

Gram-Schmidt 직교화를 이용한 LIFS 영상 부호화법에 관한 연구 - 부호화 고속 알고리즘 -

유 현 배*

A Study on LIFS Image Coding via Gram-Schmidt Orthogonalization - Fast Coding Algorithm -

Hyun-Bae Yoo*

요 약 최근, 새로운 영상 부호화로서 프랙탈 부호화가 높은 압축률의 가능성으로 폭넓게 연구되고 있다. 프랙탈 부호화는 초기에 흑백 영상과 선영상의 압축 방식으로서 검토되었다. 더욱이 Jacquin은 프랙탈 변환(LIFS)을 제안하여 프랙탈 부호화를 농담 영상으로 확장하였다. 현재, Gram-Schmidt 직교화 방식을 이용한 YTKT의 LIFS방식 등은 정지 영상 압축의 국제 표준 방식인 JPEG와 경합이 가능할 정도의 압축률이 얻어졌다. 본 연구에서는 YTKT 방식의 직교화 연산 알고리즘의 연산량을 크게 삭감할 수 있는 방법을 제안한다.

Abstract Recently, Fractal Image Coding has been studied as a way of efficient data compressing scheme. In the beginning, Fractal Image Coding has been studied for the data compressing in black & white images and linear images. A. E. Jacquin suggested LIFS which expands to Fractal Image Coding for a gray scale image. Currently, YTKT's LIFS scheme which is using Gram-Schmidt is so efficient that enough to compete with the JPEG which is the national standards. This paper investigates the way of greatly reduced calculation for the orthogonalization algorithm.

Key Words : LIFS image coding, contraction transformation, condensation transformation, average separation

1. 서 론

현재, 영상 정보를 이용한 양방향 통신이 인터넷 사용자 사이에 자주 이용되고 있으나 과도한 정보량에 기인하여 양방향 통신의 실시간성이 간과되고 있다. 이러한 문제점을 해결하기 위해 영상 정보의 압축은 JPEG 혹은 MPEG에 기반을 두고 행해지고 있다. 그러나 점차 증가하고 있는 네트워크 사용자가 실시간성을 확보하기 위해서는 더 한층 높은 압축을 실현할 수 있는 기술 개발이 요구되고 있다. 본 연구에서 다루는 프랙탈 영상 부호화법은 높은 압축을 실현하는 차세대 기술로서 주목을 받고 있다[1].

반복 함수계(IFS; Iterated Function System)에 기반을 둔 프랙탈 영상 부호화는 높은 압축률에서 JPEG와 비교해서 시간적으로 뛰어난 영상을 복원할 수 있는 압축방식으로서 많은 관심이 기울여지고 있다. Barnsley에 의해 제안된 IFS는[2, 3], 그 후 Barnsley와 Jacquin에 의해 농담 영상(gray-scale image)에 적용되었다[4, 5].

초기에는 중간조(halftone) 영상을 프랙탈로 표현하기 위한 수법으로서 확률적 방법이 고안되었다[6, 7]. 또한 웨이브렛(Wavelet) 등 다른 방식과 조합하거나 동영상에 적용하는 등 그 응용 범위가 더욱 넓어지고 있다[8, 9].

프랙탈 영상 부호화[10]는 영상 중의 자기 상사성을 이용해서 압축을 실현하고 있다. 프랙탈 부호화는 먼저 영상을 서로 겹치지 않는 블록(레인지 블록)으로 분할하여 그 각각에 대해서 블록 내의 휘도치를 가장 좋게 근사할 수 있는 다른 블록(도메인 블록)을 영상 중에 탐색한다. 도메인 블록은 레인지 블록보다도 크고 레인지 블록과 동일한 크기로 축소해서부터 아핀 변환을 실행한다. 즉, 레인지 블록은 그것보다도 크고 동일한 영상 중에 있는 영역으로부터의 축소 아핀 변환으로 표현된다. 복호화에서는 적당한 초기 영상에 대해서 관련된 도메인 블록에 아핀 변환을 실행해서 각 레인지 블록으로 대체한다. 이 조작을 유한 번 반복함으로써 원영상이 복원된다. 이와 같이 임의의 영상으로부터 출발하여 복원이 가능한 것은 콜라주 정리[2]에 의한 것이지만 압축이 달성되는 것은 적은 양의 IFS 파라미터에 의해 실제 영상 이 기술될 수 있다는 경험에 토대를 두고 있다.

*나사렛대학교 정보과학부
Tel: 041-570-7957

본 연구에서는 Gram-Schmidt 직교화를 이용한 LIFS 부호화법[11]의 부호화 처리의 문제점을 지적하고 그 개선법을 제안한다. 그 결과 부호화 효율과는 관계없이 부호화기(encoder) 및 복호기(decoder)의 산술 연산 횟수가 크게 삭감되었다.

2. 프랙탈 부호화

2.1 LIFS 부호화

LIFS 부호화법은 일반적으로 다음과 같은 순서에 의하여 이루어진다.

(1) 영상을 서로 겹치지 않는 M 개의 블록 $\{R_i\}_{i=1, 2, \dots, M}$ 으로 분할한다. R_i 를 레인지(range)로 부른다.

(2) 영상으로부터 다른 위치에 존재하는 블록 D_i 에 적절한 변환을 실행하여 R_i 를 근사하는

$$R_i \approx f_i D_i \quad (1)$$

을 고려하여 최적의 근사치를 부여하는 D_i 와 f_i 을 탐색한다. D_i 는 도메인(domain), f_i 를 국소 변환(local transformation)이라 부른다.

(3) f_i 을 기술하는 매개변수(parameter)를 R_i 에 대응하는 부호로서 저장한다.

국소 변환 f_i 는 일반적으로 영상 평면 혹은 휘도(luminance) 방향의 아핀 변환(affine transformation)의 합성으로 주어지며(Figure 1) 축소성의 가정으로부터,

① 축소 변환(Contraction Transformation): 영상 평면에서 영역 보존 및 축소적이며, 휘도 방향에 관해서도 축소적인 변환

② 응축 변환(Condensation Transformation): 레인지를 정해진 휘도 분포로 치환한 변환의 두 종류로 분류된다.

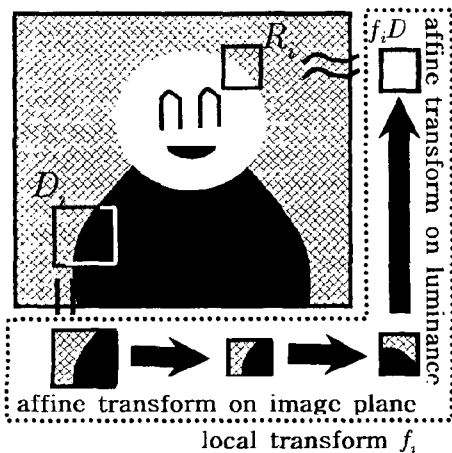


Figure 1. LIFS coding scheme

2.2 평균치 분리형 LIFS 부호화

Ida는 국소변환의 근사 정도를 개선하기 위해 직교변환 부호화법 등에서 검토되어 왔던 평균치 분리방식을 도입하여 LIFS 부호화법의 개선을 제안했다[12, 13]. 평균치 분리형(Average Separation) LIFS 부호화법에서는 레인지의 직류 성분(DC)에 해당하는 평균 휘도치로부터 교류 성분의 예측[14]에 의해서 휘도치가 일양 분포에 따른 보간 영상(가산 데이터)을 생성하여 원영상과 가산 데이터간의 차분을 축소 변환으로 근사한다. 가산 데이터 중에서 레인지 R_i 와 동일 위치 및 동일 형상의 블록을 \tilde{R}_i 로 둔다. 이 경우에 국소 변환은

$$R_i \approx g_i(D_i - \bar{D}_i) + \tilde{R}_i \quad (2)$$

으로 나타낼 수 있다. 단, g_i 는 축소 변환이며, 가산 데이터에서 \tilde{R}_i 가 R_i 의 적절한 근사가 가능할 경우에는 축소 변환은 이용하지 않고 R_i 을 \tilde{R}_i 로 치환하는 단순한 응축 변환

$$R_i \approx \tilde{R}_i \quad (3)$$

을 이용한다(Figure 2).

2.3 적응적 직교 변환

Ida의 평균치 분리형 LIFS 부호화에서 응축 변환은 단순히 축소 변환의 보조로서 이용되고 있다. 그러나 방식[15]는 축소성(contraction)의 제약이 없이 확대(enlargement)적인 아핀 변환(확장 응축 변환: extended condensation transformation)을 이용하여 응축 변환을 개선하였다. 그 결과 축소 변환보다 뛰어난 근사 정도가 얻어졌다. 더욱이 Gram-Schmidt 직교화법을 적용하여 여러 개의 확장 응축 변환을 결합함으로써 보다 높은 근사 정도를 얻을 수 있었다[11].

응축 변환의 도메인을 모두 포함하는 영상을 네스트

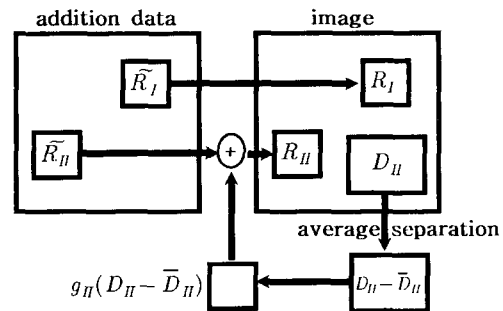


Figure 2. LIFS coding scheme with average separation. (R_i : a range block approximated by a condensation transformation, R_{ii} : a range block approximated by a condensation transformation)

(nest), 네스트 상의 도메인을 네스트 도메인 S_i 이라고 하자. 이 경우 확대 응축 변환은 네스트 전체로부터 탐색한 네스트 도메인에 아핀 변환 h_i 을 실행하여 레인지 R_i 을 근사하는 식

$$R_i = h_i(S_i - \bar{S}_i) + \bar{R}_i \quad (4)$$

으로 나타내어진다. \bar{S}_i 는 S_i 의 휘도치를 그 평균치로 치환한 블록을 나타낸다.

한편, 응축 변환의 저압축률에서의 근사 정도를 더욱 높이기 위해서 몇 개의 확장 응축 변환의 결합

$$R_i \approx \sum_{j=1}^{n_i} h_{ij}(S_{ij} - \bar{S}_{ij}) + \bar{R}_i \quad (5)$$

을 이용한다. $\{S_{ij}\}_{j=1, \dots, n_i}$ 는 R_i 에 대응하는 네스트 도메인의 집합이다. h_{ij} 는 응축 변환으로서 영상 평면 위의 아핀 변환 ϕ_{ij} 와 휘도 방향의 크기 계수 α_{ij} 로 구성되는 아핀 변환 $h_{ij} = \alpha_{ij}\phi_{ij}$ 이다. 식 (5)는 $\phi_{ij}(S_{ij} - \bar{S}_{ij})$ 을 비롯하여 모든 블록이 동일 화소 크기를 가지므로 그 화소수가 N 인 N 차원의 벡터로서 이루어진

$$r \approx \sum_{j=1}^n \alpha_j b_j + \bar{r} \quad (6)$$

으로 나타낼 수 있다. α_j, n_j 는 각각 α_j, n_j 로 간략하게 표기한 것이다. r, b_j, \bar{r} 는 각각 $R_i, \phi_{ij}(S_{ij} - \bar{S}_{ij}), \bar{R}_i$ 의 N 차원 벡터이다. 따라서 식 (6)의 국소 변환은 n 개의 네스트 도메인에 아핀 변환을 실행에 의하여 얻어진 벡터 $\{b_j\}_{j=1, \dots, n}$ 을 기저(basis)로 하는 선형 결합으로, $r - \bar{r}$ 을 근사하는 조작이라 할 수 있다.

식 (6)에서 최적의 변환 매개변수를 구하는 경우, 부호화 효율의 향상을 위하여 $n_i = 1$ 로부터 기저를 순차적으로 증가하여 최소한의 기저 수로 레인지에 대한 최적의 근사를 실현할 필요가 있다. 따라서 식 (6)으로부터 직교 기저계(basis system) $\{b_j\}_{j=1, \dots, n}$ 에 의한 근사를 고려함으로써 Gram-Schmidt 직교화법을 도입한 적응적 직교변환(adaptive orthogonal transformation)이 구성되어진다.

$$r \approx \sum_{j=1}^n \alpha_j b_j + \bar{r} \quad (7)$$

방식[11]의 결과, JPEG와 비교하여 압축률 전반에 걸쳐 경쟁이 가능할 정도의 부호화 효율이 얻어졌으며 저압축률에 있어서의 주관적인 영상 품질에서도 개선이 이루어 졌다. 그러나 영상의 품질 향상을 위해 적응적

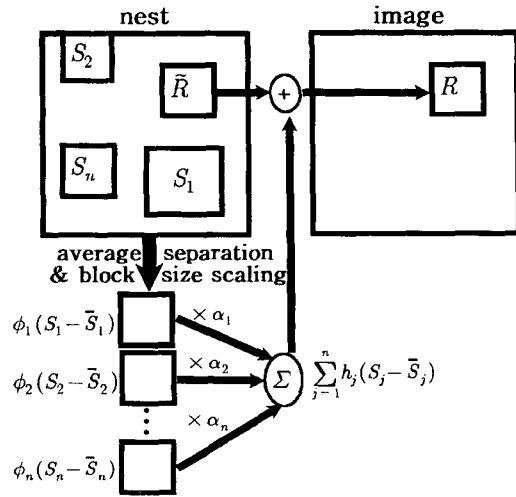


Figure 3. The combination of extended condensation transformations.

직교변환에 관한 네스트 도메인의 증가는 막대한 부호화 계산의 증가와 부호화 효율의 양면에 있어서 중복성(redundancy)으로 작용하고 있다.

3. 제안 방식 및 유효성

본 연구에서는 부호화 및 복호화에 있어서의 연산량을 크게 삭감할 수 있는 알고리즘을 검토한다.

방식 [11]에서는 네스트로부터 부 샘플링(subsampling)하여 벡터 $s \in \mathbb{R}^N$, 샘플링된 블록으로부터 평균치분리를 시행하여 기저 후보 벡터 $b \in \mathbb{R}^N$ 을 각각 구성한다. 방식 [11]의 부호기(encoder)는 기저 벡터에 관한 많은 연산이 정수 연산으로 수행됨으로서 평균치 분리된 후보 벡터는 항상 편차의 합이 영(zero)인 영평균(zero mean)이 되지 않는다. 따라서 방식 [11]에서는 기저수의 증가에 따라서 근사정도에 열화가 발생하고 저압축률에서의 영상 품질이 증가되지 않는 문제가 발생한다.

본 연구에서는 이러한 문제를 해결하기 위하여 네스트의 모든 성분의 하위 4비트를 사전에 제거하는 방법을 제안한다. 이것에 의하여 부 샘플링된 벡터 s 의 성분은 모두 N 의 배수가 되기 때문에 그 평균치는 항상 정수가 된다. 그러므로 평균치 분리된 후보 벡터 b 는 항상 영평균이 된다. 그 결과, 방식 [11]에서 일어나는 저압축률에서의 영상 품질의 열화의 문제가 다소 해결될 수 있으리라 기대된다.

또한, 방식 [11]의 부호기에서는 임의의 2개의 N 차원 벡터

$$a = (a_1, a_2, \dots, a_N), \mathbf{v} = (v_1, v_2, \dots, v_N)$$

의 내적 계산을 회의를 곱과 $N-1$ 회의 합

$$\langle \mathbf{a}, \mathbf{v} \rangle = a_1 v_1 + a_2 v_2 + \dots + a_N v_N$$

에 의해서 실행하고 있다. 따라서 가 영평균이 됨으로

$$\begin{aligned} \langle \mathbf{a}, \mathbf{v} \rangle &= a_1 v_1 + \dots + a_{N-1} v_{N-1} - a_N (v_1 + \dots + v_{N-1}) \\ &= (a_1 - a_N) a_1 + \dots + (a_{N-1} - a_N) v_{N-1} \end{aligned}$$

이 되고, $N-1$ 차원 벡터

$$\mathbf{c} = (a_1 - a_N, \dots, a_{N-1} - a_N)$$

을 사전에 준비하는 것으로서 내적 $\langle \mathbf{a}, \mathbf{v} \rangle$ 은 회의 $N-1$ 곱과 $N-2$ 회의 합

$$\langle \mathbf{a}, \mathbf{v} \rangle = \langle \mathbf{c}, \mathbf{v} \rangle = c_1 v_1 + \dots + c_{N-1} v_{N-1}$$

에 의하여 계산이 가능하게 된다. 즉, 부호기의 처리 시간을 단축시킬 수 있으며 아울러 동일한 연산 과정을 수행하는 복호기에서의 처리 시간도 단축이 가능하게 된다.

한편, 적응적 직교 변환에 있어서 부호화 대상 블록으로부터 얻어진 잔차 벡터

$$\mathbf{r} - \tilde{\mathbf{r}} = \mathbf{t} \in \mathbb{R}^N$$

에 대하여 평균치 분리된 후보 벡터의 집합 $\{\mathbf{b}_k \in \mathbb{R}^N\}_{k=1, \dots, C}$ 로부터 임의의 n 에 대하여 오차

$$\mathbf{t} - \sum_{j=1}^n \alpha_j \mathbf{b}_j^2 \quad (8)$$

을 최소화하는 적응적 직교 변환계 $\{\mathbf{b}_j\}_{j=1, \dots, n}$ 을 구할 경우,

```

PROCEDURE
// C: 후보 벡터의 총수
// j, k, max, P, x, Lk: 변수
// V, Uk: 벡터
BEGIN
max = 0
for k = 1 to k ≤ C do
    Uk = bk
    Lk = bk2 // . : 벡터의 크기
    P = ⟨t, Uk⟩
    x = P2 / Lk
    if max < x then
        max = x; I1 = k
    endif
    k = k + 1
repeat
max = 0
    
```

```

for j = 1 to j < n do
    V = UI1 / √LI1
    t = t - ⟨t, V⟩V //
    for k = 1 to k ≤ C do
        P = ⟨Uk, V⟩
        Uk = Uk - PV; Lk = Lk - P2
        P = ⟨t, Uk⟩
        x = P2 / Lk
        if max < x then
            max = x; Ij+1 = k
        endif
        k = k + 1
    repeat
    j = j + 1
repeat
END
    
```

에 의해서 구한다. 그러므로

① 제 1 기저의 계산에는 $(N+1) \times C$ 회의 승산과 $(N-1) \times C$ 회의 가감산

② 제 2 기저부터의 계산에는 $(3N+3) \times C$ 회의 승산과 $(3N-1) \times C$ 회의 가감산을 필요로 한다.

본 연구에서는 산술 연산을 보다 감소시키기 위하여 오차 (8)를 최소화하는 적응적 직교 변환계 $\{\mathbf{b}_j\}_{j=1, \dots, n}$ 을 구할 경우, \mathbf{t} 의 제 N 성분으로 각 성분의 값을 뺀 $N-1$ 차원 벡터를 \mathbf{t}' 으로 한다. 따라서 위의 연산 과정은 다음과 같이 표현된다.

```

PROCEDURE
BEGIN
max = 0
for k = 1 to k ≤ C do
    Lk = bk2
    Pk = ⟨t', bk⟩
    x = Pk2 / Lk
    if max < x then
        max = x; I1 = k
    endif
    k = k + 1
repeat
max = 0;
for j = 1 to j < n do
    
```

```

// Bj: bl을 {B1, ..., Bj-1}에 대하여 정규 직교화
한 벡터
// B': Bj의 각 성분에서 제 N성분의 값을 뺀
N-1차원 벡터
W =  $\frac{P_l}{\sqrt{L_l}}$ 
for k = 1 to k ≤ C do
    P = ⟨bb B'⟩
    Pk = Pk - WP; Lk = Lk - P2
    x =  $\frac{P_k^2}{L_k}$ 
    if max < x then
        max = x; Ij+1 = k
    endif
    k = k + 1
repeat
j = j + 1
repeat
END
    
```

즉, 산술 연산 횟수는

① 제 1 기저의 계산에는 $N \times C$ 회의 승산과 $(N-2) \times C$ 회의 가감산

② 제 2 기저부터의 계산에는 $(N+2) \times C$ 회의 승산과 $N \times C$ 의 가감산까지 감소한다.

실제, 방식 [11]에서는 화소[계조]인 표준영상 "Girl"을 이용하고 있다. 레인지의 크기는 $N=4 \times 4$ 로, 네스트 도메인의 크기는 16종류

$$4l \times 4m \text{ 화소, } l, m \in \{1, 2, 3, 4\}$$

로 구성하고 있다. 따라서 기저 후보 벡터의 총수 C는

$$C = \left(\frac{256}{4}\right)^2 \times 16 = 65,536$$

표 1. System Performance: the number of operation (The second or more basis, the number of operations are the same performance)

	proposed scheme		scheme [11]	
	multi.	add. & sub.	multi.	add. & sub.
first basis	1,048,576	917,504	1,114,112	983,040
	1,966,080		2,097,152	
second basis	1,179,648	1,048,576	3,342,336	3,080,192
	2,228,224		6,422,528	

이 된다. 표 1은 제안 방식과 방식 [11]과의 연산량을 비교한 것이다.

결론적으로 방식 [11]의 평균 기저 수 $n=5$ 의 경우, 제안 방식에 의하여 방식 [11]의 부호화 처리 시간에 결정적인 역할을 하는 적응적 응축변환의 산술 연산량이 1/3 정도까지 삭감되었다.

4. 결 론

LIFS 영상 부호화법[11]이 JPEG와 결합이 가능한 것은 Gram-Schmidt을 이용한 적응적 직교 변환의 부호화 효능이 컸기 때문이다[11]. 그러나 과도한 연산량이 문제가 되고 있다. 본 연구에서는 그러한 연산량의 문제점을 해결하기 위하여 영평균을 이용한 벡터의 내적 계산을 검토하였다. 그 결과 산술 연산량을 약 1/3정도까지 삭감이 가능하였다.

향후, 방법 [11]에 관하여

① Huffman 부호화의 적용: 모든 매개변수에 Huffman을 적용함으로써 부호화 효율은 낮고 코드북(code-book)의 양만큼 부호량이 증가한다.

② 네스트의 설정: 저주파수 성분으로 구성된 가산 데이터를 네스트로 이용하고, 고주파수 성분을 포함하는 기저 벡터를 생성하기 위해서 종류의 네스트 도메인을 설정하고 있다. 따라서 후보 기저 벡터의 수의 증가에 따른 부호기의 막대한 연산량을 초래한다.

③ 축소 변환의 적용: 축소 변환의 근사 정도는 적응적 직교 변환의 제 1 기저 정도로 낮다.

등의 검토를 통하여 한층 부호화 효율을 높일 수 있으리라 기대된다.

감사의 글

이 연구는 나사렛대학교 교내학술연구비 지원을 받음.

참고문헌

- [1] Fisher, Y. "Fractal Image Compression". SpringerVerlag, 1995.
- [2] Barnsley, M. F. "Fractals Everywhere", Academic press, San Diego, 1988.
- [3] Barnsley, M. F. "Application of Recurrent Iterated Function Systems to Image", SPIE, vol. 1001 Visual Communications and Image Processing, 1988.
- [4] Jacquin, A. E. "Image coding based on a fractal theory of iterated contractive image transformation", IEEE Trans. Image Processing, vol. 1, pp. 18-30, Jan. 1992.

- [5] Jacquin, A. E. "Fractal Image Coding: A review", Proc. IEEE, vol. 81, pp. 1451-1465, Oct. 1993.
- [6] Barthel, K. U. Voye, T. and Noll, P. "Improved Fractal Image Coding", in Proc. Picture Coding Symposium '93, 1.5, 1993.
- [7] Ohyama, H. Fujii, T. Kimoto, T. and Masayuki, T. "Variable Shaped Fractal Image Coding with Increased Variety of Block Shape", in Proc. Picture Coding Symposium '97, pp. 519-523, 1997.
- [8] Fisher, Y., Rogovin, D. N. and Shen, T. J. "Fractal (Self-VQ) Encoding of Video Sequence", in Proc. SPIE Visual Communication and Image Processing '94, SPIE vol. 2308, pp. 1359-1370, Sep, 1994.
- [9] Ida, T. and Sambnsugi, Y. "Image Segmentation and Contour Detection Using Fractal Coding", IEEE Trans. on Circuits and Systems for Video Technology, pp. 968-975, vol. 8, No. 8, December, 1998.
- [10] Barnsley, M. F. and Hurd, L. P. "Fractal Image Compression", AK Peters, 1992.
- [11] You, H., Takahashi, T., Koono, H. and Tokunaga, R. "Improving LIFS Image Coding Scheme-Combination of Extended Condensations via Gram-Schmidt Orthogonalization-", IEICE D-II, vol. J81-D-II, No. 12, pp. 2731-2737, Dec, 1998.
- [12] Ida, T. and Dachiku, K. "Image Compression using Fractal", Picture Coding Symposium in Japan (PCSJ91)6-11, pp. 149-152, 1991.
- [13] Ida, T. and Dachiku, K. "Image Compression using Iterated Transformation Theory-based Coding", The 5th Karuizawa Workshop on Circuits and Systems 21-22, pp. 137-142, April, 1992.
- [14] Takahashi, T. and Tokunaga, R. "A Fast Computation Algorithm for Predicting AC Components of Images Using Mean Values of Blocks", IEICE D-II, vol. J81-D-II, No. 4, pp. 778-780, April, 1998.
- [15] You, H., Takahashi, T., Hasegawa, Y. and Tokunaga, R. "Improving LIFS Image Coding Scheme via Extended Condensation Transforms", IEICE D-II, vol. J81-D-II, No. 7 pp. 1576-1583, July, 1998.