

프랙탈 영상 압축의 진화적인 계산에 관한 연구

유 환 영[†] · 최 봉 한^{††}

요 약

프랙탈 영상 압축(Fractal Image Compression : FIC)의 진화 계산(Evolution Computation)을 이용한 영상 분할(Image Partition)을 소개한다. 프랙탈 영상 압축에서 치역(Ranges)의 영상 분할은 꼭 필요하다[1]. 프랙탈 영상 압축은 쉽고 빠르게 복원된다는 장점을 갖는 데 비해 반복적인 프랙탈 변환의 적용으로 많은 계산량을 필요로 한다는 단점을 가지고 있다. 위와 같은 문제점을 해결하기 위한 방법으로 영상 분할을 하는 데 있어 진화 계산을 적용하는 것에 대해 제안한다. 치역 영상(Ranges Image)은 작은 사각(Square) 영상 블록들의 결합된 집합으로 구성할 수 있다. 모집단을 구성하는 하나의 N_p 는 분할되어진 하나의 코드들이다. 진화 계산에서 각각의 구성은 두 개의 이웃하는 치역은 제외하고 그들의 부모(Parents)로부터 분할을 상속받은 자식 σ 를 생성한다. 자손들의 최적의 영상은 콜라주 정리(Collage Theorem)에 기초를 둔 다음 세대 모집단을 위해 선택되어지고 처리된다. 최적의 영상은 영상 데이터에 포함된 중복성을 포함함으로써 적은 저장 공간을 차지하고, 속도 문제에 있어서 효율적이고 영상의 화질에 있어서 다른 부호화를 사용한 기법보다 우수한 성능을 갖는다. 멀티미디어 영상 처리(Multimedia Image Processing)의 진화 계산을 이용한 프랙탈 영상 압축은 영상의 복원과 영상의 질, 고 압축률을 요하는 동영상의 적용등의 많은 분야에 적용된다.

A Study on Evolutionary Computation of Fractal Image Compression

Hwan-Young Yoo[†] · Bong-Han Choi^{††}

ABSTRACT

This paper introduces evolutionary computing to Fractal Image Compression(FIC). In Fractal Image Compression(FIC) a partitioning of the image into ranges is required. As a solution to this problem, there is a propose that evolution computation should be applied in image partitionings. Here ranges are connected sets of small square image blocks. Populations consist of N_p configurations, each of which is a partitioning with a fractal code. In the evolution each configuration produces σ children who inherit their parent partitionings except for two random neighboring ranges which are merged. From the offspring the best ones are selected for the next generation population based on a fitness criterion Collage Theorem. As the optimum image includes duplication in image data, it gets smaller in saving space, more efficient in speed and more capable in image quality than any other technique in which other coding is used. Fractal Image Compression(FIC) using evolution computation in multimedia image processing applies to such fields as recovery of image and animation which needs a high-quality image and a high image-compression ratio.

* 본 연구는 1998학년도 순천제일대학 교내학술연구비의 지원을 받아 연구되었음

† 정 회 원 : 순천제일대학 컴퓨터과학과 교수

†† 준 회 원 : 순천제일대학 컴퓨터과학과 교수

논문접수 : 1999년 7월 8일, 심사완료 : 2000년 1월 9일

1. 서 론 - 진화 계산

원 영상(Initialization Image)으로부터 영상 압축을 하기 위해서 반복 함수 시스템(Iterated Function System : IFS)을 사용한다. 영상 전체를 압축 축소하는 것이 아니라 영상의 조그마한 부분은 영상의 다른 한 부분과 닮았기 때문에 그 부분의 영상을 반복 축소하여 압축할 수 있다.

원 영상의 유사한 영상을 찾기 위해 분할의 복적이 있다. 주어진 비트율(bit-rate)에서 최적의 분할을 찾는 것은 프랙탈 영상 압축에서 하나의 풀리지 않는 문제이다. 주어진 영상의 치역 공간을 분할을 하는 데 있어 처리 시간과 공간을 많이 차지한다는 것이다. 일반적으로 4분트리(quad tree)[2, 3], 직사각형(rectangular)[4], 삼각형(triangular)[5, 6], 다각형(polygonal)[7] 분할이 이용된다.

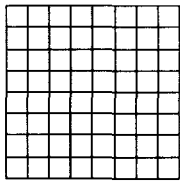
전통적인 4분 트리를 이용하여 실질적인 문제를 해결할 수 있기에 본 연구에서 최적의 영상으로 분할하기 위하여 4분 트리를 적용하였다.

영상은 간선으로 연결된 작은 사각형 영상 블록들의 결합으로 치역을 정의할 수 있다[8].

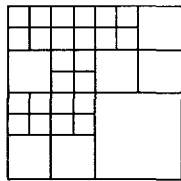
영상의 프랙탈 부호화는 선형 변환을 통한 표준의 형태를 따른다. 치역 블록(Range block) R 은 정의역 블록(Domain block)의 저장소를 선형 크기의 두배로 간주한다. 그 치역 블록들은 정의역 블록 크기에 맞추기 위해 픽셀 평균에 의해 축소되어진다.

원 영상을 $B \times B$ 화소 단위의 치역 블록 셀(cell) R_i 로 분할하고 각 R_i 의 주변에 존재하는 $2B \times 2B$ 의 정의역 셀 D_i 의 축소 사상에 의해 변환되어진다.

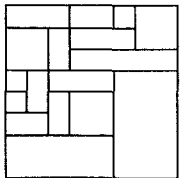
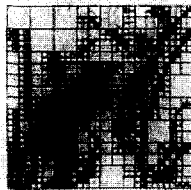
이 코드북 블록들의 저장소는 블록의 모든 8가지 선형 변환을 포함함으로써 변환되어진다. 이것은 코드북 블록들 D_1, \dots, D_{N_b} 의 저장소를 부여한다.



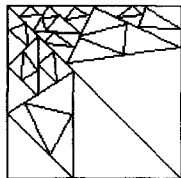
Tile



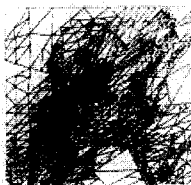
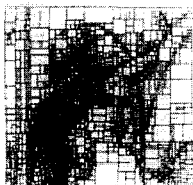
Quad-tree



Horizontal-Vertical



Triangular



변형이 없는 영상



Y 축에 대한 반사



X 축에 대한 반사



180° 회전



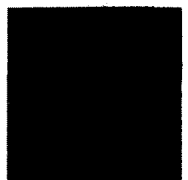
Y = X 축에 대한 반사



270° 회전
(역시계 90° 방향)



90° 회전
(시계 90° 방향)



Y = -X 축에 대한 반사

지역 R 과 코드북 블록 D 에 대해 $(s, o) = \arg \min_{s, o \in R} \|R - (sD + o1)\|^2$ 에서 1이 픽셀마다 최대의 강도를 가진 평면 블록인 곳에 허락한다. (s : 명암, o : 밝기)

상관계수 s 는 부호화 과정에서 수렴을 보장하기 위해 $[-1, 1]$ 에 고정되고 그 다음 s 와 o 둘 다 산출한 \bar{s} 와 \bar{o} 를 정량화 하게 된다.

지역 R 에 대한 콜라주 정리는 $E(D, R) = \|R - (\bar{s}D + \bar{o}1)\|^2$ 이다.

콜라주 오류 $E(D_k, R)$ 에 관하여 분류한 코드북 블록들 D_k 는 지표 k_1, \dots, k_{N_b} 를 산출한다.

지역 R 에 대한 프랙탈 코드는 최적의 지표 k_1 과 일치하는 정량화 환산과 오프셋의 매개변수 \bar{s} 와 \bar{o} 를 구성한다.

초기에 영상을 4×4 픽셀 사이즈 크기의 원자(atomic) 블록들로 하위 분할된다.

이런 표기법을 가지고 우리는 구성을 정의할 수 있다. 구성은 다음과 같이 조직된다.

- 분할, 즉 서로 해체된 지역 블록들의 집합은 전체 영상을 커버하고, 지역 블록들은 원자블록들의 간선으로 결합된 집합을 구성한다.
- 분할된 지역 블록들에 대해
 - d codebook이 지시하는 것은 k_1, \dots, k_d 의 리스트
 - codebook k_i 이 지시하는 것은 최적의 정량화 계수 \bar{s} 와 \bar{o} 이다.

모집단(population)은 분할에서 지역의 같은 수를 가진 N_b 구성의 한 집합이다. 그 진화적인 계산은 N_b 의 초기 모집단과 함께 시작되어 진다.

- 균일한 분할은 영상을 원자 블록들에서 하위 분할함으로써 획득되어진다.
- 각각의 블록(atomic block)에 대해 최적의 코드북 지표 k_1, \dots, k_d 와 정량화 계수 \bar{s} 와 \bar{o} 이다.

진화 전략에 있어서 새로운 자손 세대는 다음과 같이 만들어진다. 부모 모집단의 각각의 구성에 모든 이웃하는 지역 쌍들의 기록을 유지한다(지역들은 원자 블록의 간선을 나눌 때 이웃들로 간주되어진다).

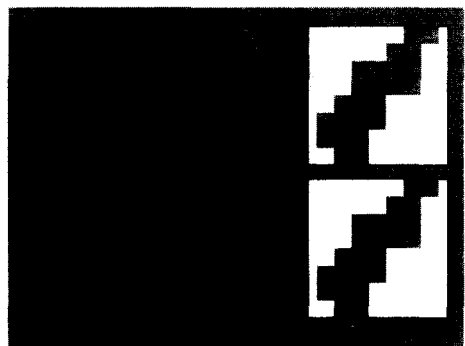
이런 기록으로부터 지역은 무작위로 선택되어진다.

이런 두 개의 지역들은 산출된 새로운 분할을 제거된 두 개의 오래된 지역들과 새로운 지역의 연합과 함께 통합되어진다.

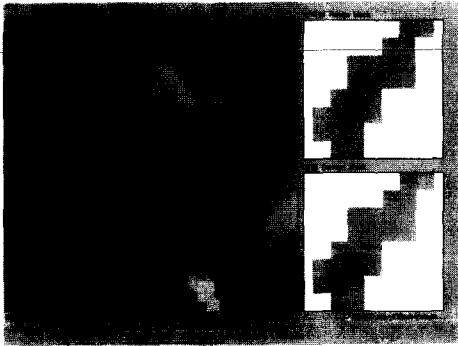
확대된 새로운 지역에 부합하는 정의역 블록을 획득하기 위해 각각의 정의역 저장소의 탐색을 하지 않지만 오히려 부모지역들로부터 유전된 정의역의 기록들에 의해 주어진 정의역들만을 간주하여 탐색한다. 물론, 이런 정의역들은 새로운 지역의 더 큰 크기에 맞게 적당히 확대되어야 한다. 그리하여, 일치하는 정량화된 계수 \bar{s} 와 \bar{o} 에 따라 새로운 지역에 맞는 d 정의역 계수를 산출하는 새로운 집합을 보다 우수한 것으로 유지함으로써 $2d$ 코드북 블록들을 획득한다.

이것은 부모 구성이 어떻게 작은 지역을 가지고 자손 구성을 유도하는가를 설명한다. 모집단 각각의 구성에 대해 이런 과정의 σ 번 반복은 σN_b 자손을 산출한다. 이것들의 최고의 N_b 만이 유지되어지고 다음 세대 모집단을 형성한다. Fitness 표준은 프랙탈 영상 압축과 일치한다. N_b 에 잔존하는 구성은 적어도 콜라주 오류 $E(D, R)$ 과 함께 한 것이고 상응하는 분할의 모든 지역에 대해 합계되어진다.

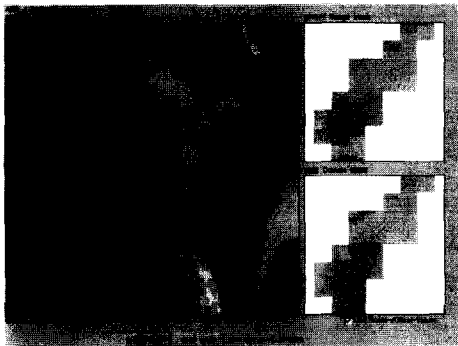
모집단 각각으로부터 가장 우수한 프랙탈 영상은 순서제어(flow control)에 맞게 쉽게 추출되어질 수 있다. 결국, 진화처리는 거대한 비트율과 작은 콜라주 오류를 가진 프랙탈 압축과 함께 시작한다. 각각의 세대는 작은 지역을 가진다. 그리하여 콜라주 오류가 상승하는 동안 비트율을 허락한다. 진화는 콜라주 오류에 대해 주어진 허용 한계값은 증가될 때나 요망되는 비트율이 성취되어질 때 정지된다.



(그림 1) LENA 원 영상



(그림 2) 1번 Iteration 처리



(그림 3) 2번 Iteration 처리

진화 처리에서 콜라주 오류 한계값과 마지막 비트올 때개변수는 모집단 크기 N_p 에서 각각의 치역에 저장된 코드북 지수의 멤버 d 이고 분기 인수 σ 이다.

무작위로 선택된 치역 블록들에 대해 위의 실험과 같이 정의역 블록에 대해 일치하는 매칭 코드를 찾을 수 있다. 반복 축소를 하면 코드와 기억 공간이 감소한다.

2. 본 론 - 진화 분할 비교

진화 분할 처리에서의 영상의 코드와 효율적으로 저장되어지는 가에 대해 4개의 알고리즘을 이용하여 비교 분석하였다(그림 4).

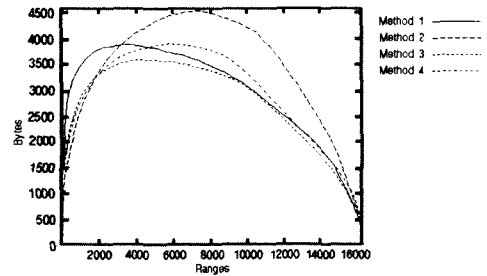
Methods 1. 원자 블록의 각각에 대해서 두 개의 비트가 오른쪽과 더 낮은 이웃들과 관련되었는지를 나타내도록 저장된다. 결과로서 나타나는 bit-stream은 바이트로 나타

내고, LZW-algorithm 사용으로 압축된다. 입력 영상(문자열)을 코드로 변환하는 변환 테이블에 입각하는 알고리즘이다.

이 방식은 입력 영상을 판독하여 이미 변환 테이블에 등록되어 있는 영상과 대조하여 압축하는 한편, 변환 테이블에 등록되어 있지 않는 영상을 변환 테이블에 등록해 나가는 방법으로서 동적으로 변환 테이블에 갱신되어 간다.

이 알고리즘의 단순성에도 불구하고 이런 방법은 다른 알고리즘과 잘 비교된다.

다음의 알고리즘들은 분할된 영상을 저장하기 위해 유도연쇄코드 Derivative Chain Code(DCC)를 사용한다. 그것들은 치역 경계 즉, 간선을 추적함으로써 진행된다.



(그림 4) 512×512 LENA 영상의 압축 분할 비교

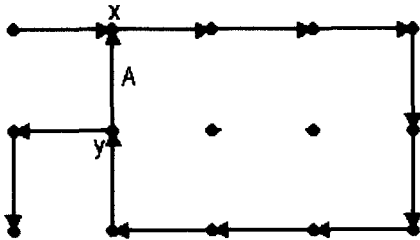
Method 2. 이 알고리즘의 절차는 치역 윤곽들이 완벽하게 기술되어질 때까지 반복 처리된다.

첫째, 치역 경계선에 방문하지 않는 점들이 선택되어지고 그것의 위치는 산출이고 스택에 더해진다. 그 스택이 비어 있지 않는 한 다음 단계는 반복되어진다. 그 스택은 나타나게 되고 주어진 위치에서 시작하면서 그 경계선은 이미 방문한 위치에 도달될 때까지 추적하게 된다. 모든 단계에서 다음에서 취할 수 있는 3가지 방향들(왼쪽으로 돌고, 앞으로 계속 가고, 오른쪽으로 돌고)을 보여주는 윤곽 분기점에서 우리는 첫 번째 분기를 취하고 나중에 읽을 수 있도록 스택에 다른 가능한 선행 세그먼트의 끝점 s 를 넣는다. 모든 단계에서 윤곽이 적어도 3가지 방향들 중에 하나로 유지해야만 하기 때문에 $2^3 - 1 = 7$ 상징들은 하나의 단계를 부호화하는 데

충분하다. 마침내, 결과로 나타나는 상징열은 부호화된 산술 엔트로피이다. 치역들이 방대하다면 이런 방법은 Method 1 보다 더 짧은 코드를 허락한다. 이것은 이 치역들의 내부에서 사용된 어떠한 비트도 존재하지 않기 때문에 기대되어진다.

이런 알고리즘은 다음 두 가지 방법들을 이끌어낸다.

Method 3. Method 2의 주요한 결함은 일부 선형 세그먼트들이 두 번 저장되는 것이다. 그러나 중복된 정보를 삭제함으로써 코드 크기를 더 줄일 수 있다.



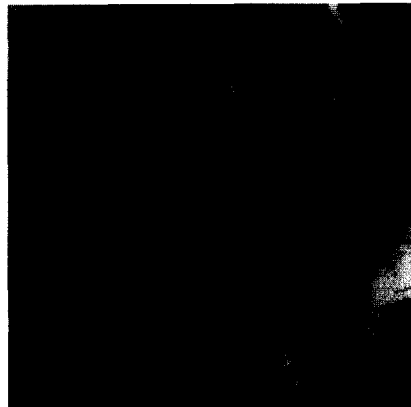
(그림 5) 치역 윤곽이 Method 2에서 압축되어진다면 매번 그 경로의 선형세그먼트들은 두 번 저장되어진다. 도형에서 세그먼트A는 두 번씩 저장되어진다. 첫 번째는 위치 X가 가로지를 때이며 두 번째는 Y가 방문되어질 때이다.

Method 4. Method 3에서처럼 bit-stream 사용하는 것 대신에 추적과정의 모든 단계에서 하나의 상징을 산출할 수 있다. 주어진 단계에 알려지지 않은 방향들 1, 2 또는 3이 존재하는지에 따라 다양한 상징 집합을 사용할 수 있다. 그러나 해독기가 기대되는 상징그룹(1,2,3)인 각각의 단계에서 알기 때문에 7개의 상징들은 Method 2에서처럼 충분하다. 산술 엔트로피 코딩의 결과로 나타난 상징열은 다시 마지막 산출을 만들어낸다.

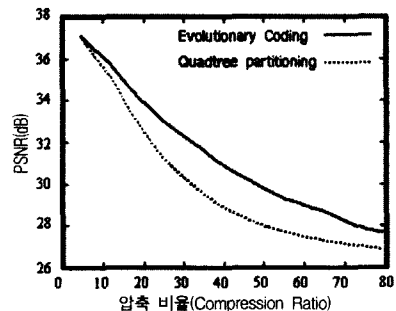
(그림 4)로부터 Method 3이나 Method 4가 모든 치역들에 대해 가장 짧은 코드를 산출할 것에 대해 명백히 알 수 있다. 실험에서 양쪽 코드가 계산되어지고 선택에 있어 복호기를 위해 부수적 비트를 쓰는 더 짧은 것을 저장한다.

3. 실험 결과

첫 번째 실험에서 512×512 LENA 영상에 대해 최적의 탐색 4분 트리를 적용하였다[9](그림 1, 2, 3). 두 번째 실험에서 256×256 LENA 영상에 대해 진화 알고리즘과 최적의 탐색 4분 트리[9]를 적용하였다(그림 6, 7). 그 처리 절차는 64×64×8 크기의 코드북의 충분한 탐색에 의해 획득된 크기 4×4픽셀의 원자 블록들을 사용하는 프랙탈 코드를 초기화하였다. 4분 트리를 가지고 우리는 4분 트리와 탐색의 단계마다 똑같은 크기의 코드북을 사용한다. 두 방법들의 수행은 기대된 것처럼 낮은 압축율(Compression Ratio)에서 비교된다. 그러나 압축율에서 우리는 PSNR의 여러 가지 dB(decibel units)에 이르는 방법에 있어서 증가한 것을 관찰할 수 있다.



(그림 6) 256×256 LENA 영상 500 Ranges 영상 분할 압축율(Compression ratio) : 70.5
원 영상의 Size : 2029 bytes
변형한 영상의 Size : 1691 bytes



(그림 7) 256×256 LENA 영상 매개변수 $N_p = d=10, \sigma=20$

원 영상에 있어서 사각형 영상 넓이 m pixel과 높이 n pixel 이라면 평균 절대 오차(Mean Absolute Error)는 다음과 같다.

$$MAE = \frac{1}{mn} \sum_{x=1}^m \sum_{y=1}^n |f(x, y) - g(x, y)|$$

여기서, $f(x, y)$ 는 원 영상이고,
 $g(x, y)$ 는 압축된 영상을 나타낸다.

영상의 유사도 측정에 사용되는 평균제곱오차(mean square error)는

$$MSE = \frac{1}{mn} \sum_{x=1}^m \sum_{y=1}^n [f(x, y) - g(x, y)]^2$$

$$RMSE = \sqrt{\frac{1}{mn} \sum_{x=1}^m \sum_{y=1}^n [f(x, y) - g(x, y)]^2}$$
 이다.

먼저 유사한 이미지(Domain)을 빨리 찾기 위하여 이미지의 부분적인 밝기의 평균 값을 기준으로 정의역들을 Domain Class로 구분하여 탐색 시간을 줄였으며, 탐색시 Domain과 Range 차이를 계산하기 위해 rms를 사용하였다. 이 rms의 값이 적을수록 영상간의 더 유사함을 나타낸다.

신호 대 잡음 비율(Signal to Noise Ratio)은

$$SNR_{rms} = \frac{\sum_{x=1}^m \sum_{y=1}^n f(x, y)^2}{\sqrt{\sum_{x=1}^m \sum_{y=1}^n [f(x, y) - g(x, y)]^2}}$$

$$SNR_{rms} = \left[\frac{\sum_{x=1}^m \sum_{y=1}^n f(x, y)^2}{\sum_{x=1}^m \sum_{y=1}^n [f(x, y) - g(x, y)]^2} \right]^{1/2}$$

$$SNR_{log} = 10 \log_{10} \left[\frac{\sum_{x=1}^m \sum_{y=1}^n f(x, y)^2}{\sum_{x=1}^m \sum_{y=1}^n [f(x, y) - g(x, y)]^2} \right]$$

peak signal to noise 의 비율은

$$\begin{aligned} SNR_{peak} &= 10 \log_{10} \left[\frac{\max[f(x, y)]^2}{\frac{1}{m} \frac{1}{n} \sum_{x=1}^m \sum_{y=1}^n [f(x, y) - g(x, y)]^2} \right] \\ &= 10 \log_{10} \left[\frac{[\max[f(x, y)] - \min[f(x, y)]]^2}{\frac{1}{m} \frac{1}{n} \sum_{x=1}^m \sum_{y=1}^n [f(x, y) - g(x, y)]^2} \right] \\ &= 10 \log_{10} \left[\frac{\text{maximum dynamic range}^2}{\frac{1}{m} \frac{1}{n} \sum_{x=1}^m \sum_{y=1}^n [f(x, y) - g(x, y)]^2} \right] \end{aligned}$$

$$\begin{aligned} PSNR &= 10 \log_{10} \left[\frac{255^2}{rmse^2} \right] \\ &= 20 \log_{10} \left[\frac{255}{rmse} \right] \end{aligned}$$

$$RMSE = \frac{255}{10 \left(\frac{PSNR}{20} \right)}$$

두 번째 구현에서 우리는 매개변수 N_p = 모집단의 크기, σ = 반복 횟수, d = 도메인 사이즈의 진화 알고리즘의 수행을 시험하였다. <표 1>과 <표 2>는 256×256 영상 LENNA의 결과치를 기록한 것이다. 매개변수의 분기요소는 σ 이다. $\sigma = 20$ 의 큰 값을 가지고 있고, 다른 두 개의 매개변수가 작을 때 가장 우수한 결과에 접근한 것을 알 수 있다. <표 1>은 실험적으로 비최적화된 압축 프로그램의 일치하는 계수 횟수의 일부를 보여주고 있다. 많은 시간이 탐색에 의해 초기 프랙탈 코드를 찾는데 소요되었다.

<표 1> 500 Ranges의 진화 계산 (Evolution Computation)

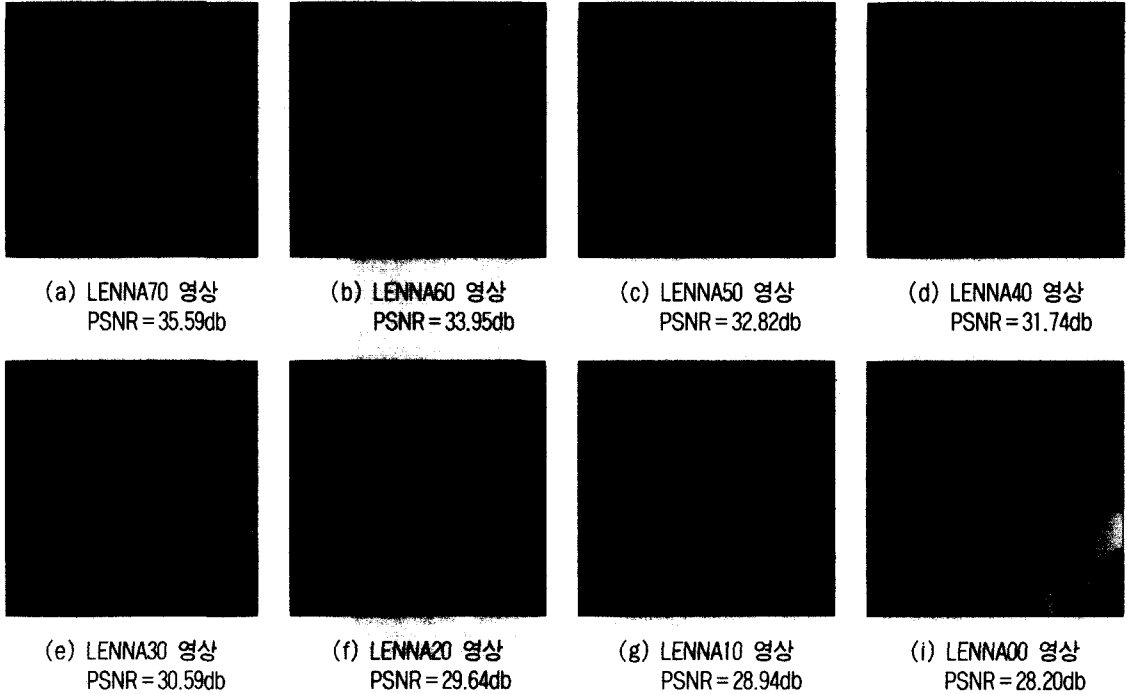
매개변수 (Parameters)			Times in sec	
N_p	σ	d	초기치 (Initialization)	진화 (Evolution)
5	5	5	1305	129
10	10	10	1305	458
20	20	20	1305	2473

<표 2> N_p = 모집단의 크기, σ = 반복 횟수, d = 도메인 사이즈

256×256 LENNA 영상의 PSNR (dB)									
	$N_p = 5$			$N_p = 10$			$N_p = 20$		
σ	$d = 5$	$d = 10$	$d = 20$	$d = 5$	$d = 10$	$d = 20$	$d = 5$	$d = 10$	$d = 20$
5	27.6	27.7	27.7	27.7	27.8	27.8	27.7	27.8	27.8
10	27.9	28.0	27.9	27.9	27.9	28.1	28.0	28.0	28.0
20	28.1	28.0	28.1	28.0	28.1	28.1	28.0	28.1	28.2

최적의 영상은 마지막 단계에서 탐색함으로써 더 향상되어질 것이다. 진화 압축기에 의해 생성된 분할에 기초를 둔 알고리즘은 최적의 치역-정의역 쌍들을 계산했다. 흥미롭게 초기의 코드를 넘어선 향상은 오히려 작아졌다.

3.1 실험에 의한 LENNA 영상



(그림 8) 압축된 영상의 256×256의 해상도와 256 밝기값을 가지는 LENNA 영상

4. 결 론

이 논문은 프랙탈 영상 압축에서 가장 중요한 문제들 중에서 영상 분할에 대한 문제를 풀기 위해 진화계산의 처리를 보여주었다. 최적의 지역 분할 작업은 이런 방향의 연구에 많은 영향을 줄 것이다.

- 초기설정은 전체 압축 시간의 큰 프랙탈 영상을 처리한다. 보다 빠른 초기설정의 결과가 시험되어야 한다.
- 진화적인 계산은 보다 단순한 것(지역블록 변수들을 이용함으로써)에 의한 결정에 바탕을 둔 오류를 교체함으로써 가속화될 수 있다.
- 오류에 바탕을 둔 Fitness 기준은 세그먼트 코드의 크기계산을 수용하는 Fitness 기준에 의해 교체할 수 있다. 이런 방법에서 우리는 엔트로피로 제한된 분할들(entropy-constrained partitions)를 소개한다.

- 자손 생성 구성에서 결합되기 위해 지역쌍을 선택하는 경쟁적 전략들의 집합으로 간주할지도 모른다. 다양한 통계학은 진화의 다양한 저장소에 적절하다는 것을 증명할 것이다.
- 원자블록들의 크기는 4×4 픽셀외의 다른 값들로 구성할 수 있다.
- 프랙탈 압축의 변형에 대한 매개변수의 엔트로피 코딩은 비트율을 훨씬 더 향상할지도 모른다.
- 마침내 암호처리의 각각의 단계에서 확률적 선택 대신 합병하도록 지역 쌍을 결정적 최적 지역쌍으로 계산할 수 있다. 즉 지역의 합병은 전반적인 오류의 가장 낮은 증가를 산출한다. 사실 첫 번째 구현은 이런 알고리즘이 결과로서 나타나는 영상 자질의 하락없이 속도에 관하여 비결정론적인 것을 수행하는 것을 제시한다.

향후, 여러 가지 이런 확장들은 앞으로의 논문에서 연구되어질 것이다.

참 고 문 헌

- [1] Jacquin, A. E., *Image coding based on a farctal theory of iterated contractive image transformations*, IEEE Trans. Image Proc. 1 (1992) pp.18-30.
- [2] Bedford, T., Dekking, F. M., Keane, M. S., *Fractal image coding techniques and contraction operation operators*, Nieuw Arch. Wisk. (4) 10,3 (1992) pp.185-218.
- [3] Jacobs, E. W., Fisher, Y., Boss, R. D., *Image compression : A study of the iterated transform method*, Signal Processing 29 (1992) pp.251-263.
- [4] Fisher, Y., Menlove, S., *Fractal encoding with HV partitions*, in [9].
- [5] Davoine, F., Antonini, M., Chassery, J.-M., Barlaud, M., *Fractal image compression based on Delaunary triangulation and vector quantization*, IEEE Trans. Image Proc. 5,2 (1996) pp.338-346.
- [6] Novak, M., *Attractor coding of images*, Licentiate Dissertation, Dept. of Electrical Engineering, Linköping University, May 1993.
- [7] Reusens, E., *Partitioning complexity issue for iterated function systems based image coding*, in : *Proceedings of the VIIth European Signal Processing Conference EUSIPCO'94*, Edinburgh, Sept. 1994.
- [8] Thomas, L., Deravi, F., *Region-based fractal image Compression using heuristic search*, IEEE Trans. Image Proc. 4,6 (1995) pp.832-838.
- [9] Fisher, Y. (ed.), *Fractal Image Compression-Theory and Application*, Springer-Verlag, New York, 1994.
- [10] Saupe, D., Hartenstein, H., *Lossless acceleration of fractal image compression by fast convolution*, Proc. 1996 Intern. Conf. Image Proc (ICIP), Lausanne, Sept. 1996



유 환 영

e-mail : wyoo@first.suncheon.ac.kr
 1981년 전남대학교 재료공학과 졸업 (학사)
 1985년 광주대학교 전자계산학과 졸업(이학석사)
 1997년~현재 조선대학교 전산통계학과 박사과정

1983년~현재 순천제일대학 컴퓨터학과 교수
 관심분야 : 알고리즘, 영상처리, 멀티미디어 등



최 봉 한

e-mail : cbh301@webmail.shinbiri.com
 1995년 호원대학교 전자계산학과 졸업(학사)
 1997년 조선대학교 전산통계학과 졸업(이학석사)
 2000년 조선대학교 전산통계학과 수료(이학박사)

1998년~현재 순천제일대학 컴퓨터학과 교수
 관심분야 : 영상 압축, 영상 처리, 유전자 알고리즘 등