

推薦論文

## グラフカットによる異常値耐性のある陰関数曲面生成法

長井 超 慧<sup>†1,\*1</sup> 大竹 豊<sup>†2</sup> 鈴木 宏 正<sup>†2</sup>

コンピュータグラフィクスや CAD を代表とする多くの分野で、表面メッシュは、実在する 3 次元物体（実物体）の形状を表現する目的で多用され重要な役割を果たしている。実物体から表面メッシュを得る手法の 1 つに、物体表面のスキャンなどで得られる点群データを入力として、その形状を等値面として持つスカラー場を構築し、その等値面を近似するポリゴンメッシュを生成するものがある。この手法はスキャンデータに一般的に含まれるノイズに比較的頑健であるものの、異常値や大きいノイズを含むデータに対し適用すると過剰な面を含むなど、元の物体形状とまったく異なる表面メッシュが生成されることがある。本稿では、等値面による近似手法の代表的な方法である Partition of Unity (PU) に大域的手法であるグラフカットを組み合わせることで、異常値を含むデータに対する頑健性を高めた表面メッシュ生成法を提案する。

### Outlier Robust Implicit Surface Reconstruction Using Graph-cut

YUKIE NAGAI,<sup>†1,\*1</sup> YUTAKA OHTAKE<sup>†2</sup>  
and HIROMASA SUZUKI<sup>†2</sup>

Surface mesh representing the shape of a real object plays very important roles in many areas involving computer graphics and CAD. Generating surface mesh from a point cloud obtained by a 3D-scanning device is called surface reconstruction. One of popular surface reconstruction methods is the implicit method. It constructs a scalar field whose an isosurface approximates the shape of the object, and then generates a surface mesh with polygonizing the isosurface. An advantage of implicit method is the robustness for noise existing in scanned data. Unfortunately, if data includes large amount of noise or outliers, implicit methods may fail to reconstruction and generate a mesh with extra components. In this paper we propose a noise robust surface reconstruction method with combining partition of unity (PU) which is an implicit method, and a popular global method, Graph-cut.

### 1. はじめに

現在、機械設計のための CAD や CAE、コンピュータグラフィクスをはじめとする多くの分野で物体の形状表現に様々な種類のメッシュが用いられている。その中でも三角形あるいは四角形などを要素とし、多面体で物体の表面形状を表すメッシュ（表面メッシュ）には高い需要がある。実物体の表面メッシュは一般的には、物体表面のスキャンなどで得た点群を基に生成される（図 1）。通常スキャン点群には異常値（表面から大きくはずれたデータ点）やスキャンの欠損箇所、複数の視点方向からスキャンすることにより生じる点群のオーバーラップ、点群密度の不均一な箇所などが存在し、これらが表面メッシュ生成を困難なものにしている。これらの問題に対処するために非常に多くのアルゴリズムが提案されているが、ここではそれらをアルゴリズムの性質に基づいて陽的な手法と陰的な手法の 2 つに分類する。

陽的な手法とは、スキャン点群の点を直接つないで表面メッシュを生成する手法である。代表的としてドロネー四面体分割を用いる手法<sup>(2),3),8),9)</sup> があげられる。手法の中には得られたメッシュの精度保証ができるものもある。しかし陽的な手法は高い点群密度<sup>4)</sup> が必要であり、またノイズに対してあまり頑健でないという欠点がある。

一方、陰的な手法は、品質の低いデータに対しても安定して表面メッシュを得られるという特長を持つ。こちらは空間を物体の内側と外側の領域として分割し、その境界として表面を定める手法である。内外判定は、まず空間に対して関数を定義しその関数値を用いて行う。関数が空間全体に対して大域的に定義されるあるアルゴリズムを大域的手法と呼ぶことにする。大域的手法<sup>(6),13),14),19)</sup> は品質の低いデータに対する頑健性が非常に高い。ただし、細かい形状が損なわれるという問題がある。それに対し、空間内の小領域に対し定義される関数を用いる手法を局所的手法と呼ぶ。局所的手法の代表は近似関数に Partition of Unity

†1 理化学研究所  
RIKEN

†2 東京大学  
The University of Tokyo

\*1 現在、ウィーン工科大学  
Presently with Vienna University of Technology

本論文の内容は 2009 年 6 月のグラフィクスと CAD シンポジウムにて報告され、グラフィクスと CAD 研究会主査・幹事全員（全員）により情報処理学会論文誌ジャーナルへの掲載が推薦された論文である。

(PU)を利用したもの<sup>18)</sup>であり、これを表面メッシュ生成に応用したものがMPU<sup>17)</sup>である。局所的手法に基づく表面メッシュ生成法<sup>1),12),17)</sup>は大域的手法ほどの頑健性はないものの、細かい形状も非常によく再現できる。

表面メッシュに求められる詳細な形状表現はPUを用いれば達成できる。しかしPUは局所的手法なので、そのままではサンプリングの欠損や異常値を含む低品質なデータに対する頑健性が低く不十分である。そこで頑健性を高めるためにこれまでに多くの研究が行われてきた。その中には、入力点の法線ベクトルを改善するもの<sup>7)</sup>、平滑化によるデノイズングを行いノイズ耐性を高めるもの<sup>16)</sup>、本稿の提案と同じアイデアに基づく大域的手法との組合せにより、欠損に対する頑健性を高めたもの<sup>13)</sup>などがある。頑健性の観点からいえばいずれの手法も目的を達成しているものの、平滑化は量の見極めが難しいため扱いにくい。また、文献13)はボクセルの内外判定を行うため十分な精度が出ない。本手法では局所的陰的手法であるPUに大域的手法を組み合わせることで、異常値に対し頑健で詳細かつ滑らかな表面メッシュを生成するアルゴリズムを構成することを提案する。大域的手法に用いたグラフカットは、画像処理で前景と背景のセグメンテーションなどの目的で多く用いられており<sup>5)</sup>、これを表面再構成において空間を物体の内外に分割するのに応用した。

本稿の構成は以下のとおりである。まずアルゴリズムの説明を2章で行う。2.1節でPU

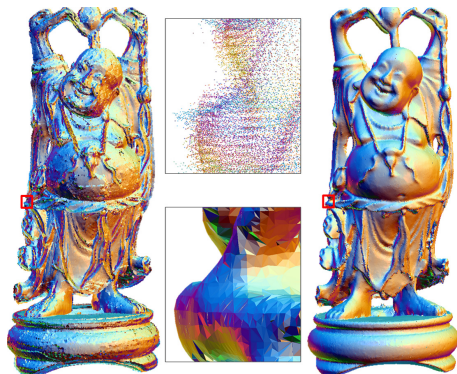


図1 提案アルゴリズムによる表面再構成の例。左：入力点群。中央：入力点群と表面メッシュの拡大。右：再構築により得られた表面メッシュ

Fig.1 An example of the results of our surface reconstruction algorithm. Left: input points. Center: close-ups of the input points and the obtained surface mesh. Right: the obtained surface mesh.

を、次いで2.2節でグラフカットによる内外判定法を説明する。表面メッシュの生成は2.3節で説明する。次に3章で結果を紹介し、最後に4章でまとめと今後の課題について述べる。

## 2. アルゴリズム

物体表面からサンプリングされた点群 $\{p_i\}$ およびこの各点における物体の内側に向き付けられた法線集合 $\{n_i\}$ から、物体の形状を表す三角形メッシュをPUとグラフカットを用いて生成する。2次元の入力に対する例を図2にあげる。

### 表面メッシュ生成アルゴリズム

- (1) PUによる関数あてはめを行い、入力点群(図2(a))のバウンディングボックスを局所近似関数のサポート球集合で被覆する(図2(b))。
- (2) サポート球の中心を頂点とする四面体メッシュを生成する。四面体メッシュの頂点と辺をグラフの頂点と辺の一部に見立て、グラフカットによりサポート球の中心の内外を判定する(図2(c))。
- (3) マーチングテトラヘドラ法により三角形メッシュを生成する(図2(d))。

### 2.1 サポート球による被覆に基づく局所近似

ここでは表面メッシュ生成アルゴリズムのステップ(1)を説明する。これは物体表面上の点からPUに基づいて陰関数を生成する手法<sup>17)</sup>である。

まず、PUを用いた陰関数曲面とは、物体表面からの符号付き距離を近似する関数 $f(x)$

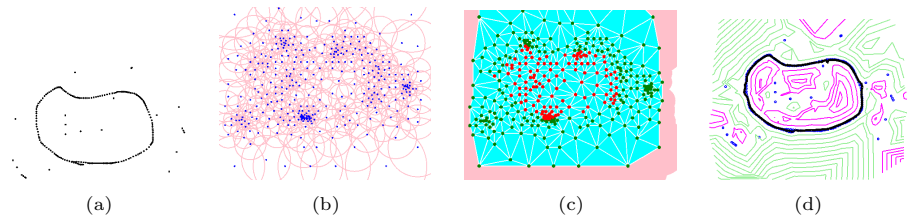


図2 2次元におけるアルゴリズムの流れ。(a) 入力点群。(b) 局所近似関数 $g_i(x)$ のサポート(ピンクの円)とその中心(青点)。(c) グラフカットによる内外判定の結果(赤点：内部、緑点：外部)。(d) 表面メッシュ(黒線)および $f(x)$ の等高線(ピンク： $f(x) > 0$ 、緑： $f(x) \leq 0$ 、入力点群(青点))

Fig.2 The overview of the proposed algorithm in 2D. (a) Sampling points. (b) Supports (pink circles) and their centers (blue points) of the local approximation functions  $g_i(x)$ . (c) The results of the Graph-cut; red points are judged as being inside the object, while the green points are outside. (d) The reconstructed surface (black). The contours of  $f(x)$  are drawn with pink and green lines according to their sign. The blue points are the sampling points.

の等値面による曲面表現である．

$$f(\mathbf{x}) = \sum_i w_i(\mathbf{x})g_i(\mathbf{x}) / \sum_i w_i(\mathbf{x}) \quad (1)$$

式 (1) から分かるように PU は任意の評価点における重みの総和が 1 になるように定められている．関数  $f(\mathbf{x})$  の定義域は入力点群のバウンディングボックスとする．関数  $g_i(\mathbf{x})$  は表面を  $g_i(\mathbf{x}) = 0$  として局所的に近似する局所近似関数であり，球状のサポートを持つ．重み関数  $w_i(\mathbf{x})$  は  $g_i(\mathbf{x})$  と同じサポートを持つ．各関数の詳細は 2.1.1 項に記す．物体の表面は等値面  $f(\mathbf{x}) = 0$  で近似される．本研究では  $f(\mathbf{x})$  は，物体内部で正，外部で負の値をとるよう定める．

入力点群のバウンディングボックスを  $f(\mathbf{x})$  の定義域とするためには，バウンディングボックス全体がサポート球で被覆されなくてはならない．また近似の効率を考えるとサポート球の分布は入力点の粗密に従っていることが望ましい．そこで MPU<sup>17)</sup> と同様にバウンディングボックスを入力点の粗密に従って八分木で分割し，得られた空間小領域（セル）の中心をサポート球中心の候補とする．ただし MPU では八分木の各セルに 1 つのサポート球が割り当てられるのに対し，本生成法では後述のアルゴリズムにより，1 つのサポート球が複数のセルを被覆する．そのため，サポート球の数をセルの数よりも減らすことができる．グラフカットは大域的計算手法であるため計算時間がかかりがちなので，グラフのノードの基となるサポートの数を削減することで計算時間を短縮する．被覆生成の詳細は 2.1.2 項に示す．

### 2.1.1 局所近似

各サポート球  $s_i$  に対し  $s_i$  内部の入力点群  $\{p_j\}$  が表す表面を値 0 の等値面で近似する関数  $g_i(\mathbf{x})$  を定める．関数  $g_i(\mathbf{x})$  のサポートは  $s_i$  である．サポート  $s_i$  の中心を  $c_i$  で表すと， $g_i(\mathbf{x})$  はサポート内部の入力点群に対する重み付き最小 2 乗フィッティングを用いて

$$g_i(\mathbf{x}) = \mathbf{m}_i \cdot (\mathbf{x} - \mathbf{d}_i), \quad (2)$$

$$\mathbf{m}_i = \frac{\sum_j w_i(p_j)\mathbf{n}_j}{|\sum_j w_i(p_j)\mathbf{n}_j|}, \quad \mathbf{d}_i = \frac{\sum_j w_i(p_j)\mathbf{p}_j}{\sum_j w_i(p_j)} \quad (3)$$

と求まる．MPU では  $g_i(\mathbf{x})$  として 2 次式を用いているため高い近似精度が期待できる一方，ノイズには弱い．本手法では 2 次式よりもノイズの影響を受けにくい 1 次式を採用した．また，出力する表面メッシュは形状の 1 次近似であり，メッシュの粗密にはサポート球の粗密が直接影響することから， $g_i(\mathbf{x})$  として 1 次式を用いるのは適切だと考えられる．

重み関数  $w_i(\mathbf{x})$  には MPU と同様に，以下の式で定義される 2 次の B-spline 関数を用い

た：

$$w_i(\mathbf{x}) = b_2 \left( \frac{3\|\mathbf{x} - \mathbf{c}_i\|}{2r_i} \right), \quad (4)$$

$$b_2(d) = \begin{cases} 0 & (3/2 \leq d) \\ (3/2 - d)^2 / 2 & (1/2 \leq d < 3/2) \\ -d^2 + 3/4 & (0 \leq d < 1/2) \end{cases} . \quad (5)$$

半径  $r_i$  はユーザ指定の許容近似誤差  $\varepsilon$  を超えない範囲でできる限り大きいものとする：

$$\max_{|\mathbf{p}_j - \mathbf{c}_i| < r_i} |g_i(\mathbf{p}_j)| = \varepsilon. \quad (6)$$

これを満たす  $r_i$  は二分法により求められる．ただし  $s_i$  が  $c_i$  の存在するセルを被覆できるよう， $r_i$  の大きさには下限を定める必要がある．本研究では， $c_i$  が存在するセルの対角線の長さの 0.75 倍に設定した． $r_i$  の大きさは八分木のセルサイズ依存なので最大分割レベルによって決まる．サポートの最大数は 3 次元の場合  $2^{3 \times \maxLevel}$  個（ $\maxLevel$  は最大分割レベル）となる．サポートの数はメモリ消費量に関わるため， $\varepsilon$  よりも最大分割レベルを優先した．

### 2.1.2 サポート球による被覆

点群のバウンディングボックスに対してサポート球による被覆を生成するアルゴリズムを以下に示す．

#### サポート球による被覆生成アルゴリズム

- (1) バウンディングボックスを八分木で分割する．サポート球の中心候補の集合  $C$  をセルの中心の集合で初期化する．
- (2)  $C$  からランダムに 1 点  $c_i$  を選ぶ． $c_i$  を中心とするサポート球  $s_i$  の半径  $r_i$  と局所近似関数  $g_i(\mathbf{x})$  をそれぞれ式 (2)，(6) により決定する．
- (3)  $s_i$  で被覆されたセルの中心を  $C$  から除く． $C = \emptyset$  ならば終了，そうでなければステップ (2) へ戻る．

ステップ (1) では，すべての入力点が最大階数（ユーザ指定とする）のセルに含まれるよう空間を八分木で分割する．すでに述べたとおり本手法では被覆を基に出力メッシュを生成する．隣り合うサポート球の大きさが極端に違くと出力メッシュでは次数の大きな頂点が現れる．そこで隣り合うサポート球の大きさの差を緩和するために，八分木の隣接セルとの

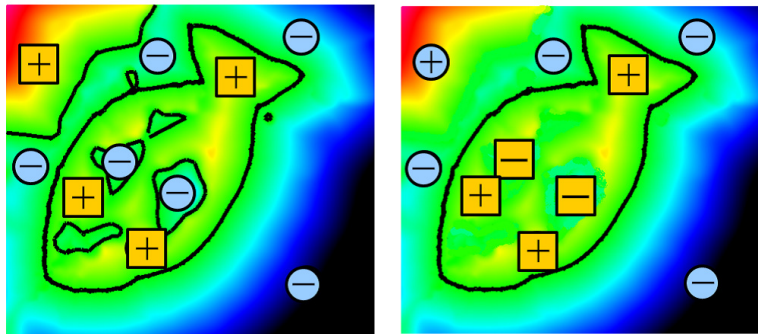


図 3 PU (左) とグラフカット (右) で内外判定を行い表面 (黒線) を再構成した結果. 背景の色は  $f(x)$  の値 (大: 赤, 小: 青) を示す. 内外判定結果を丸い青点 (外部) と四角い黄色の点 (内部) で示す.  $+/-$  は  $f(x)$  の符号を表す

Fig. 3 Results of inside/outside classification via PU (left) and the Graph-cut (right). Reconstructed surfaces are shown with black lines. The values of  $f(x)$  are indicated with red for high and blue for low. Circular blue points mean the outside vertices and square yellow points inside. The signs  $+/-$  mean the signs of  $f(x)$ .

分割レベルの差はただか 1 とする.

本手法ではステップ (2) でランダムにサポート球を生成する. 最適な中心位置を追求するならば, たとえばパブルシミュレーション<sup>20)</sup> や重心ポロノイ図<sup>10)</sup> を用いてサポート球の最適な位置を再計算することも原理的には可能である. しかし一般的な状況下ではサポート数は非常に多く計算量が膨大になってしまう. 一方, 中心をランダムに決定する手法は非常に単純かつ高速である. さらに実験で十分満足のいく結果が得られるため, 本手法ではランダムに決定するとした.

ステップ (3) では, 8 頂点すべてが  $s_i$  に含まれているか否かでセルが被覆されたかどうか判定する. 通常, 空間の被覆判定は大域的処理であるが, 八分木を用いたことおよびセル依存の最小半径値を設定したことで局所的な処理のみで判定が行える.

## 2.2 大域的内外判定

PU は局所的な手法であるため, 細部の形状を精細に近似できる半面, 大きいノイズや異常値に弱い. 特に入力データに異常値が存在するような場合には, 本来存在しなかった突出部やスキャンした物体以外の物体のような過剰な表面が生成されてしまうという問題が深刻であった. そこで近似関数  $f(x)$  を生成した後, 大域的な手法であるグラフカットを用いた内外判定を行い, 過剰な面の出ない安定した近似表面を得ることを狙う (図 3 および図 4).

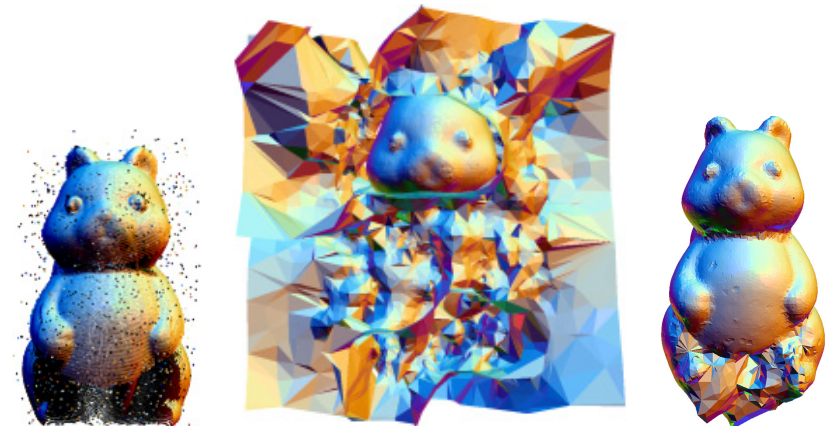


図 4 多数の異常値を含むデータ (左) から生成した表面メッシュ. PU のみで再構築した結果 (中央) と, PU とグラフカットによる再構築の結果 (右)

Fig. 4 Reconstructed surfaces for points with many outliers (left). The results with PU (center), and with PU and Graph-cut (right).

グラフカットではまず, 対象となる重み付きグラフにソースノードとシンクノードと呼ばれる 2 つのノード (ターミナルノード) を指定する (図 5 左). カットにより, グラフのその他のノード (非ターミナルノード) はソースノードを含む集合とシンクノードを含む集合の 2 つの集合に分けられる (図 5 右). 表面メッシュ生成では, サポート球の中心を非ターミナルノードに 1 対 1 に対応づけ, これらをソースノード側 (物体の内部) とシンクノード側 (物体の外部) とに分類することでバウンディングボックス内の空間分割を行う. ソースノード側のノードとシンク側のノードを両端点とするグラフの辺が境界を近似する (図 5 中央).

### 2.2.1 四面体メッシュ生成

まず, グラフの元となる四面体メッシュを生成する. 本手法では, PU で生成したサポート球の中心を頂点とし, 半径の 2 乗を重みとする重み付きドロネー四面体分割を用いる. これは半径  $r_i$  のサポート中心  $c_i$  を生成元とし, 距離<sup>\*1</sup>  $\|x - c_i\|^2 - r_i^2$  で定義されるラゲ

\*1 この距離値は負になる場合があるが, その場合は上記の距離を最大のサポート球の半径を加えた距離で置き換えればよい.

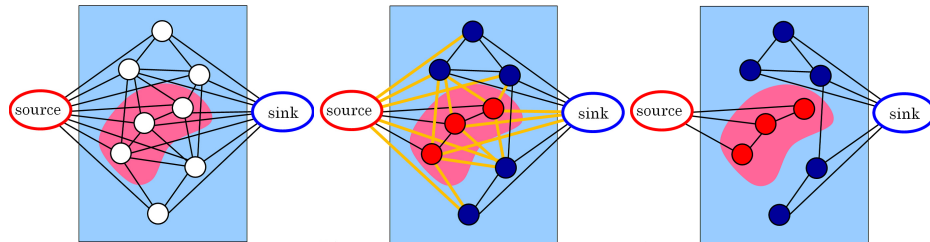


図5 表面再構成におけるグラフカットの働き．ソースノード（内部に対応）とシンクノード（外部）を指定し（左），最小カット（黄色）を求め（右），その他のノードを二分する（右）

Fig. 5 The functions of the Graph-cut for surface reconstruction. With the assigned source (inside) and sink (outside) (left), we calculate the minimum cut (yellow) (center), and then classify the other nodes into two groups (right).

ルドロネー四面体分割である<sup>11)</sup>．上記の重み付きドロネー四面体分割を用いると，ノイズの影響などによって生じた小さいサポート球は大きなサポート球に含まれやすくなり，中心が四面体メッシュの頂点として現れなくなる．その結果，メッシュの要素数を削減できる．

四面体メッシュの接続関係の構築には通常は大域的な操作が必要となる．今回はバウンディングボックスがサポート球によって被覆されているため，注目する生成元に対応するサポート球と交わるサポート球の集合からすべての3つ組みに対して空円性を調べれば十分である．つまり局所的な判定処理で接続関係を構築することができる．

### 2.2.2 グラフ生成

グラフの頂点は四面体メッシュの全頂点およびソースノードとシンクノードからなる．グラフの辺は，四面体メッシュの全辺と，ターミナルノードと四面体メッシュの各頂点に接続する辺（ターミナルエッジ）からなり，各辺には重みが与えられている．またカットの重みとは，カットに含まれる辺の重みの総和で定義される．本手法では物体表面を重みが最小のカット（最小カット）で近似する．この最小カットを求める操作はグラフカットと呼ばれる．グラフカットは以下のエネルギーを最小化することにより行う<sup>5)</sup>：

$$E = \sum_{l_t \in L_t} w(l_t) + \sum_{l_{nt} \in L_{nt}} w(l_{nt}) \quad (7)$$

ここで  $L_t$  はカットに含まれるターミナルエッジの集合を， $L_{nt}$  はカットに含まれる非ターミナルエッジの集合を表す．なおグラフカットの実装には Boykov and Kolmogorov<sup>5)</sup> を用いた．

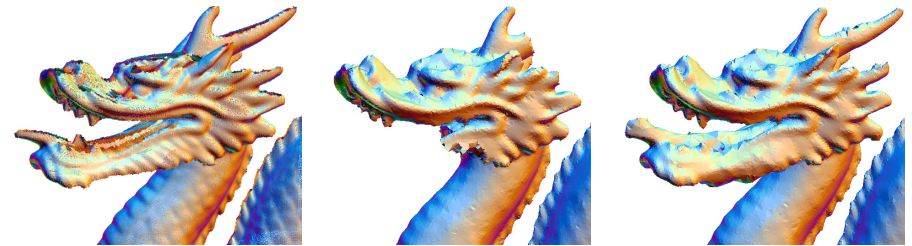


図6 入力点群（左）に対する異なる  $k$  の値による結果の比較． $k = 3$ （中央）， $k = 15$ （右）

Fig. 6 Comparison of the results for sampling points (left) with  $k = 3$  (center) and  $k = 15$  (right).

本稿では前述のとおり，ソースが物体内部，シンクが外部に対応するとする． $f(x) \geq 0$  となる頂点（PU では内部と考えられる頂点）が内部に分類されやすいよう，ターミナルノードと頂点  $v_i$  を結ぶターミナルエッジ  $l_t(v_i)$  に対する重みを以下のように設定する．ターミナルノードがソースノードの場合：

$$w(l_t(v_i)) = \begin{cases} 0 & (f(v_i) < 0) \\ k |f(v_i)| / \text{ave}(v_i) & (f(v_i) \geq 0), \end{cases} \quad (8)$$

ターミナルノードがシンクノードの場合：

$$w(l_t(v_i)) = \begin{cases} k |f(v_i)| / \text{ave}(v_i) & (f(v_i) < 0) \\ 0 & (f(v_i) \geq 0). \end{cases} \quad (9)$$

$\text{ave}(v_i)$  は  $v_i$  に接続する非ターミナルエッジの長さの平均値である．この値で割ることで重みを無次元にし，良好な実験結果が得られた．定数  $k$  はターミナルエッジの重みと非ターミナルエッジの重みとの比であり，計算機実験では  $k = 5 \sim 15$  で良い結果が得られた．図6に示すとおり， $k$  の値が小さいと欠け，大きいと膨張する傾向がある．これは  $k$  が大きくなると物体表面から離れたところまでソース（内部）に含まれやすくなるような重み付けをしているためである（式(8)，(9)）．

頂点  $v_i, v_j$  を両端点とする非ターミナルエッジ  $l_{nt}(v_i, v_j)$  の重みは

$$w(l_{nt}(v_i, v_j)) = |f(v_i) + f(v_j)| / |l_{nt}(v_i, v_j)| \quad (10)$$

とする．ここで  $|l_{nt}(v_i, v_j)|$  は辺  $l_{nt}(v_i, v_j)$  の長さを表す． $f(x)$  は表面からの符号付き距離を表すので，この重みは表面から離れた辺に対しては大きくなり，表面と交わる辺では両

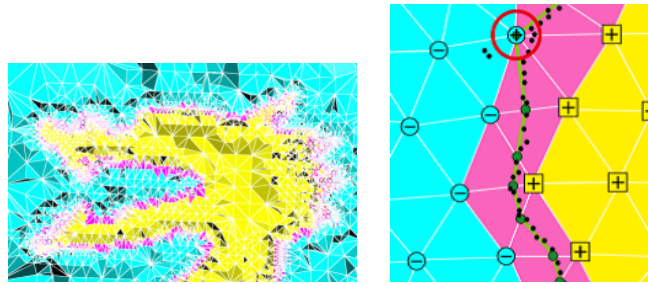


図 7 四面体メッシュ頂点に対してグラフカットによる内外判定を行い、四面体を識別した結果。四面体の塗り分けは、青：全頂点が外部、黄：全頂点が内部、ピンク：内外ともに持つ、とした。左：Stanford dragon モデルへ適用した結果の断面図。右：2次元における模式図。右図においてグラフカットの結果を、丸い青点（内部）と四角い黄色の点（外部）で示す。+/- は  $f(x)$  の符号である。黒点は入力点、緑の点および線は生成される三角形メッシュの頂点および面を表す

Fig. 7 Classification of the tetrahedra according to the inside/outside classification of the vertices via the Graph-cut. Tetrahedra with four outside vertices are shown in blue, Tetrahedra with for inside vertices are yellow, and the others are pink. Left: a cross section of a tetrahedral mesh for the head of the Stanford dragon. Right: a 2D-version example. In the right image, the results via the Graph-cut are shown with circular blue points (inside) and squared yellow points (outside). The signs +/- indicate the signs of  $f(x)$ . Black points mean the input points, and the green points and lines mean the triangular mesh.

端点の  $f(x)$  の符号が異なるため小さくなる。したがって、表面と交わる辺が最小カットに含まれやすくなる。

PU による符号付き距離  $f(x)$  が正しければ、 $f(x)$  の符号による内外判定とグラフカットによる内外判定とは一致するはずである。しかし実際には、両者の判定が異なる頂点が存在する（図 7 右の赤丸で囲まれた点）。このような現象が生じるのは、異常値などの影響で局所近似を誤るためである。このように PU のみに基づいた表面メッシュ生成では、近似に失敗した点の周辺に過剰な表面メッシュが生成されてしまう。一方グラフカットで得られる内外判定は大域的なので、異常値の影響が軽減され、過剰な面の生成が抑えられる（図 3, 4）。本項で述べたグラフカットによる内外判定を用いた表面メッシュ生成法を 2.3 節で説明する。

### 2.3 表面メッシュ生成

表面メッシュの生成はマーチングテトラヘドラ法で行う（図 8）。グラフカットで内部となった頂点と外部となった頂点を両端点とする辺上に、表面との交点（図 7 右の緑の点）を

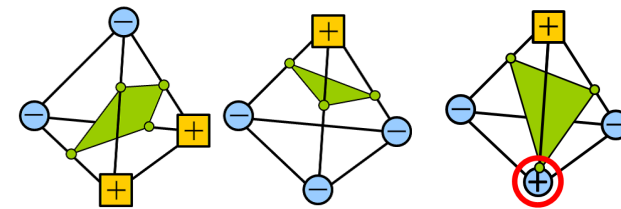


図 8 マーチングテトラヘドラのパターン。グラフカットによる判定結果を黄色の四角形（内部）、青の丸（外部）で表示。+/- は  $f(x)$  の符号を示す。赤い丸で囲った頂点はグラフカットの判定と  $f(x)$  の符号による判定が異なる頂点

Fig. 8 The surface generating patterns of the marching tetrahedra. The results with the Graph-cut are indicated with yellow squares (inside) and blue circles (outside). The signs +/- mean the signed of  $f(x)$ . The vertex surrounded by a red-circle means a vertex whose classification with the Graph-cut and the sign of  $f(x)$  are not consistent.

生成する。両端点での  $f(x)$  の符号が異なるときは、 $f(x) = 0$  となる点が交点となる（図 8）。PU による関数近似に失敗している場合には、両端点の  $f(x)$  の符号が同じになる。形状は後で平滑化により整えることにして、まずは水密なメッシュを作ることを目指して  $|f(x)|$  の値が小さい方の端点上に交点を生成する。これは、 $f(x)$  は表面からの符号付き距離を近似するので、 $|f(x)|$  が小さい頂点の方がもう一方よりも表面に近いと考えられるためである。

上記の同符号の  $f(x)$  の端点を持つ辺上に生成した交点と、スキャンによるサンプリングが不足している個所では、得られた表面メッシュの形状は滑らかでない（図 9）。そこでこれらの箇所にバイラブラシアンスムーシング<sup>15)</sup>を行う。バイラブラシアンスムーシングは特定の頂点のみ移動して周囲の動かない部分と滑らかに接続するようにすることで、メッシュの滑らかさを向上させる。本手法では PU で詳細に得られた形状はそのままに、異常値やサンプリング不足のために PU では正確に得られなかった箇所の頂点をバイラブラシアンスムーシングで平滑化することで、望ましい形状の表面メッシュを生成する。本手法では、エッジに関する処理であるグラフカットを採用しているため、サンプリング不足の判定もエッジごとに行うこととする。エッジの端点が入力点を 10 個以上含まないサポートに対応する場合、そのエッジはサンプリングが不足している個所にあると考え、エッジ上に生成した表面メッシュの頂点に対してバイラブラシアンスムーシングを行う。

バイラブラシアンスムーシングの計算方法として、正方向のラブラシアンスムーシング  $v'_i = v_i + \lambda u$  と逆方向のラブラシアンスムーシング  $v''_i = v_i - \lambda u$  を交互に行う方法を用

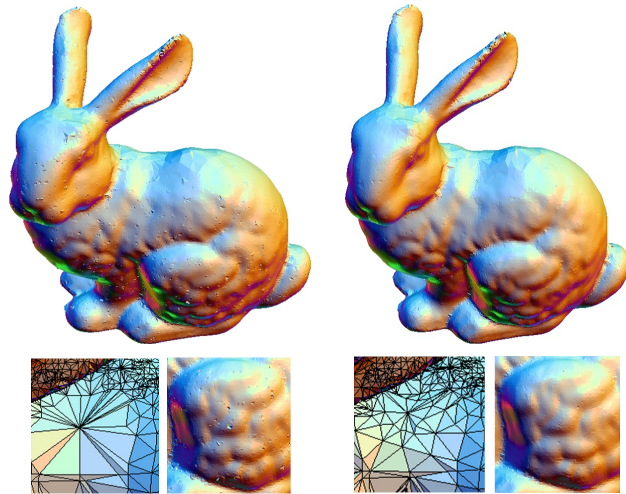


図9 バイラプリアンスムービング前(左)および後(右)の表面メッシュ. 下段は各メッシュの背中部分のワイヤフレーム表示と左後脚の拡大図

Fig. 9 Triangular mesh before bi-Laplacian smoothing (left) and after (right). The close-ups of the backs and the left hind feet are shown in the bottom.

いた<sup>15),21)</sup>. ここで  $U = \frac{1}{n} \sum_j v_j - v_i$ ,  $U' = \frac{1}{n} \sum_j v'_j - v'_i$  であり,  $n$  は頂点  $v_i$  の接続数,  $v_j$  は  $v_i$  に隣接する頂点を表す. 上述のとおりこのスムービングは対象となる頂点のみに対して行われ他の頂点は動かないので, PU により詳細に近似されたと考えられる個所の形状が平滑化されることがない. 本稿に示した結果は,  $\lambda = 0.5$ , 繰返し回数を 30 回として得られたものであり, スムービング操作による頂点の移動はほぼ収束した状態である.

### 3. 結 果

本研究と同じく表面再構成を目標としてグラフカットを用いた手法<sup>13)</sup>は大域的手法のみに頼っているために, 局所的手法をベースとした本手法の方が精度の点で勝っていることは比較するまでもなく明らかである. また PU を基とした局所的手法のうち, 文献 16) は点群のノイズ除去のための手法であり, 本研究とは目的が異なる. 同じく PU に基づく局所的手法である文献 7) は法線の向きを揃えているが, それだけでは欠損部分に生じた穴が埋まらない. これらの理由から本研究では提案手法の有効性を示すにあたり比較対象とし

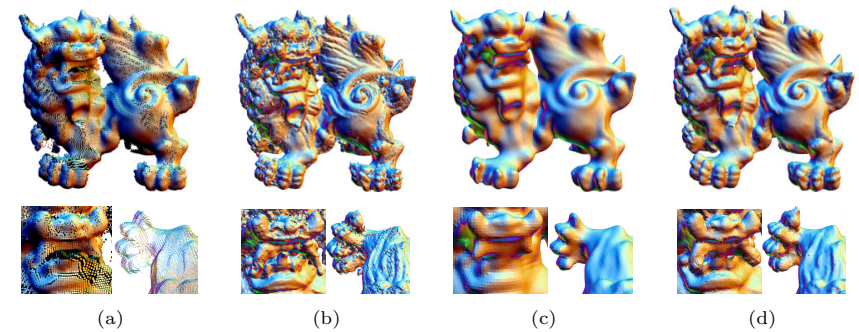


図10 シーサのスキャンデータ (a) に対する結果. MPU (b), Poisson 表面メッシュ生成法 (c), 本手法 (d) による結果. 下段は口と右前脚の拡大図

Fig. 10 Results for a scanned data of the Siisa-A (a). The result by MPU (b), Poisson surface reconstruction (c), and proposed algorithm (d). The close-ups of the mouths and the right forefeet are shown in the lower column.

て, 提案手法の基となった MPU<sup>17)</sup> と, 大域的な手法ながら非常に詳細に表面を再構成する Poisson 表面メッシュ生成法<sup>14)</sup> を選んだ.

素焼のシーサのスキャンデータに対する結果を, Ohtake らによる MPU による結果と, Kazhdan らによる Poisson 表面メッシュ生成法の結果と比較した (図 10). このデータはノイズが多く, 脚先や口の部分に欠損があり, また異常値を含んでいる. MPU の誤差許容値は  $1.0 \times 10^{-3}$ , Poisson 表面メッシュ生成法では八分木の最大階数を 10, 提案手法では誤差許容値を  $1.0 \times 10^{-3}$ , 八分木の最大階数を 8,  $k = 10$  とした. また, 本稿中のデータはすべてバウンディングボックスの最大辺長が 1 になるようにスケールしている. MPU では尾の模様のような非常に細かい形状も再現できる. しかし, ノイズの影響で近似が不安定になり, 頭部にみられるように本来の物体には存在しない小さな凹凸が多数出現している. 一方 Poisson 表面メッシュ生成法はこのようなノイズの多いデータに対しても滑らかな表面メッシュを生成できる. しかしデータに欠損や異常値が存在する場合, それらに対する平滑化の影響が全体に及んで細かい形状まで過剰に平滑化してしまう傾向がある. 本データに対してもその傾向は明らかで, 顔周辺のノイズの影響は軽減されているものの, 尻尾の模様などは非常に弱い凹凸としてしか再現されていない. またサンプリングが不足している部分で縮む傾向もある. サンプリング密度の少ない右前脚と比較すると, Poisson 表面メッシュ生成法の結果は明らかに縮んでしまっていることが観察される. それに対し本手法の結

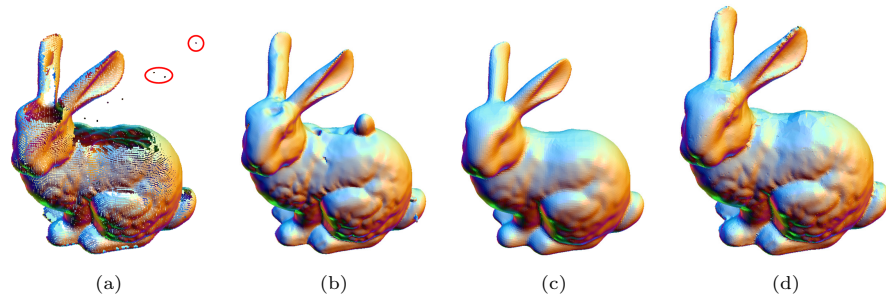


図 11 異常値 (赤丸で囲んだ点) と欠損を含むデータ (a) に対する結果の比較. MPU (b), Poisson 表面メッシュ生成法 (c), 提案手法 (d) による結果

Fig. 11 Comparison for a low quality data with outliers (red-circled) and lacks of sampling (a). The results obtained with MPU (b), Poisson surface reconstruction (c), and proposed algorithm (d).

果ではこのような縮みは起こらない. MPU で顔周辺に生じたようなノイズの影響もほとんどみられない. 胴体部分も滑らかな表面で再現されている. さらに, 脚の先や尻尾にある線状の模様も再現できている.

図 11 は, 頭と背中部分に大きな欠損と複数の異常値を含むデータに対する結果の比較である. いくつかの異常値は背中欠損部分に近い位置にある. MPU の誤差許容値は  $1.0 \times 10^{-3}$ , Poisson 表面メッシュ生成法では八分木の最大分割レベルを 9 とし, 提案手法の誤差許容値は  $1.0 \times 10^{-3}$ , 八分木の最大分割レベルは 8,  $k = 10$  とした. MPU ではウサギの目や, 胸・腿にある凹凸が精細に表現できている. しかし欠損箇所である背中に, 欠損と付近の異常値による大きな凹凸ができており, 頭も欠損の影響を受け凹んでいる. また, 尾の付近に過剰な面が生成されている. それに対し Poisson 表面メッシュ生成法で生成した表面メッシュには目立った変形がない. 全体としての形状は望ましいものの, 異常値の影響を軽減するためのスムージングにより, 目や腿の様子は失われている. 提案手法では MPU を基に欠損や異常値の影響をグラフカットで軽減するため, 細かい形状を損なうことなく変形の少ない表面メッシュを生成できる. 結果を観察すると, 目や腿の様子は MPU と同程度の細かさまで表現されているながらも大きな変形がないことが分かる.

表 1 に高精度で表面再構成した場合の処理時間を示す. 処理時間の半分程度が四面体メッシュ生成に費やされていることが分かる.

表 1 入力パラメータ, 計算時間及び生成されたメッシュの頂点数

Table 1 Examples of the values of parameters, computational speed and the numbers of vertices of the generated surface meshes.

Model	#Points	Error tol.	Max level	#Supports	Timings	
					(tet)[sec]	#Vertices
Dragon	2,106 K	$4.0 \times 10^{-3}$	9	257 K	2,648 (1,147)	317 K
Buddha	3,260 K	$4.5 \times 10^{-3}$	9	207 K	2,963 (1,459)	261 K
Si-sa-a	429 K	$1.0 \times 10^{-3}$	8	374 K	3,057 (2,305)	352 K

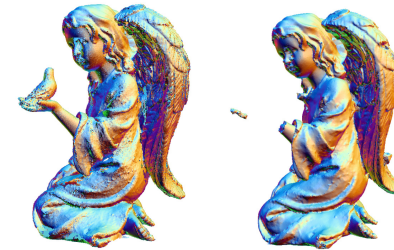


図 12  $k$  を一定値とすることによる問題例. 左: 入力点群. 右:  $k = 2.3$  とした結果

Fig. 12 A problem caused by a globally constant  $k$ . Left: input points. Right: result obtained with  $k = 2.3$ .

#### 4. ま と め

局所的な手法である PU に大域的な手法であるグラフカットを組み合わせることで, 細かい形状を再現可能かつ異常値や欠損に対して頑健な表面メッシュ生成手法を提案した. 実行時間は入力点数が数百万点のデータに対して数十分程度であり, その多くが四面体メッシュ生成に費やされる. 実装における今後の課題として, 四面体メッシュの計算の高速化があげられる.

本手法は PU に基づいているため, 局所近似において法線が重要な役割を果たしている. したがって入力点に与えられた法線が不正確なときは関数近似がうまくできない場合があり, その対処が求められる. また, グラフの辺の重みづけに用いる  $k$  の決め方も課題である. 図 12 は  $k = 2.3$  とした結果である. 左腕が欠けてしまっているが, 顔や羽など他の部分は表現されていることが観察される. このことから, 現在  $k$  はグラフ全体で一定値としているが, 適応的に値を定めればより良い結果が得られると期待される.



謝辞 本研究の一部は、科学研究費補助金（基盤（A））、No.22246018 によって行われた。また本研究で用いたデータの一部は the Stanford 3D Scanning Repository および Berkeley Computer Animation and Modeling Group より提供されたものである。

### 参 考 文 献

- 1) Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D. and Silva, C.T.: Point set surfaces, *IEEE Visualization 2001*, pp.21–28 (2001).
- 2) Amenta, N., Bern, M. and Kamvysselis, M.: A new Voronoi-based surface reconstruction algorithm, *Proc. ACM SIGGRAPH 1998*, pp.415–421 (1998).
- 3) Amenta, N., Choi, S. and Kolluri, R.: The Power Crust, *Proc. 6th ACM Symposium on Solid Modeling*, pp.249–260 (2001).
- 4) Amenta, N. and Bern, M.: Surface reconstruction by Voronoi filtering, *SCG '98: Proc. 14th Annual Symposium on Computational Geometry*, pp.39–48, ACM, New York, NY, USA (1998).
- 5) Boykov, Y. and Kolmogorov, V.: An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.26, No.9, pp.1124–1137 (2004).
- 6) Carr, J.C., Beatson, R.K., Cherrie, J.B., Mitchell, T.J., Fright, W.R., McCallum, B.C. and Evans, T.R.: Reconstruction and Representation of 3D Objects With Radial Basis Functions, *Proc. ACM SIGGRAPH 2001*, pp.67–76 (2001).
- 7) Chen, Y.-L. and Lai, S.-H.: A Partition-of-Unity Based Algorithm for Implicit Surface Reconstruction Using Belief Propagation, *SMI '07: Proc. IEEE International Conference on Shape Modeling and Applications 2007*, Washington, DC, USA, IEEE Computer Society, pp.147–155 (2007).
- 8) Dey, T.K. and Goswami, S.: Tight Cocone: A water-tight surface reconstructor, *Proc. 8th ACM Sympos. Solid Modeling Applications*, pp.127–134 (2003).
- 9) Dey, T.K. and Goswami, S.: Provable surface reconstruction from noisy samples, *Proc. 20th ACM Sympos. Comput. Geom.* (2004).
- 10) Du, Q., Faber, V. and Gunzburger, M.: Centroidal Voronoi Tessellations: Applications and Algorithms, *SIAM Review*, Vol.41, No.4, pp.637–676 (1999).
- 11) Edelsbrunner, H.: *Geometry and Topology for Mesh Generation (Cambridge Monographs on Applied and Computational Mathematics)*, Cambridge University Press (2006).
- 12) Guennebaud, G. and Gross, M.: Algebraic point set surfaces, *SIGGRAPH '07: ACM SIGGRAPH 2007 Papers*, pp.23.1–23.9 (2007).
- 13) Hornung, A. and Kobbelt, L.: Robust reconstruction of watertight 3D models from non-uniformly sampled point clouds without normal information, *SGP '06: Proc. 4th Eurographics Symposium on Geometry Processing*, pp.41–50 (2006).
- 14) Kazhdan, M., Bolitho, M. and Hoppe, H.: Poisson surface reconstruction, *SGP '06: Proc. 4th Eurographics Symposium on Geometry Processing*, pp.61–70 (2006).
- 15) Kobbelt, L., Campagna, S., Vorsatz, J. and Seidel, H.-P.: Interactive multi-resolution modeling on arbitrary meshes, *SIGGRAPH '98: Proc. 25th Annual Conference on Computer Graphics and Interactive Techniques*, pp.105–114 (1998).
- 16) Li, J. and Wang, R.: Robust denoising of point-sampled surfaces, *WSEAS Trans. Comp.*, Vol.8, No.1, pp.153–162 (2009).
- 17) Ohtake, Y., Belyaev, A., Alexa, M., Turk, G. and Seidel, H.-P.: Multi-level partition of unity implicits, *ACM Trans. Graphics*, Vol.22, No.3, pp.463–470 (2003).
- 18) Renka, R.J.: Multivariate interpolation of large sets of scattered data, *ACM Trans. Mathematical Software*, Vol.14, No.2, pp.139–148 (1988).
- 19) Sharf, A., Lewiner, T., Shklarski, G., Toledo, S. and Cohen-Or, D.: Interactive topology-aware surface reconstruction, *ACM Trans. Graphics*, Vol.26, No.3, pp.43.1–43.9 (2007).
- 20) Shimada, K. and Gossard, D.C.: Bubble mesh: Automated triangular meshing of non-manifold geometry by sphere packing, *SMA '95: Proc. 3rd ACM Symposium on Solid Modeling and Applications*, pp.409–419 (1995).
- 21) Taubin, G.: A signal processing approach to fair surface design, *Computer Graphics (Proc. SIGGRAPH '95)*, pp.351–358 (1995).

(平成 22 年 11 月 1 日受付)

(平成 23 年 4 月 8 日採録)

### 推 薦 文

本論文では、計測された点群データをもとに陰関数曲面をフィッティングする手法を提案しています。計測点群には、その性格上、ノイズの混入が避けられませんが、最適化手法としてグラフカットを用いることにより、高品質の陰関数曲面を比較的高速に生成することを可能にしています。これは情報処理学会会員にとっても興味深くかつ有益な研究であり、論文誌での掲載価値を十分に有していると判断しました。よって、本研究会は本論文を推薦論文として推薦するものです。（グラフィクスとCAD研究会主査 山口 泰）



長井 超慧 (正会員)

2010年東京大学大学院博士課程修了・工学博士, 2010年理化学研究所研究員, 2011年よりウィーン工科大学博士研究員. 研究分野: 形状モデリング, コンピュータグラフィックス. 加入学会: 応用数学会, 精密工学会.



大竹 豊

2002年会津大学大学院博士課程修了・コンピュータ理工学博士, 2002年マックスプランク情報科学研究所博士研究員, 2004年理化学研究所研究員, 2007年東京大学講師 (精密機械工学専攻). 2011年同准教授. 研究分野: 形状処理, コンピュータグラフィックス. 加入学会: 精密工学会.



鈴木 宏正 (正会員)

1986年東京大学大学院博士課程修了・工学博士, 1987年東京大学助手 (教養学部), 1988年同講師, 1990年同助教授, 1994年東京大学助教授 (精密機械工学専攻), 2003年同教授, 2004年より同大先端科学技術研究センター. 研究分野: 形状モデリング, デジタルエンジニアリング. 加入学会: ACM, IEEE, 日本機械学会, 精密工学会等.