

# 적응 직교변환을 이용한 LIFS 부호화의 고속화에 관한 연구

유현배<sup>†</sup>, 박경남<sup>\*\*</sup>, 박지환<sup>\*\*\*</sup>

## 요 약

영상 데이터의 압축을 위하여 영상이 갖고 있는 자기 유사성(self similarity)을 이용하는 프랙탈 부호화가 활발하게 연구되고 있다. 본 연구에서는 부호화 효능에 있어서 JPEG과 경합이 가능한 적응 직교변환에 의한 LIFS 부호화에 주목하여 적응 직교변환의 연산 과정의 문제점을 지적하고 그 개선법을 제안하였다. 또한 직교변환의 모든 도메인(domain)을 포함하는 축소된 영상 데이터의 구성법을 제안하였다. 그 결과, 부호화 성능의 열화 없이도 부호기(encoder) 및 복호기(decoder)의 산술 연산량이 크게 삭감되었다.

## A Study on Fast LIFS Image Coding Using Adaptive Orthogonal Transformation

Hyun-bea You<sup>†</sup>, Kyung-nam Park<sup>\*\*</sup>, Ji-hwan Park<sup>\*\*\*</sup>

## ABSTRACT

For digital image compression, various fractal image coding schemes using the self-similarity of image have been studied extensively. This paper discusses the problem that occurs during the calculating process of adaptive orthogonal transformation and provides improvements of LIFS coding scheme using the transformation. This proposed scheme has a better performance than JPEG for a wide range of compression ratio. This research also proposes an image composition method consisting of all domains of the transformation. The results show that the arithmetic operation processes of the encoder and the decoder become much smaller even without the distortion of the coding performance.

**Key words:** LIFS image coding scheme(LIFS 영상 부호화법), condensation transformation(응축 변환), adaptive orthogonal transformation(적응 직교변환)

## 1. 서 론

영상압축은 영상을 저장 및 전송할 경우 중요한 기술이며 그 연구가 폭 넓게 이루어지고 있다. 농담 영상(gray scale image)을 대상으로 하는 압축 방법

으로는 변환 부호화, 벡터 양자화 등이 있지만, 비교적 최근에 제안된 프랙탈 부호화[1-8]는 그 중의 하나이다.

Mandelbrot[9]에 의해 제안된 프랙탈은 과학을 초월하여 공학 분야에서도 주목을 받고 있다. 그 초기에 컴퓨터 그래픽스를 적극적으로 이용하여 시각화된 프랙탈은 먼저 컴퓨터 그래픽스의 분야에서 형상 표현, 텍스처 표현을 위한 수리모델로서 이용되었다. 또한 컴퓨터 비전의 분야에서 MIT의 Pentland[10]는 그 특징량인 프랙탈 차원을 영상 분할에 도입하였다. 이와 같은 프랙탈 응용은 80년대 후반부터 Barnsley을 중심으로 하는 응용 수학자 그룹에 의해 적극적으로 전개되었다.

※ 교신저자(Corresponding Author): 유현배, 주소: 충남 천안시 쌍용동 456(330-718), 전화: 041)570-7957, FAX: 041)570-7957, E-mail: hbyoo@kornu.ac.kr

접수일: 2003년 5월 2일, 완료일: 2003년 8월 6일

<sup>†</sup>정회원 나사렛대학교 정보과학부 조교수

<sup>\*\*</sup>정회원 나사렛대학교 정보과학부 전임강사  
(E-mail: knpark@kornu.ac.kr)

<sup>\*\*\*</sup>중신회원, 부경대학교 컴퓨터멀티미디어공학전공 교수  
(E-mail: jpark@pknu.ac.kr)

※ 본 논문은 2004년도 나사렛대학교 50주년 기념 논문임.

프랙탈 영상 부호화는 영상이 갖고 있는 자기 유사성을 추출하여 이 성질을 기술하는 축소적인 차분 역학계(IFS: Iterated Function System)를 구성하는 방식이다. 현재 그 주류가 되고 있는 것이 블록 분할된 영상 사이의 유사성을 이용하는 Jacquin의 LIFS (Local IFS) 부호화이다[2-4]. 이것을 기반으로 하는 개선법이 다양하게 검토되고 있다[5-8,11-18,21-25].

본 연구는 부호화 효율에 있어서 JPEG(Joint Photographic Experts Group)과 경합이 가능한 Gram-Schmidt 직교화 방식에 의한 LIFS 부호화(YTKT 방식[21])에 주목하여 적응 직교변환의 산술 연산과정에 있어서의 문제점을 지적하고 그 개선법을 제안하였다. 또한 고주파 성분을 포함하는 축소된 직교변환의 네스트의 구성에 관하여 연구하였다. 그 결과 YTKT 방식[21]의 부호화 성능을 열화 시키지 않으면서 부호기(encoder) 및 복호기(decoder)의 산술 연산의 횟수가 크게 삭감되었다.

## 2. 프랙탈 부호화

### 2.1 LIFS 부호화

LIFS 부호화는 일반적으로 다음과 같은 순서에 의하여 이루어진다.

1. 영상을 서로 겹치지 않는  $M$ 개의 정방형 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  $R_i$ 을 근사하는

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

을 고려하여 최적의 근사치를 부여하는  $D_i$ 와  $f_i$ 을 탐색한다.  $D_i$ 를 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 변환은 일반적으로 그림 1과 같이 영상 평면 혹은 휘도(luminance) 방향의 아핀 변환(affine transformation)의 합성으로 주어지며, 축소성의 가정으로부터

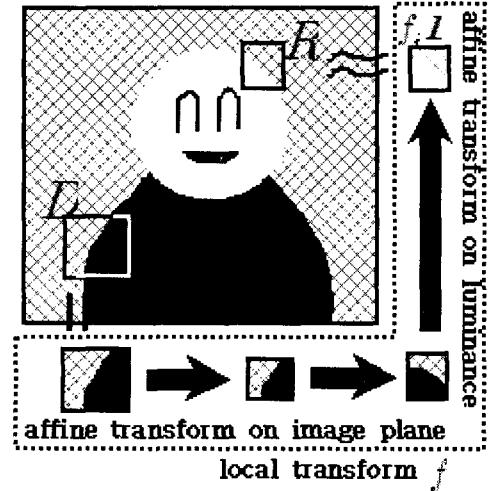


그림 1. LIFS 부호화의 개념

- 축소 변환(contraction transformation)  
영상 평면에서 영역을 축소하여 보존하고, 휘도 방향으로 축소적인 변환
- 응축 변환(condensation transformation)  
레인지를 정해진 휘도 분포로 치환한 변환의 두 종류로 분류된다.

### 2.2 평균치 분리형 LIFS 부호화

Ida은 국소 변환의 근사 정도를 개선하기 위해 직교변환 부호화 등에서 검토되어 왔던 평균치 분리방식을 기반으로 하는 LIFS 부호화를 개선시켰다[11-13]. 평균치 분리형(average separation) LIFS 부호화법에서는 레인지의 직류 성분(DC)에 해당하는 평균 휘도치로부터 교류 성분의 예측[19,20]에 의해서 휘도치의 일양 분포에 따른 보간 영상(가산 데이터)을 생성하여 영상과 가산 데이터 사이의 차분을 축소 변환으로 근사시킨다.

가산 데이터 중에서 레인지  $R_i$ 와 동일 위치 및 동일 형상의 블록을  $\tilde{R}_i$ 로 둔다. 이 경우에 국소 변환은

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

로 나타낼 수 있다.  $\overline{D}$ 는  $D_i$ 의 휘도치를 그 평균치로 치환한 블록을 나타낸다. 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

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

이용한다. 그림 2의  $R_I$ 와  $R_{II}$ 는 레인지  $R_i$ 중에서 식(3)과 식(2)에 의해 근사되는 레인지를 각각 나타낸다.

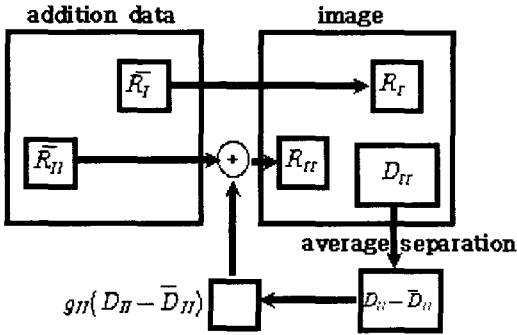


그림 2. 평균치 분리형 LIFS 부호화법 ( $R_I$  : 응축 변환에 의해 근사된 블록,  $R_{II}$  : 축소 변환에 의해 근사된 블록)

2.3 적응 직교변환에 의한 응축변환의 일반화

Ida의 평균치 분리형 LIFS 부호화에서 응축변환은 단순히 축소 변환의 보조로서 이용되고 있다. 그러나 개선 방식[21]은 축소성(contraction)의 제약이 없는 확대(enlargement) 아핀 변환인 확장 응축변환(extended condensation transformation)을 이용하여 개선하였다. 그 결과 축소변환보다 뛰어난 근사 정도가 얻어졌다. 더욱이 Gram-Schmidt 직교화를 적용하여 여러 개의 확장 응축변환을 결합함으로써 보다 높은 근사 정도를 얻을 수 있었다.

응축변환의 도메인을 모두 포함하는 영상을 네스트(nest), 네스트 상의 도메인을 네스트 도메인  $S_i$ 라고 하자. 이 경우 확장 응축변환은 네스트 전체로부터 탐색한 네스트 도메인에 아핀 변환  $h_i$ 을 실행하여 레인지  $R_i$ 을 근사화하는 아래의 식

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

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

한편, 낮은 압축률에서 보다 높은 근사 정도를 얻기 위하여 여러 개의 확장 응축변환을 결합한 변환

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

을 고려한다.  $S_{ij}, j=1, \dots, n_i$ 는  $R_i$ 에 대응하는 네스트 도메인의 집합이다.  $h_{ij}$ 는 응축변환이며, 영상 평면에서의 아핀 변환  $\phi_{ij}$ 와 휘도 방향의 크기 계수  $\alpha_{ij}$ 로 구성되는 아핀 변환  $h_{ij} = \alpha_{ij}\phi_{ij}$ 이다(그림 3). 식(5)는  $\phi_{ij}(S_{ij} - \bar{S}_{ij})$ 을 비롯하여 모든 블록이 동일 화소 크기를 가지므로 그 화소수가  $N$ 인  $N$ 차원 벡터로서 이루어진

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

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

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

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

YTKT 방식[21]의 결과, JPEG와 비교하여 압축률 전반에 걸쳐 경쟁이 가능할 정도의 부호화 효율이 얻어졌으며, 낮은 압축률이 있어서의 주관적인 영상 품질에서도 개선이 이루어졌다. 그러나 영상의 품질 향상을 위해 적응 직교변환에 관한 네스트 도메인의 증가는 막대한 부호화 계산의 증가와 부호화 효율의 양면에 있어서 중복성(redundancy)으로 작용하고 있다.

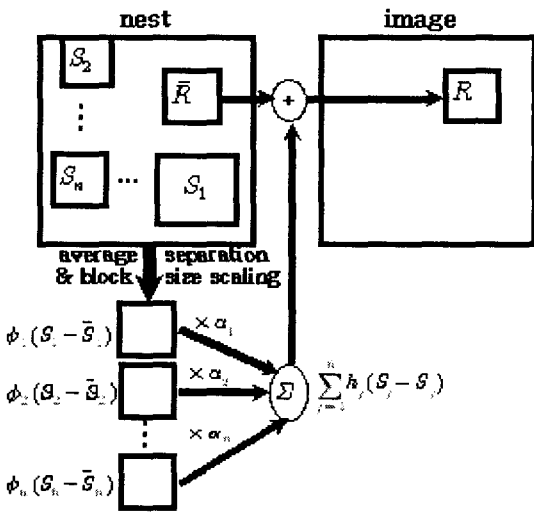


그림 3. 확장 직교변환의 결합

### 3. LIFS 부호화 고속 알고리즘 제안

본 연구에서는 고주파 성분으로 구성된 네스트의 설정 방법과 부호화 및 복호화에 있어서의 연산량을 크게 삭감할 수 있는 적용 직교변환의 고속화 알고리즘을 제안한다.

#### 3.1 네스트의 설정

YTKT 방식[21]에서는 저주파 성분으로 구성된 가산 데이터를 네스트로 이용하고, 고주파 성분을 포함하는 기저벡터를 생성하기 위해서 16종류의 네스트 도메인을 설정하고 있다. 따라서 후보 기저벡터의 개수의 증가에 따른 부호기의 막대한 연산량을 초래한다.

본 연구에서는 고주파 성분의 후보 벡터를 얻을 수 있도록 각 레인지의 평균 휘도치로 구성된 축소 네스트를 구성하였다. 그림 4(a)는 본 연구에서 사용하고 있는 표준 영상 Girl(256×256 화소)에 대하여 4×4화소의 레인지 일 경우의 네스트를 나타낸 것이다. 제안 네스트는 YTKT 방식[21]의 1/16의 크기의 영상을 이용함으로써 기저의 후보 벡터의 개수를 크게 줄일 수 있다.

#### 3.2 적용 직교변환의 고속화

YTKT 방식[21]에서는 네스트로부터 부 샘플링(subsampling)하여 벡터  $s \in \mathbb{R}^N$ , 샘플링 된 블록으



그림 4. (a) 영상 "Girl" (b) 네스트 영상

로부터 평균치를 분리하여 기저의 후보 벡터  $b \in \mathbb{R}^N$ 을 각각 구성한다.

YTKT 방식[21]의 부호기(encoder)는 기저벡터에 관한 많은 연산이 정수 연산으로 수행됨으로서 평균치 분리된 후보 벡터는 항상 편차의 합이 0인 영 평균(zero mean)이 되지 않는다. 따라서 YTKT 방식[21]에서는 기저수의 증가에 따라 근사 정도에 열화가 발생하고 낮은 압축률에서의 영상 품질이 증가되지 않는 문제가 발생한다.

본 연구에서는 이러한 문제를 해결하기 위하여 네스트의 모든 성분의 하위  $\log_2 M$ 비트를 사전에 제거하는 방법을 제안한다. 이것에 의하여 부 샘플링 된 벡터  $s$ 의 성분은 모두  $N$ 의 배수가 되기 때문에 그 평균치는 항상 정수가 된다. 그러므로 평균치 분리된 후보 벡터  $b$ 는 항상 영 평균이 된다. 그러나 하위 비트의 제거에 따른 고주파수 성분의 감소로 적용 직교변환의 근사 정도가 감소되는 문제점이 발생한다. 본 연구에서는 3.1절의 축소된 네스트의 구성으로 그 문제점을 최소화하였다.

또한, YTKT 방식[21]의 부호기에서는 임의의 2개의  $N$ 차원 벡터

$$a = (a_1, a_2, \dots, a_N), \quad v = (v_1, v_2, \dots, v_N)$$

의 내적 계산을  $N$ 회의 곱과  $N-1$ 회의 합

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

에 의해서 실행하고 있다. 따라서  $v$ 가 영 평균이 되므로

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

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  $N-1$ 회의 곱과  $N-2$ 회의 합

$$\langle \mathbf{a}, \mathbf{v} \rangle = \langle \mathbf{c}, \mathbf{v} \rangle = c_1 v_1 + c_2 v_2 + \dots + c_{N-1} v_{N-1}$$

에 의하여 계산이 줄어들게 된다. 따라서 부호기의 처리 시간을 단축시킬 수 있으며, 동일한 연산 과정을 수행하는 복호기에서의 처리시간도 단축이 가능하게 된다.

한편, 적응 직교변환에 있어서 부호화 대상 블록으로부터 얻어진 오차 벡터  $\mathbf{r} - \tilde{\mathbf{r}} = \mathbf{t} \in \overline{\mathbf{r}^N}$ 에 대하여 평균치 분리된 후보 벡터의 집합

$$\mathbf{b}_k \in \overline{\mathbf{r}^N}_{k=1, \dots, C} \text{로부터 임의의 } n \text{에 대하여 오차}$$

$$\left| \mathbf{t} - \sum_{j=1}^n \alpha_{I_j} \mathbf{b}_{I_j} \right|^2 \quad (8)$$

을 최소화하는 적응 직교변환계  $\mathbf{b}_{I_j, j=1, \dots, n}$ 는

**PROCEDURE**

//  $C$  후보 벡터의 총수  
 //  $j, k, \max, x, L_k, P$  변수  
 //  $V, U_k$  벡터

**BEGIN**

max = 0  
**for**  $k=1$  **to**  $k \leq C$  **do**  
      $U_k = \mathbf{b}_k$   
      $L_k = |\mathbf{b}_k|^2$  //  $|\cdot|$ : 벡터의 크기  
      $P = \langle \mathbf{t}, U_k \rangle$   
      $x = \frac{P^2}{L_k}$   
     **if** max  $\langle x \rangle$  **then**  
         max =  $x$ ;  $I_1 = k$   
     **endif**  
      $k = k + 1$

**repeat**

max = 0  
**for**  $j=1$  **to**  $j < n$  **do**  
      $V = \frac{U_{I_j}}{\sqrt{L_{I_j}}}$   
      $\mathbf{t} = \mathbf{t} - \langle \mathbf{t}, V \rangle V$   
     **for**  $k=1$  **to**  $k \leq C$  **do**  
          $P = \langle U_k, V \rangle$   
          $U_k = U_k - PV$   
          $L_k = L_k - P^2$   
          $P = \langle \mathbf{t}, U_k \rangle$   
          $x = \frac{P^2}{L_k}$   
         **if** max  $\langle x \rangle$  **then**  
             max =  $x$ ;  $I_{j+1} = k$   
         **endif**  
      $k = k + 1$   
     **repeat**  
          $j = j + 1$   
     **repeat**  
         **END**

의 연산 알고리즘에 의해 구해진다. 그러므로

- 제1기저의 계산에는  $(N+2) \times C$ 회의 승제산과  $(N-1) \times C$ 회의 가감산
- 제2기저부터의 계산에는  $(3N+4) \times C$ 회의 승제산과  $(3N-1) \times C$ 회의 가감산을 필요로 한다.

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

**PROCEDURE**

//  $P_k$  변수  
**BEGIN**  
 max = 0  
**for**  $k=1$  **to**  $k \leq C$  **do**  
      $L_k = |\mathbf{b}_k|^2$   
      $P_k = \langle \mathbf{t}', \mathbf{b}_k \rangle$

$$x = \frac{P_k^2}{L_k}$$

if max < x then

max = x; I<sub>1</sub> = k

endif

k = k + 1

repeat

max = 0

for j = 1 to j < n do

// V<sub>j</sub>; b<sub>I<sub>j</sub></sub>을 V<sub>1</sub>, ..., V<sub>j-1</sub>에 대하여 정규 직교화  
한 벡터

// V': V<sub>j</sub>의 각 성분에서 제 N성분의 값을 뺀  
N-1차원 벡터

$$W = \frac{P_{I_j}}{\sqrt{L_{I_j}}}$$

for k = 1 to k ≤ C do

P = < b<sub>k</sub>, V' >

P<sub>k</sub> = P<sub>k</sub> - 8

L<sub>k</sub> = L<sub>k</sub> - P<sup>2</sup>

$$x = \frac{P_k^2}{L_k}$$

if max < x then

max = x; I<sub>j+1</sub> = k

endif

k = k + 1

repeat

j = j + 1

repeat

END

즉, 산술 연산의 횟수는

· 제1기저의 계산에는 (N+1) × C회의 승제산과 (N-2) × C회의 가감산

· 제2기저부터의 계산에는 (N+3) × C회의 승제산과 N × C의 가감산까지 감소된다.

한편, YTKT 방식[21]에서 최대 기저수 n이 8이상인 레인지의 네스트 도메인의 위치와 네스트 도메인의 크기를 고정장 부호화법으로 부호화할 경우, 16byte이상의 부호량이 필요하게 된다. 256계조의 영상에 대한 4 × 4 화소의 레인지를 부호화할 경우는

레인지의 각각의 화소에 1byte 씩이 할당된 16byte의 화소치로 직접 부호화하는 것이 보다 효율적이다. 따라서 YTKT 방식[21]의 최대 기저수 15는 부호화 효율과 산술 계산량의 양면에 있어서 중복성으로 작용하고 있다. 본 논문에서는 최대 기저수를 7로 제한하여 이러한 문제점을 해결하였다.

### 3.3 부호화 알고리즘

부호화 알고리즘은 YTKT 방식[21]을 기반으로 구성한다. 영상을 B × B 화소로부터 구성되는 M개의 정방형 블록으로 분할하여 레인지 R<sub>i=1,2,...,M</sub>로 한다. 각 레인지의 평균 휘도(DC)를 원소로 하는 축소 네스트와 가감산의 교류 성분예측에 의한 가산 데이터를 구성한다. 축소 네스트는 모든 성분의 하위 log<sub>2</sub>M비트를 사전에 제거한다. 다음은 국소 변환의 근사정도를 평가하는 기준으로서 블록의 평균자승 오차 d(·, ·)을 고려하고 허용치 Z를 설정한다.

국소변환은 다음의 3종류

- 단순한 응축변환
- 축소변환
- 적응 직교변환

을 이용한다. i = 1, 2, ..., M에 관하여 다음의 순서에 따라 R<sub>i</sub>를 근사하는 국소 변환 f<sub>i</sub>를 구한다.

(i) 가산 데이터를 네스트로 하는 응축변환에 관한 근사를 실행한다. d(R<sub>i</sub>, A<sub>i</sub>) < Z일 경우, 이것을 f<sub>i</sub>로 설정하고 (iv)로 이동.

(ii) 축소변환에 의한 근사오차를 최소로 하는 (D<sub>i</sub>, g)를 탐색한다. d(R<sub>i</sub>, A<sub>i</sub>) < Z일 경우, 이것을 f<sub>i</sub>로 설정하고 (iv)로 이동.

(iii) DC 성분만으로 구성된 축소 네스트 상의 적응 응축변환에 의한 근사를 실행한다. d(R<sub>i</sub>, A<sub>i</sub>) < Z가 만족할 때 까지 네스트 도메인을 추가하여 구성된 변환을 f<sub>i</sub>로 설정한다.

(iv) R<sub>i</sub>의 DC 성분, 선택된 국소변환 f<sub>i</sub>의 종류와 f<sub>i</sub>을 기술하는 매개변수가 부호로서 저장된다.

### 3.4 국소변환 매개변수

3종류의 국소 변환의 매개변수는 다음과 같이 선택한다.

· 단순 응축변환 :  $R_i$ 의 DC 성분

· 축소변환 : 도메인 크기는  $2B \times 2B$ 로 고정하고 휘도 방향의 크기 계수는 18종류의 양자화 값  $\pm 0.9, \pm 0.8, \dots, \pm 0.1, 0$ 으로부터 선택한다. 매개변수는 휘도 크기계수와 도메인 위치.

· 적응 직교변환 : 네스트 도메인  $S_j$ 의 크기는  $\beta_x B \times \beta_y B$ 화소 ( $\beta_x, \beta_y \in 1, 2, 3$ )과

“ $\beta_x \neq 3 \vee \beta_y \neq 3$ ”을 동시에 만족하는 8종류이며, 이것을 부 샘플링 하여 레인지와 동일한 크기의 블록을 생성한다. 휘도 크기 계수는 선택된 기저에 대한 최적치를 직접 계산한다. 따라서  $j=1, 2, \dots, n_i$ 의 각각에 관하여 네스트 도메인의 위치, 부 샘플링 간격의 종류 그리고 크기 계수가 변환의 매개변수가 된다.

#### 4. 실험 결과 및 고찰

실험영상은  $256 \times 256$ 화소[256계조]의 표준영상 SIDBA “Girl”과 “Lena”를 이용한다. 제안방식과 YTKT 방식[21]의 부호화 성능의 평가기준은

$$\text{압축률} = \frac{\text{실험 영상의 화소수} \times 8 [\text{bit/화소}]}{\text{압축 후의 총 부호량}}$$

에 대한 왜곡률

$$\text{SNR} = 20 \log_{10} \frac{255}{\text{오차의 표준 편차}} [dB]$$

으로 한다. 부호화를 실행할 경우, 국소변환의 연산량을 삭감하기 위해 도메인과 네스트 도메인의 탐색 간격을 각각 4, 1화소로 설정하였다. 복호의 경우는 각 화소 값이 0인 초기 영상을 이용하고 축소 변환의 반복 횟수 3회로 한다. 이러한 조건으로부터 3.3의 부호화 순서에 의해 얻어진 각 레인지의 부호를 다음과 같이 재 압축한다.

· 레인지 DC 성분 : 2차원 DPCM 부호화와 허프만 부호화에 의하여 압축한다. 표 1은 압축 효율 향상을 위하여 허용오차  $Z$ 에 따른 DPCM의 양자화 계수  $\sigma$ 를 나타낸 것이다.

· 변환 종류 판별 부호 : zero Run-Length를 고려하여 허프만 부호화에 의해 압축한다.

· 네스트 도메인 위치, 부 샘플링 간격의 종류 : 고정장 부호화법에 의해 압축한다.

표 1. 허용 오차와 양자화 계수

	허용 오차 $Z$
$\sigma=16$	2000
$\sigma=8$	1000, 500, 200
$\sigma=4$	50, 100
$\sigma=2$	10, 25

· 도메인 위치, 축소 변환의 전개 계수와 적응 직교변환의 전개계수 : 개별적으로 허프만 부호화에 의해 압축한다.

표 2와 그림 5, 6은 영상 “Girl”과 “Lena”의 실험 결과를 나타내고 있다. 본 연구에서 제안한 축소된 네스트의 구성에 의해서도 모든 구간에서 기존의 YTKT 방식[21]과 동등한 부호화 성능을 나타내고 있다.

본 연구에서는 고속 부호화를 위하여 적응 직교변환의 산술 연산량의 감소와 더불어 후보 벡터의 개수를 감소시켰다. YTKT 방식[21]의  $\beta_x, \beta_y \in 1, 3, 5, 7$ 에 대하여 적응 직교변환의 후보 벡터의 총수는

$$\begin{aligned} & \sum_{\beta_y} \sum_{\beta_x} \max_{N_{\beta_x} \in \vec{n}} 4(N_{\beta_x} - 1) + 3\beta_x \leq 255 \\ & \times \max_{N_{\beta_y} \in \vec{n}} 4(N_{\beta_y} - 1) + 3\beta_y \leq 255 \\ & = 60, 516\text{개} \end{aligned}$$

가 되며, 제안 방식에서는

$$\begin{aligned} & \sum_{\beta_x} \sum_{\beta_y} \max_{N_{\beta_x} \in \vec{n}} N_{\beta_x} - 1 + 3\beta_x \leq 63 \\ & \times \max_{N_{\beta_y} \in \vec{n}} N_{\beta_y} - 1 + 3\beta_y \leq 63 \\ & = 27, 489\text{개} \end{aligned}$$

된다.  $\vec{n}$ 는 자연수이다. 즉, YTKT 방식[21]보다 벡터의 개수가  $\frac{1}{2.2}$ 까지 감소되었다.

부호화 과정에 있어서 각 레인지에 대한 적응 직교변환의 평균 산술 연산 횟수는

$$\frac{1}{M} \sum_{j=1}^M N_{B(j)} (N_{M(j)} + N_{A(j)})$$

이 된다.  $N_{M(j)}$ 과  $N_{A(j)}$ 는 최대 기저수가  $j$ 일 때, 승계의 산술 연산 횟수와 가감의 산술 연산 횟수를 각

표 2. 영상 "Girl"과 "Lena"의 성능 평가

	proposed scheme		YTKT scheme[21]	
	PSNR	com. ratio	PSNR	com. ratio
Girl	27.81	59.74	27.83	58.88
	31.64	22.26	31.76	21.11
	40.20	4.59	40.30	4.54
Lena256	25.4	47.7	25.4	47.5
	30.90	14.51	30.91	14.36
	40.18	3.85	40.20	3.86

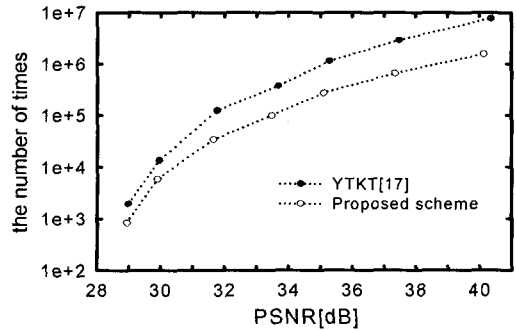


그림 7. 왜곡률에 대한 적응 직교변환의 평균 산술 연산 횟수 (영상 "Girl")

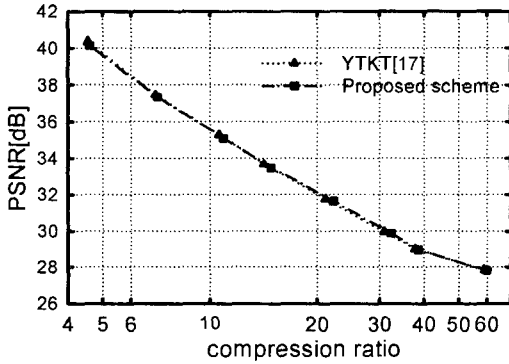


그림 5. 영상 "Girl"에 대한 부호화 성능 평가

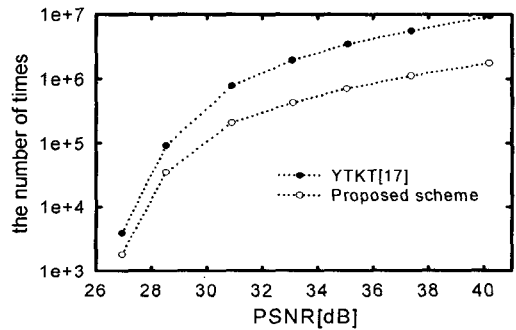


그림 8. 왜곡률에 대한 적응 직교변환의 평균 산술 연산 횟수 (영상 "Lena")

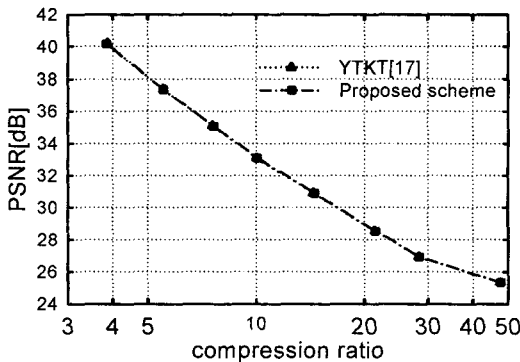


그림 6. 영상 "Lena"에 대한 부호화 성능 평가

각 나타낸다. 그리고  $N_{B(j)}$ 는 최대 기저수를  $j$ 로 갖는 레인지의 개수를 나타낸다. 그림 7, 8은 영상 "Girl"과 "Lena"에 있어서의 왜곡률에 대한 적응 직교변환의 평균 산술 연산 횟수를 각각 나타내고 있다. 직교변환의 산술 연산량이 영상 "Girl"과 "Lena"에 있어서 1/5.0, 1/5.3까지 각각 감소되었다.

제안방식 및 YTKT 방식[21]에서는 축소 변환, 적응 직교변환의 순서로 최적의 블록을 탐색하고 있다. 따라서 적응 직교변환은 영상의 품질 개선에 크게 기여하고 있다. 실제, 40dB 이상의 영상 품질을 얻기 위한 적응 직교변환의 비율은 "Girl"의 경우 41%, "Lena"의 경우는 45% 이상을 차지하고 있다. 그러므로 적응 직교변환의 산술 연산량과 후보 벡터의 개수를 감소시킴으로서 부호기 및 복호기의 처리 시간이 단축될 수 있다.

### 5. 결 론

LIFS 영상 부호화[21]가 JPEG과 결합이 가능한 것은 Gram-Schmidt를 이용한 적응 직교변환의 부호화 효능이 컸기 때문이다. 그러나 LIFS 부호화는 과도한 연산량이 문제가 되고 있다. 본 연구에서는 그러한 연산량의 문제점을 해결하기 위하여 영 평균을 이용한 벡터의 내적 계산과 축소된 네스트의 구성



법에 관하여 검토하였다. 그 결과, 부호화 성능은 떨어뜨리지 않으면서 산술 연산량 및 후보 벡터의 개수를 감소시킴으로서 부호화 및 복호화기의 고속화를 얻을 수 있게 되었다.

금후, 부호화 및 복호의 시간 측정을 통하여 본 연구에서 제안한 고속 알고리즘의 유효성을 입증한다. 또한, 국소 변환의 구성법과 대부분의 매개변수에 허프만 부호화법을 적용함으로써 발생하는 부호화 효율의 감소와 코드북(code-book) 만큼의 부호량 증가 등의 단점을 개선할 수 있는 재 압축에 관한 검토를 통하여 부호화 효율을 더욱 높일 수 있으리라 기대된다.

### 참 고 문 헌

- [1] M.F. Barnsley, *Fractals Everywhere*, Academic press, San Diego, 1988.
- [2] A.E. Jacquin, "A Novel Fractal Block-Coding Technique for Digital Image," Proc. ICASSP-90, pp. 2225-2228, 1990.
- [3] A.E. Jacquin, "Image Coding Based on a Fractal Theory of Iterated Contractive Image Transformation," IEEE Trans. Image Processing, Vol.1, pp. 18-30, Jan. 1992.
- [4] A.E. Jacquin, "Fractal Image Compression: A Review," Proc. IEEE, Vol.81, pp. 1451-1465, Oct. 1993.
- [5] L. Thomas and F. Deravi, "Pruning of The Transform Space in Block-Based Fractal Image Compression," Proc. ICASSP-93, Vol.5, pp. 341-344, 1993.
- [6] K.U. Barthel, T. Voyer and P. Noll, "Improved Fractal Image Coding," Proc. Picture Coding Symposium(PCS), Section 1.5, 1993.
- [7] Y. Fisher, D. Rogovin, and T.P. Shen, "Fractal (Self-VQ) Encoding of Video Sequence," Proc. SPIE Visual Communications and Image Processing. '94, SPIE Vol.2308, pp. 1359-1370, Sep. 1994.
- [8] Y. Fisher, *Fractal Image Compression*, SpringerVerlag. 1995.
- [9] B. Mandelbrot, *The Fractal Geometry of Nature*, San Francisco, W.H.Freeman and Co., 1982.
- [10] A. Pentland, "Fractal-Based Description of Natural Scenes," IEEE Trans. on PAMI, Vol.PAMI-6, No.6, pp. 661-674, 1984.
- [11] T. Ida and K. Dachiku, "Image Compression Using Fractal," Picture Coding Symposium in Japan(PCSJ91)6-11, pp. 149-152, 1991.
- [12] T. Ida and K. Dachiku, "Image Compression Using Iterated Transformation Theory-based Coding," The 5th Karuizawa Workshop on Circuits and Systems April 21-22, pp. 137-142, 1992.
- [13] H. You, T. Takahashi, Y. Hasegawa and R. Tokunaga, "Improving LIFS Image Coding Scheme via Extended Condensation Transforms," IEICE D-II, Vol.J81-D-II, No.7 pp. 1576-1583, July 1998.
- [14] G.E. Oien, S. Lepdoy, and T.A. Ramstead, "An Inner Product Space Approach to Image Coding by Contractive Transformations," Proc. ICASSP-91, pp. 2773-2776, 1991.
- [15] 박승준, 김상현, 김남철, "프레임간 프랙탈 근사화를 이용한 동영상 부호화", Telecommunications Review(한국이동통신), 제1권, 제4호, pp.42-61, 1995.
- [16] 김정일, 송광석, 강경인, 박경배, 이광배, 김현욱, "고속 프랙탈 영상 부호화 기법", 한국정보처리학회 논문지 A, 제3권, 제5호, pp.1317-1328, 1996.
- [17] 김상현, 김남철, "LMSE 해석 및 부블록 특징에 근거한 고속 프랙탈 부호화", 한국통신학회 논문지, 제22권, 제6호, pp.1279-1288, 1997.
- [18] 박용기, 박철우, 김두영, "블록 근사화식의 적응적 선택을 이용한 프랙탈 영상 부호화", 한국정보처리학회 논문지 A, 제4권, 제12호, pp.3185-3199, 1997.
- [19] T. Watanabe and K. Ohzeki, "A Study of AC Prediction Method Making Use of Mean Value," Picture Coding Symposium in Japan (PCSJ89)2-2, pp. 29-30, Oct. 1989.
- [20] T. Takahashi and R. Tokunaga "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.

- [21] H. You, T. Takahashi, H. Koono and R. Tokunaga, "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.
- [22] 김영봉, 이윤정, "레인지 블록 유사성을 이용한 프랙탈 압축", 한국정보과학회 논문지 A, 제27권, 제2호, pp.0117-0122, 2000.
- [23] 강현수, 김성대, 최대각, "기준영상들을 이용한 프랙탈 부호화 방식", 통신학회 논문지, 제25권, 제3호, pp.0519-0528, 2000.
- [24] 김종구, 함도용, 위영철, 김하진, "정규화된 분산을 이용한 프랙탈 압축방법", 한국정보처리학회 논문지 A, 제8-A권, 제4호, pp.0499-0502, 2001.
- [25] 곽성근, "프랙탈 기반 저전송율 영상 압축 부호화", 컴퓨터산업교육기술학회 논문지, 제3권, 제8호, pp.1085-1092, 2002.



**유 현 배**

1992년 부경대학교 응용수학과 졸업  
 1994년 부경대학교 대학원 응용수학과 졸업(이학석사)  
 2000년 일본 Tsukuba대학교 대학원 공학연구과 졸업(공학박사)

2000년 일본 통신종합연구소(CRL) 연구원  
 2001년 순천향대학교 BK21 전임교수  
 2002년~현재 나사렛대학교 정보과학부 조교수  
 관심분야: 영상처리 및 이해, 가상현실, 프랙탈 기초 및 응용, 웹3D 애니메이션

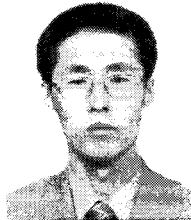


**박 경 남**

1992년 2월 경북대학교 전자공학과 졸업  
 1997년 2월 경북대학교 대학원 전자공학과 졸업(공학석사)  
 2003년 2월 경북대학교 대학원 전자공학과 졸업(공학박사)  
 2003년 3월~현재 나사렛대학교

정보과학부 전임강사

관심분야: 영상처리, 영상통신, 디지털 TV 등



**박 지 환**

1990년 3월 일본 요코하마국립대 전자정보공학 졸업(공학박사)  
 1994년 9월~1995년 3월 동경대 생산기술연구소 방문 연구  
 1998년 1월~1998년 2월 전기통신대학(일본), 방문연구

1999년 7월~1999년 8월 Monash University, Australia, Visiting Research  
 2001년 2월~2001년 3월 Communication Research Lab (CRL) Japan, STA Fellowship  
 1996년 4월~현재 동경대학 생산기술연구소 협력연구원  
 1990년 3월~현재 부경대 컴퓨터멀티미디어공학부 교수  
 1997년 3월~현재 한국정보보호학회 이사  
 2002년 3월~현재 한국정보보호학회 영남지부장  
 1998년 12월~현재 한국멀티미디어학회 운영위원 논문지 편집위원  
 1999년 3월~현재 한국정보처리학회 논문지 편집위원  
 2002년 3월~현재 한국정보보호학회 논문지 편집위원  
 2002년 1월~2월 CRL 방문연구 JSPS Fellowship  
 관심분야: 멀티미디어 콘텐츠 보호 및 응용, 암호학