상보다 떨어지는 것을 볼 수 있으며, 분할되는 블록의 개수가 많아 더 많은 비트가 필요한 것을 알 수 있다. 두 영상의 경우 프랙탈 근사화

표 4. Range 영상에 대한 프랙탈 근사화 결과 (Lenna:0.36bpp, Bridge:0.75bpp)

Table. 4. Fractal approximation result for the range image (Lenna :0.36bpp,Bridge:0.75bpp)

블럭크기 \ 영상	Lenna	Bridge
16×16	34.52	29.48
8×8	34.41	30.29
4×4	31.19	27.51
전체 영상	33.31	28.10

에 의한 성능 향상이 6-7dB 정도이며 특히 4×4 블럭에 대한 성능 향상이 가장 높다. 이는 다항 근사화에서 비트수를 줄이기 위하여 최소블럭에 대한 근사화 차수를 0으로 한 결과이며 블럭의 크기가 작을수록 프랙탈 근사화가 잘 이루어지기 때문이다. 전체적인 영상의 화질은 최소 블럭의 화질에 크게 영향을 받는 것을 알 수 있다. 4×4 블럭의 경우 프랙탈 근사화는 평균을 제거한 영상 표현의 형태를 이용함으로써 0차보다 높은 차수의 나머지 정보들을 원영상으로부터 찾아낸다. 즉, 프랙탈 근사화가 높은 차수에 해당하는 원영상의 정보를 충분히 이용할 수 있음을 나타낸다. Lenna 영상의 16×16 블럭의 경우 오히려 프랙탈 근사화에 의한 성능이 낮아지는 것을 볼 수 있다. 그러나 다항 근사화에 의한 비트율을

표 5. 제안한 방법의 영상 복원 결과 (Lenna:0.49bpp, Bridge :0.97bpp)

Table. 5. Reconstruction result by the proposed method (Lenna :0.49bpp,Bridge:0.97bpp)

블럭크기 \ 영상	Lenna	Bridge
16×16	34.52	29.50
8×8	34.82	30.30
4×4	30.68	27.51
전체 영상	33.31	28.11

줄이기 의해 기저 함수의 갯수를 적게 설정하였기 때문에 다항 근사화만으로는 부자연스러운 밝기값 변화와 경계효과가 나타난다. 따라서 프랙탈 근사화에 의한 결과는 residual 영상에 대한 표현으로 보다 자연스럽게 영상을 복원한다. 블럭 분할시 최대 블럭의 크기를 8×8 보다 큰 16×16으로 설정한 결과이다. 표 5에서 보듯이 대체적으로 16×16과 8×8 블럭에서의 성능이 4×4로 한 경우를 나타내었으며, Jacquin과 Lepsoy 등의 방법에서는 8×8과 4×4블럭에 대한 성능보다 높게 나타나며, 이는 분할된 4×4 블럭이 에지나 불규칙 텍스처 등으로 이루어진 영역으로 성능향상을 위해 많은 비트가 필요한 반면, 성능은 거의 향상되지 않는 특성을 보이기 때문이다.^[18]

표 6. 여러 방법의 성능 비교

Table. 6. Performance comparison of various methods

방법 \ 영상 성능	Lenna		Bridge	
	비트율(bpp)	PSNR(dB)	비트율(bpp)	PSNR(dB)
Monro(4×4)	1.38	34.34	1.38	26.45
Monro(8×8)	0.44	30.02	0.44	23.55
Lepsoy 등 (8)	0.71	33.20	1.33	27.93
Lepsoy 등 (12)	0.58	32.24	1.09	27.40
Jacquin (8)	0.63	31.90	1.40	27.15
Jacquin (12)	0.51	31.20	1.14	26.59
제안한 방법	0.49	33.31	0.97	28.11

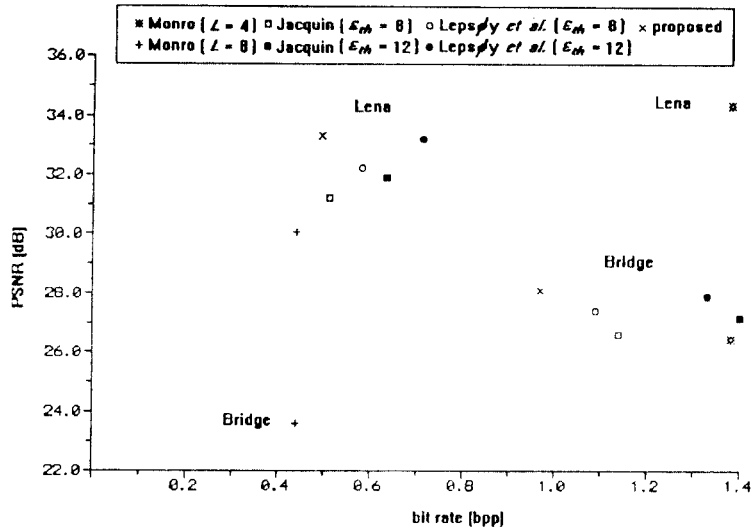


그림 7. 성능 비교
Fig. 7. Performance comparison

표 6에 기존의 방법과 제안한 방법의 PSNR과 비트율을 나타내었으며 Lena 영상과 Bridge 영상에 대한 결과를 그림 7에 나타내었다. Monro와 Dudbrige 방법의 경우 range 블록의 크기를 8×8 과 4×4 의 가변블록을 나누기 위한 문턱값 RMSE(Root Mean Square Error)를 8과 12로 한 두 경우를 나타내었다. 두 방법의 경우 낮은 RMSE에 대해서는 4×4 블록으로 분할되는 영역의 갯수가 많아져 상대적으로 높은 비트율과 좋은 성능을 나타낸다.

Monro와 Dudbrige 방법의 경우 조각 변환에 비해 상당히 빠른 수행시간을 보이는 반면, 블록의 크기가 클 경우 에지나 불규칙 텍스처에서의 근사화 오차 등으로 영상 전체의 뭉그러짐 현상을 나타낸다. 반면 작은 블록 크기에 대해서는 다른 방법에 비해 많은 비트수를 필요로 하는 단점이 있다. Jacquin의 방법은 블록간의 경계효과가 적게 나타나며 연속적인 밝기값에서는 자연스러운 형태의 모습을 복원하는 장점이 있으나, 사용되는 변화과 분할되는 블록의 형태가 다양하여 많은 계산과 비트를 요구하는 단점이 있다. Lepsoy 등의 방법은 Jacquin의 방법에 비하여 domain pool에 대한 제한 조건을 더 많이 요구한다. 특히 영상의 밝기 특성이 불규칙적이고 높은 주파수 신호가 많은 영상에 대해서는

domain pool에 대한 제한은 실제의 영상 정보를 효과적으로 이용할 수 없게 한다. 제안한 방법에서는 작은 탐색 영역을 이용하나 영상의 밝기 형태를 보다 효과적으로 이용함으로써 전체적으로 가장 좋은 성능을 보인다.

그림 8과 9에 복원영상과 원영상과의 오차영상을 각각의 방법에 따라 나타내었으며, 제안한 방법과 비슷한 비트율에서의 결과를 보였다. Lena 영상의 경우에는 오차의 절대값을 10배한 경우를 보였으며, Bridge 영상의 경우에는 큰 오차값들로 인해 절대값을 10배할 경우 대부분이 255를 넘어서며, 작은 스케일에 대해서는 각 방법들과의 차이가 나타나지 않아 절대값 대신 제곱을 하여 0.4배한 경우를 나타내었다. Monro와 Dudbrige의 경우 밝기값이 평활한 부분에서의 오차는 심하지 않으나 에지나 불규칙 텍스처에서의 오차가 상대적으로 높아 다른 방법보다 낮은 PSNR을 보인다. 이는 가변 블록을 이용한 부호화 기법을 사용하지 않기 때문이다. Jacquin의 경우 에지 영역뿐만 아니라 평활화 영역에서도 어느 정도의 오차를 나타내나 에지 등에서의 오차가 심하지 않아 전체적으로 좋은 성능을 나타내며 Lepsoy 등의 방법이 Jacquin의 경우보다 좀 더 좋은 성능을 나타낸다. Jacquin의 방법이 Lepsoy 등의 방



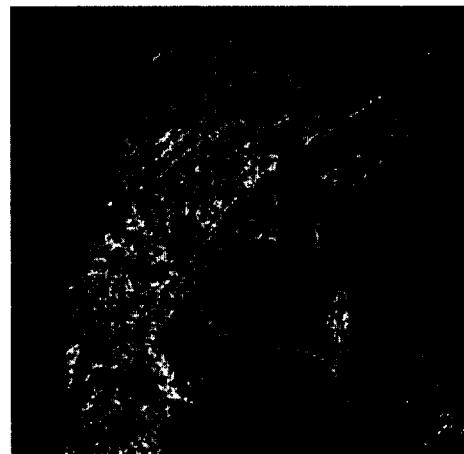
(a) Monro와 Dudbridge (8×8)



(b) Lepsy 등 (12)



(c) Jacquin (12)



(d) 제안한 방법

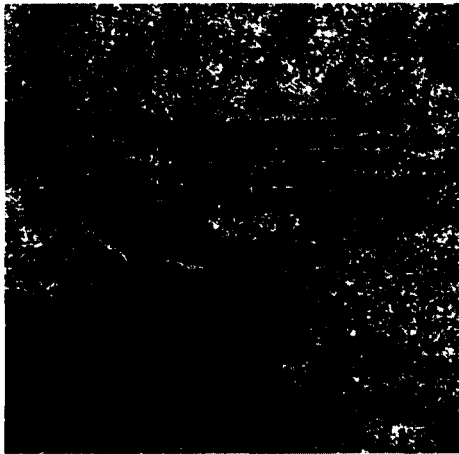
그림 8. 복원 오차영상 (Lenna 영상)
Fig. 8. Reconstruction error image (Lenna)



(a) Monro와 Dudbridge (4×4)



(b) Lepspy 등 (12)



(c) Jacquin (12)



(d) 제안한 방법

그림 8. 복원 오차영상 (Bridge 영상)

Fig. 8. Reconstruction error image (Bridge)

법보다 나은 형태의 domain 정보를 사용하나 탐색을 위한 블럭의 갯수가 상대적으로 작아 성능이 떨어지기 때문이다. 제안한 방법은 최소 블럭의 에지에 대한 결과가 좋게 나타나는 반면, 큰 블럭과 작은 블럭들간의 끊어짐 등이 나타나는 것을 볼 수 있다. 따라서 블럭들간의 평활화⁽¹⁸⁾를 후처리로 사용할 경우 보다 나은 성능을 보일 것으로 기대된다.

VI. 결 론

제안한 방법에서는 근사화된 원영상 정보를 직접적으로 사용함으로써 원영상의 밝기값 특성을 동적인 형태의 부호책으로 사용한다. 따라서 제안한 방법에서는 근사화를 위한 정적 부호책과 프랙탈 근사화를 위한 동적 부호책을 함께 사용함으로써 보다 효과적으로 영상을 부호화하였다. 즉, 적절한 블럭의 선택이 용이하며 변환에 대한 축소 매핑 조건이 없다. 또한 기저 함수의 직교성을 이용하는 직교 다항 변환을 사용함으로써 기존의 다항식을 이용한 근사화 변환을 다항 직교 변환으로 확장하였다. 그리고 영상 표면의 특성을 나타내는 프랙탈 차원을 새로운 척도로 사용하여 블럭을 분할함으로써 프랙탈 차원이 영상 특성을 효과적으로 나타낼 수 있음을 보였으며, 영상의 프랙탈 차원을 구하기 위하여 반복 변환 시스템을 이용하는 방법을 제안하였다. 앞으로 동적 부호책 개념을 이용한 동영상 부호화와 domain 블럭을 선택하기 위한 빠른 탐색 알고리즘의 개발 등에 대한 연구가 진행되어야 할 것이다. 또한 제안한 변환 시스템을 대역 분할과 다해상도 기법⁽²³⁾ 등을 적용함으로써 다른 부호화의 특성을 이용하는 연구도 이루어져야 할 것이다.

참고문헌

1. M. F. Barnsley, *Fractals Everywhere*, San Diego: Academic Press, 1988.
2. H.-O. Peitgen and D. Saupe, *The Science of Fractal Images*. New York: Springer Verlag, 1988.
3. H.-O. Peitgen, H. Jürgens, and D. Saupe, *Chaos and Fractals*. New York: Springer Verlag, 1992.

4. A. E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transforms," *IEEE Trans. Image Process.*, vol. IP-1, pp.18-30, Jan. 1992.
5. A. E. Jacquin, "Fractal image coding: A review," *Proc. IEEE*, vol. 81, pp.1451-1465, Oct. 1993.
6. D. M. Monro and F. Dudbrige, "Fractal approximation of image block," in *Proc. Int. Conf. Acoust., Speech, Signal Processing '92*, vol. 3, pp.485-488, San Francisco, California, Mar. 1992.
7. D. M. Monro, "A hybrid fractal transform: A review," in *Proc. Int. Conf. Acoust., Speech, signal processing '93*, vol. 5, pp.169-172, Minneapolis, Minnesota, Apr. 1993.
8. S. Lepsøy, G. E. øien, and A. Ramstad, "Attractor image compression with a fast non-iterative decoding algorithm," in *Proc. Int. Conf. Acoust., Speech, Signal Processing '93*, vol. 5, pp.337-340, Minneapolis, Minnesota, Apr. 1993.
9. M. G. Alkhansari and T. S. Huang, "A fractal-based image block-coding algorithm," in *Proc. Int. Conf. Acoust., Speech, Signal Processing '93*, vol. 5, pp.345-348, Minneapolis, Minnesota, Apr. 1993.
10. G. Vines and M. H. Hayes, III, "Adaptive IFS image coding with proximity maps," in *Proc. Int. Conf. Acoust., Speech, Signal Processing '93*, vol. 5, pp.349-352, Minneapolis, Minnesota, Apr. 1993.
11. H. Li, M. Novak, and R. Forchheimer, "Fractal-based image sequence compression scheme," *Optical Engineering*, vol. 32, pp.1588-1595, July 1993.
12. B. Hrütgen and P. Büttgen, "Fractal approach to low rate video coding," in *Proc. SPIE Visual Communications and Image Processing '93*, pp.120-131, Cambridge, Massachusetts, Nov. 1993.

13. E. Reusens, "Sequence coding based on the fractal theory of iterated transforms systems," in *Proc. SPIE Visual Communications and Image Processing '93*, pp.132-140, Cambridge, Massachusetts, Nov. 1993.
14. G. Vines and M. H. Hayes, III, "Nonlinear address maps in a one-dimensional fractal model," *IEEE Trans. Signal Process.*, vol. SP-41, pp. 1721-1724, Apr. 1992.
15. I. K. Kim, R.-H. Park, "Image coding based on fractal approximation and vector quantization," in *Proc. 1994 IEEE Int. Conf. Image Processing*, vol. III, pp.132-136, Austin, Texas, Nov. 1994.
16. K. E. Atkinson, *An Introduction to Numerical Analysis*. Singapore: John Wiley & Sons, 1988.
17. A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Dordrecht: Kluwer Academic Publishers, 1992.
18. J. Vaisey, "Image compression with variable block size segmentation," *IEEE Trans. Signal Process.*, vol. SP-40, pp.2040-2060, Aug. 1992.
19. A. P. Pentland, "Fractal-based description of natural scenes," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp.661-674, Nov. 1984.
20. S. Peleg, J. Naor, R. Hartley, and D. Avnir, "Multiple resolution texture analysis and classification," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp.518-523, July. 1984.
21. M. Rabbani and P. W. Jones, *Digital Image Compression Techniques*. Washington: SPIE, 1991.
22. Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Comm.*, vol. COM-28, pp.84-95, Jan. 1980.
23. G. W. Worneil, *Synthesis, Analysis, and Processing of Fractal Signals*, RLE Tech. Rep. No. 566, MIT, 1991.



金仁權 (In Kwon Kim) 정회원
 1970년 2월 16일생
 1992년 2월 : 서강대학교 물리학과 졸업(이학사)
 1994년 2월 : 서강대학교 대학원 전자공학과 졸업(공학석사)
 1994년 3월~현재 : 서강대학교 대학원 전자공학과 박사과정



朴來洪 (Rae-Hong Park) 정회원
 현재 : 서강대학교 전자공학과 교수
 19권 5호 참조

※ 주관심분야 : 영상 부호화, 프랙탈 이론 등