

무반복 프랙탈 복호화 알고리즘 기반의 영상 복호화기의 설계 및 구현

정회원 김 재 철*

Design and Implementation of Image Decoder Based on Non-iterative Fractal Decoding Algorithm.

Jae-Chul Kim* Regular Members

요 약

본 논문에서는, 무반복 복호화 알고리즘을 제안하고 이를 이용한 범용 DSP칩을 사용한 프랙탈 영상 복호화기를 설계하고 구현하였다. 무반복 복호화 알고리즘은 치역블록에 대한 정의역 블록이 고정된 Monro and Dudbridge의 알고리즘으로 부호화된 영상에 대하여 끌개영상은 해석식으로 얻을 수 있음을 보여준다. 그리고 얻어진 해석식을 이용하여 끌개영상은 반복없이 얻을 수 있었다. 또한 일반화된 식을 유도하였다. 해석식의 성능평가를 위하여, 여러 테스트 영상에 대한 해석식을 이용하는 방법과 이전의 반복적인 방법의 시뮬레이션 결과를 비교하였다. 그 결과 해석식을 이용하는 방법은 복호화 시간을 5배 이상 향상시킬 수 있었다. 제작된 프랙탈 영상 복호화기는 2개의 ADSP2181을 가지고 3단계 파이프 라인 구조로 복호화 과정이 수행된다. QCIF 형식에 대한 정지영상에 대하여, 구현된 복호화기의 성능은 프레임당 최대 31.2ms 소요된다. 그 결과, 초당 30 frames/sec 이상의 실시간 처리 가능함을 보였다.

ABSTRACT

In this paper, algorithm for non-iterative decoding method is proposed and fractal image decoder based on non-iterative fractal decoding algorithm used general purpose digital signal processors is designed and implemented. The algorithm is showed that the attractor image can be obtained analytically when the image is encoded using the fractal algorithm proposed by Monro and Dudbridge, in which the corresponding domain block for a range block is fixed. Using the analytical formulas, we can obtain the attractor image without iteration procedure. And we get general formulas of obtained analytical formulas. Computer simulation results for various test images show that we can increase the image decoding speed by more than five times when we use the analytical formulas compared to the previous iteration methods. The fractal image decoder contains two ADSP2181's and perform image decoding by three stage pipeline structure. The performance tests of the implemented decoder is elapsed 31.2ms/frame decoding speed for QCIF data when all the frames are decoded. The results enable us to process the real-time fractal decoding over 30 frames/sec.

I. 서 론

프랙탈 영상 부호화 기법은 매우 복잡하고 불규

칙한 자연 현상을 묘사하고 예측하기 위하여 수학의 기하학에서 출발한 프랙탈 이론^[1-3]을 기본으로 한다. 프랙탈이론은 Mandelbrot^[1]에 의해 처음으로

* 구미1대학 정보통신네트워크전공(blackcow@kumi.ac.kr)

논문번호 : 020455-1017, 접수일자 : 2002년 10월 17일

정립되었으며 Barnsley^[3-5]에 의해 영상 압축에 대한 적용 가능성이 제안되었고 이후 Jacquin^[6-8]에 의하여 구체화되었다. 이 기법은 영상 내에 존재하는 자기 유사성(self-similarity)을 이용하여 변환하고 임의의 초기 영상에 변환식을 반복적으로 적용함으로써 원 영상을 복원하는 부호화 기법이다.

Jacquin의 알고리즘에서는 넓은 정의역 영역의 탐색으로 인해 계산량이 많고 부호화 시간이 오래 걸리는 단점이 있다. 또한 복호화 과정은 임의의 영상의 전영역에 대하여 정의역 블록을 가져와 축소 변환을 반복하여 이루어짐으로 영상복원 시간이 비교적 오래 걸리는 단점이 있다. Jacquin의 알고리즘을 개선하기 위하여 Monro와 Dudbridge^[10-12]의 알고리즘에서는 영상을 일정한 크기의 중첩되지 않는 정의역 블록으로 나누고 이를 같은 크기의 치역 블록 4개로 다시 나누어 해당되는 치역블록들의 정의역으로 고정시켜서 최적의 축소 변환 계수를 구하였다. 따라서 Monro와 Dudbridge의 알고리즘은 Jacquin의 알고리즘과 같이 탐색영역이 넓은 경우와 비교하여 볼 때 부호화 시간이 짧은 장점을 갖고 있으나 화질이 떨어지고 복호화 과정은 축소 변환의 반복으로 인해 시간이 오래 걸리는 단점이 있다. 또한 Fisher^[13], Oien^[14-16], 그리고 Viens^[17,18] 등에 의해 프랙탈 영상 부호화 기법은 화질과 비트율에서 점진적으로 발전을 이루어왔다.

이러한 개선 기법들은 대부분 프랙탈 영상 압축 기법에 관한 연구였고 프랙탈 영상 복호화 기법에 관한 연구와 하드웨어에 관한 연구는 미미한 상태이다. 그러나 고속 프랙탈 복원 기법에 관한 연구는 동영상에 대한 프랙탈 영상 압축 기법의 적용에 있어 소프트웨어 측면 뿐 만 아니라 하드웨어 측면에서도 절대적으로 요구된다.

위에서 언급된 대부분의 프랙탈 부호화 기법에 대한 복호화 과정은 임의의 영상에 축소 변환을 반복 적용함으로써 시간이 오래 걸리는 단점과 종료 시점이 명확하지 않음에 문제점이 있다. 이러한 반복에 의한 복호화 과정의 문제점을 극복하기 위하여 Oien과 Lepsoy^[15]는 기저 블록(basis block)와 내적 공간(inner product space)의 직교성(orthogonality)을 이용하여 반복 적용이 필요 없는 복호화 알고리즘을 제안하였다. 그러나 이 방법들은 치역블록이 8×8 이하로 제한되는 단점을 갖는다.

본 논문에서는 Oien과 Lepsoy의 알고리즘과 달리 끌개영상(attractor image)의 해석식을 유도하여 적용한 무반복(non-iterative) 프랙탈 복호화 알

고리즘^[19]과 해석식의 일반화 식을 제안한다. 그리고 제안된 알고리즘을 이용한 무반복 프랙탈 복호화 알고리즘 기반의 영상 복호화기를 설계 제작하였다.

제안한 무반복 프랙탈 복호화 알고리즘은 Monro와 Dudbridge 알고리즘^[10]에 의해서 부호화된 프랙탈 영상의 복호화 방법에 대한 것으로 하나의 치역 블록에 대한 정의역 블록이 고정되어 부호화했을 때 얻어지는 끌개영상은 축소변환 연산자의 부호화 계수로부터 해석적으로 유도됨으로 프랙탈 영상의 복원 영상이 반복적용이 필요 없음을 보였다. 따라서 해석적으로 얻어지는 끌개영상은 반복 적용되는 수렴 영상과 정확하게 일치한다. 이 알고리즘의 검증은 위하여 여러 테스트 영상들에 대하여 시뮬레이션을 수행한 결과, 초기영상과 관계없이 축소 변환의 반복 적용이 필요 없기 때문에 복호화 시간이 일정하고 아주 짧은 장점과 하드웨어 및 소프트웨어 측면에서도 반복 적용되는 영상을 위한 임시 메모리가 필요 없으므로 메모리 관리 면에서도 아주 뛰어난 장점을 갖는다.

제안한 무반복 프랙탈 복호화 알고리즘의 영상 복원 성능을 확인하기 위하여 프랙탈 영상 복호화기의 하드웨어를 설계 제작하였다. 제작된 프랙탈 영상 복호화기의 성능은 QCIF의 Claire 영상에 대하여 Monro와 Dudbridge 알고리즘에 의해서 부호화된 프랙탈 정지영상에 대한 복호화 시간이 최대 31.2ms소요되었다. 그 결과 제작된 프랙탈 영상 복호화기는 초당 30 프레임의 복호화 속도를 가짐으로서 동영상 처리가 가능함이 확인되었다.

본 논문의 II장에서는 프랙탈 이론과 무반복 복호화 알고리즘에 대하여 서술하고 III장에서는 제작된 프랙탈 영상 복호화기의 하드웨어 구조에 대해 설명한다. 그리고 IV장에서 무반복 프랙탈 복호화 알고리즘 및 제작된 영상 복호화기의 성능을 분석하고 V장에서 결론을 맺는다.

II. 프랙탈 이론 과 무반복 복호화 알고리즘

1. 반복함수계 이론

완전 척도 공간(complete metric space) (M,d)에서 정의되는 변환 $W:M \rightarrow MX$ 에 대해

$$d(W(X), W(Y)) \leq s \cdot d(X, Y) \quad \forall X, Y \in M \quad (2-1)$$

을 만족하는 상수 $0 \leq s < 1$ 가 존재하면, 변환 W 를 축소 변환[2]이라 한다. 여기서 $d(X, Y)$ 는 두 영상 X, Y 사이의 거리를 나타내는 척도이고, s 는 변환 W 에 대한 축소비(contractivity factor)를 의미한다. 이러한 유한기의 축소 변환으로 이루어진 계를 반복함수계(Iteration Function System)라 한다.

축소변환 W 는

$$W(X) = LX + T \quad (2-2)$$

로 나타내고 축소선형변환영상 LX 와 영상신호 X 에 무관한 평행변환 영상 벡터 T 로 나눌 수 있다.

프랙탈 변환 W 를 초기영상 X_0 에 대하여 무한히 반복하여 얻어지는 끌개영상(attractor image) X_T 는

$$X_T = (1 + L + L^2 + \dots + L^\infty)T + L^\infty X_0 \quad (2-3)$$

와 같다.

식 (2-3)에서 L 의 놈(norm) $\|L\|$ 이 1 보다 작으면 우측항 $L^\infty X_0$ 은 0이 되고 끌개영상 X_T 는

$$X_T = (I - L)^{-1}T \quad (2-4)$$

와 같이 표시된다.

식 (2-4)에서 I 는 단위행렬(identity) 연산자를 의미한다. 따라서 복호화시 임의의 초기영상을 이용하여 프랙탈 변환을 무한히 반복하면 초기영상과 무관한 끌개 영상을 얻게된다.

2. 타일링(Tiling) 구조와 프랙탈 변환

그림 2-1은 타일링 구조^[12]와 프랙탈 변환이 수행되는 과정을 표시한다. 이 구조는 원 영상을 $2B \times 2B$ 의 크기를 갖는 정의역 블록 D 로 나누고, 이를 가로 세로 각각 2등분하여 그림 2-1을

$$D = \sum_{n=0}^{2B-1} \sum_{m=0}^{2B-1} d(n, m) \quad (2-5)$$

$$= \bigcup_{k=1}^4 R^{(k)} = \bigcup_{k=1}^4 \sum_{p=0}^{B-1} \sum_{q=0}^{B-1} r^{(k)}(p, q)$$

과 같이 $R^{(1)}, R^{(2)}, R^{(3)}, R^{(4)}$ 의 네 개의 치역블록으로 나눈다. 정의역 블록 D 에서 각 치역 $R^{(k)}$ ($k=1,2,3,4$)로의 프랙탈 축소 변환 $W^{(k)}(D)$ 는

$$R^{(k)} = W^{(k)}(D) = s_k D_c + T^{(k)} \quad (2-6)$$

와 같이 표시된다. 여기서 식 (2-5)에서 $R^{(k)}$ 는 정의역 블록을 이루는 치역블록, 식 (2-6)에서 $R^{(k)}$ 는

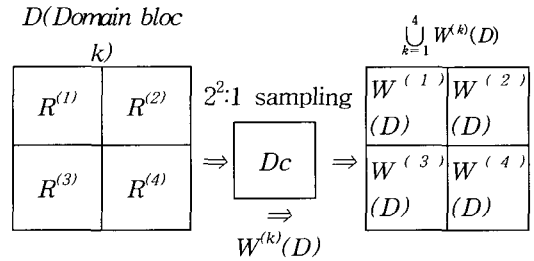


그림 2-1. 타일링 구조와 프랙탈 변환
Fig 2-1. Tiling structure and fractal transform.

복원된 치역 블록 그리고 D_c 는 $2^2:1$ 공간축소변환된 정의역 블록, s_k 는 축소변환 상수를 표시하고 $T^{(k)}$ 는 평행 변환 연산자이다.

하나의 정의역 블록에서 영상의 부호화는 원 영상에 대하여 식 (2-6)에서 $R^{(k)}$ 에 최적으로 근사시키는 s_k 값과 $T^{(k)}$ 의 파라미터 값을 구함으로 프랙탈 부호화가 이루어진다. 프랙탈 복원화는 콜라주 이론에 의한

$$W(D) = \bigcup_{k=1}^4 W^{(k)}(D) = \bigcup_{k=1}^4 R^{(k)} \quad (2-7)$$

와 같이 표현된다.

식 (2-6)에서 D_c 는 $2^2:1$ 로 공간적 축소변환 영상으로서

$$D_c = \sum_{x=0}^{B-1} \sum_{y=0}^{B-1} d_c(x, y) = \sum_{x=0}^{B-1} \sum_{y=0}^{B-1} d(2x, 2y) \quad (2-8)$$

로 표시된 바와 같이 샘플링(sampling) 방법과

$$D_c = \sum_{x=0}^{B-1} \sum_{y=0}^{B-1} d_c(x, y) = \sum_{x=0}^{B-1} \sum_{y=0}^{B-1} \sum_{z=0}^1 \frac{d(2x+z, 2y+z)}{4} \quad (2-9)$$

와 같이 인근 화소 값의 평균을 구하는 방법이 널리 사용된다.

3. 무반복 프랙탈 복호화 알고리즘

프랙탈 이론을 이용한 영상 압축의 복원은 임의 영상에 대하여 프랙탈 축소 변환 파라미터를 반복 적용함으로써 끌개영상을 얻음으로 이루어진다. 본 장에서는 이러한 프랙탈 변환의 반복 적용되는 복원 영상을 위하여 치역에 대하여 정의역 탐색이 없는 타일링 구조에서 끌개 영상의 해석식과 해석식의 일반화 식 유도를 통하여 무반복의 고속 프랙탈 복호화 알고리즘을 제안한다.

1) 4×4 정의역 블록 끝개 영상의 해석식

Monro와 Dudbridge 알고리즘으로 부호화된 정의역 내의 복호화를 위한 임의의 초기 영상 벡터 $X^{(0)}$ 의 각 요소를 $x_{pq}^{(0)}$ ($p,q=1,2,3,4$)라 정의하고, 축소 변환된 정의역 블록 영상 벡터 $D_c^{(0)}$ 의 각 요소는 $d_{pq}^{(0)}$ ($p,q=1,2$)라 정의한다. 그림 2-1에서 정의역에서 부호화된 k 번째 치역 블록의 축소 변환 상수가 s_k 이고 평행변환 연산자 T 의 (p,q) 요소를 $t_{pq}^{(k)}$ 이라 하면, 식 (2-8)을 이용하여 임의의 초기 축소 변환된 영상을 한번 프랙탈 변환하였을 때의 정의역 블록 영상 $X^{(1)}$ 는

$$\begin{aligned}
 [X^{(1)}] &= \sum_{p=1}^4 \sum_{q=1}^4 x_{p,q}^{(1)} \quad (2-10) \\
 &= S \otimes D_c^{(0)} + T_4 \\
 &= \begin{bmatrix} s_1 & s_2 \\ s_3 & s_4 \end{bmatrix} \otimes \begin{bmatrix} x_{11}^{(0)} & x_{13}^{(0)} \\ x_{31}^{(0)} & x_{33}^{(0)} \end{bmatrix} + \begin{bmatrix} t_{11}^{(1)} & t_{12}^{(1)} & t_{11}^{(2)} & t_{12}^{(2)} \\ t_{21}^{(1)} & t_{22}^{(1)} & t_{21}^{(2)} & t_{22}^{(2)} \\ t_{31}^{(1)} & t_{32}^{(1)} & t_{31}^{(2)} & t_{32}^{(2)} \\ t_{41}^{(1)} & t_{42}^{(1)} & t_{41}^{(2)} & t_{42}^{(2)} \end{bmatrix} \\
 &= \begin{bmatrix} s_1 x_{11}^{(0)} + t_{11}^{(1)} & s_1 x_{13}^{(0)} + t_{12}^{(1)} & s_2 x_{11}^{(0)} + t_{11}^{(2)} & s_2 x_{13}^{(0)} + t_{12}^{(2)} \\ s_1 x_{31}^{(0)} + t_{21}^{(1)} & s_1 x_{33}^{(0)} + t_{22}^{(1)} & s_2 x_{31}^{(0)} + t_{21}^{(2)} & s_2 x_{33}^{(0)} + t_{22}^{(2)} \\ \dots & \dots & \dots & \dots \\ s_3 x_{11}^{(0)} + t_{11}^{(3)} & s_3 x_{13}^{(0)} + t_{12}^{(3)} & s_4 x_{11}^{(0)} + t_{11}^{(4)} & s_4 x_{13}^{(0)} + t_{12}^{(4)} \\ s_3 x_{31}^{(0)} + t_{21}^{(3)} & s_3 x_{33}^{(0)} + t_{22}^{(3)} & s_4 x_{31}^{(0)} + t_{21}^{(4)} & s_4 x_{33}^{(0)} + t_{22}^{(4)} \end{bmatrix}
 \end{aligned}$$

와 같이 표시된다. 여기서, S 는 축소 변환 벡터, D_c 는 공간축소 정의역 영상 벡터, T_4 는 평행변환 벡터, 그리고 T_{C4} 는 공간축소 평행변환 벡터이다.

정의역 블록 $X^{(1)}$ 의 2:1 축소 변환된 정의역 블록 $D_{C4}^{(1)}$ 을

$$\begin{aligned}
 [D_c^{(1)}] &= \sum_{p=1}^2 \sum_{q=1}^2 d_{p,q}^{(1)} \\
 &= \begin{bmatrix} s_1 x_{11}^{(0)} + t_{11}^{(1)} & s_2 x_{11}^{(0)} + t_{11}^{(2)} \\ s_3 x_{11}^{(0)} + t_{11}^{(3)} & s_4 x_{11}^{(0)} + t_{11}^{(4)} \end{bmatrix} \quad (2-11) \\
 &= S \otimes d_{11}^{(0)} + T_{C4}
 \end{aligned}$$

와 같이 나타난다. 같은 방법으로 프랙탈 변환을 n 번 반복 수행하였을 때 얻어지는 2²:1 공간축소 변환된 정의역 블록 영상 벡터 $D_c^{(n)}$ 에 대한 끝개영상 $D_c^{(T)}$ 의 화소값 $d_{pq}^{(T)}$ 와 일반화 식은

$$\begin{aligned}
 [D_c^{(T)}] &= \lim_{n \rightarrow \infty} \sum_{p=1}^2 \sum_{q=1}^2 d_{p,q}^{(n)} \\
 &= S \otimes d_{11}^{(T)} + T_{C4} \quad (2-12) \\
 &= \begin{pmatrix} \frac{t_{11}^{(1)}}{1-s_1} & & & \\ & s_2 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(2)} & & \\ & & & \\ & s_3 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(3)} & s_4 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(4)} & \end{pmatrix}
 \end{aligned}$$

와 같이 표현된다. 식 (2-12)의 끝개 영상의 공간 축소 변환된 정의역 블록의 영상 $D_c^{(T)}$ 을 프랙탈 변환을 하면 정의역 블록의 끝개 영상 $X^{(T)}$ 과 치역 블록의 끝개영상 $R^{(T)}$ 해석식을

$$\begin{aligned}
 [X^{(T)}] &= \bigcup_{k=1}^4 R^{(T,k)} \quad (2-13) \\
 &= S \otimes D_c^{(T)} + T_4 \\
 &= \begin{bmatrix} s_1 & s_2 \\ s_3 & s_4 \end{bmatrix} \otimes \begin{bmatrix} \left(\frac{t_{11}^{(1)}}{1-s_1} \right) & s_2 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(2)} \\ s_3 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(3)} & s_4 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(4)} \end{bmatrix} \\
 &\quad + \begin{bmatrix} t_{11}^{(1)} & t_{12}^{(1)} & t_{11}^{(2)} & t_{12}^{(2)} \\ t_{21}^{(1)} & t_{22}^{(1)} & t_{21}^{(2)} & t_{22}^{(2)} \\ t_{31}^{(1)} & t_{32}^{(1)} & t_{31}^{(2)} & t_{32}^{(2)} \\ t_{41}^{(1)} & t_{42}^{(1)} & t_{41}^{(2)} & t_{42}^{(2)} \end{bmatrix}
 \end{aligned}$$

으로부터 [19]에서와 같은 해석식을 얻을 수 있다.

따라서 Monro와 Dudbridge 알고리즘과 같이 탐색 정의역이 고정되어 부호화되면 영상의 복원시 임의의 초기영상에 대하여 반복적으로 프랙탈 변환을 수행하지 않고도 부호화 계수만으로 표현되는 해석식을 이용하여 끝개 영상을 얻을 수 있다.

2) 8×8 정의역 블록 끝개 영상의 해석식

정의역의 크기가 8×8 블록일 때의 끝개영상 요소를 얻는 과정은 4×4블록일 때와 같이 임의의 초기 영상에 대하여 프랙탈 변환을 계속 반복 수행하여 얻어진다. 그 결과는 수렴된 영상의 축소 변환된 4×4 정의역 블록 $D_{C8}^{(T)}$ 은

$$\begin{aligned}
 D_{C8}^{(T)} &= \bigcup_{i=1}^4 D_{C8}^{(T,i)} \quad (2-14) \\
 &= S \otimes (S \otimes d_{11}^{(T)} + T_{C4}) + T_{C8}
 \end{aligned}$$

와 같이 표현되며, T_{C8} 은 8×8 평행변환 벡터의 공간 축소 벡터이다.

식 (2-15)에서 $D_{C8}^{(T(i))}$ ($i=1,2,3,4$)는 그림 1의 정의역내의 i 번째 블록과 같은 형식으로 축소 변환된 정의역 블록내의 부블록의 확인번호(ID)를 나타낸다. 또한 $d_{pq}^{(T)}$ ($p,q = 1,2,3,4$)는 축소 변환된 끝개영상에 대한 정의역 블록의 화소 값을 의미한다.

끝개 영상 화소값 $d_{pq}^{(T)}$ ($p,q = 1,2,3,4$)의 해석식은

$$D_{C8}^{(T(1))} = \begin{bmatrix} \frac{t_{11}^{(1)}}{1-s_1} & s_1 \left(s_2 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(2)} \right) + t_{13}^{(1)} \\ s_1 \left(s_3 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(3)} \right) + t_{11}^{(1)} & s_1 \left(s_4 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(4)} \right) + t_{13}^{(1)} \end{bmatrix} \quad (2-15a)$$

$$D_{C8}^{(T(2))} = \begin{bmatrix} s_2 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(2)} & s_2 \left(s_2 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(2)} \right) + t_{13}^{(2)} \\ s_2 \left(s_3 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(3)} \right) + t_{11}^{(2)} & s_2 \left(s_4 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(4)} \right) + t_{13}^{(2)} \end{bmatrix} \quad (2-15b)$$

$$D_{\alpha}^{(\pi_{31})} = \begin{bmatrix} s_3 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(3)} & s_3 \left(s_2 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(2)} \right) + t_{13}^{(3)} \\ s_3 \left(s_3 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(3)} \right) + t_{31}^{(3)} & s_3 \left(s_4 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(4)} \right) + t_{33}^{(3)} \end{bmatrix} \quad (2-15c)$$

$$D_{\alpha}^{(\pi_{41})} = \begin{bmatrix} s_4 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(4)} & s_4 \left(s_2 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(2)} \right) + t_{13}^{(4)} \\ s_4 \left(s_3 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(3)} \right) + t_{31}^{(4)} & s_4 \left(s_4 \left(\frac{t_{11}^{(1)}}{1-s_1} \right) + t_{11}^{(4)} \right) + t_{33}^{(4)} \end{bmatrix} \quad (2-15d)$$

처럼 축소 변환된 정의역 블록을 4개의 부분블록 $D_{\alpha}^{(T(i))}$ ($i=1,2,3,4$)으로 표현한다

식 (2-15)의 축소변환된 정의역 블록의 해석식을 이용하여 그림 2-1과 같이 프랙탈 변환을 한 번 더 행하여 정의역 블록의 쪼개 영상을 얻을 수 있다. 이러한 정의역 블록의 쪼개영상 $X^{(n)}$ 의 식은

$$\begin{aligned} X^{(n)} &= \prod_{i=1}^4 W^{(i)}(D) \\ &= \prod_{i=1}^4 (s_i D_{\alpha}^{(T(i))} + T^{(i)}) \\ &= \prod_{i=1}^4 R^{(\pi(i))} \\ &= S \otimes D_{\alpha}^{(T)} + T_8 \end{aligned} \quad (2-16)$$

와 같이 표현되고, 쪼개기 영상 $X^{(n)}$ 의 해석식은 [19]에서와 같이 표현할 수 있다.

3) 임의의 정의역 블록 쪼개영상의 일반식

임의의 정의역 블록에 대한 쪼개영상 일반식을 1)와 2)의 식들로부터 구한다. 공간 축소된 정의역 블록의 쪼개영상 일반식의 유도는 상당한 의미가 있다. 왜냐하면, 공간 축소된 정의역 블록의 쪼개영상은 1 번의 프랙탈 변환으로 정의역 블록의 쪼개영상은 구해진다. 그러므로 공간 축소된 정의역 영상 블록은 가장 먼저 구해져야 한다. 수립된 공간 축소 변환된 4x4 정의역 블록 D_{α} 는

$$D_{\alpha} = \begin{bmatrix} s_1 D_{\alpha} + T_{\alpha}^{(1)} & s_2 D_{\alpha} + T_{\alpha}^{(2)} \\ s_3 D_{\alpha} + T_{\alpha}^{(3)} & s_4 D_{\alpha} + T_{\alpha}^{(4)} \end{bmatrix} = \begin{bmatrix} s_1 & s_2 \\ s_3 & s_4 \end{bmatrix} \otimes D_{\alpha} + \begin{bmatrix} T_{\alpha}^{(1)} & T_{\alpha}^{(2)} \\ T_{\alpha}^{(3)} & T_{\alpha}^{(4)} \end{bmatrix} \quad (2-17)$$

$$= S \otimes D_{\alpha} + T_{\alpha} \quad T_{\alpha}^{(1)} = \begin{bmatrix} t_{11}^{(1)} & t_{13}^{(1)} \\ t_{31}^{(1)} & t_{33}^{(1)} \end{bmatrix} \quad (2-18a)$$

$$T_{\alpha}^{(2)} = \begin{bmatrix} t_{11}^{(2)} & t_{13}^{(2)} \\ t_{31}^{(2)} & t_{33}^{(2)} \end{bmatrix} \quad (2-18b)$$

$$T_{\alpha}^{(3)} = \begin{bmatrix} t_{11}^{(3)} & t_{13}^{(3)} \\ t_{31}^{(3)} & t_{33}^{(3)} \end{bmatrix} \quad (2-18c)$$

$$T_{\alpha}^{(4)} = \begin{bmatrix} t_{11}^{(4)} & t_{13}^{(4)} \\ t_{31}^{(4)} & t_{33}^{(4)} \end{bmatrix} \quad (2-18d)$$

와 같이 Kronecker 곱으로 표현이 가능하다.

정의역 블록 4x4, 8x8에 대한 공간 축소변환된 정의역 블록 D_{α} , D_{α} 식을 유도하였고, 8x8 정의역 블록의 일반식은

$$\begin{aligned} D_8 &= S \otimes D_{\alpha} + T_8 \\ &= S \otimes (S \otimes D_{\alpha} + T_{\alpha}) + T_8 \\ &= S \otimes S \otimes D_{\alpha} + S \otimes T_{\alpha} + T_8 \end{aligned} \quad (2-19)$$

와 같이 유도할 수 있다. 그리고 $2n \times 2n$ 정의역 블록의 일반식은

$$\begin{aligned} D_{2^n} &= S \otimes D_{2^{n-1}} + T_{2^n} \\ &= S \otimes (S \otimes D_{2^{n-2}} + T_{2^{n-1}}) + T_{2^n} \\ &= (S \otimes \dots \otimes S \otimes D_{2^2}) + (S \otimes \dots \otimes S \otimes D_{2^3}) \\ &\quad + (S \otimes \dots \otimes S \otimes D_{2^4}) + \dots + T_{2^n} \end{aligned} \quad (2-20)$$

와 같이 표현할 수 있다.

III. 무반복 프랙탈 복호화 알고리즘 기반의 복호화기 설계 및 제작

본 논문에서 제작된 무반복 프랙탈 복호화 알고리즘 기반의 프랙탈 복호화기는 영상 처리 속도를 증가시키기 위하여 복수 개의 프로세서를 사용하는 병렬 프로세서 구조로 이루어져있다. 즉 구조적으로는 공유메모리를 중심으로 독립적인 수행이 가능한 다중프로세서로 이루어진 다중버스 구조이다.

프랙탈 영상 복호화기에 장착된 프로세서들은 MIMD(Multiple Instruction streams over Multiple Data streams)^[20] 구조를 갖는다. 그리고 제작된 프랙탈 영상 복호화기는 영상 처리 능력을 향상시키기 위하여 기능별로 파이프라인 동작이 되도록 설계되었다.

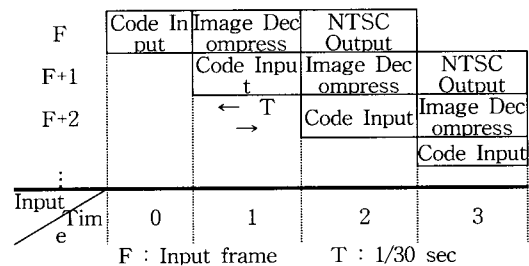


그림 3-1. 제작된 프랙탈 영상 복호화기의 파이프라인 동작 Fig 3-1. Pipeline operation of constructed fractal image decoder.

그림 3-1은 설계 제작된 프랙탈 영상 복호화기의

파이프라인 동작을 나타내고 있다. 또한 프로세서들의 영상 처리 속도를 저하시키는 요인인 프로그램 길이의 증가를 피하고 최고의 처리속도를 갖기 위하여 모든 프로그램은 어셈블리어로 작성하여 고수준(high level) 언어에서 나타나는 프로그램의 중복성(redundancy)을 제거하였다.

본 논문에서 설계 제작된 무반복 프랙탈 복호화 알고리즘을 이용하는 프랙탈 영상 복호화기는 구조적으로는 영상복원보드(image decoding board)와 영상신호출력보드(NTSC signal output board)로 이루어져 있으며 기능적으로는 복호화 입력제어부(decoding input controller), 복호화 주제어부(decoding main controller) 그리고 NTSC 부호화부(NTSC encoder)로 이루어져 있다.

영상복원보드는 2개의 ADSP2181^[21]을 이용하여 파이프라인 구조로 설계되었다. 첫 번째 ADSP2181 칩은 프랙탈 영상 부호코드를 RS422 입력 포트로부터 입력받아 허프만 복호화를 행한 후 역양자화를 수행하는 복호화 입력제어부 역할을 한다. 두 번째 ADSP2181 칩은 영상의 복원 및 NTSC 부호화의 감시 및 제어, PC와의 RS232C를 통한 접속 및 복호화 모드시 영상 부호 코드를 무반복 프랙탈 복호화 변환을 수행하여 복원영상을 만드는 복호화 주제어부 역할을 한다. 영상신호출력보드는 복원된 디지털 영상을 Brooktree사의 Bt856^[22] 칩을 이용하여 NTSC 표준영상신호를 발생시키는 NTSC 부호화부 역할을 수행한다. 이 영상신호 출력보드는 복원될 영상의 크기를 복호화 주제어부에서 소프트웨어적으로 가변시킬 수 있도록 하였다. 설계 제작된 프랙탈 영상 복호화기의 하드웨어 사양은 표 3-1과 같다.

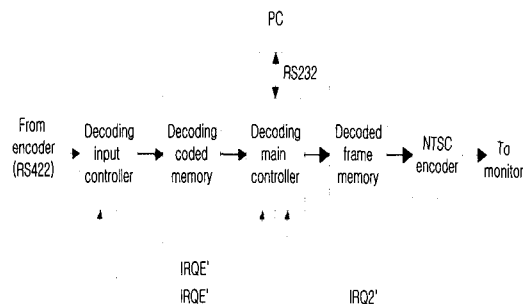


그림 3-2. 제작된 프랙탈 영상 복호화기의 하드웨어 블록 다이어그램.
Fig 3-2. Hardware block diagram of constructed fractal image decoder.

그림 3-2의 프랙탈 영상 복호화기의 하드웨어 블

표 3-1. 제작된 프랙탈 영상 복호화기의 하드웨어 사양
Table 3-1. Hardware specifications of constructed fractal image decoder.

DSP	ADSP2181 (Input & Main controller)
Main Memory	48kbytes (Internal RAM) 64kbytes (EEPROM)
Decoder Memory	512kbytes (UM611024AK-20*4SRAM)
Coded Memory	256kbytes (UM611024AK-20*2SRAM)
DAC	1 (Bt856 : NTSC encoder)
Input channel	1 (RS422 : Max 1Mbps)
Output channel	1 (NTSC)
Serial com.port	1 (RS232)

록 다이어그램이다. 이는 두 개의 ADSP2181 프로세서는 복호화 입력 제어부와 복호화 주제어부로 동작하며, 복호화 코드 메모리(decoding coded memory)를 중심으로 시스템 버스는 복호화 입력 제어부 버스(decoding input controller bus) 와 복호화 주제어부 버스(decoding main controller bus)로 구분되는 다중버스 구조를 갖는다.

1. 복호화 입력제어부

RS422 포트로부터 입력되어오는 프레임 단위의 코드들을 허프만 복호화와 역양자화를 통하여 복호화 코드 메모리에 프레임 단위로 저장한다. 한 개의 프레임 단위로 프랙탈 코드 데이터가 저장되어 완료되면 복호화 주제어부로 프레임 코드 데이터가 수신되어 완료되어 복호화 코드 메모리에 저장되어 완료를 알리는 인터럽트를 발생시킨다. 복호화 입력 제어부는 ADSP2181 프로세서의 내부 메모리에 있는 프랙탈 계수를 BDMA를 통해서 복호화 코드 공통 메모리에 저장한다. RS422 포트를 통하여 연속적으로 입력되는 데이터는 프레임 단위로 2개의 복호화 코드 메모리 बैं크를 교대로 사용한다. 이는 하나의 메모리 बैं크에 복호화 코드를 입력할 때 다른 메모리 बैं크는 복호화 주제어부에서 사용하여 프랙탈 복호화를 수행하도록 하기 위해서다.

복호화 입력제어부의 RS422 포트를 이용한 전송은 ADSP2181 프로세서의 부하를 줄이기 위하여 MC68681을 사용하도록 설계되었다.

2. 복호화를 위한 부호화 메모리

수신된 부호화 데이터를 저장하는 곳으로 128kbyte 메모리 बैं크 2개로 이루어져 있다. 그림 3-1에서 볼 수 있듯이 고속의 파이프 라인 방식으로 복

호화 입력제어부와 복호화 주제어부가 동시에 작업을 수행하므로 입력제어부가 수신된 부호화 데이터를 저장하는 메모리와 복호화 주제어부가 읽어오는 메모리가 각각 필요하다. 따라서 2개의 메모리 뱅크를 사용하였다. 각각의 제어부는 데이터의 충돌을 피하기 위하여 메모리를 접근하기 전에 다른 제어부가 사용하기를 원하는 메모리 뱅크의 사용권을 확인한 후에 BDMA를 이용하여 데이터의 읽어오기를 시작하게 된다.

3. 복호화 주제어부

복호화 주제어부의 역할은 복호화기의 초기화와 수신된 코드를 프레임 변환을 통하여 복호화를 행하여 복원영상을 만들고, 복호화된 디지털 데이터를 복호화 프레임 메모리(decoding frame memory)에 저장한다. 복호화기의 초기화는 Bt856 NTSC 부호화기를 초기화시키고, 복원 영상의 크기를 결정하는 수평크기, 수직크기등을 결정해 주는 것을 포함한다.

복호화 주제어부는 영상 데이터의 복호화를 위해서 복호화 코드 메모리로부터 데이터를 읽어올 때와 복호화된 데이터를 복원 프레임 메모리(decoded frame memory)에 저장할 경우, 복호화 입력 제어부의 프레임 코드 수신 저장완료 신호인 인터럽트 요구에 의해 기동되며, 복호화가 끝나면 복원된 영상 데이터를 복원 프레임 영상 메모리에 저장한다.

PC와의 RS232 통신은 ADSP2181 프로세서 내부의 직렬 통신 장치, 타이머(timer) 그리고 flag I/O 를 사용하여 소프트웨어로 이루어지도록 하였다. 제작된 복호화기에서는 BDMA를 이용한 부팅을 사용하였다.

4. 복호화 프레임 메모리

복호화 프레임 메모리는 복원 영상을 저장하고, NTSC 부호화기가 디지털 영상을 아날로그 영상으로 변환시 사용되도록 한다. 복호화 프레임 메모리는 128kbyte SRAM 2개가 하나의 뱅크로 사용되며 각 메모리 뱅크(bank)는 하나의 프레임에 대한 정보를 저장한다. 복호화 프레임 메모리는 2개의 메모리 뱅크로 이루어져 있다. 이는 복호화 주제어부가 복호화를 수행한 후 복원영상을 저장하기 위하여 메모리 뱅크를 하나 사용하고 나머지 하나의 메모리 뱅크는 NTSC 부호화기에서 NTSC 영상출력 신호 발생을 위하여 사용한다. 메모리 뱅크의 사용권은 NTSC 부호화기에서 발생하는 프레임 신호에

의해 복호화 주제어부에 주어지며, 복호화 주제어부는 인터럽트 신호로 사용되는 프레임 신호에 의해 NTSC 영상신호의 프레임 신호의 동기(sync) 시간 간격에 메모리 뱅크의 사용권을 교환시킨다.

복호화 주제어부가 NTSC 부호화기와 같은 프레임 메모리 뱅크를 접근(access)하는 것을 방지하기 위하여, 복호화 주제어부는 프레임 메모리를 접근하기 전에 NTSC 부호화기가 사용하고 있는 메모리 뱅크를 확인한 후에 프레임 메모리에 접근을 시작한다. 복호화 주제어부가 코드 메모리로부터 프랙탈 코드를 읽어올 때 그리고 프레임 메모리로 복원 영상 데이터를 쓸 때도 BDMA를 이용한다.

5. NTSC 부호화부

NTSC 표준 영상 신호 부호화부의 블록 다이어그램은 그림 3-3과 같다. 그 기능은 다음과 같이 요약된다. NTSC 표준영상 신호 부호화부는 디지털 영상 신호를 아날로그 NTSC 영상신호로 변환한다. 이를 위하여 제어 신호 발생단, 프레임 메모리 어드레스 발생단, D/A 변환단으로 구성되었다.

제어신호 발생단(control signal generator)은 출력되는 영상의 크기를 결정하는 수직 동기 신호를 발생시키는 수직 동기 신호 분리단(vertical sync separator)과 수평 동기 신호를 발생시키는 수평 동기 신호 분리단(horizontal sync separator)으로 이루어진 조합논리회로로 구성된다. 제어신호 발생단에서는 NTSC 부호기에서 발생하는 필드동기신호, 수평동기신호, 수직동기신호를 이용하여 원하는 크기만큼의 D/A 변환과 어드레스 발생을 수행하도록 D/A 변환가능신호(enable)를 발생시킨다.

프레임 메모리 어드레스 발생단은 프레임 메모리에 저장된 영상 데이터를 NTSC 부호화기가 읽을 수 있도록 어드레스를 발생시키는 역할을 한다.

D/A 변환단(D/A converter)는 디지털 영상 신호를 D/A 변환하여 아날로그 NTSC 표준영상 신호를 발생시키는 역할을 하는 NTSC 부호화기이다.

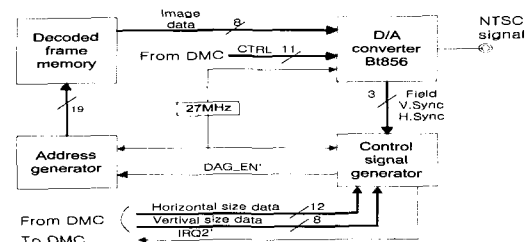


그림 3-3. NTSC 부호화부의 블록 다이어그램
Fig 3-3. Block diagram of NTSC encoder.

그림 3-3에서 수평 크기 데이터는 출력 영상의 가로 화소의 수, 수직 크기 데이터는 출력 영상의 세로 화소의 수를 결정한다. 이러한 값들은 시스템의 초기화 과정에서 복호화 제어부에 의해서 결정된다. 제작된 복호화기는 초당 처리되는 프랙탈 복원 영상과는 관계없이 30 프레임을 NTSC 표준 영상 신호가 출력된다.

6. 프랙탈 영상 복호화기의 소프트웨어

제작된 프랙탈 영상 복호화기의 모든 프로그램은 최고의 복호화 속도를 얻기 위하여 어셈블리어로 작성하였다. 또한 부동소수점에 대한 ADSP2181 프로세서의 연산에 대한 부담을 줄이기 위하여 정수연산으로 변환하여 II장 3절에서 정의역 블록의 끝개영상 일반식의 해석식을 [19]에서와 같이 유도 적용하여 프로그램을 작성하였다.

유도된 해석식은 [19]에서 $t_{11}^{(k)}$ 는 $a_{1,n}^{(k,m)} \cdot x_i + a_{2,n}^{(k,m)} \cdot y_j + b_n^{(k,m)}$ 에서 8×8 치역 블록의 확인번호와 4×4 치역 블록의 확인번호를 무시하면

$$t_{11}^{(k)} = a_{1,1}^{(k,m)} \cdot x_1 + a_{2,1}^{(k,m)} \cdot y_1 + b_1^{(k,m)} \quad (3-1)$$

같이 나타낼 수 있다.

식 (3-1)에서 화소값 이동 $b_n^{(k,m)}$, X축 비례 축소 $a_{1,n}^{(k,m)}$, Y축 비례 축소 $a_{2,n}^{(k,m)}$, 화소값 비례 축소 $a_{3,n}^{(k,m)}$ 값들을 이용한 영상 복원 속도를 향상시키기 위하여 복원영상의 화질에 영향이 최소화하는 일정한 상수를 곱하여 실수의 프랙탈 코드를 정수화하여 복호화를 수행한다. 그 이유는 복호화기에 사용된 ADSP2181은 부동소수점을 지원하지 않기 때문이다. 또한 실수 연산이 필요한 부분에서는 사용자 정의 부동 소수점 형식^[23]을 사용하였다.

그림 3-4은 제작된 프랙탈 영상 복호화기의 입력 제어부와 복호화 제어부의 프로그램 흐름도이다.

IV. 무반복 프랙탈 알고리즘 기반의 복호화기 성능평가

본 장에서는 II장 3절에서 유도된 정의역 블록의 끝개 영상 해석식을 이용한 무반복 프랙탈 복호화 알고리즘을 검증하기 위하여 테스트 영상으로는

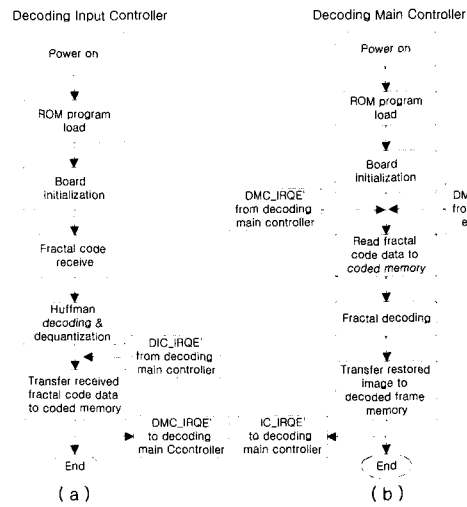


그림 3-4. (a) 복호화 입력 제어부 (b) 복호화 제어부의 프로그램 흐름도

Fig 3-4. Flowchart of (a) decoding input controller, and (b) decoding main controller program.

256×256 크기의 Lady, Lena, Boat, Face 영상을 이용하였다. 그리고 무반복 프랙탈 복호화 알고리즘을 기반으로 하여 제작된 프랙탈 영상 복호화기의 성능을 검증하기 위하여 MPEG-4의 표준규격인 QCIF 포맷(144×176)의 Claire 영상 100 프레임을 사용하여 설계 제작된 복호화기를 이용하여 복호화를 수행하였다.

1. 해석식을 이용한 무반복 프랙탈 복호화 알고리즘의 성능 평가

부호화시 평행변환 연산자 T 는 X 및 Y 축에 대하여 1차원 식

$$W^{ki}(D) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (s_k d_c(i, j) + a_k x_i + b_k y_j + c_k) \quad (4-1)$$

을 적용하였다.

테스트 영상들에 대한 해석식을 이용한 끝개영상의 복원화 과정의 성능 평가를 위한 시뮬레이션 결과를 표 4-1에서 나타내었다. 부호화시 사용된 정의역 블록의 크기는 8×8 이다. 표 4-1에서 proposed method행은 식 (2-16)의 해석식을 이용하여 얻은 끝개영상들의 PSNR 값을 표시하였고 iteration method행은 기존의 반복에 의한 복호화 과정으로써 초기 영상을 사용하여 축소변환을 수행하였을 때 축소변환 회수에 따른 PSNR값의 변화를 나타

표 4-1. 제안된 복호화 알고리즘 및 기존의 반복에 의한 복호화 알고리즘의 PSNR 값 비교
Table 4-1. Compare PSNR values using the proposed method and conventional iteration methods.

Source image	Initial image	Proposed method	Average method	Iteration method (dB)							
				1st	2nd	3rd	4th	5th	6th	7th	8th
Lady	Lena	31.72dB	31.49dB	21.40	28.41	30.98	31.54	31.69	31.70	31.71	31.71
	Boat			20.19	27.77	30.78	31.61	31.67	31.70	31.70	31.71
	Face			18.92	26.68	30.42	31.53	31.64	31.68	31.70	31.70
Lena	Lady	28.39dB	28.16dB	20.09	26.08	27.96	28.36	28.41	28.42	28.42	28.42
	Boat			20.17	26.07	27.96	28.38	28.41	28.42	28.42	28.42
	Face			20.15	26.04	27.90	28.28	28.35	28.37	28.38	28.39
Boat	Lady	28.81dB	28.48dB	19.02	25.52	28.19	28.74	28.80	28.81	28.82	28.82
	Lena			20.47	26.52	28.44	28.77	28.81	28.81	28.81	28.81
	Face			18.22	25.11	27.94	28.65	28.75	28.79	28.80	28.80
Face	Lady	33.86dB	33.60dB	19.03	27.44	32.10	33.60	33.79	33.84	33.85	33.60
	Lena			21.63	29.32	32.71	33.62	33.78	33.82	33.84	33.85
	Face			19.24	27.72	32.24	33.56	33.77	33.82	33.85	33.85

낸다. 예상되는 바와 같이 반복회수가 증가할수록 P SNR 값은 해석식을 이용한 값에 수렴함을 볼 수 있다. 표 4-1에 나타난 바와 같이 반복적 방법으로 영상 복원을 수행하여 PSNR값이 0.1dB 이하로 충분히 수렴되는 영상을 얻기 위해서는 5~6회 정도의 반복적인 축소변환이 요구되므로 해석식을 이용하여 영상복원을 수행할 경우 기존의 방법보다 복원속도가 약 5배 빨라짐을 알 수 있다.

Average method행은 식 (2-9)에 표시된 평균 화소값을 이용하여 끝개영상을 복원하였을 때의 PSNR 값을 표시된다. 시뮬레이션 결과 사용된 테스트 영상들에 대하여 식 (2-8)을 이용한 샘플링방법이 식 (2-9)를 이용한 경우보다 0.2~0.3dB 정도 PSNR 값이 높음을 볼 수 있다.

표 4-1에서 초기 영상에 대한 수렴속도의 차이에 대하여 생각해 보면, 스케일링 상수 s_k 값의 크기가 작으면 반복회수의 증가에 따라 빠르게 끝개영상으로 수렴하게되나 s_k 값의 크기가 1에 접근하게되면 수렴속도는 현저히 떨어짐을 알 수 있다^[9]. 또한 s_k 값이 1에 근접하면 초기영상에 따라 수렴속도가 많은 차이가 난다. 특히 영상의 변화가 거의 없는 블록에 대하여 영상복원을 수행할 때 s_k 의 값이 1에 근접하면 초기영상에 따라서 수십 번 프랙탈 변환을 반복수행을 하여도 끝개영상에의 수렴이 이루어지지 않아 평탄한 복원 영상부분에 모자이크 형태의 블록들이 나타난다. 이러한 현상은 부호화 과정에서 s_k 의 크기가 1에 가까울 때 강제적으로 $s_k = 0.5$ 로 축소하여 고정시킨 다음 a_k, b_k, c_k 를 최소 자승법으로 구하여 모자이크 형태의 블록들을 제거

시킬 수 있었다. 그러나 해석식을 이용하여 복원할 경우에는 직접 복원영상을 해석적으로 얻을 수 있으므로 위에 언급된 반복과정이 필요없다.

표 4-2는 해석식을 이용한 영상복원과 기존의 방법을 이용하여 영상복원을 수행하였을 때 필요한 연산량에 대하여 나타내고 있다. 즉 제안된 방법에 대한 연산량이 기존의 방법에 비해 연산량이 다소 많은 편이나 제안된 방법은 복원 과정이 1회만 수행되나, 기존의 방법은 표 4-1에서 나타난 바와 같이 5회이상 반복 적용해야 함으로 연산량에서 보여주는 상당한 이점이 속도 향상을 가져옴을 알 수 있다.

표 4-2. 8x8 정의역블록에 대한 제안된 방법과 반복 방법과의 연산량 비교

Table 4-2. Comparison operation of proposed method with iteration method for 8x8 domain block.

Method Computation	Proposed method	Iteration method	
		1 time	5 times
+	174	192	960
-	37	.	.
×	130	192	960
÷	1	.	.
Shift	82	.	.

2. 동영상 프랙탈 영상 복호화기의 성능평가

제작된 프랙탈 영상 복호화기의 성능을 평가하기 위하여 II장 3절에서 유도된 정의역 블록의 끝개영상 해석식을 이용하여 구현된 복호화기를 검증하기 위하여 ADSP2181 프로세서 시뮬레이터를 이용하여 제작된 복호화기의 성능 평가를 위하여 복원영상 화질을 컴퓨터 시뮬레이션 결과와 비교하였고 제작된 복호화기의 복호화 시간이 최대 31.2ms로 처리할 수 있어 초당 30프레임 처리 가능함을 확인하였다.

표 4-3에서는 테스트 영상들에 대한 PC 시뮬레이션과 제작된 복호화기의 복원영상의 화질을 나타내었다. 부호화시 사용된 치역블록의 크기는 4×4이다. 표 4-3에서 constructed decoder행은 식 (2-16)의 해석식을 이용하여 제작된 복호화기를 통해 얻은 끝개영상들의 PSNR 값을 표시하고, PC simulation행은 PC 시뮬레이션을 통해 얻은 끝개영상들의 PSNR 값을 표시하였다. Average method행은 식 (2-9)에 표시된 평균화소값을 이용하여 끝개영상을 복원하였을 때의 PSNR 값을 표시된다. 시뮬레이션 결과 사용된 테스트 영상들에 대하여 PC 시뮬레이션과 제작된 복호화기의 복원 영상은 같음을 알 수 있다. 표 4-4에서는 제작된 복호화기를 이용한 복원에 소요되는 시간은 해석식을 이용할 경우 8×8 및 16×16의 정의역에 대하여 각각 31.2ms 와 32.1ms가 소요되었다.

본 논문에서 설계 제작된 무반복 프랙탈 복호화 알고리즘 기반의 복호화기는 프랙탈 부호 수신, 영상 복원, NTSC 표준영상 출력등의 기능을 갖는 프랙탈 영상 복호화기이다. 그림 4-1는 제작된 프랙탈 영상 복호화기의 사진이다.

표 4-3. 제작된 복호화기와 PC 시뮬레이션의 복원영상 비교
Table 4-3. Comparison decoding image of the constructed decoder with simulated PC.

Source image	Constructed decoder	PC Simulation	
		Proposed method	Average method
Lady	31.73dB	31.72dB	31.49dB
Lena	28.40dB	28.39dB	28.16dB
Boat	28.82dB	28.81dB	28.48dB
Face	33.86dB	33.86dB	33.60dB

표 4-4. 제안된 알고리즘과 반복적 방법에 의하여 복호화시 소요 되는 시간의 비교
Table 4-4. Comparison decoding time of the proposed with iteration method for QCIF.

Method	Domain block size	Block	Frame
Constructed decoder	8×8	78.7μs	31.2ms
	16×16	324μs	32.1ms

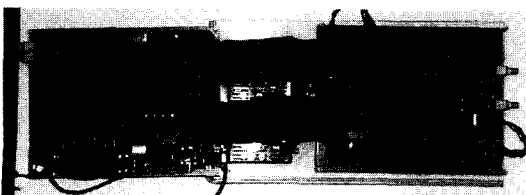


그림 4-1. 제작된 프랙탈 영상 복호화기의 사진
Fig 4-1. Photograph of the constructed fractal image decoder.

V. 결 론

본 논문에서는 무반복 프랙탈 복호화 알고리즘을 제안하고, 이를 이용한 프랙탈 영상 복호화기를 설계 제작하였다.

본 논문에서는 제안된 무반복 프랙탈 복호화 알고리즘으로 빠른 복원 영상을 얻기 위하여 정의역 블록의 끝개영상 해석식과 해석식에 대한 일반화식을 유도하였다. 본 논문에서 유도된 해석식을 이용한 복호화 과정과 프랙탈 영상의 일반적인 반복적인 축소변환에 의한 복호화 과정과 비교하여 볼 때, 복원영상의 화질은 같았으며, 유도된 해석식을 이용할 경우 무반복 복호화 속도는 약 5배정도 향상됨을 시뮬레이션을 통하여 확인할 수 있었다. 또한 제안된 무반복 프랙탈 영상의 복호화 알고리즘을 기반으로 하는 프랙탈 영상 복호화기의 하드웨어를 설계 제작하였다. 제작된 프랙탈 영상 복호화기의 성능은 QCIF의 Claire 영상에 대하여 제안된 무반복 프랙탈 복호화 알고리즘의 유도된 해석식을 이용할 경우, 정지영상에 대한 부호화시 최대 31.2ms로 초당 30 프레임 이상의 복호화 속도를 가진다. 이는 움직임이 있는 일부 영상만을 부호화하여 전송하는 동영상에 대한 본 논문에서 설계 제작된 프랙탈 영상 복호화기는 QCIF영상에 대한 동영상 처리가 충분함도 확인하였다.

참고문헌

- (1) B. Mandelbrot, The Fractal Geometry of Nature, W. H. Freeman and Co., San Francisco, 1982.
- (2) Peitgen, Jurgen, and Saupe, Chaos and Fractal New Frontiers of Science, Springer Verlag, New York, 1992.
- (3) M. F. Barnsley, Fractals Everywhere, Academic Press, 1988.
- (4) M. F. Barnsley and A. D. Sloan, "A better way to compress images," Byte 13(1), pp. 215-224, 1988.
- (5) M. F. Barnsley and L. P. Hurd, Fractal Image Compression, AK Priters, Ltd., 1992.
- (6) A. E. Jacquin, A Fractal Theory of

