

フラクタル図形を利用した3次元構造データの マルチレゾリューション可視化に関する研究

47-076785 岩丸 雅紀
指導教員 岡本 孝司 教授

Visualization of volume data is important issue in scientific visualization. However, there are few visualization methods for volume data except for volume rendering or streamlining. In this study we propose a novel method for visualizing volume data as a two-dimensional image by expanding 3D voxels onto 2D squares of a fractal diagram, called Sierpiński Carpet. The proposed method uses the self-similarity of the fractal diagram and achieves multi-scale visualization of volume data in 2D. With this method, we can browse the entire domain of volume data without occlusion. To achieve multilateral visualization, we combined the proposed method and existing 3D-based methods. In addition, we propose some useful functions for interactive visualization and discuss the uniqueness and effectiveness of our method.

Key words : Multi-resolution Volume visualization, Fractal diagram, Sierpiński Carpet

1 緒言

ボリュームデータは3次元空間的構造をもつデータであり現実空間における温度・密度・速度といった物理量の分布を記録したものである。従来科学的可視化の対象とされてきたボリュームデータの可視化手法としてはボリュームレンダリング¹⁾や流線が一般的であるが、対象データの特徴を正しく把握するためには可視化精度に注意を払うとともに複数の可視化手法を並列的に用いて多角的な可視化を行う必要がある²⁾。従って既存の3次元可視化手法と親和性の高い新しい可視化手法の開発が求められる。

2 目的

本研究の目的は既存の3次元可視化手法との併用に適した新規手法を開発することである。そのためには既存の3次元可視化手法とは異なる観点からのボリュームデータ可視化・閲覧を実現する必要がある。そこで情報可視化的発想を取り入れた新しい可視化手法を開発する³⁾⁴⁾。またそのような手法は3次元可視化の問題点である視点依存性・遮蔽性を克服するものでなければならない。我々は提案手法を視点依存性・遮蔽性のない2次元可視化とすることでこれらの欠点を克服し、既存の3次元可視化手法と組み合わせる際に多角的なデータ可視化を実現することを目指した。

3 手法

3.1 概念

情報可視化手法の多くが階層構造をもつデータを可視化対象としている。ボリュームデータのボクセル構造を階層構造とみなすことでボリュームデータ可視化に情報可視化的発想を持ち込むことが容易になる。さらに本来3次元的であるボリュームデータを2次元平面で一覧可視化するためにフラクタル図形を活用してボクセルを2次元平面に展開するという新しい手法を提案する。

まずボクセルの階層性は解像度 R を用いて Fig. 1 のように定義できる。

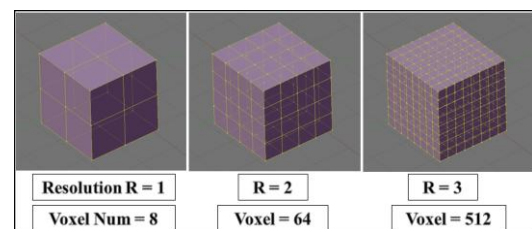


Fig. 1 Subdivision of voxels

ここでは各ボクセルを $2 \times 2 \times 2$ に等分割する過程をボクセルの再分割と呼び、これにより解像度 R が1上昇するものとする。このとき解像度 R におけるボクセルの総数は 8^R 個である。一般に高解像度のデータを用いればデータ値分布を高精度に記録できるがデータ量および可視化処理負荷は解像度 R に対し指数関数的に増加する。

次にシェルピンスキーのカーペット (Sierpiński Carpet 以下 SC と略記) というフラクタル図形について述べる。SC は Fig. 2 に示すような以下の描画アルゴリズムを再帰的に適用することで描画可能である

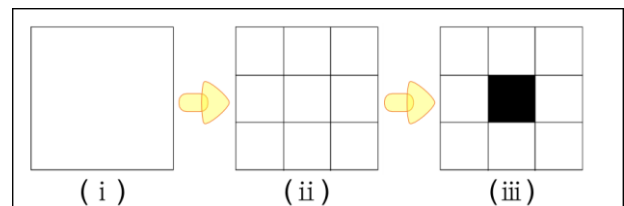


Fig. 2 Drawing algorithm for SC

- (i) 初期状態は1個の矩形
- (ii) 図形中の各矩形を 3×3 に9分割する
- (iii) 9分割後の矩形群から中心の矩形を除去する

(ii), (iii)の操作を再帰的に施すことで高解像度の SC を描画することができる。(ii), (iii)の操作を施すことを SC の再分割と呼ぶことにする。ここで SC の解像度 R を Fig. 3 のように定義する。

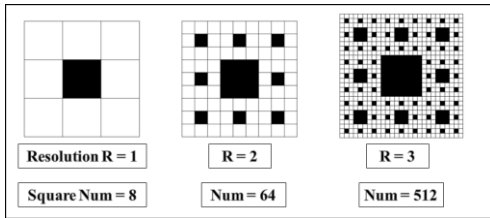
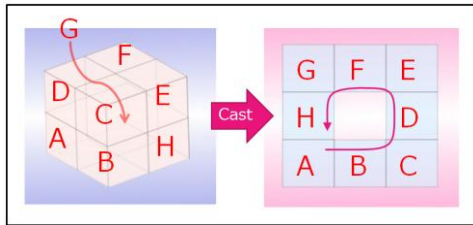


Fig. 3 Subdivision of SC

すなわち再分割により SC の解像度 R が 1 上昇する。このとき解像度 R の SC における矩形領域の数は 8^R 個である。本手法の核となる考え方は 8^R 個のボクセルをこの 8^R 個の矩形に展開することである。

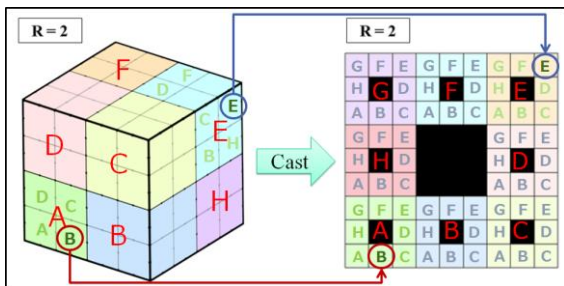
3.2 基本展開ルール

まずは $R=1$ における 2 次元展開のルールを定める。このルールはボクセルと矩形が 1 対 1 の対応している限り任意である。ここでは Fig.4 のように展開ルールを定めることにする。このルールを基本展開ルールとする。

Fig. 4 Mapping rule for $R=1$

3.3 高解像度データのための展開ルール

より高解像度のデータ ($R \geq 2$) に対してはボクセル・SC 両方に再分割を施すことで対応する。この際に $R=1$ で定めた展開ルールを再帰的に適用することで各ボクセルを SC の矩形に対応付けることができる。Fig.5 は $R=2$ の 2 次元展開ルールである。

Fig. 5 2D Mapping for $R=2$ volume data

上左図において赤丸で囲んだボクセルは $R=1$ において A に属し $R=2$ において B に位置するボクセルである。これをボクセル {AB} と呼称することにする。このボクセルが 2 次元展開される先は、上図右の赤丸で示した矩形であり、これは $R=1$ において A に属する領域内の、さらに $R=2$ において B に位置する矩形 {AB} である。

この展開の様子を段階を追って説明する。まず $R=1$ の

展開でボクセル {A} (ボリューム左下の緑色のボクセル群) は矩形 {A} (SC 左下の緑色の領域) へと展開される。ここまでは $R=1$ のボリュームデータに基本展開ルールを適用したにすぎない (Fig.4 参照)。さらにボクセル {A} に再分割を施し 8 個のサブボクセルに分割する。同様に矩形 {A} にも再分割を施すことで 8 個の矩形に分割される。ここで生じた 8 個のボクセルと 8 個の矩形に対し再度基本展開ルールを適用する。こうしてボクセル {AB} は矩形 {AB} へと展開される。

3.4 2次元展開の有する特徴

2 段階の展開を適用した Fig.5 においてボクセルの内包関係が 2 次元展開後は矩形の内包関係として現れていることに注目する。例えばボクセル {AB} はボクセル {A} に内包されるボクセルであるが、その展開先の矩形 {AB} もやはり矩形 {A} に内包されている。同様のことが青丸で示したボクセル {EE} と矩形 {EE} にも言える。このようにボクセルの包含関係は 2 段階の 2 次元展開を経た後も矩形の包含関係として保たれる。この特性はさらに多段階の 2 次元展開を経ても維持される。

さらにここでボクセル・SC の再分割は選択的・局所的に適用可能であることに注目する。内包するボクセル値が一樣なボクセルはそれ以上分割せず、一樣でないボクセルのみに再分割を施すことで SC の見かけの情報量を減らし認知負荷・描画負荷を低減することができる。このとき SC には解像度の異なるボクセルが共存して可視化されており、マルチレゾリューションな可視化が実現される。

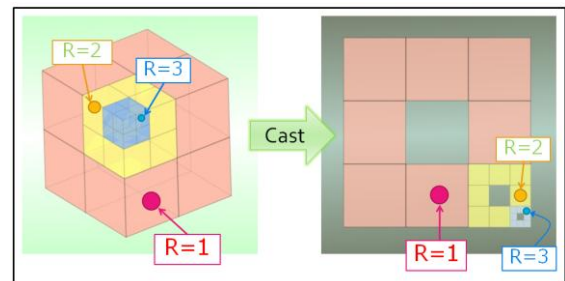


Fig. 6 Multi-resolution 2D Mapping

最後にブランク領域の活用について言及する。原理上 SC には多数のブランク領域が生じる。これらの領域に補助的な情報を可視化することで有効活用することが可能である。ここではブランクの周囲の矩形に相当するボクセルの平均値を格納する方法をとる。これは矩形 {AA} ~ {AH} に囲まれるブランクにはボクセル {AA} ~ {AH} の平均値に相当する値を色で可視化することを意味する。本手法では親ボクセルには子ボクセルの平均値が格納されているため、このブランクには親ボクセル {A} の格納値が可視化される。これを $R=0$ まで遡って適用すればブランクには全ての親ボクセルの格納値が可視化されていくため、結果的に全解像度における可視化結果が一枚の SC 上に可視化されることになる。

4 実装

前段で述べた 2 次元展開による可視化手法をアプリケーションとして実装し、既存の 3 次元可視化手法（スカラーデータに対してはボリュームレンダリング、ベクターデータに対しては流線）と相互運用可能な独自の可視化システムを構築した。実装にはプログラミング言語に C# を、グラフィクス API に Direct X 9.0c を、GPU のプログラマブル・シェーダのプログラミングに HLSL を用いた。次の Fig.7 はシステムのスクリーンショットである。

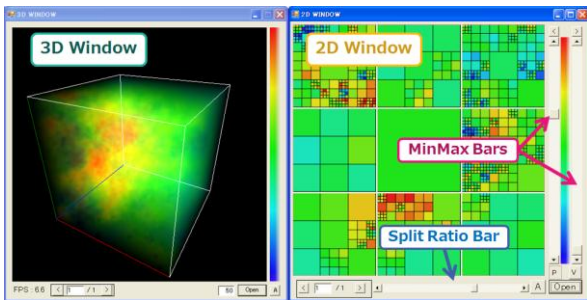


Fig. 7 The screenshot of the application

ボクセルに格納されたデータ値は色として可視化される。アプリケーションは 3D Window (左) と 2D Window (右) からなり、それぞれ 3 次元可視化・2 次元可視化像を描画する。2D Window には 3 つのスクロールバーが備えられている。左端の 2 本の垂直スクロールバーはボクセル値と色との対応関係を調節するバーであり Min Max Bar と呼ぶ。ユーザーはこのバーを操作することで可視化色を調節し、より視覚的に理解しやすい可視化像を得ることができる。この操作は 3D Window ・ 2D Window 両方の可視化結果に反映される。下端にある水平スクロールバーは Split Ratio Bar と呼ばれ、SC の分割具合を調節する。このバーを操作することで適度な分割度合いの 2D 可視化像を得ることができる。

まずはスカラーデータの可視化を試みた。ボクセルの分割は格納されたデータ値の分散に基づいて行った。色変換にはグラデーションテクスチャを用いて、大きな値は赤色へ小さな値は青色へ変換されるようにした。ボリュームレンダリングを行う際にはこれに不透明度を加え、半透明化することで内部構造を可視化した。

さらにボリュームデータを可視化する上で有益と思われる機能をいくつか実装した。ここでは一例として**指定したボクセル値の減算処理**を挙げる。本システムでは任意のボクセルをマウス操作によって直接指定することができる。SC の矩形とボクセルは 1 対 1 に対応しているため、SC 上である矩形を選択すれば対応するボクセルを指定したことになる。この特徴を活かして、指定したボクセルの格納値を全ボクセルから減算する機能を提供する。この機能を用いることで指定したボクセルよりも大きな値をもつボクセルだけが可視化結果に反映される。Fig.8 はこの機能をスカラーデータに対して適用した例である。

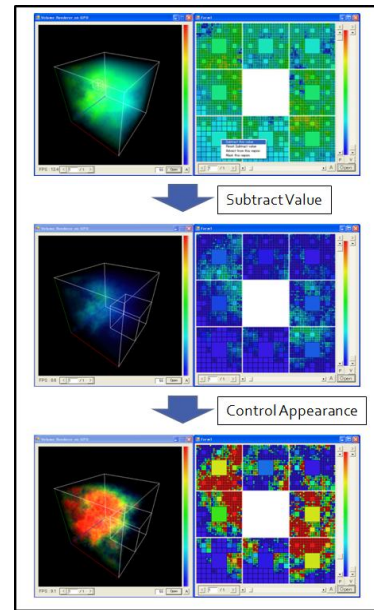


Fig. 8 Demo of “subtract value” function

同様にベクターデータの可視化も行った。用いたデータは流体数値計算により得られた速度場のデータである。ここでも分散値に従ってボクセルを分割した。ベクター値から色への変換は速度ベクトルの (u, v, w) 成分をそれぞれ矩形色の (r, g, b) 成分に対応させることで行った。

ベクトル場の可視化では 3 次元手法として流線描画を併用した。流線を生成するためには始点を定めて仮想的な粒子を移流させる必要があるが、この始点領域を SC を用いて直接的に指定できるようにした。ユーザーはまず SC の結果を眺め速度場が特徴的な部位を特定し、その部位を直接指定して流線を描画することができる。Fig.9 はこの機能により特徴領域の流れを可視化した例である。

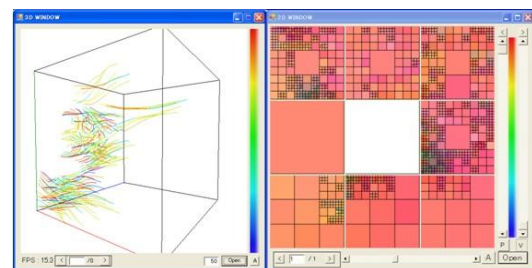


Fig.9 Vector Visualization

SC 全域がピンク色 $(R, G, B) = (1.0, 0.5, 0.5)$ に近い色であることは流れ場全体において X 軸正方向の流れが支配的であることを表している。細かく分割された色変化の激しい領域は流れ場が複雑であるため、この部位を指定して流線の始点とすることで特徴領域に流線を生成できる。

5 評価

2 次元展開と既存の 3 次元可視化の定性的相違を明らかにするために被験者によるユーザーテストを実施した。評価項目は「全体把握」「局所把握」「特徴値把握」の 3 項目とし、可視化された 4 つのスカラーボリュームデータを

以下に述べる指標に基づいて並べ替えるタスクを行った。各実験でタスク正答率と回答に要した時間を調査した。正答率(点数)は1位および4位に回答すべきデータが正しく並べられているかを元に1.0点満点で算出した。

実験1の全体把握ではボリューム全域の平均値を可視化結果から目視により推測、その値が大きい順にデータを並べ替えるというタスクを行った。可視化手法にボリュームレンダリングを用いた場合と2次元展開を用いた場合で正答率および回答時間がどう異なるかを調べた。

実験2の局所把握ではボリューム内の一部領域を指定し、その領域における平均値を推測・比較するタスクを行った。被験者は指定した領域の平均値を見積もり高い順にデータを並べ替える。指定した領域はボリューム内部にあるためVRでは必然的に遮蔽の問題が生じることになる。

実験3の特徴値把握では各データに含まれるボクセル格納値の最大値同士を比較するタスクを行った。MinMaxBarにより可視化像を調節することで最大値を抽出し比較する。

実験の結果を以下のTable.1に示す。用いた可視化手法は3種ありそれぞれVR(ボリュームレンダリング)、SC I(ブランクには何も可視化しないSC)、SC II(ブランクに平均値を可視化したSC)である。実験3のみVRとSC Iのみの比較となっている。

Table.1 Results of User Tests

Test Results	VR	SC I	SC II
Test 1 Ave. Score	0.81	0.85	0.88
Test 1 Ave. Time(sec)	62.6	61.2	48.0
Test 2 Ave. Score	0.72	0.94	0.97
Test 2 Ave. Time(sec)	97.6	46.0	58.6
Test 3 Ave. Score	0.75	0.94	
Test 3 Ave. Time(sec)	80.7	61.8	

実験1ではブランクに平均値を可視化したSC IIが最も正答率が高く、回答に要した時間も最短となった。SC IIでは中央のブランクにデータ全域の平均値大きく可視化されているため比較が容易であったことが理由である。VRとSC Iでは回答時間に差はないものの点数では若干SC Iが優位である。この理由として視点依存性と遮蔽が無いことがデータ同士の比較を容易にする点が挙げられる。このメリットは実験2および実験3の結果により明確に表れている。一方で全域平均を推測するためにはSC Iの全ての矩形を考慮する必要があるためSC Iの方がVRに比べ回答に時間のかかるケースや正確に平均値を推測できないケースもあった。ボリュームレンダリングの各ボクセル色は複数のボクセルの情報が集約された結果であり、その解釈に一意性を求めることはできないがそれと引き換えに認知負荷を軽減する効果があると言える。

局所比較(実験2)・最大値の比較(実験3)では正答率・回答時間共にSCを用いた手法が優位な結果となった。

2次元展開手法は個々のボクセルを独立して可視化するためボリュームレンダリングでは可視化結果に表れにくい特徴、特に少数のボクセルによる小さな領域における特徴を捉えることを可能にする。実験2・実験3のように少数のボクセルだけを考慮すれば良い比較では2次元展開を用いた可視化は特に効果的であったと言える。

以上が今回明らかにできた2次元展開とボリュームレンダリングの定性的相違である。加えて形状把握には3次元空間の連続性が保たれていることが重要であるため2次元展開よりも3次元可視化手法のほうが有効であることは自明である。

6 結論

本研究ではまず従来の可視化における手法およびその発展のオーバービューを通じて多角的可視化の必要性、科学的可視化・情報可視化両手法の融合、2次元可視化・3次元可視化の融合といった新しい可視化の試みが求められていることを確認した。

こうした試みに向けた第一歩としてフラクタル図形を利用した2次元展開という情報可視化の考え方に基づく手法開発しボリュームデータに適用するというアプローチをとった。手法の開発にあたり既存の可視化手法を研究しそれらの共通点や相違を考察することで有益な着想を多数得た。その結果提案手法はこれまでの可視化手法の特徴を取り入れつつもフラクタル図形の自己相似性を可視化そのものに利用した点、SCを入力インターフェースに活用することで3次元ボリューム空間にダイレクトなアクセスを提供した点などにおいて新規性のあるものになったと自負する。

さらにユーザーテストにより従来の3次元手法との定性的相違を明確化した。その結果提案手法は個々のボクセルを独立して認識する目的に適することが実証された。遮蔽性・視点依存性の不在によりデータ間比較の際の認識コスト・インタラクションコストが低減されることも確認した。既存の3次元可視化手法にないこれらの特徴を有する本手法を併用することでより、効率的かつ新規性のある可視化ワークフローが構築可能となると結論付ける。

文献

- 1) Robert A. Drebin et al.: "Volume Rendering", ACM SIGGRAPH, Vol.22 Issue 4 (1988), pp 65-74.
- 2) 白山 晋:「ボリュームレンダリングの可視化精度について」, 日本機械学会第13回計算力学講演会講演論文集(2000), pp. 109-110.
- 3) 岩丸雅紀, 岡本孝司, 「フラクタル図形を用いたボリュームデータの2次元展開」, 第36回可視化情報シンポジウム講演論文集, pp. 75-78, July, 2008
- 4) 岩丸雅紀 岡本孝司:「ボリュームデータ可視化装置および方法並びにプログラム」, 特願2008-142786.