

Bath Fractal 변환에 의한 영상압축 기법의 성능 분석 (Performance Analysis of Bath Fractal Image Compression)

강현철 문영식
한양대학교 전자계산학과

요 약

본 논문은 고속의 프랙탈 영상압축 기법으로 알려져 있는 Bath 프랙탈 영상 압축 기법의 성능을 여러 가지 측면에서 분석한다. Bath 프랙탈 영상 압축 기법은 영상의 빠른 복호화가 가능하므로 미래의 다양한 형태로 요구되는 정보서비스, VOD(Video On Demand), CD-ROM등과 같이 저장되어 있는 영상 정보의 빠른 복원이 요구되는 곳에 적합한 부호화 기술이므로 그 성능에 대한 분석이 중요하다. 본 논문에서는 Bath 프랙탈 압축 기법의 양자화 방법에 따른 성능 분석, 프랙탈 계수 값의 분포에 따른 성능 평가, 사용된 어떤 맵핑식에 따른 성능 비교, 영상내의 에지 빈도수에 따른 성능 변화, 쿼드트리 구조의 작은 블록들에 대한 BFT의 성능 평가등을 고찰한다.

1. 서론

영상정보의 저장이나 전송을 위해서는 데이터 압축이 필수적이고, 이런 이유에서 지금까지 주로 변환 부호화 기법에 대한 활발한 연구가 진행되어 왔다. 변환 부호화 기법은 영상 데이터가 가지고 있는 신호의 시간 영역과 공간 영역의 중복성을 줄이는 기법으로 많은 데이터 압축을 할 수 있다. 최근 들어서는 변환 부호화 기법의 일종인 DCT(Discrete Cosine Transform)를 근간으로 하는 압축방법이 영상정보 압축기법의 표준안으로 채택되었는데, 정지 영상 압축 기법인 JPEG(Joint Photographic Experts Group), 영상회의나 화상통신을 위한 영상 압축 기법인 CCITT H.261, 그리고 동영상 압축 기법인 MPEG(Moving Picture Experts Group)이 그것이다. 그러나 이 변환 부호화 기법은 일정한 압축비 이상에서는 복원영상의 저질화가 심하게 나타나며 위에서 설명한 DCT 기반의 표준안들은 높은 압축률에서 블록징 효과(blocking effect)가 현저하게 나타나는 문제점이 있다.

본 논문에서 고찰하고자 하는 프랙탈 영상 압축 기법은 1990년 전후로 그 연구가 활발히 진행된 영상 압축 기법이다. 1988년 Barnsley에 의해 디지털 영상 압축에 이 방법이 소개되었고[1], 그 후 Jacquin, Fisher 등에 의해 발전되었다[2], [3], [4]. 이 압축법은 높은 압축률에서도 복원영상의 화질을 일정 수준 이상으로 유지할 수 있는 부호화 알고리즘으로 평가되고 있다. 프랙탈 영상 압축 기법은 높은 압축률 때문에 영상신호 부호화의 표준화 기구인 MPEG-4에서 미래의 표준 부호화 방법중의 하나로 검토되고 있다. 프랙탈 영상 압축 기법은 기존의 고압축 기법과 달리 복원 과정이 빠르고 간단하여 미래의 다양한 형태로 요구되는 정보서비스, VOD(Video On Demand), CD-ROM등과 같이 저장되어 있는 영상 정보의 빠른 복원이 요구되는 곳에 적합한 부호화 기술로 주목받고 있다.

프랙탈 영상 압축 기법이란 영상 데이터를 전체와 부분간의 자기 닮음성(self similarity)을 고려한 수축변환의 변

환 계수(parameter)들만으로 표현함으로써 정보량을 줄이고자 하는 방법이다. 일반적으로 프랙탈 영상 압축의 부호화 과정은 압축할 영상을 고정점(fixed point)으로 하는 수축변환을 구하는 것이다. 한편 복호화 과정은 콜라주 이론에 근거하여 압축시 구한 수축변환을 임의의 초기 영상에 반복적으로 적용함으로써 얻어지게 된다. 전체와 부분간의 자기 닮음성이 강한 이진 영상의 경우는 자기 닮음성을 만들어 내기가 쉽고 적은 수의 계수 값을 사용하여 영상을 표현할 수 있지만, 자연 세계에 존재하는 영상들은 자기 닮음성이 그리 강하지 못하며 닮음성이 있다고 하더라도 그 닮음성을 찾아내기가 쉽지 않다. 또한 색정보를 가지고 있으므로 단순한 형태의 수축 변환식으로는 표현하기 어렵다. 그렇기 때문에 초기의 프랙탈 영상 압축 기법을 제안한 Jacquin이나 Fisher의 방법들은 부호화 과정에서 자기 닮음성을 찾기 위해 많은 탐색과정이 필요하여 부호화 속도에서 문제가 있었다. 한편 Jacquin이나 Fisher가 제안한 방법의 문제점들을 극복하기 위해서 Monro, Dudbridge 등은 이론적 연구와 함께 개선된 알고리즘을 발표하였는데, 이들의 알고리즘을 보통 Bath 프랙탈 변환(BFT: Bath Fractal Transform)이라고 한다.

본 논문에서는 Monro와 Dudbridge 등에 의해 제안되었던 BFT 알고리즘을 여러 가지 방법을 통해 성능을 분석한다. 성능 분석을 위한 평가 척도로는 양자화 방법에 따른 성능 분석, 프랙탈 계수 값의 분포에 따른 성능 평가, 사용된 어떤 맵핑식에 따른 성능 비교, 영상내의 에지 빈도수에 따른 성능 변화, 쿼드트리 구조의 작은 블록들에 대한 BFT의 성능 평가를 사용한다.

2. BFT

2.1 고전적 방법의 문제점

기존의 프랙탈 영상 압축법의 주류를 이루었던 Fisher나 Jacquin의 방법은 IFS(Iterated Function System)를 작

성하기 위한 핵심 과정이 지역 블록과 닮은 정의역 블록을 탐색하는 과정이다. 이 방법들은 1차원의 선형방정식으로 정의역 블록과 지역 블록사이의 근사화를 위해 그레이 레벨을 조정하는 방법을 사용하고 있다. 근사화가 잘 이루어지기 위해서는 두 블록의 그레이 레벨 분포가 거의 같은 형태를 가지고 있어야 한다. 그러므로 각 지역 블록의 그레이 레벨 분포가 가장 유사한 형태를 가지는 정의역 블록을 찾기 위한 탐색과정이 요구된다. 탐색해야하는 정의역 블록의 계수는 두 방법에서 약간의 차이는 있지만 고가의 컴퓨터 시간을 요구하게된다.

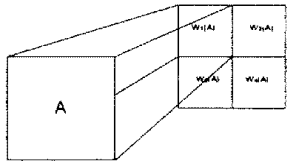
부호화과정에서 소요되는 시간은 Alkhansari, Lepsoy, Zhang, Kawamata 등에 의해 블록 탐색의 방법을 달리하는 방법으로 개선시켰다[5], [6], [7], [8]. 그러나 이런 방법들 역시 부호화과정에서 후보 블록을 찾기 위해 탐색 시간을 요구하기 때문에 소요 시간의 근본적인 해결방법은 될 수 없었다.

2.2 SAS(Self Affine System)에 기초한 프랙탈 영상 압축 기법

이러한 문제점을 해결하기 위해 Monro, Dudbridge등은 압축성 탐색과정을 필요로 하지 않으면서 자기 압축성을 찾을 수 있는 방법을 구현하였다[9], [10], [11]. 즉, Fisher 나 Jacquie이 제안했던 그레이 레벨 조정식을 식 2.1과 같이 n 차의 다항식을 사용함으로써 지역 블록의 근사화 과정에서 정의역 블록을 탐색하는 과정을 제거할 수 있게 하였다. 이와 같은 방법을 자기 어떤 시스템(SAS: Self Affine System) 이라고 한다. 자기 어떤 시스템은 그림 2.1과 같이 하나의 상위 블록(정의역 블록) A를 4개의 작은 하위 블록(지역 블록)으로 나누고, 상위 블록을 변형시켜 하위의 블록에 근사화시키는 방법이고, 여기서 $W_k(A)$ 는 수축변환을 나타낸다. 식 2.1은 Monro가 제안한 2차 접근식의 예이다.

$$v_i(x, y) = a_1^{(i)}x^2 + a_2^{(i)}y^2 + a_1^{(i)}x + a_2^{(i)}y + a_3^{(i)}f(x, y) + b^{(i)} \dots\dots\dots(\text{식 2.1})$$

여기서, f 는 원영상이고 v_i 는 f 의 SAS을 만들어 내기 위한 어떤 맵핑이다. (x, y) 는 공간좌표이고, 6개의 계수들이 부호화에 사용될 데이터이다.



<그림 2.1> Self Affine System

이 방법은 탐색과정이 전혀 없기 때문에 부호화시간이 대단히 빠르지만 경계면이나 복잡한 영역을 표현하는데 문제가 많다. 이 방법은 빠른 부호화 때문에 동영상의 실시간 전송에 유용하지만[12], [13], 정지영상의 정밀한 묘사에는 효과가 떨어진다. 한편, Monro는 정확한 영상 복원을 위해 이 방법에 탐색 과정을 추가하여 복잡한 영상을 복원하는데 적합한 알고리즘을 제안하였고[14], [15], 압축률을 높이기 위해 쿼드트리구조를 적용하기도 하였다[13].

3. BFT의 성능분석

영상의 부호화에 사용되는 내용물은 단지 각 블록에 대해서 식 2.1에서 구해진 프랙탈 계수들뿐이다. 그러나 이 프랙탈 계수들은 실수 값을 가지므로 저장하기 위해 양자화가 필요하다. 각 계수들의 양자화(quantization)를 위해 Lloyd-Max 양자화기를 사용하며, 양자화된 계수들은 블록 순서대로 저장되게 된다. 각 계수들은 m 비트를 사용하는 2^m 개의 심벌로 표현된다. 또한 f 에 대한 SAS를 유추하기 위해 V 를 구성하는 모든 v_k 를 평탄(flat) 접근이나 1차(bilinear) 접근이나 2차(biquadratic) 접근 방식으로 어떤 맵핑(affine mapping)을 구성한다. 다음 식 3.1은 각 접근식이다.

$$\begin{aligned} v_k(x, y, z) &= a_3^{(k)}z + b^{(k)} \\ v_k(x, y, z) &= a_1^{(k)}x + a_2^{(k)}y + a_3^{(k)}z + b^{(k)} \\ v_k(x, y, z) &= a_1^{(k)}x^2 + a_2^{(k)}y^2 + a_1^{(k)}x + a_2^{(k)}y + a_3^{(k)}z + b^{(k)} \end{aligned} \dots\dots\dots(\text{식 3.1})$$

또한, 양자화된 계수들은 계수들의 중복성(redundancy)을 줄이기 위해 허프만 무호화기(Huffman Coder)와 같은 무손실 부호화기(lossless coder)를 사용하여 압축의 효율을 극대화한다.

3.1 양자화 방법에 따른 성능 평가

3.1.1 일정한 크기의 블록을 가지는 BFT

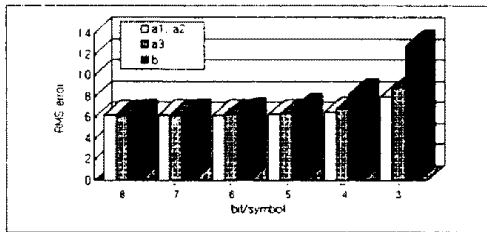
심벌을 표현하는 m 값에 대한 BFT의 성능 평가가 Monro와 Wolley에 의해 연구되었는데[11], 그들은 할당 m 값에 따라 High, Medium, Low, Poor의 단계로 나누어 그때의 RMSE(Root Mean Square Error)변화를 비교하였다. 각 블록내의 계수들은 차등 부호화기(differential coder)와 허프만 부호화기(Huffman coder)에 의해 최종 부호화 과정을 마치게 된다. 이 알고리즘을 사용해 512x512 크기의 "Gold Hill" 그레이 영상과 "Lenna" 그레이 영상을 압축했을 때 다음 표 3.1과 같은 결과를 보인다. 표 3.1은 1차 접근식을 사용했을 때의 각 복원된 영상의 PSNR(Peak-to-peak Signal to Noise Ratio)값과 대응하는 화소당 할당 비트 수(bit/pixel)를 보여준다.

<표 3.1> 각 양자화 방법에 따른 "Gold Hill"과 "Lenna" 영상의 PSNR과 bit/pixel의 변화정도

양자화 방법		양자화 레벨(bit)				1차 접근	
		a_1	a_2	a_3	b	PSNR	bpp
Gold Hill	양자화 하지 않음					31.89	
	High	5	5	6	6	31.00	0.79
	Medium	4	4	5	5	29.84	0.67
	Low	3	3	4	4	27.80	0.56
	Poor	2	2	3	3	24.33	0.43
Lenna	양자화 하지 않음					34.63	
	High	5	5	6	6	33.71	0.77
	Medium	4	4	5	5	32.27	0.65
	Low	3	3	4	4	29.58	0.56
	Poor	2	2	3	3	25.68	0.42

3.1.2 쿼드트리 구조를 가지는 BFT

본 논문에서는 각 레벨당 성능의 비교분석을 뚜렷이 하기 위해 기본적으로 5단계의 쿼드트리 레벨을 사용하며, 일반적인 쿼드트리 부호화기의 방법처럼 본 논문에서도 각 레벨에 차별적인 비트 할당을 사용하였다. 이것은 큰 영역을 차지하는 상위 레벨의 큰 블록들에게는 많은 비트를 할당하고 반대로 하위 레벨의 작은 블록에는 적은 비트를 할당하는 방법이다. 본 논문에서는 다음 표 3.2와 같은 방법으로 각 레벨의 블록에 대한 프랙탈 계수 $a_1^{(k)}$, $a_2^{(k)}$, $a_3^{(k)}$ 와 $b^{(k)}$ 에 차별적으로 비트를 할당한다. 이 방법은 각 계수에 할당되는 비트수에 따른 복원영상의 화질에 대한 Monroe 등의 연구에 기초한다[11]. 그림 3.1은 1차 접근식을 사용했을 때 BFT 알고리즘의 각 계수들에 대한 양자화 레벨에 따른 복원영상의 화질저하 정도를 보여준다. 특히 $b^{(k)}$ 의 값은 5비트 이하가 되면 현저하게 화질이 저하되며, 이런 현상은 4비트 이하의 $a_3^{(k)}$ 에서도 나타난다. 이와 같이 복원 화질을 척도로 사용하여 표 3.2와 같이 차별적으로 할당했을 때, 좋은 결과를 얻을 수 있음을 실험적으로 확인했다.



<그림 3.1> BFT 알고리즘의 각 계수들에 대한 양자화 레벨에 따른 복원영상의 화질 변화

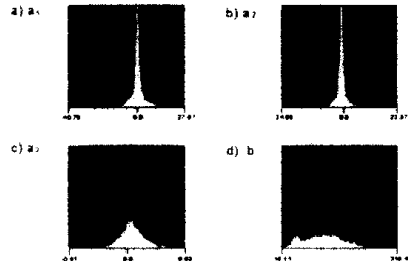
<표 3.2> 각 블록에 대한 할당 비트

블록의 크기	할당 비트수			
	$a_1^{(k)}$	$a_2^{(k)}$	$a_3^{(k)}$	$b^{(k)}$
32×32	5	5	6	6
16×16	5	5	6	6
8×8	4	4	5	6
4×4	4	4	5	6
2×2	3	3	4	5

3.2 프랙탈 계수 값의 분포에 따른 분석

각 블록에 대한 프랙탈 계수들은 서로 다른 분포를 보이며, 그 경계(boundary) 또한 서로 다르다. 프랙탈 계수 $a_1^{(k)}$, $a_2^{(k)}$, $a_3^{(k)}$ 와 $b^{(k)}$ 의 분포도는 그림 3.2와 같다. 이 분포도는 256×256 크기의 "Lenna" 영상을 1차 접근 방식으로 부호화했을 때 나타나는 각 프랙탈 계수들의 분포도이다. 분포의 모양이나 경계의 차이가 각각 심각하게 나타남을 알 수 있다.

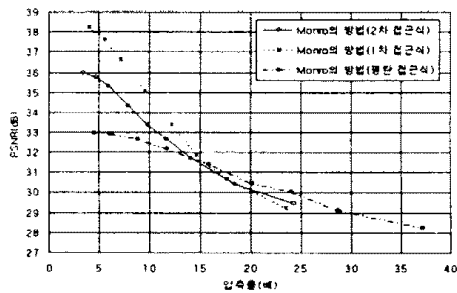
그러므로, $a_1^{(k)}$, $a_2^{(k)}$, $a_3^{(k)}$ 와 $b^{(k)}$ 각각에 대해 다른 형태의 Lloyd-Max 양자화기를 사용해야 한다. $a_1^{(k)}$ 과 $a_2^{(k)}$ 는 Laplacian 분포형태의 Lloyd-Max 양자화기를 사용하며, $a_3^{(k)}$ 은 Laplacian 분포형태나 Gaussian 분포형태, $b^{(k)}$ 는 Gaussian 분포형태나 평탄(flat) 분포형태의 Lloyd-Max 양자화기를 사용했을 때 가장 좋은 결과를 얻을 수 있다.



<그림 3.2> 계수 값들의 분포도

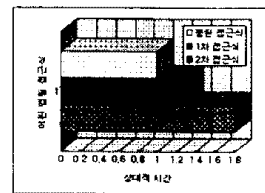
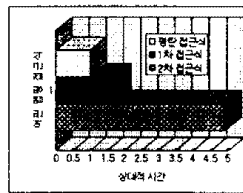
3.3 사용한 어떤 맵핑식에 따른 성능 비교

SAS를 유추해 내기 위해 사용된 어떤 맵핑식에 따른 압축성능의 차이는 그림 3.3(512×512 Lenna)과 같다. 실험에 사용한 어떤 맵핑식으로는 평탄, 1차, 2차 접근식을 사용했고, 5개 레벨의 쿼드트리 구조를 가지는 BFT를 사용했다. 각 성능의 차이는 접근식이 복잡영역을 얼마나 잘 만들어 내는가에 따라 차이가 생기게 된다. 예를 들어 낮은 압축률에서는 대부분의 블록이 쿼드트리 하위 레벨에 주로 나타나게 되며, 이 부분들은 대부분 복잡영역이 된다. 그러므로 비록 많은 비트를 할당하지만 복잡영역 표현에 강한 2차 접근식이 우수한 성능을 가진다.



<그림 3.3> 어떤 맵핑식에 따른 성능 비교

부호화 시간과 복호화 시간은 당연히 많은 차수의 계산을 요하는 2차 접근식이 상대적으로 많이 걸리게 된다. 그림 3.4와 그림 3.5는 평탄접근식의 부호화·복호화 시간을 1로 보았을 때 나머지 방법들의 상대적인 부호화·복호화 시간을 나타낸다.



<그림 3.4> 부호화 시간 <그림 3.5> 복호화 시간

3.4 영상내의 에지영역의 빈도에 따른 성능 평가

이 부호화기는 같은 영상에 대해서 영상의 크기가 증가할수록 더 높은 PSNR 값을 가지는 복원영상을 만들어 내는데, 그 이유는 이 부호화기가 에지 영역에 대해 근사치를 잘 만들어내지 못하기 때문이고, 같은 블록 크기 내에 상대적으로 에지가 적게 나타나는 큰 영상에 대해서는 복원력이 높게 된다. 즉 256×256 크기의 "Gold Hill" 영상과 512×512 크기의 "Gold Hill" 영상의 복원된 PSNR 값은 표 3.3과 같이 후자의 영상의 경우가 더 좋게 나타난다. 표 3.3

은 1차 접근법을 사용하여 Medium 방법으로 양자화했을 때의 영상에 대한 PSNR값의 변화 정도를 보여준다.

<표 3.3> 영상의 크기에 따른 BFT의 성능 변화

Image	256×256		512×512	
	PSNR	bpp	PSNR	bpp
Gold Hill	27.14	0.67	29.84	0.67
Lenna	27.45	0.66	32.27	0.65

3.5 2×2 블록에 대한 BFT의 성능 분석

압축률의 문제 때문에 2×2 크기의 블록에 대해서는 많은 비트를 할당할 수 없다. 그런데 대부분의 2×2 블록들은 복원된 화질에 문제가 있어서 큰 블록에서 부호화되지 못한 에지 영역이다. 한편 BFT는 에지영역의 부호화에 문제가 있으므로 2×2 블록들은 BFT로 복원이 힘든 에지영역이며, 압축률문제 때문에 많은 비트를 할당할 수 없다. 결과적으로, 제안된 트리구조의 2×2 블록에서는 현저하게 복원 영상의 화질저하가 발생한다. 표 3.4는 복원영상과 원영상을 비교하여 분리과정을 수행한 256×256 크기의 "Lenna" 영상의 각 크기의 블록 개수, 해당 블록내의 PSNR 값, 그리고 해당 블록들이 전체영상에서 차지하는 비율을 나타낸다. 표 3.4에서 볼 수 있듯이 2×2 블록이 전체 영상에서 차지하는 비율이 24.1% 임에도 불구하고 다른 블록에 비해 PSNR 값이 현저하게 낮아짐을 볼 수 있으며, 이 때문에 전체 PSNR도 낮아지게 된다.

<표 3.4> 전 블록을 BFT로 부호화한 경우 각 블록에 대한 PSNR

블록의 크기	개수	전체 영상에서 차지하는 비율	해당 블록의 PSNR	전체 PSNR	bpp
32×32	8	12.5%	29.5	30.05	1.09
16×16	57	22.3%	30.2		
8×8	247	24.1%	30.78		
4×4	698	17.0%	31.56		
2×2	3944	24.1%	28.8		

4. 결론 및 향후 연구방향

본 논문에서는 BFT를 여러 측면에서 분석해 보았다. 우선 일정한 크기의 블록을 가지는 BFT와 쿼드트리 구조를 가지는 BFT로 나누어 성능을 비교하였다. 또한, 양자화 방법에 따른 성능 분석, 프랙탈 계수 값의 분포에 따른 성능 평가, 사용된 어떤 맵핑식에 따른 성능 비교, 영상내의 에지 빈도수에 따른 성능 변화, 쿼드트리 구조의 작은 블록들에 대한 BFT의 성능 평가 등을 고찰해 보았다.

BFT는 대단히 빠른 압축 알고리즘인데 비해 본문에서 설명했듯이 블록내의 에지를 제대로 복원해낼 수 없는 단점을 가지고 있으며, 이를 해결하기 위해 쿼드트리 구조를 적용한다. 그러나, 이와 같이 블록 내 화소값의 분산값에 기초로한 쿼드트리 구조는 비록 블록의 분산값은 크더라도 BFT 알고리즘에 의해 충분히 복원될 수 있는 경우에도 작은 서브블록으로 분리를 하게 된다. 반대로 BFT에 의해 복원이 잘 되지 못하는 블록인데도 분산값이 작다는 이유로 서브블록으로 분리하지 않을 수도 있다. 이것은 특히 복잡영역을 잘 복원해내지 못하는 BFT에게는 치명적인 약점이 될 수 있으며, 복원영상의 질을 떨어뜨리는 원인이 된다. 이런 이유에서 블록의 분산값을 기초로한 쿼드트리 구조는 기존의 BFT 알고리즘을 효율적으로 개선시키지 못하

고 있다. 향후에는 이와 같은 비효율적인 쿼드트리 구조가 아닌 BFT의 특성을 잘 살릴 수 있는 영상 분할구조에 대한 연구가 필요하겠다.

참고문헌

- [1] F. Barnsley, *Fractals Everywhere*, AP Professional, 1988
- [2] A. E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Trans. Image Process.*, vol. 1, no. 1, pp.18-30, Jan.1992.
- [3] A. E. Jacquin, "Fractal Image Coding: A Review," *PROCEEDINGS OF THE IEEE*, vol. 81, no. 10, OCTOBER 1993.
- [4] Y. Fisher, *Fractal Image Compression*, siggraph, 1992 Course Notes.
- [5] M. G. Alkhansari, T.S. Huang, "A fractal-based image coding algorithm," *ICASSP 93*, vol. 5, pp. 345-348, Apr. 1993
- [6] S. Lepsoy, G. E. Oein, A. Ramstad, "Attractor image compression with a fast non-iterative decoding algorithm," *ICASSP 93*, vol. 5, pp.337, Apr. 1993.
- [7] Zhengbing Zhang, Yaoting Zhu, Guangxi Zhu, Hanqiang Cao, Donghui Xue, "Hybrid fractal image coding method," *VCIP 96*, vol. 2727, pp. 1360-1366, 1996.
- [8] Masayuki Kawamata, Masaki Nagahisa, Tatsuo Higuchi, "Multi-resolution tree search for iterated transformation theory-based coding," *ICIP 94*, vol. 3, pp. 137-141, Nov. 1994.
- [9] D. M. Monro, F. D. Dudbridge, "Fractal approximation of image blocks," *ICASSP 92*, vol. 3, pp. 485-488, 1992.
- [10] F. Dudbridge, "Image approximation by self affine fractals," *Ph.D. Thesis*, University of London, 1992.
- [11] D. M. Monro, Wolley, "Fractal image compression without searching," *ICASSP 94*, vol. 5, pp. 557-560, 1994.
- [12] D. L. Wilson, F. D. Dudbridge, D. M. Monro, "Rate buffered fractal video," *ICASSP 94*, vol. 5, pp. 505-509, 1994.
- [13] D. M. Monro, J. A. Nicholls, "Low bit late colour fractal video," *ICIP 95*, vol. 3, pp. 264-267, 1995.
- [14] D. M. Monro, "A hybrid fractal transform," *ICASSP 93*, vol. 5, pp. 169-172, 1993.
- [15] S. J. Wolley, D. M. Monro, "Optimum parameters for hybrid fractal image coding," *ICASSP 95*, pp. 2571-2574, 1995.