

八分木を用いたラジアル基底関数による高速 3D 形状再構築

三浦 憲二郎[†] 曾根 順治^{††} 増田 宏^{†††}

この論文で、3D 点群データから RBF を用いて形状を再構築するために八分木が高速化に寄与することを示す。八分木により分割された個々のセルに対して、台が無限に広がったラジアル基底関数 (infinitely supported radial basis function) の一つである triharmonic ラジアル基底関数により、各セルに割り当てられた点群を内挿する。八分木のセルに割り当てる点群の数と処理時間との関係を理論的に、また実験的に調べ、点群の総数に依存しない最適な数値があることを示した。

An Octree-based Approach to fast reconstruction of 3D objects with Radial Basis Functions

KENJIRO T. MIURA,[†] JUNJI SONE^{††} and HIROSHI MASUDA^{†††}

The paper proposes a simple and fast surface reconstruction method from point-cloud data using radial basis functions with octree-based space decompositions. For each cell of the octree decompositions, we use triharmonic radial basis functions to interpolate the given point data. We have analyze the relationship between the number of points assigned to the octree cells and the processing time theoretically and experimentally and found an optimized number of points for each cell.

1. はじめに

CG や CAD において点群を内挿する問題は基本的な課題の 1 つである。ラジアル基底関数による陰関数表現は再メッシュ化や面の平滑化など様々な問題を統一的に解決する手法となっている。本研究では、八分木による空間分割 (space decomposition) に基づく点群から形状を高速に再構築する簡潔な手法を提案する。八分木により分割された個々のセルに対して、台が無限に広がったラジアル基底関数 (infinitely supported radial basis function) により、各セルに割り当てられた点群を内挿する。

2. 関連研究

高速多重極子 (fast multipole) 法といった洗練された数学的手法を用いるならば、大域的なラジアル基底関数により実用的な内挿や近似問題が解決できること

が Carr¹⁾ 等により示された。しかしながら、彼等の方法は非常に複雑であり、また点群の数が百万個を越えると数時間かかる。

Ohtake 等⁵⁾ は内挿時間を短縮するために台がコンパクトなラジアル基底関数 (CSRBF) を階層的に用いる方法を提案し、処理時間を実質的に数時間から数分に短縮した。CSRBF の外挿能力の低さを補うために、多重尺度 (multi-scale) 内挿を用い、処理の高速化を目的としてオフセット点を不要とするために、各々の面上点に局所直交座標を設定し、その座標系で定義される 2 次関数を導入した。しかしながら、彼等の方法では CSRBF の係数とともに局所座標を定義するためのベクトルと 2 次関数の係数を持たねばならないので多量のメモリを消費する。さらに、我々の実装では再構築された曲面が指定された CSRBF の有効半径に強く依存し、特に相対的に点群の密度が低い場所では不必要に波打つ場合があった。

3. 再構築法

3 次元の点群から形状を再構築する我々の手法は以下のように記述される: 与えられた点群に対して、

- (1) オフセット点を算出する。
- (2) 八分木を用いて点群の存在する空間を適当な大きさのセルに分割し、そのセルに含まれる点群と近傍の点群を割り当てる。

[†] 静岡大学工学部機械工学科

Department of Mechanical Engineering, Shizuoka University

^{††} 東京工芸大学工学部光工学科

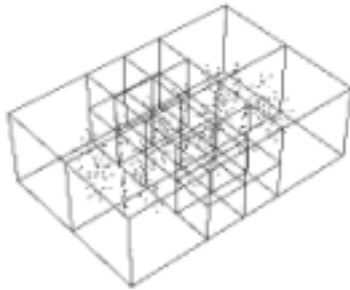
Department of Photonic Information and Media Engineering, Tokyo Polytechnic University

^{†††} 東京大学大学院環境海洋工学専攻

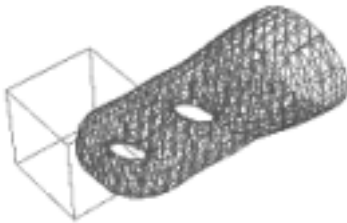
Department of Environmental and Ocean Engineering, University of Tokyo



(a) ポリゴン



(b) 点群と八分木



(c) 1つのセルで再構築した曲面



(d) 再構築した曲面

図 1 3 穴トーラス
Fig. 1 3 hole torus

- (3) 各セルに対して、前ステップでセルに割り当てた点群を用いて triharmonic ラジアル基底関数を構築する。

triharmonic RBF の計算のための行列の大きさを小さくするために、Iske 等³⁾ が提案しているようにステップ 2) で点群が存在する空間を八分木のセルに分割する。一般に離れた部分の形状は関連性が低いのでこの処理は正当化される。例えば、人物モデルでその顔の形状と足の形状に強い相関 (陰関数表現としての相関) 関係はない。顔と足の形状は基本的に独立であり、顔の形状を変化させずに立っているモデルを足を曲げて座らせることもけして不自然ではない。

ユーザが指定すべき変数は以下の 3 である。

- (1) 各セルに割り当てる最大の点群数: n_{max}
- (2) オーバラップ係数: α
- (3) 活性係数: β

ステップ 2) において、まず始めにオフセット点を含む点群は単位立方体に含まれるように移動拡大・縮小される。点群の境界箱を利用しながら、辺が 1 より小さな立方体 (辺の長さはユーザが指定可能) により境界箱を含むように空間を分割する。小さな立方体の一つ一つが八分木の最初のノード (根) となり、そのセルに割り当てるべき点群の数が指定された最大値 n_{max} を越えた場合は再帰的に 8 つに分割される。もし、セルに割り当てられる点は何もない場合は “dead” とラベル付けされ、そのセルに対しては RBF は計算されない。ある点の位置がセルの中に含まれるか、あるいは x と y, z 軸方向においてオーバラップ係数の半分をセルの 1 辺の長さに掛けた距離以内に存在する場合、その点をセルに割り当てる。

個々のセルに対して独立に RBF を計算するので、隣接するセルの境界では値が微妙に異なる。それらの関数が十分に重なっていないと違いが大きくなる可能性がある。その違いを抑えるために、オーバラップ係数が必要となる。オーバラップ係数の適切な値は点群の分布や密度に依存すると考えられるが、検証に用いた点群では $\alpha = 1.3 \sim 1.4$ の場合に内挿結果と処理速度の良いバランスが得られた。

活性係数 β はセルを “dead” にするための係数であり、点群が何も割り当てられていない場合や面上点がセルの中や近傍に含まれていない場合はそのセルを殺す。セルが面上点を持たない場合は、そのセルに交差する面が存在しないと考えるのが自然であり、セルを “dead” とする。しかしながら、セルが点を持たなくとも、セルの中に面が存在する可能性がある。例えば、セルのコーナが内挿すべき曲面を鋭くカットするかもし

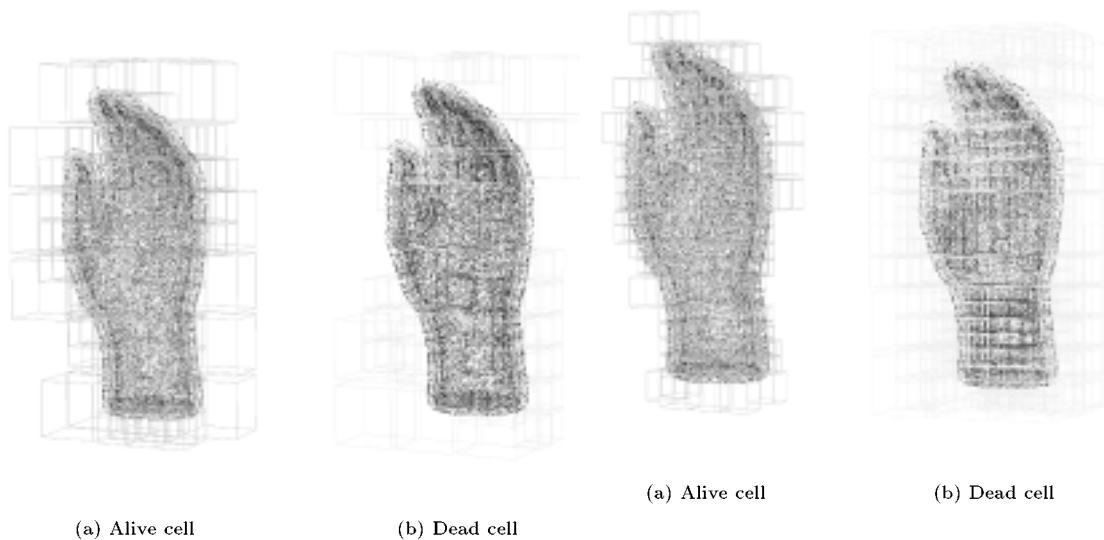


図 2 手モデル: $n_{max}=1000$
Fig. 2 Hand model: $n_{max}=1000$

図 3 手モデル: $n_{max}=250$
Fig. 3 Hand model: $n_{max}=250$

れない。そのような場合を避けるために、仮想的にセルを β だけ拡大し、拡大したセル内に面上点があればそのセルは生かす。この論文で示すすべての例に対して $\beta = 1.05$ を用いた。

最終的な形状評価のために、理想的には各セルに生成された RBF の関数値を混ぜ合わせるにより、隣接するセルの関数値の違いを吸収させることが望ましい。その方法としては、例えば、関数値を計算したい位置 P を含むセルとその近傍の $3 \times 3 \times 3 - 1$ 個のセルを用い、 P とセルの中心との x と y , z 軸方向の距離をパラメータとし、3 次 B-spline volume を用いて値を計算することが考えられる。ただし、この方法では空間内の 1 点の値を計算するために 27 個の RBF を計算する必要があり処理が遅くなる。したがって、実装においてはその点を含むセルが生きている場合はそのセルに生成された RBF の関数値を、死んでいる場合には再構築した形状の外側では正の大きな値 ($\gg 1$)、内側では負の小さな値 ($\ll -1$) をそこでの形状を決める (面上か、面から離れているか) 値としている。

これまでに説明した手順を実施した模式的な例を図 1 に示す。この例では点群ではなく、左上に示す 3 つ穴トーラスのポリゴンメッシュから面上点とオフセット点を生成し、形状再構築システムの入力としている。右上は八分木により空間を分割している様子を示している。面上点を緑で、外側へのオフセット点を赤、内側へのオフセット点を青でそれぞれ示している。左下の図は赤線に示した 1 つのセルに割り当てられた点群から再構築された曲面を示している。局所的な点群のみを用いているので 3 つの穴すべてが再構築されるわけではなく 2 つの穴のみが再現されている。最終的な形状の評価は有効なセルの結果を統合するので右下の図のように適切に再構築される。

4. 結果と議論

図 2 と図 3 は手のモデルの内挿例を示す。図 2 の例は $n_{max} = 1000$ を指定した結果である。図 3 の例は $n_{max} = 250$ であり、図 6 に示したグラフからわかるようにこの場合に最短の処理時間が得られた。“生きている”セルはシアンで、“死んだ”セルは再構築した形状の外側は黄色で、内側はマゼンタで表示している。図 4 は Stanford bunny の例であり、 $n_{max} = 300$ としている。図 5 は Happy budda の例であり、 $n_{max} = 250$ としている。表 1 に処理時間を示しそのグラフを図 6 に示す。表中の 150 等の数字は n_{max} の値であり、s は秒を意味する。使用した計算機は Pentium4 2.53GHz で 512MB のメモリを搭載している。

Model	on-pnt	off-pnt	150	200
Hand	13,530	13,478	10.4s	9.50s
Bunny	34,825	34,679	28.7s	26.7s
Budda	543,652	543,196	261s	231s

表 1 処理時間その 1

Table 1 Processing time no.1

Model	250	300	350
Hand	8.47s	8.55s	9.38s
Bunny	25.3s	24.8s	26.8s
Budda	230s	247s	267s

表 2 処理時間その 2

Table 2 Processing time no.2

実験に用いた RBF は次式で与えられ、

$$f(x) = \sum_{i=1}^n d_i \phi_i(\|x - c_i\|) + P(x). \quad (1)$$

ここで、 $\phi_i(r) = r^3$ であり、 $P(t)$ は 1 次多項式である。この式中の RBF の係数 d_i と 1 次多項式の係数を計算するために連立線形 1 次方程式を対称 LU 分解を用いて求めた。対称ソルバの処理時間は $N^3/6 + \Omega(N^2)$ に比例する。以下の分析では議論の単純化のためにオーバーラップ係数は 1 と仮定する (この仮定により議論の本質は変わらない)。

もし、点群の存在する空間が理想的に分割され n 個のセルに $a = N/n$ 個の点群が割り当てられたとする。八分木の生成時間を無視すれば、処理時間の支配的な部分は $(a^3/6)N/a = a^2N/6$ に比例する。これは n_{max} を少なくすればするほど処理時間が短縮されると解釈される。実験結果が示すようにこれは正しくない。したがって、行列の生成 (N^2 に比例) やセルへの点の割り当て (N に比例)、その他 N^2 や N 、定数に比例する処理が考えられるので、1 つのセルの処理時間を $N^3/6 + c_0N^2 + c_1N + c_2$ に比例すると仮定する。この仮定より、全体の処理時間は $(a^2/6 + c_0a + c_1 + c_2/a)N$ に比例すると考えられる。この式が意味するように、処理時間が最短となる n_{max} の値は 3 例ではほぼ等しく約 250 であり、点群の総数にあまり依存しない。

5. 結 言

ラジアル基底関数による陰関数表現は再メッシュ化や面の平滑化など様々な問題を統一的に解決する手法となっており、本研究では点群を内挿する問題に適用した。この論文で、3D 点群データから八分木により分割された個々のセルに対して、台が無限に広がったラジアル基底関数 (infinitely supported radial basis function) の一つである triharmonic ラジアル基底関

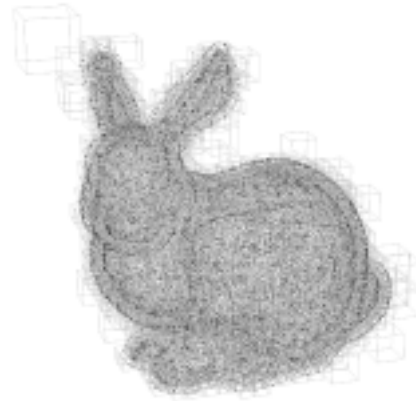
数により, 各セルに割り当てられた点群を内挿し, RBF を用いて形状を再構築するために八分木が高速化に寄与することを示した. また, 八分木のセルに割り当てる点群の数と処理時間との関係を理論的に, また実験的に調べ, 点群の総数に依存しない最適な数値があることがわかった.

6. 謝 辞

図 4 は Stanford Bunny であり, Happy Buddha とともに Stanford 3D scanning repository に所属する. 手のモデルについては Fairfield Technology に感謝いたします. メッシュからの点群の作成にはラティステクノロジー (株) の XVLKernel を使用させていただきました. 感謝いたします.

参 考 文 献

- 1) J.C. Carr, R.K. Beatson, J.B. Cherrir, T.J. Mitchell, W.R. Fright, B.C. McCallum, and T.R. Evans. Reconstruction and representation of 3D objects with radial basis function. In *Computer Graphics (Proceedings of SIG-GRAPH2001)*, pages 67-76, August 2001.
- 2) J. Duchon. Sur l'erreur d'interpolation des fonctions de plusieurs variables par les d^m splines. *R.A.I.R.O Analyse numerique*, 12(4):325-334, 1978.
- 3) A. Iske, and J. Levesley. Multilevel scattered data approximation by adaptive domain decomposition. Technical report, University of Leicester, April 2002.
- 4) L. Kojekine, I. Hagiwara, and V. Savchenko. Software tools using CSRBFs for processing scattered data. *Copmuters & Graphics*. Accepted.
- 5) Y. Ohotake, A. Belyaev, and H.-P. Seidel. A multi-scale approach to 3D scattered data interpolation with compactly-supported basis functions. *Shape Modeling International 2003*, Seoul, Korea, May 2003 Accepted.



(a) Alive cell



(b) 再構築曲面

図 4 バニーモデル: $n_{max}=300$
Fig. 4 Bunny model: $n_{max}=300$



図 5 仏陀モデル: $n_{max}=250$
Fig. 5 Happy budda model: $n_{max}=250$

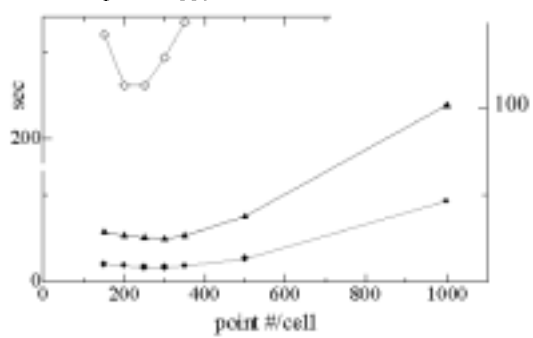


図 6 計算時間 ●:手, ▲:Stanford Bunny, □:Happy Buddha
Fig. 6 Processing time ●:手, ▲:Stanford Bunny, □:Happy Buddha