

再帰的反復関数系を用いたハイパーテクスチャリング

長峯 望[†] 望月 茂徳[‡] 堀江 大輔[‡] 蔡 東生^{*}

筑波大学理工学研究科[†] 筑波大学システム情報工学研究科[‡] 筑波大学電子・情報工学系^{*}

1. はじめに

ソリッドノイズとは、3次元空間の各点にランダムな値が割り当てられる関数であり、コンピュータグラフィックスにおいては不可欠である。これは物体に対して自然な揺らぎを持たせることが可能で、サーフェス・テクスチャ、確率モデリング、自然現象のアニメーションなどにおいて即時的で強力なツールであることが知られている。本研究では、再帰的反復関数系(Recurrent Iterated Function System: Recurrent IFS)と3次元のChaos Game Algorithmによって生成されるソリッドノイズについて述べ、これを用いたハイパーテクスチャリングを行う。

2. ソリッドノイズ

ソリッドノイズとは、3次元上の実数を返す関数 $f: R^3 \rightarrow R$ で定義され、コンピュータグラフィックスに導入されているソリッドテクスチャの概念の一部であり、3次元物体のテクスチャリングに用いられる。また、「自然」は複雑さと規則性を同時に兼ね備えている。このような自然画像を作り出すことの出来るソリッドノイズはコンピュータグラフィックスやモデリングにおいて非常に有効であり、これを用いて1980年代にPerlinらによってハイパーテクスチャ[1][2]が提唱された。ハイパーテクスチャにおけるPerlinらのノイズは有用ではあるが、物体への適応において経験的な知識を必要とする故に、任意の複雑なテクスチャノイズを思い通りに設計することは容易ではない。

そこで、本研究では、2次元や3次元の物体や風景をモデリングできる利点を持つ再帰的反復関数系による任意のノイズ設計を行うことを考える。

3. 再帰的反復関数系 (Recurrent Iterated Function System: Recurrent IFS)

本研究における再帰的反復関数系 (Recurrent IFS) [3] とは IFS $\{X; \omega_i; i=1, 2, \dots, N\}$ と確率行列 $p_{ij} \in [0, 1]$ ($i, j=0, 1, \dots, N$)

から成り以下の2つの条件を満たすものである

$$(i) \quad p_{i1} + p_{i2} + \dots + p_{iN} = 1 \quad (i=1, 2, \dots, N)$$

(ii) どんな i, j を選んでも次を満たすような有限な整数の数列 $k, l, \dots, m \in 1, 2, \dots, N$ が存在する

$$p_{ik} p_{kl} \dots p_{mj} > 0$$

4. Borel Measure Rendering Algorithm

Borel Measure は実際に測ることが出来ず、Borel Measure Rendering Algorithmによって以下のように近似される。

(X, d) をコンパクトな距離空間とし、 $\{X; \omega_i; p_i; i=1, 2, \dots, N\}$ を Recurrent IFS としたときに、この Recurrent IFS に対して Chaos Game Algorithm を施して生成された点列を $\{z_i\}$ ($i=0, 1, \dots, \infty$) とする。このとき、空間 X の Borel 部分集合 B に Borel Measure μ が存在し、以下のように近似される[3]。

$$\mu(B) \cong \lim_{n \rightarrow \infty} \left\{ \frac{N(B, n)}{(n+1)} \right\}$$

(ただし $N(B, n) = \{z_i\} \cap B$ に含まれる点の総数)

本研究では Measure をノイズ値として扱い、Recurrent IFS と Borel Measure Rendering Algorithm によって生成されたソリッドノイズを便宜上、フラクタルソリッドノイズと呼ぶ。次に IFS コードの取得する方法を記す。

5. フラクタル画像符号化

本研究では IFS コードを3次元のフラクタル画像符号化の手法を用いて取得する。また得られた IFS コードに対して blowing 理論を用い、連続的に変化するノイズを生成する。

フラクタル画像符号化は LIFS (局所反復関数系) [4] および局所コラージュ定理[4]に基づいた符号化法で、符号化は概ね以下の過程で行われる。

- ① 画像を領域分割する(分割された領域をレンジブロックと呼ぶ)。
- ② 各領域に対して局所コラージュ定理における局所コラージュ誤差が最小となる領域(ドメインブロック)および写像パラメータを決定する。
- ③ ②で決定した写像パラメータ等の情報を符号語として記録する。

また、任意の画像に対して符号語の写像パラメータで決定される写像を反復し得られた極限(アトラクタ)が復元画像であ

Hypertexturing using Recurrent Iterated Function System

[†]Nozomi Ngamine [‡]Shigenori Mochizuki, Daisuke Horie

^{*}DongSheng Cai

[†]Master's Program in Science and Engineering at University of Tsukuba

[‡]Graduate School of Systems and Information Engineering at University of Tsukuba

^{*}Institute of Information Sciences and Electronics at University of Tsukuba

る。

6. Blowing 理論

パラメータ p を持つ IFS $[X; \omega_i, i=1,2,\dots,N]$ において, 縮小写像 ω_i がパラメータ p によって連続的に変化するならば, アトラクタも p に依存し連続的に変化する[3]. これを blowing 理論と呼ぶ。

本研究では, フラクタル画像符号化で取得した IFS コードに対し, パラメータ p を付加し, 連続的に変化するノイズを生成することを考える。

以上をもとに取得した IFS コードによって生成されたノイズと blowing 理論を用いて IFS コードに変化を与えたノイズを図 1 に示す。

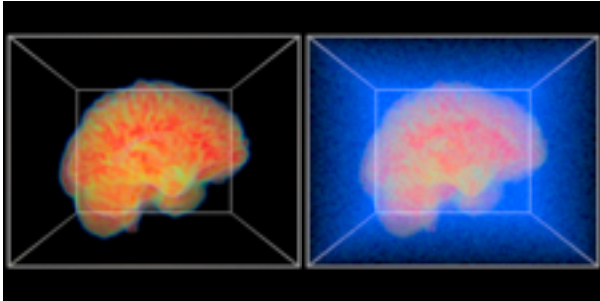


図 1. (左: 3 次元復号画像 右: blowing 理論によって変化が加わった 3 次元復号画像)

7. ハイパーテクスチャリング

フラクタルソリッドノイズによるハイパーテクスチャを行うために, $[0,1]$ の 範囲の値を持ったオブジェクトに対する密度関数 $D(x)$ を導入する. $D(x)$ は R^3 の点 x についての密度を返し, オブジェクトの形状を決定する。

さらに, 密度調整関数 (Density Modulation Function : DFM) f_i を導入する. これはオブジェクトの密度を調整するために使用される。

これにより, ハイパーテクスチャは次のように定義される。

$$H(D(x), x) = f_n(\dots f_2(f_1(D(x))))$$

次に, 本研究で対象とするオブジェクトを示す. 対象とするオブジェクトは soft region を持つシンプルな球体とし, 中心座標を c , 半径を r , soft region を作る強度を s として次式の密度関数で定義する。

$$\begin{aligned} sphere_{[c,r,s]}(x) \\ r_1^2 &:= (r - s/2)^2 \\ r_0^2 &:= (r + s/2)^2 \\ r^2 &:= (x_x - c_x)^2 + (x_y - c_y)^2 + (x_z - c_z)^2 \end{aligned}$$

$$\begin{aligned} sphere &:= \text{if } r_x^2 < r_1^2 \text{ then } 1.0 \\ &\quad \text{if } r_x^2 > r_0^2 \text{ then } 0.0 \\ &\quad \text{else } (r_0^2 - r_x^2) / (r_0^2 - r_1^2) \end{aligned}$$

また, ハイパーテクスチャリングを行うために図 1 で示されるフラクタルソリッドノイズを DMF として使用し, $fractal\ noise(x)$ として表す。

次にこれらをもとに生成されたハイパーテクスチャと Perlin らによるハイパーテクスチャを比較する. 図 2(左)は Perlin らによって提案された turbulence 関数によるノイズを加えることによって生成されたオブジェクトであり, 図 2(右)はフラクタルソリッドノイズを加えることによって生成されたオブジェクトである. このとき, それぞれの密度関数は DMF を用いて次のように表される。

$$\begin{aligned} H_1(sphere(x), x) &= gain(bias(turbulence(x), 0.3), 0.9) \cdot sphere(x) \\ H_2(sphere(x), x) &= gain(bias(fractal\ noise(x), 0.3), 0.9) \cdot sphere(x) \end{aligned}$$



図 2. (左: turbulence 関数ノイズによるハイパーテクスチャ 右: フラクタルソリッドノイズによるハイパーテクスチャ)

8. おわりに

本研究では, ハイパーテクスチャにおけるノイズを再帰的反复関数系 (Recurrent IFS) と Borel Measure に基づいてノイズ分布関数を作ること提案した. このノイズ分布関数を用いることにより, ソリッドノイズの種類が多様化されるため, 様々なハイパーテクスチャが生成できる. また, フラクタル画像符号化法により, 任意のノイズ設計が可能であるため, これまで表現が困難であったハイパーテクスチャや不可能であったハイパーテクスチャを生成することが可能であると考えられる。

参考文献

- [1] Ken Perlin and Eric M. Hoffert, "Hypertexture," Computer Graphics, Vol. 23, No. 3, July 1989 (SIGGRAPH Conference Proceedings).
- [2] Ebert, D., Musgrave, F., Peachey, P., Perlin, K., and Worley, S., "Texturing and Modeling: A Procedural Approach, Third Edition", with contributions from W. Mark and J. Hart, Morgan Kaufman, November 2002
- [3] M. F. Barnsley : "Fractals Everywhere", Academic Press, 1988.
- [4] Y. Fisher, Ed., Fractal Image Compression: Theory and processing operation, Springer-Verlag, New York, 1994.