

メタボールとボクセルデータの統合化造形システムの開発

福山大学工学部 松田 亮 治
 福山大学工学部 正会員 西田 友 是

〈あらまし〉 本論文では、造形デザインや人体モデルの作成に有用なインタラクティブモデリングシステムを提案する。3次元モデルの表現法には、多角形、ボクセル表現、メタボールおよびパラメトリック曲面などがあるが、創造的なデザインを行うには、メタボールやボクセルデータを扱えるシステムが有用である。提案法は、メタボール、ボクセルデータおよび多角形データが統合して扱えるシステムであり、次の特徴をもつ。3次元モデルの編集(追加、修正、自由形状変形)をスクリーン上でインタラクティブに直接操作でき、データの表現形式相互間の変換(ボクセルデータからメタボールへの変換など)が行える。さらに、提案法の有効性を示すために、筆者らが開発している衣服の着装シミュレーションに用いるメタボール人体モデルを適用例として示した。

〈Summary〉 We propose a useful interactive modeling system for a sculptural design or for a human body model. We have representation methods for three-dimensional object using polygon meshes, volume (voxel) data, metaballs and parametric surfaces. For creative design, a modeling system handling metaballs and voxel data would be very useful, so this paper proposes such a system which integrates metaballs, voxel data and polygon meshes. This has the following advantages: A three-dimensional model can be edited (e.g., addition, removal, free-form deformation) on a screen interactively. The various representation forms can be mutually converted (e.g., from voxel data to metaballs). Furthermore, the metaball human body model to be used for the dress simulation we developed serves as an excellent example to show the effectiveness of the proposed method.

1. ま え が き

計算機支援形状設計(CAD)は色々な分野で行われている。自動車、電化製品などの工業製品、アパレル製品など種々の分野で曲面処理は重要である。曲面の処理は工業デザインに欠かせないものである。Bézier曲面などのパラメトリック曲面を対象としたCADシステムは多く開発されているが、それに比べボクセルデータやメタボールを対象としたCADシステムは、まだこれからの発展が期待されている。本稿では、造形デザインや人体モデルの作成に有用なインタラクティブモデリングシステムを提案する。なお、人体モデルは衣服の着装シミュレーションのようなアパレルCAD技術に応用できる。

3次元モデルの表現法には、多角形、ボクセル表現、

メタボールおよびパラメトリック曲面などがある。自由曲面を多く含む物体を多角形で表現する場合、非常に多くのデータ量となり、その変形まで考慮すると限界がある。メタボールは人体、動物などの表現に優れており、そのデータ量は多角形表現に比べると1桁少ない量での表現が可能と言われている。ボクセル表現は、ボクセルのサイズにもよって異なるが、比較的記憶容量が大きくなるが、不規則な形状や半透明物体を含み種々のものを表現できる。

本論文では、こうした種々の3次元データの表現法の特徴を活かせる形式で3次元物体を生成・修正し、その応用目的に応じて表現法間のデータの相互変換を行うことが可能なシステムについて論じる。この相互変換が必要なものとして、次のようなものがある。例えば、衣服の着装シミュレーション^{1)~3)}の場合、それに用いる人体モデルにはメタボール表現が適している^{2),3)}。実測された人体の距離データに基づいた多角形表現されたデータを入力し、それをボクセルデータに変形し、必要に応じ

“Development of a Modeling System Integrating Metaballs and Voxel Data” by Ryouji MATSUDA, Tomoyuki NISHITA (Member) (Fukuyama University).

て体型の局所変形を会話的にを行い、メタボールに変換することができる。

提案システムの機能および特徴は次のようである。

1) メタボール、ボクセル、および多角形(ポリゴンメッシュ)表現による3次元データが表示でき、許容範囲の誤差を含んで相互にデータ変換できる(対象物の精度に応じて、変換後のボクセルの分割数やメタボールの数も変化する)。

2) 3次元形状の編集(追加, 修正, 自由形状変形)をスクリーン上でインタラクティブに直接操作でき、修正の都度高速表示ができる。

3) スクリーン上で指定した点の3次元座標の検出(逆透視変換)が容易で、断面表示や等高線表示により、データの把握が容易である。

なお、開発したシステムでは、メタボールはレイトラッキング法によって、またボクセルデータはレイキャスティング法によって表示される。本論文では、効率のよい表示法のほか、メタボールへのデータ変換、ボクセルデータのスムーズな形状変形法を論じる。メタボールの表示およびFFD(自由形状変形)は、Bézier Clipping法⁹⁾を適用することにより実現できた。

2. 3次元データの表現形式と表示法

2.1 曲面の表現形式

造形デザインシステムにおいては自由曲面の表示が不可欠である。関数で曲面を表現する方法としては、パラメトリック表現とインプリシット(陰関数)表現がある。前者はBézier曲面、B-スプライン曲面、NURBSが、後者においてはメタボールが代表的な例である。一方、関数で表現できないものに対しては、ボクセルデータ(ボリュームデータ)による表現がある。本論文では、これらの表現形式について、形状の入力(モデリング)および表示法について論じるものである。

ボリュームデータは、データ量が多くかつ精度が劣る問題があるが、データ構造が簡単で、集合演算がきわめて簡単である。また、非常に不規則な形状を表すのに有効である。

メタボールは、人体などの生物、液体などの表現に適し、恐竜などの表現に多用されている。メタボールはBlinn⁵⁾、西村ら⁶⁾によって開発されたものである。3次元空間内に点電荷分布を与え、その等電位面として曲面形状を与えるものである。その際、球の中心座標、電荷の強さ(濃度)とその分布で与えられる。

多角形による自由曲面の表現は、グラフィクスワークステーションのGL(グラフィックライブラリ)を利用できるようになり、高速表示が可能となる。しかし、ス

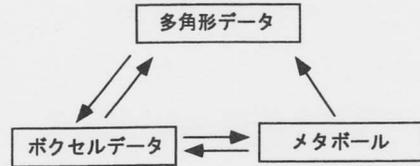


図1 データ形式の相互変換

Fig.1 Mutual conversions of object representation formats

ーズな曲面を精度よく表現するにはデータ量が多くなる問題がある。

本研究の目的の1つに、人体データのモデリングがあり、実用システムでは、実測データに基づいたものが有用である。実測されたものはレンジデータやCT画像などから得られる。これらは、多角形データあるいはボクセルデータとして提供されている。これらの表現形式はデータ量が多く、メタボールへ変換して保存する試みもなされている(4.1節参照)。本研究では、図1に示すように、相互にデータの表現形式が変換できるシステムを構築した。なお、多角形データ、ボクセルデータ、メタボールの三者の間で相互に全ての直接的な変換ができる必要はない。例えば、多角形データを直接メタボールに変換するのは困難であるが、多角形データを一度ボクセルデータに変換し、変換されたボクセルデータをさらにメタボールに変換することで可能になる。

メタボールで曲面を表現する場合、滑らかな曲面を表現するには多くのメタボールが必要となる。このような、多量のデータ入力が大変であるのはもちろんのこと、データの修正に手間がかかる。こうした、多量のデータを簡易に編集できるプログラムを作成した。なお、適用例として筆者らが開発した着装シミュレーション¹⁰⁾に用いる人体モデルではメタボールデータが適しており、この方法の場合パラメトリック曲面を必要としないので、ここでは除外して考えた。

2.2 メタボールの表示法

メタボールの表示法は、基本的にはすでに開発している方法を用いるが、説明の都合上、その概要を記す。

分布関数としては、正規分布⁵⁾、区分2次関数⁶⁾、4次関数、6次関数^{7),8)}が開発されている。濃度分布関数が高次の場合、スムーズな曲面が得られる。本論文においては、表現力の豊かな以下の6次関数を用いた。

$$f_i(r) = (-4/9)(r/R_i)^6 + (17/9)(r/R_i)^4 - (22/9)(r/R_i)^2 + 1 \quad (1)$$

ここで、 R_i はメタボール i の有効半径、 r はメタボールの中心 (x_i, y_i, z_i) から任意の点 P までの距離である。なお、 $r > R_i$ の範囲では濃度は0である。 n 個のメ

ボールがある場合、次式を満たす等値面(iso-surface)で自由曲面が定義される。

$$f = \sum_{i=1}^n q_i f_i - T = 0 \quad (2)$$

ここで、 T はしきい値、 q_i は中心での濃度である。また、負の濃度をもつメタボールにより凹ませたり穴をあけたりできる。

メタボールの表示には、レイ(視線)と濃度分布関数との交差判定が必要である。しかし、高次の場合には、レイと濃度関数との交点を解析的には求めることができない。この交点計算においては6次式を解く必要がある。開発した方法⁹⁾では、レイ上の濃度分布をBézier関数で表現し、Bézier曲線の凸包の性質を利用することにより、解の有無を高速に判定し、必要な1根のみを効率よく求めることができる。

3. 3次元データの編集および断面表示

造形システムにおいては、粘土細工にみられるような操作、すなわち、形状の変形(粘土の追加、削除、変形)が会話的に、ユーザにストレスを与えることなく行えることが望まれる。開発したシステムにおいては、スクリーン上で指定した位置における3次元空間での座標の検出、粘土の追加、削除、自由変形などの機能を持たせた。また、断面表示や等高線表示によりデータの把握を容易にし、データの入力ミスなどを検出しやすくした。

まず、メタボールの編集機能について論じる。

3.1 メタボールの編集

1) 座標検出

スクリーン上で指定した点に相当する等値面上での3次元座標、法線、その点に影響しているメタボール番号およびそれらの寄与率、さらに、その点を通過するレイ上に存在するすべてのメタボール番号を検出できるようにした。これは、いわゆる、逆透視変換である。これは、スクリーン上で指定した画素を通過するレイと等値面との交点を計算することに等しい。すなわち、レイトレーシングにより実現できるから文献⁹⁾のBézier Clipping法で交点を算出した。

2) メタボールの追加

スクリーン上において物体をクリックすると、その点に相当する3次元空間上にメタボールを追加できる。この場合、メタボールの半径、濃度、色およびオフセットを指定する。オフセットとは、メタボールの中心の等値面からの法線方向の変位である。一般に、オフセット値は0で、この場合、等値面上にメタボールの中心が配置される。

図2に示すように、スクリーン上において点Qを指

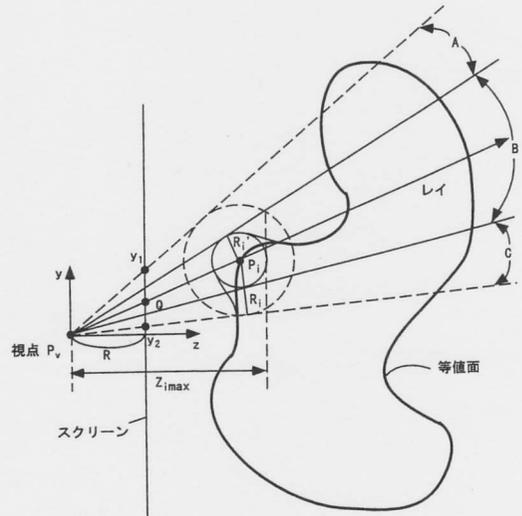


図2 追加ボールの走査
Fig.2 Scanning an added ball

定すると、等値面上での座標 P_i を検出し、その点にメタボールを生成する。確認のため、修正と同時にこのメタボールによって影響を受ける領域を局部的に再表示する。高速表示するには、いかに計算領域を最小にするかが重要である。すなわち、追加したメタボールの領域内のみ計算する。メタボールを包含するバウンディングボックスを利用して計算領域を判定する方法もあるが、この方法は最適な方法とは言えない。メタボールは球であるが、透視投影の場合、投影面上でこの領域は完全な円ではない。したがって、その領域を計算して効率化を計る。投影面(スクリーン)上で走査する y の範囲および各走査線での x の範囲は付録を参照されたい。この範囲のみについてレイトレーシングを行い部分表示すればよい。

上記のレイトレーシングについても不透明体の場合最適化できる。メタボールが他のメタボールと交差しない場合には、球として表示されるが、その半径はしきい値(式2参照)により決まる。この半径をしきい値半径 R_i' とする。したがって、この半径と有効半径 R_e からなる2重構造の球とみなして処理すると効率化が計れる。図2に示すように、レイとこれらの球との交差領域を考えると、次の最適化が計れる。(1)しきい値半径内の領域については(図2の領域B)、この球の後方のものは必ず不可視である。したがって、しきい値球より後方(図中の z_{max})のメタボールは無視してレイトレーシングをすればいい。(2)しきい値球の外部で有効半径球の内部(図中のA)については、有効半径球とレイとの交差区間に他のメタボールが存在しない場合、他のメタボール

メタボールとボクセルデータの統合化造形システムの開発

と融合する可能性はないので、その部分の可視面は変化しない(追加前にすでに表示されているものが可視)ので、それ以上の処理は不要である。

以上のように、スクリーンの計算領域を最小にし、かつレイトレーシングを最適化することにより、適用例で示すように高速表示が実現できた。

負の濃度を持つメタボールを用いることにより、3次元形状を削り取る(凹ませるあるいは穴を開ける)ことができる。また、既に入力したメタボールを検出し(前記1)参照)、そのパラメータ(位置、濃度、半径)を変更、あるいはメタボールを削除することもできる。

3) 断面表示

断面を表示することで、内部構造を把握することが容易となり、データエラーの発見に有用となる。集合演算などにより3次元データを切断し、それを再表示する方法もあるが、ここでは、データを変更しないで断面部分のみを表示する。

切断面をどのように指定するかも重要である。断面の平面方程式を入力する方式はユーザに負担である。そこで、スクリーン上において、物体上の3点をクリックすると、それらの3点を通過する平面を切断面とする方式とした。断面部分の表示は以下の手順で行う(図3参照)

(1) メタボールを包含するバウンディングボックスと切断平面との交差多角形(以下切断多角形と呼ぶ)を求め、その面に交差するメタボールを抽出する。

(2) 切断面上のメタボールのスクリーン上での y の範囲(付録参照)を算出し、 y の最大値および最小値(図の y_{min} , y_{max})の間を走査する。

(3) 各走査線においては、走査線と切断多角形との交線(図の P_1P_2)を求め、それに交差するメタボールを抽出し、各メタボールと走査線との交差区間を求めておく。

(4) 走査線上の各画素について、濃度を計算して内外判定を行う。

なお、(4)において、メタボールの走査線上の存在区間がわかっているので、計算中の画素を含むメタボールのみを対象に濃度計算を行うことが可能であるから高速処理が実現できる。

3.2 ボクセルデータの編集

ボクセルデータの場合も、ほぼ同様な手法で処理できる。メタボールの場合、メタボール単位で追加・削除を行った。ボクセルの場合、ボクセル単位で追加・削除を行うと、ボクセルは微小立体なので、形状を変化させるのに手間がかかる。そこで、メタボールと同様に球を単位に追加などを行う。すなわち、指定した半径の球状の粘土を物体表面に追加する(球の半径のほか、オフセッ

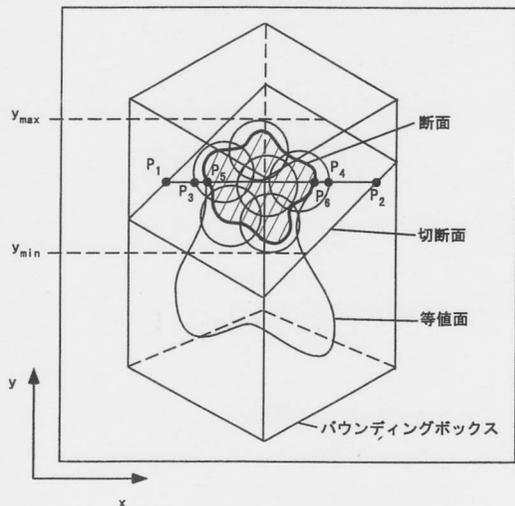


図3 メタボールデータの断面表示

Fig. 3 Rendering a cross-section of metaball data

トも与える)。これは、スクリーン上で物体上の点を指定し、その点に相当する物体境界面上の3次元座標を中心に球を生成し、その内部に存在するボクセルに内部フラグをセットする。一方、削る場合は、負のメタボールと類似した方法とする。すなわち、球の内部を外部フラグにセットするこの球の大きさはのみ(彫刻刀の刃)あるいは筆の大きさに相当する。

ボクセルデータの断面表示に関しては、メタボールの場合と類似した方法を用いる。すなわち、図3の断面多角形の領域内のみを走査し、断面と走査線との交線上(図中 P_1P_2)の各画素に相当するボクセル(画素を通過する視線と切断面との交点を含むボクセル)を求め、そのボクセル値が内部か否かを判定する。このように、簡易な方法によって断面を表示できる。

4. データ変換

図1に示したように、多角形、ボクセルデータ、メタボールの表現形式相互間のデータ変換を考える。

1) 多角形からボクセルへの変換

多角形で構成される物体のバウンディングボックスをボクセルに分割し、各ボクセルが多角形で構成される物体の内部か否かを判定すればよい。これは、例えば、 xy 平面に平行な平面で物体をスライスし、その断面多角形を走査変換することによって、物体内部のボクセルを塗りつぶすことができる。

2) メタボールおよびボクセルデータから多角形への変換

ボリュームデータもメタボールも基本的には空間中の

濃度分布を表現するものであるから、この変換は、マーチンキューブ法¹⁰⁾による多角形法を利用した。

3) メタボールからボクセルデータへの変換

すべてのメタボールに対するバウンディングボックスを格子状に分割して、格子点でのメタボールの密度によりボクセルの内外判定を行う。すなわち、密度がしきい値以上の値ならば内部ボクセル、逆に近い値ならば外部ボクセルとして記憶する。

4) ボクセルデータからメタボールへの変換

ボクセルデータからメタボールへの完全な変換というより、ボクセルデータに基づいてある許容範囲内においてメタボールを生成する方法である。ボクセルから直接メタボールへの変換、または多角形データからボクセル表現を経由してのメタボールへの変換について以下に論じる。

4.1 メタボール生成の従来法

3次元物体を測定データから陰関数(メタボール)を用いて再構成する方法を、最初村木¹¹⁾が開発した。この方法は1つのメタボールから始まり、測定データとの誤差が少なくなるまで、分裂を繰り返す方法である。これは誤差に関してエネルギー最小問題とし、疑似ニュートン法を利用して解いている。しかし、残念ながらかなり時間がかかる方法である(1日以上を要す)。また、結果も満足できるものではない。

Ranjaら¹²⁾はメタボールのように融合はしないが、球の集合として物体(特にCT画像の入力)を再構成する方法を開発した。彼らは、境界を構成する頂点を含むドロネ3角形を利用して、それに基づいた球を生成した。この方法は球なので、メタボールのようになめらかな曲面は表現できない。

最近、Bitterら¹³⁾によって、与えられた離散データからメタボールを生成する方法が開発された。この方法では、一旦ボクセルデータを生成し、それを基にデータの骨格に相当するメジアンアクシスと呼ばれる点の集合を抽出し、それらを中心としたメタボールのうち、データ点での濃度がある誤差内になる条件を満たすメタボールを選択する方法により、メタボールを生成している。

一方、2次元において、与えられた境界内を埋め尽くす2次元メタボール(メタサークル)を生成し、データ圧縮をはかる方法¹⁴⁾も提案されているが、2次元問題にも関わらずかなりの時間を要す。

4.2 提案法の基本的な考え方

インタラクティブなシステムにおいては、ユーザーに負担にならない時間で処理できること望ましい。本論文では、Bitterらの方法に類似しているが、一部会話的に行うことにより効率的に行う方法である。全自動で行う

場合、一様にある精度を保証する必要から時間がかかる傾向にある。部分ごとにメタボールの近似精度を指定できる方が、実システムには有効である。また、メタボールの生成過程を可視化することにより、ユーザーのストレスを軽減している。

基本的には、ボクセルデータで表現される物体表面をメタボールで構成される等値面に変換する方法である。なお、内部の濃度分布まで等価なメタボールを生成するものではないものとする。簡単のためボクセル値は2値として説明する。ボクセルデータには物体の内部か外部を示すフラグがあるものとし(物体内部が1、外部が0)、それぞれを内部ボクセル、外部ボクセルと定義し、隣接するボクセルが一つでも外部にあるものを境界ボクセルとする。なお、本論文では、隣接するボクセルとは、面で接するボクセルを対象として考える。まず、物体の背骨に相当するスケルトンを指定する。なお、このスケルトンは折線で与える。スケルトンは、物体形状の複雑さによっては複数個必要になる。スケルトン付近に中心があり、半径が最も大きくボクセル内部に存在するメタボールを求める。これらのメタボールをスケルトンに沿って求める。これにより物体内部の主要な部分はこれらのメタボールで占有される。次に、境界部の微妙な形状を表現するための比較的小さなメタボールを生成する。メタボールはその目的から2種類を考え、スケルトンボールと表面ボールと呼ぶことにする。

4.3 概略手順

提案法は以下の手順によりなる。

- (1) ボクセルデータを半透明表示させる。
- (2) スケルトンの線分の二端点をマウスで指定し、スケルトンボールを生成する。その際、表示に関しては生成されたメタボールは透明として扱われ、メタボールに包含されない部分のボクセルデータは半透明表示のままである。
- (3) 表面ボールを生成する。その際、境界面でのサンプリング点における誤差が設定した値以下になるまで追加、修正を繰り返す。また、表示に関しては(2)と同様に生成されたメタボールは透明として扱われる。

(1)において、半透明表示するのは次の理由による。ステップ(2)、(3)においてはボクセルデータ内でメタボールが生成されると、その部分にフラグを立てる。メタボールによる占有領域は透明体とみなして表示することにより、メタボールに変換されていない空間が半透明表示により容易に可視化できる。ここでは、メタボールを1つ追加するごとに再描画され、どの部分にメタボールが生成されたか、リアルタイムに観察できる方法とした。

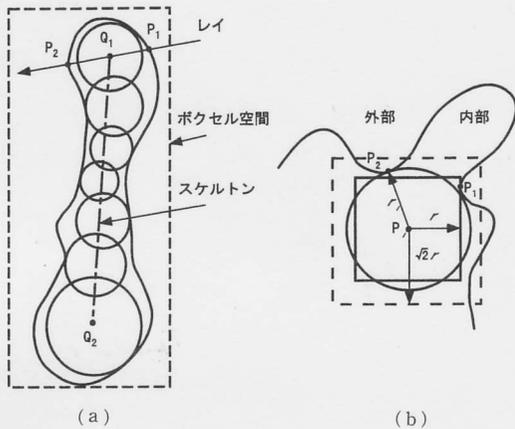


図 4 ボリュームデータ内のメタボールの生成法
 Fig. 4 A creation method of metaballs based on volume data

(2)において、スケルトンを指定するのは次の理由による。本システムでは、基本的には人体などを対象としている。人体のアニメーションを多く作成しているThalmannら^{15),16)}は、まずスケルトンを指定し、それをもとにメタボールを用いて肉づけをして、基本的な人体モデルを生成後、多角形化して表示するシステムを開発している。本論文においても、骨組みとなるメタボールの回りに小メタボールを生成する方式をとる。図4(a)に示すように、スケルトンを表現するための線分の2端点をスクリーン上においてマウスで指定する。半透明ボクセルの場合、レイと物体の交差区間は容易に判定できる。この交差区間の中点をスケルトンの端点としてみなせる。図の例では、レイと物体との交点 P_1P_2 から中点 Q_1 を算出できる。スケルトン上の1端点から他の端点(例えば図の Q_1Q_2)に向かって進みながらメタボールを生成する。メタボールの場合、半径、位置 (x, y, z) 、中心での濃度と5変数必要であるが、ここでは、濃度はユーザーが指定した一定値とし、他の変数の最適解を算出する。なお、なるべく少ない数のメタボールとするために、なるべく大きな半径のメタボールを生成するように最適化を行う。

4.4 スケルトンボールの生成法

スケルトン上のあるボクセル付近を中心とし、物体内を占有する最大半径の球を求めることを考える(図4(a)のスケルトンの周辺メタボール参照)。位置と半径の最適化をするため、スケルトン上にあるボクセルの中心をメタボールの中心の初期値とし、近傍にあるボクセル(例えば $5 \times 5 \times 5$ ボクセル)中でメタボールの中心 P_i (ボクセルの中心とする)を移動し、メタボールの半径が最大となるボクセルを探索する。なお、各中心点 P_i

における最大半径 R_i は次の方法によって算出できる。この半径は、すべての境界ボクセルと点 P_i との距離の最小値として求まる。しかし、全てのボクセルとの距離の計算を行うと計算量が多くなる。そのため、図4(b)に示すように、 P_i を中心とした立方体(キューブ)をまず求める。この立方体は、外部ボクセルを含まない最大のサイズの立方体である。これは、箱のサイズを1層づつ増加し、外部ボクセルを発見するまで(図ではボクセル P_1)、層を拡大する。その立方体のサイズ(ボクセル数)を $2r$ とすると、メタボールの最大半径は $2r$ から $2\sqrt{2}r$ の範囲にあることが言えるから、 $2r$ から $2\sqrt{2}r$ の範囲(図中の実線と点線で示される立方体に挟まれた空間)内の外部ボクセルを抽出し、それらと P_i の距離を計算すると、それらの距離の最小値(図では点 P_2)が求める半径である。

4.5 表面ボールの生成法

ボクセルデータの境界とメタボールの等値面が等価になるように、メタボールを生成するのが目的である。前述の方法で中心部がメタボールで充填されているから、ここでは微妙な面の変化を表現するための表面付近のメタボールを生成する。この場合、境界部に存在するすべてのボクセルを等値面と一致するかどうか判定するのは困難であるから、境界ボクセルをサンプリングして、境界ボクセルの中心において判定する。

あるサンプリング点 P において、その点に接し、かつ最大の半径(物体外部のボクセルを含まない範囲の最大半径)となるメタボールを生成する。このメタボールは次のように求める。最小半径をあらかじめユーザーが指定しておき、その半径のメタボールを初期値としてはじめ、その P を含み法線方向の半直線を考え、その半直線を物体内部に向かって進み、その半直線を中心として外部ボクセルを含まない最大半径のものを算出する。ここで、最大半径の算出は前述と同じ方法を用いる。この際、メタボールが追加されることによって、他のサンプリング点の濃度も影響される。したがって、既に生成している付近のメタボールによる濃度も考慮する。すなわち、サンプリング点における濃度を求めておき、補うべき濃度を求め、それに必要な半径を補正する。

4.6 誤差評価および補正

4.4および4.5節の2つの方法によって、メタボールの基本的配置が求まる。しかし、メタボールが互いに影響(融合)しあうので、メタボールによる等値面とボクセル境界に誤差が生じる場合がある。したがって、ボクセル境界部のサンプリング点において誤差評価し、(式(2)を利用)、誤差がある場合はメタボールを追加・補

正する。これは、ある与えられた濃度値よりサンプリング点の濃度が低い場合はメタボールを追加し、ある許容誤差以外の場合、基本的に最も近いメタボールの半径を補正し、1つのメタボールの補正でもまだ誤差がある場合はさらに複数のメタボールを順次補正する。これを、すべてのサンプリング点について行う。この操作を何回か行い、誤差が減少するまで繰り返す。

物体表面に相当する境界ボクセルは数万個にも及ぶので、これらの全ての点をサンプリング点として選択することはできないので、千個程度になるように数個おきを選択した。また、すべての空間で精度が保証される必要があるとは限らないので、ボクセルデータのある方向から見た投影面上においてマウスを用いて領域指定し、その範囲を重点的に、精度の評価および補正を可能とした。誤差の評価法としては、式(2)においてしきい値 T に対する誤差の割合を指定する方式とした(今回の例では5%)。

5. 自由形状変形法

人体モデルを作成する場合、基本的な形状を入力しそれを変形する方が容易である。また、着装シミュレーション

を行うには実測データに基づいた高精度の形状が望まれる。しかし、各個人についてその都度実測するにも限界がある。そこで、標準データを基に、相似変換(拡大・縮小)あるいは局所変形する方法が有効的である。本論文では、ボクセルデータを対象とし、自由形状変形を精度よく実現できる方法を提案する。

本論文では Sederberg らの開発した Free Form Deformation (FFD) 理論¹⁷⁾を用いて変形を行う。ただし、この理論は順変換にしか適用できない。すなわち、ある頂点を変形後の新しい頂点に移動することができるが、その逆変換はできない。ポリゴンデータの場合はポリゴンの構成頂点を移動させればいいが、ボクセルデータの場合、容易ではない。メタボールの場合には、ボールの中心と半径を変える必要があり、これらを精度良く変形させるには問題があり、ボクセルデータの方が自由形状変形に適しているので、ボクセルデータの自由形状変形について論じる。

FFD は基本的には次のようである。変換前の原形状をソースデータ、変形後をディスティネーションデータと呼ぶことにする。図5に示すように、ソースのボクセルデータをメッシュで構成される箱で囲み、その格子点

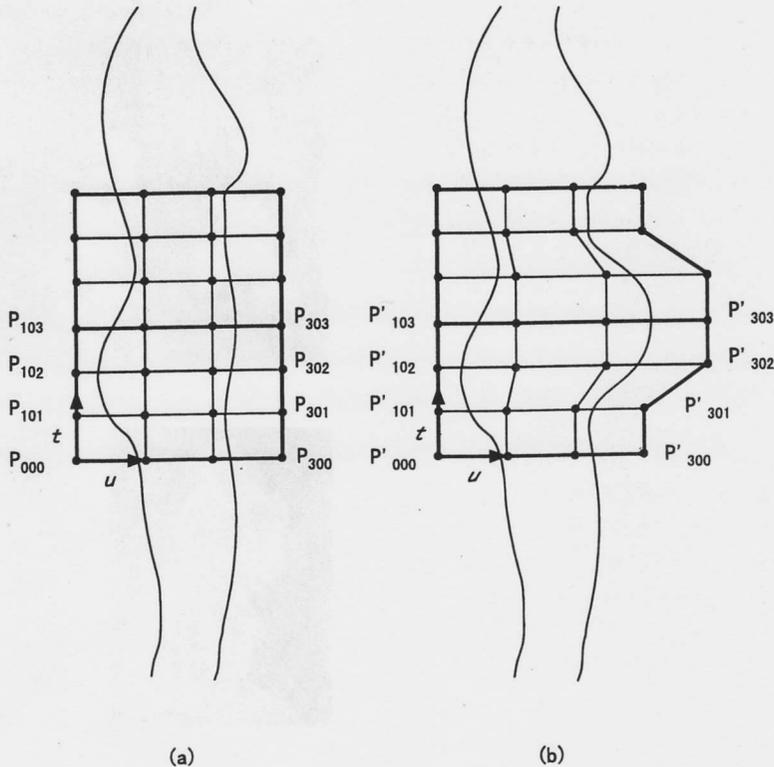


図5 ボリュームデータの自由形状変形
Fig. 5 Free form deformation of volume data

メタボールとボクセルデータの統合化造形システムの開発

を移動させることにより変形を行う。

ソースデータにおけるバウンディングボックス内のパラメータ座標系 (u, v, t) を考える。 $n \times n \times n$ 次の格子の場合、格子点の座標は次式で表される。

$$P_{ijk} = O + (i/n)U + (j/n)V + (k/n)T \quad (3)$$

($i=0, 1, 2, \dots, n, j=0, 1, 2, \dots, n, k=0, 1, 2, \dots, n$)

ここで、 O はバウンディングボックスの原点、 U, V, T は基準ベクトルで、ここではそれぞれ x, y, z 軸に平行にとる。ソースデータのバウンディングボックスは直交格子なのでこの空間内のある点 P のパラメータ座標は簡単に求まる。すなわち、点 P の座標は次式で表される。

$$P = O + uU + vV + tT \quad (0 \leq u, v, t \leq 1) \quad (4)$$

変形後の格子点を P'_{ijk} とすると、点 P は次の位置 P' に変位する。

$$P'(u, v, t) = \sum_{i=0}^n \sum_{j=0}^n \sum_{k=0}^n P'_{ijk} B_i^n(u) B_j^n(v) B_k^n(t) \quad (5)$$

ここで、 B は Bernstein の基底関数である。

変形の際、変形の境界部においてスムーズに変化することが望まれる。したがって、 C^1 接続が保証される変形を施すことを考える。そのため、ここでは3次の Bézier 関数を使用する。Bézier 曲線をスムーズに接続するには、端点が一致し、端点に隣接するそれぞれの曲線の制御点が直線上に並ぶ必要がある。したがって、図 5(a) のように2組の隣接するバウンディングボックス (4×4 の制御点からなる) を準備する。変形後もスムーズとなる条件を満足するためには、例えば変形後も下2列の制御点 (図中の $P'_{000} P'_{101}, \dots, P'_{300} P'_{301}$) は不変にするなどの条件を満たして、制御点を移動すればよい。

この変換はテクスチャマッピングと類似している。すなわち、曲面に例えると、変形後の曲面上のある点のパラメータ (u, v) を算出し、その u, v に相当する原画像の色をマッピングする。この場合、任意の点のパラメータ (u, v) を算出するのが主たる問題である。Kajiya によると、このパラメータを解析的に求めるには18次式 (双3次曲面の場合) を解く必要がある¹⁸⁾。

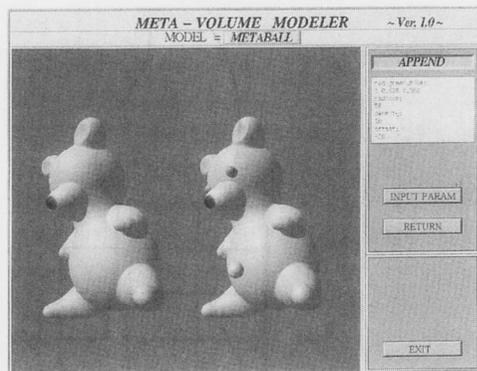
ボクセルデータの場合も、これと同じようにある点 (すなわちボクセル) のパラメータ (u, v, t) を算出し、その点に相当するソースデータのボクセル値をコピーすればよい。

ここで、問題を簡単にするため、変形の際には少なくとも1変数は固定して行う方法とした。すなわち、2段階に分けて変形する。例えば、 x, y 成分についてまず変形し、次に z 成分についての変形を行う。この場合、パラメトリック曲面のレイトレーシングと同じ問題と解

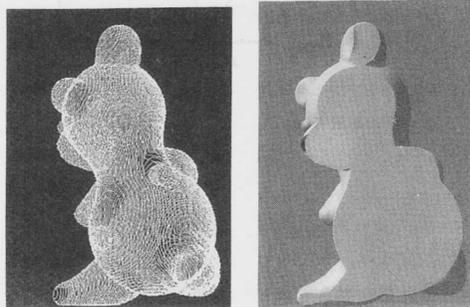
積できる。3次元空間のある点の (x, y, z) 座標を与えた際のパラメータ (u, v, t) を (z および t は固定) を算出することになる。すなわち、変形後のバウンディングボックスを xy 平面に平行な面によって切断すると、その面上での Bézier 曲面の制御点が得られる。この制御点ネットを利用し、あるボクセルの xy 座標から、そのネットで定義されるパラメトリック座標 (u, v) を算出し、その (u, v) に相当するソースデータにおけるボクセル値をコピーすればよい。この際に必要な逆変換 (xy から uv 座標への変換) は、すでにレイトレーシングのために開発された、Bézier Clipping 法⁴⁾ が適用できる。Bézier Clipping 法の適用によって、18次式の解法は1次式の収束計算に置き換えることができ、効率よく解を算出できる。この処理は、ディスティネーションデータのバウンディングボックス内のすべてのボクセルについて順次、ソースデータにおけるボクセル番号を算出することにより、実現できる。

6. 適用例

図 6 から図 9 に本システムによる表示例を示す。図 6(a) は、熊の縫いぐるみを作成した例であり、左側の



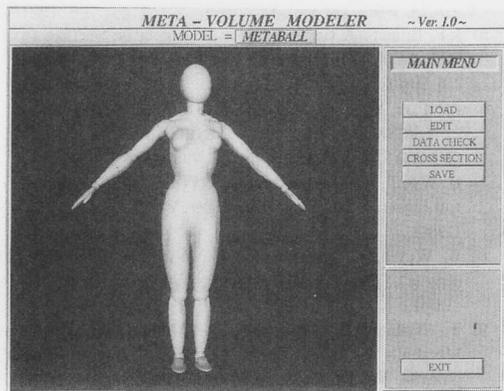
(a)



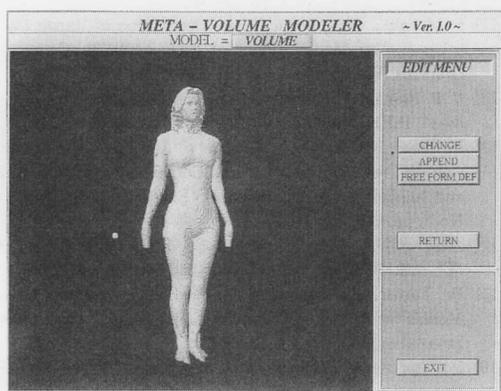
(b)

(c)

図 6 メタボールの表示例
Fig. 6 Example of metaballs



(a) メタボールによる人体



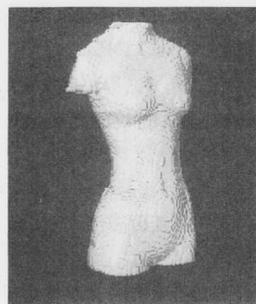
(b) ボクセルデータによる人体

図7 人体の表示例

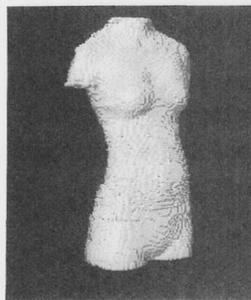
Fig.7 Examples of human body

物体にメタボールを会話的に追加(目やへそなど)したものが右側に示されている。これは、メタボールの追加位置をスクリーン上でマウスにより簡単に指定した例である。図(b)は等高線表示したもので、(c)は断面を表示したものである。図6(a)の計算時間が12.6秒である(メタボール数48個)。例えば、へそ部分を追加するための計算時間が0.16秒であった。追加メタボールのバウンディングボックスを用いる方法(3.1節の2参照)に比べ19%減少した。この例のように、ユーザがインタラクティブに操作するのにストレスを感じない範囲であると判断する。

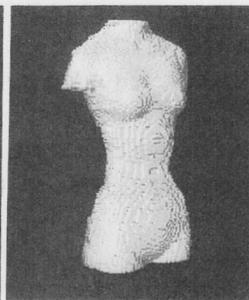
図7(a)はメタボールおよびメタ楕円体を組合せて人体を作成したものである。これは人体計測の資料からのスケッチを座標読み取り装置で計測したものに基づいており、着装シミュレーション^{2),3)}で用いた基本人体モデルを本システムにより局所的な変更を加えたものである。図(b)は実測データ(人間生活工学研究センターよ



(a)



(b)



(c)

図8 自由形状変形の例

Fig.8 Examples of free-form deformation



(a)



(b)



(c)

図9 メタボールの生成

Fig.9 Examples for generation of metaballs

り入手)に基づく多角形データを、ボクセルデータに変換したものである。測定の際の雑音により部分的に不自然な部分があったので、これを本システムの編集機能に

よって修正した。また、顔の部分は市販されている多角形表現の顔を結合したものである。

図8はFFDの例であり、胴体部を変形したものである。図(a)は原形、図(b)は腹部を太らせたもので、(c)は腹部を痩せさせた例である。

図9はボクセルデータを基にメタボールを生成したものである。(a)、(b)は頭蓋骨で(aはボクセルデータ)、(c)は胴体部の例(図8が元のボクセルデータ)である。これらの例は、元データが多角形表現されており、メタボールへの変換によりデータ量はテキスト行数に関して1桁(10^4 から 10^3 のオーダー)減少する結果が得られた。今回得られたメタボールはスムーズでない部分もあり、まだ十分とは言えないが、良い結果が得られた。適用する対象に応じて変換精度を変える必要があるが、その方法としてボクセルの分割数やメタボールの数を増減し実用上問題のない程度に変換する。

7. む す び

本論文では、種々の表現形式による3次元物体を会話的に操作できる、造形デザインシステムを提案した。以下の成果を得ることができた。

メタボール、ボクセル、およびポリゴン表現による3次元データが表示でき、相互にデータ変換が可能になった。3次元形状の変形(追加, 修正, 自由形状変形)をスクリーン上でインタラクティブに操作でき、修正部の高速表示が実現できた。スクリーン上で指定した点の3次元座標の検出(逆透視変換)が容易で、断面表示や等高線表示により、データの把握が容易となった。

適用例の一つとして、衣服の着装シミュレーションに用いる人体モデルを作成することで、提案法の有効性を示した。

謝 辞 本研究の中で用いられた Bézier Clipping 法のアイディアの一部は、米国ブリガムヤング大学のトーマス・シーダバーク教授との議論によるものであり、彼に感謝する。また、プログラムの一部あるいはデータ作成に協力してくれた、山阿、神原および岩崎君に感謝する。

参 考 文 献

- 1) H. Okabe, H. Imaoka, T. Tomiha, H. Niwaya: "Three Dimensional Apparel CAD System", Proc. of SIGGRAPH'92, Computer Graphics, Vol. 26, No. 2, pp. 105-110 (July-1992).
- 2) 松田亮治, 今岡春樹: "衣服設計のための濃度球を用いた人体形状モデリング", 繊維学会誌, Vol. 50, No. 5, pp. 221-228 (May-1994).
- 3) 松田亮治, 今岡春樹: "CGによる姿勢変化可能な人体モデルへの衣服の着付法", 繊維学会誌, Vol. 51, No. 5, pp. 225-233 (May-1995).
- 4) T. Nishita, T. W. Sederberg, M. Makimoto: "Ray Tracing Rational Trimmed Surface Patches", Computer Graphics,

- Vol. 24, No. 4, pp. 337-345 (1990).
- 5) J. F. Blinn: "A Generalization of Algebraic Surface Drawing", ACM Transaction on Graphics, Vol. 2, No. 3, pp. 235-256 (1980).
- 6) 西村仁志, 平井 誠, 河合利幸, 河田 享, 白川 功, 大村皓一: "分布関数による物体モデリングと画像生成の一手法", 電子通信学会論文誌, Vol. J68-D, pp. 718-725 (1985).
- 7) B. Wyvill, C. McPheeters, G. Wyvill: "Data structure for soft objects", The Visual Computer, Vol. 2, No. 3, pp. 227-234 (1986).
- 8) B. Wyvill, C. McPheeters, G. Wyvill: "Animating soft objects", The Visual Computer, Vol. 2, No. 3, pp. 235-242 (1986).
- 9) T. Nishita, E. Nakamae: "A Method for Displaying Metaballs by using Bezier Clipping", Proc. of Eurographics'94, Vol. 13, No. 3, pp. 271-280 (1994).
- 10) W. E. Lorensen, H. E. Cline: "Marching Cubes: a High Resolution 3D Surface Construction Algorithm", Computer Graphics, Vol. 21, No. 4, pp. 163-169 (1987).
- 11) S. Muraki: "Volumetric Shape Description of Range Data using Blobby Model", Proc. of SIGGRAPH'91, Computer Graphics, Vol. 25, No. 4, pp. 227-235 (July-1991).
- 12) V. R. Ranjan, A. Fournier: "Volume Models for Volumetric data", IEEE Computer, Vol. 27, No. 7, pp. 28-36 (1994).
- 13) E. Bitter, N. Tsingos, M. Gascuel: "Automatic Reconstruction of Unstructured 3D Data: Combining a Medial Axis and Implicit Surfaces", Proc. of Eurographics'95, Vol. 14, No. 3, pp. 457-468 (1995).
- 14) 木本, 浅井, 安田: "楕円体によるシルエット画像の記述と表現", 画像符号化シンポジウム(PCSJ92), pp. 121-124 (1992).
- 15) B. Turner, D. Thalmann: "The Elastic Surface Layer Models for Animated Character Construction", Proc. of CG International'93, pp. 399-412 (1993).
- 16) D. Thalmann, J. Shen, E. Chauvaneau: "Fast Realistic Human Body Deformations for Animation and VR Applications", Proc. of CG International'96, pp. 166-173 (1996).
- 17) T. W. Sederberg, S. R. Parry: "Free-Form Deformation of Solid Geometric Models", Computer Graphics, Vol. 20, No. 4, pp. 151-160 (1986).
- 18) J. Kajiya: "Ray Tracing Parametric Patches", Computer Graphics, Vol. 16, No. 3, pp. 245-254 (1982).

付 録

メタボールの走査範囲

視点座標系を考え、メタボールの中心位置を (x_i, y_i, z_i) 、有効半径を R_i 、また、スクリーンと視点の距離を R とすると、スクリーン上での走査線の範囲は次式で計算できる(図2参照)。

$$y = \frac{(y_i z_i \pm R_i \sqrt{y_i^2 + (z_i^2 - R_i^2)}) R}{z_i^2 - R_i^2}$$

走査面と球との交点の投影面での x 座標は次式で計算できる。

$$x = \frac{(x_i z_i \pm r_i \sqrt{x_i^2 + (z_i^2 - R_i^2)}) R}{z_i^2 - R_i^2}$$

ただし、 $r_i = \sqrt{R_i^2 - d^2}$ 、 $z_i = z_i + dz_s$ 、 $d = y_s y_i + z_s z_i$ ここで、 r_i は走査面とボールの交差円の半径、 d は走査面からメタボール中心までの距離であり、また、 $(0, y_s, z_s)$ は走査面の単位法線である。

松田 亮治



昭55, 福井大・工・繊維工卒, 昭57, 同大学院修士課程修了. 同年, 広島県立福山繊維工業試験場(現広島県立東部工業技術センター)に入所, 平4, 奈良女子大学に一年間内地留学. CGを用いた衣服の着装シミュレーションや着心地などの研究開発に従事. 現在, 福山大学大学院工学研究科電子情報工学専攻博士課程在籍中. 情報処理学会, 繊維学会, 日本繊維機学会, 各会員.

西田 友是 (正会員)



昭46, 広島大・工・電気工卒, 昭48, 同大学院修士課程修了. 同年, マツダ入社. 昭54, 福山大学工学部電子電気工学科講師. 昭59, 同大学助教授. 昭63, 米国 Brigham Young 大学客員研究員. 平2, 同大学教授. CGに関して, 3次元物体のリアルな表現法, 照明シミュレーション, 景観予測などの研究に従事. 昭62, 情報処理学会研究賞受賞. 平8, 画像電子学会論文賞受賞. 工学博士. 情報処理学会, 電気学会, 電子情報通信学会, IEEE, ACM, 各会員.