프랙탈 차원의 보다 정확한 계산

*김종구, 함도용, 남현우, 김하진

아주대학교 대학원 컴퓨터공학과

The Preciser Estimation of the Fractal Dimension

Jongkoo Kim, Do Yong Hahm, Hyeonwoo Nam, Ha-Jine Kimn

Department of Computer Engineering, Ajou University Graduate School

요 약

여러 다양한 프랙탈 구조의 차원을 측정하는 개량된 프랙탈 차원 측정 방법을 제안 하였다. 기존의 box counting 방법은 사용상의 편리성은 있으나 측정에 사용되는 데이터에 의존적이어서 기존 box counting 방법의 약점을 보완, 개량한 방법의 적용으로 프랙탈 차원의 보다 정확한 측정 결과를 얻었다

1. 서론

프랙탈 그래픽의 기본개념인 IFS(Iterated Function System)를 이용하여 이미지의 구성 정보를 알아 내려는 시도는 특히 이미지 처리의 경우 그 효율성 및 생산성 측면에서 효과를 충분히 기대할 수 있다[3]. 이에 따라 많은 연구가 국내외에서 이루어지고 있다 프랙탈의 개념을 이용하는 경우 이미지의 내부에 서로 유사한 부분이 있다는 가정 하에 정보를 얻을 수 있으며, 기본적으로 자기 유사성을 이용하게되어 기존의 방법보다 적은 메모리를 사용하며 이미지의 재생의 경우 그 배율에 관계없이 확대 및 축소가 가능하다는 장점이 있다[2]. 여러 가지 다양하고 복잡한 이미지의 처리에 있어서 원래의 이미지에 대한 충실도를 유지하면서 동시에 압축상의 효율성까지도 고려한 압축 기법에 이러한 프랙탈을 이용한 여러 시도들 중 Barnsley 는 원래의 이미지에 근사하는 프랙탈 이미지를 직접 생성하는 알고리즘을 제안하였다[3,4]. 간단한 프랙탈 알고리즘이 아주 복잡한 이미지를 생성할 수 있기 때문에, Barnsley 는 이미지의 저장을 위하여 알고리즘과 관계되는 변수들만 저장하여 저장 메모리를 감소시키는 것을 제안하였다 [6] 일반적으로 이미지의 정확한 정보에 근거한 효율적인 처리 기법은 향상된 전송 결과와 충실한 재생을 허용한다. 그러나 이미지의 경우, 프랙탈이 아닌 일반 이미지에 이러한 프랙탈에 기초한 알고리즘을 적용하는 경우 작은 오차를 가지는 이미지의 처리는 어려운 일이다 효율적인 프랙탈 이미지 처리 알고리즘과 그것들의 변수를 다루는 것이 프랙탈 이미지 처리에 있어서의 inverse 문제이며 이러한 inverse 문제의 해결을 위하여 A Jacquin 은 Barnsley 와 함께 적은 개수의 변수로 표현될 수 있는 함수들의 집합으로서 piecewise affine 변환을 연구하였다[5].

2. 프랙탈 이미지 분류

프랙탈 이미지 처리에 적용되는 IFS의 개념은 임의의 다양한 이미지의 집합이 어떠한 모양의 변환들의 집합으로도 재생이 가능하며 원 이미지보다 더 작은 크기의 메모리로도 생성 이미지의 저장이 가능하다는 것이다 이미지 처리의 경우에 프랙탈의 개념을 적용하면 기본

적으로 다음과 같은 식이 사용 된다[1,2].

$$T_i(D_k) = P(R_i) + sO \circ G(D_k)$$

이 때, 원래의 이미지 $P(R_i)$와 변환식 $sO \circ G(D_k)$의 사용에서 분류에 필요한 이미지 요소의 블록 개수가 결정된다. 따라서 프랙탈 기법을 이러한 이미지의 처리에 이용하는 경우 위의 식으로 정보의 검색을 위한 블록의 개수와 각 블록의 비교 횟수를 줄여 이미지 처리시의 처리 시간을 단축시킬 수 있으며 블록의 비교를 하는 경우 본 연구에서는 이미지의 밝기를 이용한다. 이러한 방법으로 이미지를 처리하는 경우 전체를 하나의 구성 요소로 단순히 검색, 평균 밝기를 이용하는 기존의 처리 방법 대신에 제안하는 방법을 이용하는 경우 프랙탈 기법의 적용과 정확한 분류 기준으로 변환식에 의한 분류 정보가 많아져 목적하는 바를 얻을 수 있다. 그러나 처리를 위한 이미지 요소의 분류를 위해서는 각 이미지의 비교 블록 개수에 대한 최적화가 필요하다. 이 과정에서 일반적으로 wavelet 방법 또는 각 비교 블록에 가중치(weight value)를 주어 분류하게 되나 본 논문에서는 이러한 블록의 분류에 프랙탈 차원을 이용하여 두 이미지 사이에서 서로 같은 차원의 크기를 갖는 요소들의 블록에 대해서만 유사성을 측정한다. 결과적으로 검색에 필요한 블록 개수의 감소가 이루어지게 된다. 프랙탈 차원의 보다 정확한 값을 얻는다면 이를 이용하여 보다 세분화된 분류 기준에 따라 보다 정보의 처리가 가능하게 된다. 따라서 프랙탈 차원의 개념을 이용하여 이미지들의 정보를 처리하는 경우 프랙탈 차원의 정확한 계산이 필요하게 된다

3. 프랙탈 차원

본 연구에서 제안하는 방법은 Barnsley 등이 제안한 프랙탈의 개념을 이용하여, 압축은 하되 이미지의 처리전의 영역과 처리후의 영역을 비교하여 이미지의 품질을 높이기 위해 기존의 분류 방법을 적용하지 않고 밝기의 변화에 따른 프랙탈의 차원 개념을 적용하는 것이다. 프랙탈 차원을 계산에서 일반적으로 쓰이는 방법은 그 간편성 때문에 Box counting 방법을 많이 사용한다[2]. 그러나 기존의 일반적인 box counting 방법은 Hausdorff 차원을 이용한 방법으로 단순히 그 개념을 적용한 결과 실제 계산시의 계산 결과는 측정하는 이미지의 요소 크기에 따라 각각 달라지게 되며 결과적으로 유사 요소로 이루어진 블록들의 분류시 분류 그룹의 개수에 많은 영향을 미치게 된다. 따라서 보다 정확한 프랙탈 차원 계산 방법이 요구된다

기존의 Hausdorff 차원은 수치적인 계산으로는 정확한 차원의 값을 구하는 것이 실제 적용이 어려우므로 기존 Hausdorff 차원의 실제 계산상의 문제와 이에 따른 부정확성을 보완하여 차원 측정값의 정확성을 높이는 개량된 부분 Hausdorff 차원(local Hausdorff dimension)을 제안한다

이미지의 경우에 각 이미지가 가지고 있는 각 특성은 차원을 이용하여 측정할 수 있으며 이 경우 기존의 유클리드 차원이 아닌 프랙탈 차원을 적용하게 된다. 여러 종류의 프랙탈 차원 중에서는 Hausdorff - Besicovitch 가 그 적용이 용이하여 많이 사용된다.

프랙탈 차원의 계산에 많이 사용되는 Box counting 차원 계산은 Hausdorff 차원의 개념을 이용하는 데 이 방법은 structured - walk 기법을 사용한다. 그러나 이 방법은 이미지의 차원계산 측정시 구간의 크기인 ε 에 의존하게 되어 적용은 간단하나 반면에 계산 알고리즘에서의 비효율성으로 차원의 계산 값이 부정확하게 된다. 이에 따라 개량된 차원 계산법을 제안한다.

기존의 Hausdorff 차원의 값은 그 계산식을

$D_H = \lim_{\varepsilon \to 0} \frac{\ln N(\varepsilon)}{\ln(1/\varepsilon)}$ 로 나타낸다.

이러한 Hausdorff 차원의 계산 방법은 실제 계산시에는 Box counting 방법을 사용하게 된다.

프랙탈 차원의 계산은 D_B로 주어지게 되는데, 이때의 D_B는 $N(\varepsilon) = k\,\varepsilon^{-D}$의 log-log 기울기의 비로 나타난다. $N(\varepsilon) = k\,\varepsilon^{-D}$는 log를 취하여 $\ln N(\varepsilon) = D \ln\frac{1}{\varepsilon} + \ln k$ 로 되며, 이때 k 는 이미지의 기하학적인 특성 상수이다.

Box counting 방법에 의한 차원은 결과적으로

$D_B = \lim_{\varepsilon \to 0} \frac{\ln N(\varepsilon)}{\ln(\frac{1}{\varepsilon})}$ 가 된다..

$N(\varepsilon)$는 구간 크기 ε를 가지며 이미지를 cover하는 최소 box의 개수를 의미한다.

4. 제안한 프랙탈 차원의 계산방법

제안하는 개량된 box counting 방법에 의한 차원의 계산은 Attractor A가 존재하는 구간의 최소 반경을 측정 box의 크기로 하는 것이다 Hausdorff 차원의 계산을 위한 다른 방법으로서 attractor를 이용하는 경우 이는 이미지 또는 곡선의 간격 자체를 이용하지 않고 각 구성 이미지의 경계와 근방 영역, 그리고 그 이외의 부분으로 나누어 측정을 하게되는 것이다 따라서 attractor의 각 존재 반경을 기준으로 삼아 차원을 측정하게 되며 그 이외의 부분 또한 attractor 존재 반경의 complement로서 차원의 측정에 이용된다. 이와 같이 제안된 차원의 새로운 계산 방법은 다음과 같다

$\delta(y, S) = \inf_{x \in S} \| x - y \|$ 일 때,

$A_\varepsilon \equiv \{ y \in R^n \mid \delta(y, A) < \varepsilon \}$ 라 정의 하자.

n 을 위상적 차원의 수리 할 때, $D_A = \lim_{\varepsilon \to 0} \frac{\ln A_\varepsilon}{\ln \varepsilon}$ 로 정의하면, $D_B = n - D_A$ 이다 따라서 D_B 를 아래와 같은 순서로 계산할 수 있다

Attractor A가 구간 ε의 크기를 한 변으로 하는 $N(\varepsilon)$개의 box로 cover 된다면, 이 때 attractor A를 가지는 영역 내부의 각 점들 또한 $N(\varepsilon)$개의 box들 내부에 있게 된다 각 box는 자신을 포함하여 3^n개의 근방을 가지게 되며 프랙탈 차원의 정의에 의해 우리는 D_B 를 계산 할 수 있다

5. 제안 방법의 적용

우리가 제안한 개량된 차원 계산법을 적용하여 다음과 같은 결과를 얻었으며 보다 정확한 프랙탈 차원을 계산할 수 있다 기존의 결과와 비교하면 다음과 같다 의다.

< KOCH 곡선의 차원>

log4 / log3 = 1 261860

box 크기	기존방법	개량방법
2	701	639
차원	1.153135	1.286733

box 크기	기존방법	개량방법
4	402	346
차원	1.371402	1.512993

box 크기	기존방법	개량방법
6	296	258
차원	1.063996	1.020151

box 크기	기존방법	개량방법
평균차원	1 196178	1.273292

위 결과는 제안한 방법을 적용하면, 기존의 box counting 방법과 비교하여 보다 정확한 차원 값을 얻을 수 있음을 보인다

6. 결론

개량된 프랙탈 차원의 계산 방법으로서 제안된 box counting 방법으로 측정된 차원의 결과는 기존의 방법에 의한 결과값보다 정확한 결과를 얻었다. 이러한 결과를 이미지의 처리에 적용하는 경우 이미지 처리의 근간이 되는 이미지의 각 구성 요소의 분류를 보다 정확하게 할 수 있고 하여 이미지 자체가 가지는 정보를 읽는데 도움이 될 수 있다고 본다. 따라서 각 이미지의 분류 또는 압축/처리등에 적용하는 경우 보다 바람직한 결과를 기대할 수 있을 것이며 이것에 대한 연구가 필요할 것이다

참고문헌

[1] John C Hart, Fractal Image Compression and Recurrent Iterated Function System, IEEE Computer Graphics and applications 1996

[2] Yuval Fisher, Fractal Image compression -Theory and Application, Springer - Verlag, 1994.

[3] M F Barnsley, L P Hurd, Fractal Image Compression, AK Peters, Ltd,1993

[4] R. M. Gray, P C Cosman, E. A. Riskin, Image Compression /Tree- structured Vector Quantization, Image and Text compression, Kluwer Academic Pub.,1992

[5] M F Barnsley, Fractals Everywhere, Academic Press,1988.

[6] J. E. Hutchinson , Fractals and Self-similarity, Indiana University Mathematics Journal, Vol. 35 No.5. 1981