

프랙탈 화상압축을 이용한 기밀 데이터 합성법

이혜주 박지환

부산수산대학교 전자계산학과

Embedding method a Secret Data into Fractal Image Compression

Hye Joo Lee Ji Hwan Park

Dept. of Computer Science Nat'l Fisheries Univ. of Pusan

요약

화상 내의 중복도를 의미하는 자기 상사성(self-similarity)을 제거하여 화상압축을 수행하는 프랙탈 화상압축 방식에 대한 연구가 최근에 활발히 진행되고 있다. 프랙탈 화상압축은 비가역 부호화 방식으로 화상을 일정블럭으로 분할하여 상사영역을 탐색하고 그 상사영역에 대한 파라메터를 압축을 위한 정보로 저장한다. 전송하고자 하는 데이터가 화상과 기밀 데이터의 2가지가 혼재할 경우 일반적으로 각 데이터를 개별적으로 압축하고 전송하게 된다. 또한, 화상의 특징을 이용하여 기밀 데이터를 화상 내에 몰래 집어 넣은 후, 그 화상을 압축하여 전송하는 방식이 있다. 본 논문에서는 이러한 방식과는 달리 기밀 데이터를 프랙탈 화상압축을 수행함과 동시에 예측부호화를 이용하여 제 3자가 인식할 수 없는 잡음의 형태로 합성하는 방법에 대하여 고찰하고 시뮬레이션 결과를 나타낸다.

1. 서론

정보화 사회로 발전함에 따라 팩시밀리, 화상회의 등 여러 분야에서 화상을 이용하는 기회가 증가되고 있다. 화상은 데이터량이 많고 중복도(redundancy)가 높으며, 최종적으로 인간의 시각에 의하여 인식되어지는 특징을 가지고 있다. 네트워크를 통해 화상을 전송할 경우 많은 데이터량으로 인해 통신 비용이 증가하게 되므로 통신 비용의 절감을 위하여 필요한 것이 화상압축이다. 일반적으로 화상은 최종적으로 인간의 시각에 의해 인식된다는 점에 주목하여 압축율의 향상을 위해 어느 정도의 왜곡을 허용하는 비가역 부호화(lossy compression)를 수행한다.

한편 정보의 보안을 위하여 제 3자가 알면 곤란한 데이터를 전송할 경우, 제 3자의 공격에 대비하여 암호화를 수행하여 전송하게 된다. 그러나 최근에 기밀 데이터를 화상의 압축단계에서 몰래 집어넣어 전송하는 방식이 주목되고 있다^[1]. 뿐만 아니라, 화상 자체가 가치를 지니고 있는 경우 그 화상의 도용을 방지하기 위해 화상 내에 저작자의 인증을 위한 기밀 데이터(인증자)를 추가하는 방식^[2]이 연구되고 있다.

본 논문에서는 화상압축을 수행함과 동시에 기밀 데이터를 합성하는 방식을 제안한다. 2장에서는 화상압축 중에서 현재 활발한 연구가 진행되고 있는 프랙탈 화상압축 방식에 대하여 알아본다. 3장에서는 프랙탈 화상압축을 이용한 기밀 데이터 합성 방식을 제안하며, 컴퓨터 시뮬레이션에 의해 평가한 결과를 4장에 나타낸다. 마지막으로 5장에서는 결론 및 향후 연구과제를 도출한다.

2. 프랙탈 화상압축

지금까지 다양한 화상압축 방식이 제안되어져 왔는데, 최근에는 자기 상사성을 이용하여 압축율을 향상시키는 새로운 방식의 프랙탈 화상압축에 대한 연구가 활발히 진행되고 있다.

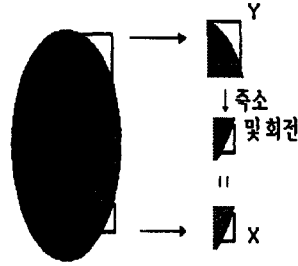


그림1. 부분적 자기상사성

자기 상사성이란 그림1과 같이 화상 내에서 어떤 화상패턴 X 에 주목할 때, 동일 화상 내에 X 와 유사한 화상패턴 Y 가 존재함을 의미한다. 디지털 화상에서는 완전한 자기 상사성이란 존재하지 않고 부분적인 자기 상사성을 가지게 된다. 이 부분적 자기 상사성은 중복도라는 개념으로 볼 수 있으며, 프랙탈 화상압축은 이 중복도를 제거하여 압축을 수행하게 된다. 프랙탈 화상압축은 Jacquin^[3,4]에 의해 농도 화상(gray scale image)에 적용된 이래 다양한 방식이 제안되고 있다. 여기서서는 화상 재생 시에 수렴속도가 빠른 평균치 분리방식과 가산 데이터^[5,6]를 이용한 프랙탈 화상압축을 대상으로 한다.

프랙탈 화상압축은 비가역 부호화 방식으로 자기 상사성을 갖는 상사영역을 탐색하여 압축을 수행한다. 압축을 위해 먼저 $M \times N$ 크기의 원화상을 $m \times n$ 화소의 블록 B_k ($1 \leq k \leq M/m, 1 \leq l \leq N/n$)로 분할하고, 블록마다 크기가 $g \times h$ 인 영역 D_k 의 화소값을 추출한다. 단, g 와 h 는 m 과 n 의 정수배이다. 여기에 간단한 선형변환 T_k 을 수행한 후, 변환 데이터와 블록 데이터의 2승오차가 최소가 되는 영역을 상사영역으로 결정한다. 이 때 압축 데이터는 상사영역의 위치, 선형 변환을 지시하는 파라메터 등이 된다.

재생시에는 압축 데이터로 임의의 초기화상에 변환 T 를 반복 수행한다. 변환 T 는 식(1)과 같이 각 블록에 대응하는 선형변환 T_k 의 집합으로 초기 화상에 반복적으로 수행하면 어떤 일정한 화상에 수렴하게 되고, 이 수렴화상은 원화상에 가깝게 된다.

$$T = \cup T_k \quad (1)$$

2.1 블록과 상사영역의 설정 및 선형변환

프랙탈 화상압축에서는 블록과 상사영역, 선형변환에 의해 압축율과 화질이 좌우된다.

(1) 블록과 영역의 설정

블록의 크기는 정방형으로 고정한다.

$$n = m = C_b = 4 \quad (2)$$

영역의 크기는 블록 크기의 가로, 세로 2배로 설정하거나 재생시 수렴속도를 빠르게 하기 위해 2배 이상으로 설정할 수도 있다. 여기서는 식(3)과 같이 영역의 크기를 설정한다.

$$g = h = 2C_b \quad (3)$$

상사영역의 위치를 나타내기 위해 식(4)와 같이 파라미터 α_M, β_M 을 설정한다.

$$\alpha_M = 4i \ (0 \leq i < M/m), \ \beta_M = 4j \ (0 \leq j < N/n) \quad (4)$$

(2) 선형변환

프레임 압축에서는 블록 B_M 의 상사영역을 결정하기 위하여 전체 영역에 변환을 수행한다. 즉, 영역의 모양, 크기, 화소값등을 변환시켜 B_M 과의 2승오차가 최소인 영역을 상사영역으로 결정하게 된다. 영역의 변환 방법에 따라서 적절한 상사영역의 탐색여부가 좌우된다. 즉 복잡한 변환을 이용하면 상사성이 높은 상사영역을 결정할 수 있는 반면 압축에 소요되는 시간은 증가된다. 따라서 압축 속도를 빠르게 하기 위해 여기서는 아래와 같은 3가지의 간단한 선형변환을 수행한다.

① 화상 평면내의 축소

영역 D_M 의 크기를 블록 B_M 의 크기로 축소한다. 영역의 화소패턴이 유지될 수 있도록 축소하여야 하므로 일반적으로 대응되는 화소의 평균값을 이용한다.

② 화소값의 변환

축소된 영역 $\overline{D_M}$ 의 각 화소값에서 $\overline{D_M}$ 의 평균값을 분리한 후, 각 화소값에 δ_M 을 곱하여 화소의 진폭치 방향을 변환한다. 변환 데이터와 블록 데이터의 2승오차 E_M 를 최소로 하는 δ_M 의 값을 구하고 그 양자화치를 이용한다.

$$\delta_M = \{-0.9, -0.8, \dots, -0.1, 0.1, 0.2, \dots, 0.9\} \quad (5)$$

③ 화소배치 변환

화소배치 변환을 위해 표1과 같이 ϵ_M 로 주어지는 회전과 반전의 조합을 수행한다. ϵ_M 은 8개의 값 중에서 2승 오차 E_M 을 최소로 하는 것을 선택하게 된다.

표 1. 화소배치변환

* 회전은 시계방향 *

ϵ_M	회전	반전
1	0°	하지않음
2	0°	함
3	90°	하지않음
4	90°	함
5	180°	하지않음
6	180°	함
7	270°	하지않음
8	270°	함

2.2 평균치 분리와 가산 데이터

프랙탈 화상압축은 상사영역의 데이터를 블록 데이터로 치환하여 화상을 재생하게 된다. 재생 화상을 빨리 수렴시키기 위한 방법으로 평균치 분리와 가산 데이터를 이용한 방식이 제안되어 있다.

(1) 평균치 분리

Jacquin이 제안한 방식에서는 블록을 평탄부(shade), 중간부(mid-range) 및 윤곽부(edge)로 분류한다. 이 때 중간부와 윤곽부에서는 블록과 영역의 평균값을 근사적으로 같게 하기 위해서 Δg_i 라는 파라미터를 이용한다. 이 Δg_i 는 블록 데이터의 평균값으로부터 화소값 변환된 영역 데이터의 평균값을 감소한 것으로 상사영역일 때의 Δg_i 를 저장하고, 화상 재생시에 상사영역의 데이터에 Δg_i 를 가산한 결과를 블록 데이터로 치환하게 된다.

그러나, 재생의 초기단계에서 임의 화상의 상사영역 데이터는 원화상과 많은 차이가 생기게 된다. 따라서, 상사영역의 데이터에 Δg_i 를 가산한 후, 그 결과를 블록 데이터로 치환하는 Jacquin의 방식은 수렴 속도가 느려지게 된다. 이 문제점을 해결하기 위하여 Δg_i 대신에 블록 데이터의 평균값 γ_{kl} 을 이용하여 수렴의 속도를 빠르게 하는 평균치 분리방식이 제안되어 있다^[6]. 이 방식은 블록의 각 화소값에서 평균값 γ_{kl} 을 분리(감산)한 목적 데이터 M_{kl} 과 영역과의 2승오차가 최소가 되는 상사영역을 탐색하게 된다. 이때 평균값 γ_{kl} 을 압축 데이터로 저장하게 되며, 재생시에는 상사영역의 데이터에 Δg_i 대신에 γ_{kl} 을 가산한다. 그 결과, 1회의 반복으로 블록의 평균값은 압축시의 블록 평균값 γ_{kl} 과 거의 일치하기 때문에 수렴속도가 빠르게 된다.

(2) 가산 데이터

블록의 평균값 γ_{kl} 을 가산할 경우 블록내에서 그 값을 인정하기 때문에 블록 형태의 왜곡이 발생하게 된다. 따라서 가산치를 평균값 대신에 평균값으로부터 교류성분을 예측하는 방식^[7]을 적용한다. 교류성분을 예측하기 위해 각 블록을 서브 블록으로 분할한다. 서브 블록이 포함된 블록과 주변 블록의 평균값에 가중치를 주어 평균하고 그 값을 서브 블록의 새로운 교류성분으로 한다. 이 과정을 서브 블록이 단위 화소가 될 때까지 재귀적으로 수행하면 그 블록의 교류성분을 구할 수 있게 된다. 이것을 가산 데이터 A_{kl} 이라 하고, 목적 데이터 M_{kl} 은 블록 데이터에서 γ_{kl} 대신에 A_{kl} 을 가산하여 상사영역을 탐색한다. 재생시에는 전송된 γ_{kl} 로 부터 다시 A_{kl} 을 구하여 γ_{kl} 대신에 가산하게 된다. 계산된 가산 데이터는 블록의 크기와 같은 $m \times n$ 화소로 구성되어지며, 각 화소값은 서로 다른 값을 가지게 되기 때문에 블록형태의 왜곡이 감소하게 된다.

목적 데이터의 분산에 의해 블록의 형태를 분류하게 되는데, M_{kl} 의 분산이 임계값 Z_{an} 보다 작은 경우에는 상사영역을 탐색하지 않으며, 반대로 큰 경우에 상사영역을 탐색하게 된다. 전자의 경우는 Jacquin 방식의 평탄부 블록에 해당하고, 후자는 중간부, 윤곽부 블록에 해당하는 처리를 의미한다.

3. 프랙탈 화상압축상에 기밀 데이터 집어 넣기

2장에서 서술한 프랙탈 화상압축에 있어서 예측 부호화 방식을 이용한 기밀 데이터의 합성방식을 제안한다. 프랙탈 화상압축의 압축 데이터 중에서 평균값 γ_M 에 예측 부호화를 적용한다. 즉 블록 B_M 의 평균값 γ_M 과 블록 평균 예측치 $\overline{\gamma_M}$ 의 예측오차 $e_M (= \gamma_M - \overline{\gamma_M})$ 을 구한다. 여기서 e_M 은 $-K+1 \sim K-1$ 로 확대되기 때문에 양수의 범위로 제한시키기 위하여 식(6)에 의해 조절한다^[8]. 이때, k 는 화소를 구성하는 비트수로서 $K(=2^k)$ 는 원화상의 레벨수를 나타낸다.

$$e_{M'} = \begin{cases} e_M, & e_M \geq 0 \\ e_M + K, & e_M < 0 \end{cases} \quad (6)$$

기밀 데이터를 합성하기 위한 할당표는 오차를 줄이기 위하여 $e_{M'}$ 의 전 범위에 대해 1 또는 0의 비트를 표3와 같이 임의로 할당한다. 이때 기밀 데이터와 일치하는 비트가 할당된 $e_{M'}$ 를 압축 데이터로 저장하게 된다.

표2 할당표

$e_{M'}$	0	1	2	3	4	5	...	252	253	254	255
b_i	0	0	1	0	1	0	...	1	0	1	0

이와 같은 방식은 1 블록당 1비트의 기밀 데이터가 합성되나, 제3자의 공격을 고려한 임의의 조건을 만족하는 블록에 대해서만 합성하게 되면 안전성의 강도를 높일 수 있게 된다.

(1) 압축 알고리즘

데이터의 길이 L_i (비트 단위)와 복호시 가산 데이터를 계산하기 위해 첫번째 블록의 평균값 γ_M 이 추가 되어진다.

step 1 : 초기화

- L_i 비트의 기밀 데이터 I_i 를 입력
- 화상을 버퍼에 저장, 블록 $B_M(4 \times 4)$ 분할
- 가산 데이터 A_M 를 계산

step 2 : 기밀 데이터의 부가처리

- 예측오차 $e_M = (\gamma_M - \overline{\gamma_M})$ 에 대한 $e_{M'}$ 계산
- $b = b_i$ 이면 $e_{M'}$ 저장
- $b \neq b_i$ 이면 $b = b_j (i \neq j)$ 가 되는 $\overline{e_{M'}}$ 저장

step 3 : 블럭의 적용 치환

- 목적 데이터 M_M 의 계산 : $M_M = B_M - A_M$
- M_M 의 분산 $\leq Z_{av}$ 이면 $D_b = 0$, step 5 수행
- M_M 의 분산 $> Z_{av}$ 이면 $D_b = 1$

step 4 : 상사영역의 탐색

- 선형변환 T_M 을 수행
- $Min(E_M = (T_M(D_M) - B_M)^2)$ 인 영역 D_M 선택
- 압축 데이터 $\alpha_M, \beta_M, \epsilon_M, \delta_M, D_b, e_M'$ 저장

step 5 : 전체 블럭에 대해 step2, step4 수행

(2) 복호 알고리즘

첫번째 블럭의 평균값 γ_M 과 기밀 데이터의 길이 L_s , 그리고 각 블럭에 주어지는 $(e_M', \alpha_M, \beta_M, \epsilon_M, \delta_M, D_b)$ 의 압축 데이터를 이용하여 화상을 재생한다. 임의의 초기 화상에 대하여 반복 횟수 M 만큼 아래의 과정을 반복한다.

step 1 : 초기화

- 초기 화상 준비
- 압축 데이터(파라미터) 입력

step 2 : 기밀 데이터 추출

- 압축 데이터 e_M' 의 b_i 을 기밀 데이터 I 에 부가
- 블럭 평균값을 계산

$$temp = \overline{\gamma_M'} + e_M'$$

if ($temp \geq K$) {

$$e_M = e_M' - K, r_M' = \overline{r_M'} + e_M$$

else $r_M' = \overline{r_M'} + e_M'$

- 가산 데이터 A_M' 계산

step 3 : 블럭의 판단

- $D_b = 0$ 이면 $B_M \leftarrow A_M'$, step 5 수행
- $D_b = 1$ 이면 step 4 수행

step 4 : 상사영역과 블럭 데이터의 치환

- 압축 데이터 α_M, β_M 이 지시하는 영역 D_M 을 추출, 선형변환 T_M 을 수행

- $B_{kl} \leftarrow (A_{kl}' + T_{kl}(D_{kl}))$
- 전체 블록에 대해 step 2, step 4 수행

4. 시뮬레이션 및 고찰

그림2와 같은 128×128 화소, 8[bit/pixel]인 원화상을 이용하여 3장에서 제안한 기밀 데이터 부가 방식의 성능을 평가하기 위하여 시뮬레이션 하였다. 표3은 원화상을 임계값 $Z_w = \infty$ 로 하여 기밀 데이터를 집어 넣지 않은 경우와 전 블록에 1비트씩 총 1024비트의 기밀 데이터를 집어 넣어 압축을 수행한 재생 화상의 SNR을 비교한 것이다.

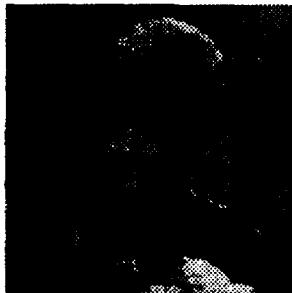


그림2. 원화상

표 3. SNR의 비교
 $SNR = 20 \log(255/RMS)$

	기밀 데이터를 넣지 않은 경우	기밀 데이터를 집어 넣은 경우
SNR[dB]	24.8	23.8

그 결과, 약 1[dB] 정도의 SNR이 떨어짐을 알 수 있다. 이것은 블록의 평균값 γ_{kl} 을 예측 부호화하여 예측 오차 e_{kl} 에 대한 할당표의 비트 b_i 와 기밀 데이터의 비트 b 가 일치하지 않을 경우, 일치한 비트를 가진 새로운 $\overline{e_{kl}}$ 을 찾아 그 값을 저장하기 때문이다. 즉 기밀 데이터를 집어 넣은 경우, 블록의 평균값이 원래의 γ_{kl} 이 아닌 오차가 생긴 γ_{kl}' 이 되기 때문에 평균값으로부터 계산되는 가산 데이터 역시 원래의 A_{kl} 과는 차이가 발생한다. 그 차이만큼 SNR이 기밀 데이터를 집어 넣지 않은 경우보다 떨어지게 된다.

그림3은 $Z_w = 25$ 로 하여 화상을 압축한 후, 반복 횟수를 증가시켜 화상을 재생한 SNR의 변화를 나타낸 것이다. 평균치 분리와 가산 데이터를 이용하였기 때문에 2회의 반복으로 화상은 일정한 화상으로 수렴됨을 알 수 있으며, 기밀 데이터를 집어 넣지 않았을 경우와 1024비트의 기밀 데이터를 집어 넣은 경우의 SNR은 약 0.8 dB 정도의 차가 생기게 된다.

표4는 임계값 Z_w 의 변화에 따른 압축율을 나타내었다. 탐색을 수행하게 되면 step 4의 각 파라메터를 위해 5바이트와 탐색여부를 위한 1비트가 블록당 필요하며, 탐색을 수행하지 않게 되면 블록당 e_{kl}' 을 위한 1바이트와 탐색여부 1비트가 저장되어 진다. 따라서 임계값에 따라 압축율이 변하게 된다. 여기에 기밀 데이터를 집어 넣은 경우 첫번째 블록의 평균값을 위한 1바이트와 기밀 데이터의 길이를 명시하기 위한 2바이트가 추가되어질 뿐이므로 압축율은 거의 변화되지 않음을 알 수 있다. 한편, 저장되는 각 파라메터에 대해 엔트로피 부호화를 수행하면 압축효과를 더욱 향상시킬 수 있게 된다.

각각의 경우에 화상을 압축하고 임의의 초기화상으로부터 재생한 수렴화상을 그림4에 나타내었다. 그 결과, 시각적으로는 2가지 경우의 수렴화상의 화질 차이를 거의 식별할 수 없으나, 원화상과 수렴화상의 화질은 많은 차이가 있음을 알 수 있다. 이것은 상사영역을 탐색하기 위해 영역의 크기를 블록 크기의 2배로 고정시켰기 때문이다. 따라서 영역의 크기를 2배 이상으로 설정하여 상사영역 후보의 수를 증가시키면 화질 개선이 될 수 있으나, 영역의 탐색 시간이 늘어나 압축 시간이 증가하게 되므로 보다 효율적인 탐색 알고리즘이 필요하게 된다.

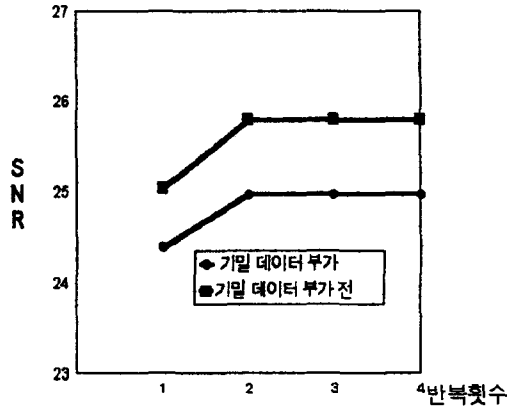


그림3. SNR의변화

표 4. 압축율(%)

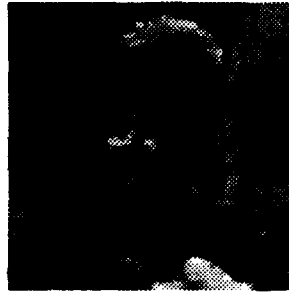
$$(\text{압축율} = (1 - \frac{\text{압축 화일 크기}}{\text{원 화일의 크기}}) \times 100)$$

기밀 데이터를 넣지 않은 경우	Z_{∞}	압축율	N_{∞}	D_b	α_{kl}	β_{kl}	γ_{kl}	ϵ_{kl}	δ_{kl}		
	$Z_{\infty} = \infty$	92.96	1024	1[b]	-	-	1	-	-		
	$Z_{\infty} = 25$	77.0	270	1[b]	1	1	1	1	1		
기밀 데이터를 넣은 경우	Z_{∞}	압축율	N_{∞}	D_b	α_{kl}	β_{kl}	e_{kl}'	ϵ_{kl}	δ_{kl}	r_{kl}	L_s
	$Z_{\infty} = \infty$	92.95	1024	1[b]	-	-	1	-	-	1	2
	$Z_{\infty} = 25$	76.9	270	1[b]	1	1	1	1	1	1	2

N_{∞} : 상사영역 탐색하지 않는 블록의 수, 단위 : [b]:bit, 그외 byte



초기화상



(a) 기밀 데이터 집어 넣지 않은 경우



(b) 기밀 데이터 집어 넣은 경우

그림4. 수렴 화상

5. 결론

평균치 분리 방식과 가산 데이터를 이용하는 프래탈 화상압축에서는 압축 데이터로써 $(\alpha_{kl}, \beta_{kl}, \gamma_{kl}, \delta_{kl}, \epsilon_{kl})$ 이 저장되어 진다. 여기에 기밀 데이터를 집어 넣기 위해 블록의 평균값인 γ_{kl} 을 예측 부호화하여 미리 작성되어져 있는 할당표에서 기밀 데이터 1비트와 일치하는 비트값을 가진 e_{kl}' 를 찾아 γ_{kl} 대신에 저장하고, 화상 재생시에 이 값을 이용하여 기밀 데이터를 추출하는 방식을 제안하였다.

제안 방식에 의해 재생된 화상의 화질은 약간 떨어지지만 압축율을 저하시키지 않으면서 기밀 데이터를 집어넣을 수 있음을 알 수 있다. 재생시 화상의 화질은 기밀 데이터와 압축 화상에 의해 좌우가 된다. 즉, 기밀 데이터와 할당표를 비교할 때 1회의 비교로써 일치되는 비트를 찾게 된다면 압축시와 재생시의 오차는 발생되지 않는다. 이와 반대로 기밀 데이터와 할당표의 1비트가 일치되지 않아 새로운 값을 찾아야 하는 경우에 오차는 피할 수 없게 된다. 또한, 일치되지 않을 경우 원래의 평균값과 1만큼의 차이가 발생되어 적절한 할당이라고 할 수 있으나, 제 3자의 공격에는 약하게 된다. 따라서 오차를 작게 하는 방법과 제 3자가 기밀 데이터를 알아 내고자 공격하는 경우에 대한 연구가 계속되어야 할 것이다.

[참고 문헌]

1. 松井 甲子雄, “畫像深層暗號”, p.185, 森北出版(株) (1993)
2. T.Hashiguchi et al, “Authentication Methods for Document Image by Using Concealed Image”, Crypto and Information Security Symposium (CIS'88)(in Japanese)
3. A.E.Jacquin, “ A Novel Fractal Block-coding Technique for Digital Images”, ICASSP 90 Proceedings, M8.2, pp.2225-2228, 1990
4. A.E.Jacquin, “Image Coding Based on a Fractal Theory of Iterated Contractive Image Transformations”, IEEE Trans. on Image Processing, Vol.1, No.1, pp.18-30, 1992
5. T.Ida and K.Dachiku, “Image Coding using Fractal”, Picture Coding Symposium in Japan(PCSJ'91), pp.149-152(in Japanese)
6. T.Ida and K.Dachiku, “Image Compression Using Iterated Transformation Theory-based Coding”, Proc. The 5th Karuizawa Workshop on Circuits and Systems”, pp.137-142, 1992(in Japanese)
7. T.Watanabe, K.Ohzeki, “A Study of AC Prediction Method Making use of Mean Value”, Picture Coding Symposium in Japan(PCSJ89), pp.29-30, 1989(in Japanese)
8. 박지환, 김진홍, 김두영, “단계적 전송기능을 갖는 영상 데이터의 가역 부호화”, 한국통신학회 논문지 Vol.19, No.1, pp.111-118(1994.1)