

自由形状モデリングのための 陰関数曲面を用いた仮想粘土細工システム

松宮 雅俊[†] 竹村 治雄[†] 横矢 直和[†]

3次元コンピュータグラフィクスにおいて、従来のモデラによる自由形状のモデリングは難しく、より簡単な操作でモデリングを行えるモデラが望まれる。本論文では、粘土細工を行うような感覚で自由形状のモデリングが行える仮想粘土細工システムについて述べる。本システムでは、仮想現実感技術と陰関数曲面を用いることにより、手を用いて直接的に仮想粘土へ凹凸を加えるという形状の変形を実現している。形状変形に際しては、陰関数曲面の定義を利用して、形状の衝突判定や変形処理に複雑な処理を必要とせず、その定義から自然に変形を表現している。一方で、仮想空間では、自由形状の変形やユーザの視点移動による画像を実時間で提示する必要がある。そこで、陰関数曲面を効率良くポリゴン表現に変換する手法を提案し、高速な描画を実現する。最後に、実装したシステムによるモデリング例を示し、粘土細工のような手を用いた対話的かつ直感的な操作によって、自由形状を短時間でモデリングできることを示す。

A Virtual Clay Modeling System for 3D Free-form Design Using Implicit Surfaces

MASATOSHI MATSUMIYA,[†] HARUO TAKEMURA[†]
and NAOKAZU YOKOYA[†]

We present a new interactive free-form modeling system based on the metaphor of clay work. This paper discusses design issues and an immersive modeling system which enables a user to design intuitively and interactively 3D solid objects with curved surfaces by using one's hand. Shape deformation can be expressed by simple formulas without complex calculations due to the nature of skeletal implicit surfaces employed to represent smooth free-form surfaces. A polygonization algorithm that generates polygonal representation from implicit surfaces is developed to reduce the time required for rendering curved surfaces, since conventional graphics hardware is optimized for displaying polygons. The prototype system has shown that a user can design 3D solid objects composed of curved surfaces in a short time by deforming objects intuitively using one's hand in real time.

1. はじめに

3次元コンピュータグラフィクス(CG)を効果的に使用した画像や映像は、プレゼンテーション、映画、コマーシャル、ゲームなどに積極的に活用され、3次元CGの必要性は高まる一方である。それにともない、自由形状をモデリングするための技術も、ますます重要になっている。

従来の3次元CGソフトウェアにおけるモデラでは、複雑な幾何操作や位相操作によって自由形状のモデリングを行う。そのため、ユーザには幾何や位相に関する数学的知識が必要なうえ、多くの複雑な操作を

覚えなければならない。さらに、3面図や投影図などの2次元表示を用いて3次元形状を把握する能力も要求される。すなわち、従来のモデラでは、自由形状のモデリングに要するユーザへの負担が非常に大きい。これは、これから3次元CGを始めようとする初心者には敷居の高いものである。また、ユーザは形状をモデリングする際、どのような幾何、位相操作を組み合わせるべきなのかを考えなければならず、形状そのものに集中することができない。

このような自由形状のモデリングにおける問題点を解決するために、粘土細工や彫刻のようなだれもが経験のある造形手法を取り入れることで、直感的な形状モデリングを目指す研究が多数行われている。

まず、彫刻を模倣したモデリングに関する研究として、形状をボクセルデータで表現し、3次元マウスを

[†] 奈良先端科学技術大学院大学
Nara Institute of Science and Technology

用いて操作される切削ツールの動きに従ってボクセル値を更新することで、仮想空間で対話的な形状モデリングを実現しているもの¹⁾がある。しかし、生成される形状は現実の彫刻のような細密なものではなく、形状操作においても立方体形状をした3次元マウスを操作するという現実にはない独特なものである。一方、楕円体で表現される彫刻刀と、平面または2次曲面で表現される彫刻素材との論理演算によって、現実の彫刻に似た形状を生成する研究²⁾がある。しかし、ユーザの視点移動に対する画像の描画に時間がかかる描画手法であるため、現実空間での造形作業のように形状を自由に眺めながらモデリングを行うことができない。

また、粘土細工を模倣したモデリングに関する研究としては、ボクセル空間に配置された3次元セル構造オートマトンを応用したもの³⁾や、粒子ベースモデルによって粘土の形状変形を実現しているもの⁴⁾がある。しかし、これらは対話的な形状変形や可視化を考慮していない。ほかに、双三次ベジエ曲面で表現された形状を直観的に変形させる研究⁵⁾があるが、変形手段や変形モデルの提案であり、自由形状のモデリングを行うには至っていない。メタボールを対話的に操作することによってモデリングを行う研究⁶⁾では、メタボールを移動させることによる形状の変化が粘土に似ている。しかし、操作する対象はあくまでメタボールであり、メタボールを意識しながらのモデリングは、粘土細工の造形作業とは異なる。

その他の造形手法を模倣したモデリングに関する研究として、仮想ろくろを用いた回転体形状モデラ⁷⁾がある。これは回転体の輪郭線を体積を一定に保ちつつ変形することで、ろくろを用いて作成したような形状を生成している。しかし、モデリング可能な形状は回転体に限られ、任意の自由形状のモデリングができない。

以上のように、様々な形で自由形状のモデリングの問題点を解決する試みが行われているが、我々は、より現実空間での作業に近づけたモデリング環境が必要であると考えた。たとえば、現実空間での造形手法では、素材を手で持って移動、回転をさせることや、頭を動かすことによって、好きな方向から眺めながら、手や道具を使って直接的に素材に変形を加えて造形していく。これをコンピュータ上の仮想空間で、現実空間と同様に行うことができれば、特別な知識や難解な操作を新たに覚えなくても、なじみのある方法でより直感的に形状をモデリングすることができる。

そこで、我々は、なじみのある造形手法として粘土細工を取り上げ、コンピュータ上で現実空間と同様の

粘土細工が行える環境を目指し、研究を進めている。本論文では、仮想現実感技術と陰関数曲面を用いて、仮想粘土を手で直接「凹ませる」とこと、「つまみ出す」ことによって、粘土細工を行うような感覚で自由形状のモデリングが可能な仮想粘土細工システムについて述べる。

このようなモデリングシステムを開発するにあたって、まず、ユーザ・インタフェースを考える必要がある。我々は、仮想現実感技術^{8),9)}を用いて、計算機の生成する仮想空間に没入して3次元形状のモデリングが行えるシステムとした。HMD(Head Mounted Display)を装着し仮想空間に没入したユーザが、手形状入力装置を装着した手で直接的に仮想粘土に変形を加えることによってモデリングを行う。また、3次元位置センサを用いて、頭部と手の位置および姿勢を計測する。

次に、仮想の粘土と手との対話を表現しなければならない。現実空間での粘土細工では、粘土に指を押し込んでいくと、粘土が指の形状に沿って凹んでいく。また、指でつまみ出すと、つまんでいる指と指の間に粘土がつまみ出される。さらに、このような変形を加えても、粘土形状は滑らかである。このような仮想粘土を表現するために、陰関数曲面(Implicit Surfaces)の一種である、スケルトルサーフェス(Skeletal Implicit Surfaces)^{10),11)}を利用する。スケルトルサーフェスは、距離場関数に従って、その骨格となるスケルトンを包み込むように生成される曲面である。このスケルトルサーフェスで仮想の粘土と手を表現し、適切な距離場関数とスケルトンの生成手法を開発することで、先述のような仮想粘土を表現する。

最後に、仮想粘土の変形やユーザの視点移動による画像を実時間で提示するために、スケルトルサーフェスによる形状を高速に描画する手法についても検討する必要がある。本論文では、陰関数曲面を効率良く実時間でポリゴンに変換する手法を応用し、高速な描画を実現する。

以下では、まず、2章で仮想粘土細工システムのユーザ・インタフェースについて述べる。次に、3章で仮想粘土の表現手法や描画手法の詳細を述べた後、4章でモデリング例を示し、最後にまとめとする。

2. 仮想粘土細工システムのユーザ・インタフェース

本章では、仮想粘土細工システムのユーザ・インタフェースについて述べる。まず、2.1節でシステムの概要とともにハードウェア構成について述べ、次に、

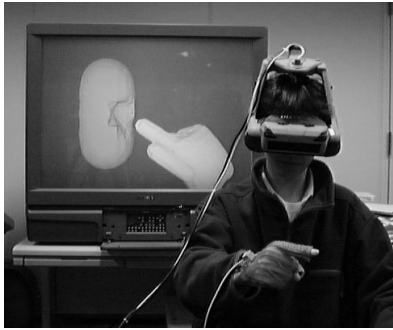


図 1 システムの様子

Fig.1 An appearance of the system.

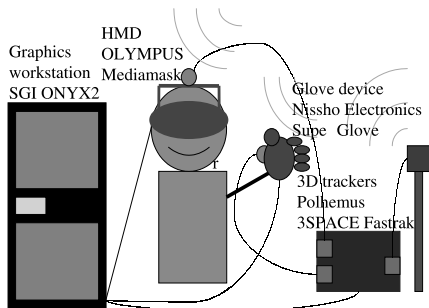


図 2 システム構成

Fig.2 The system components.

2.2 節でモデリング操作について述べる。

2.1 仮想粘土細工システムの概要

本システムでは、仮想空間にあらかじめ用意された球や円筒などのプリミティブ形状に対し、HMD(Head Mounted Display)を装着し仮想空間に没入したユーザが、手形状入力装置を装着した手で変形を加えることによってモデリングを行う。HMD と手形状入力装置には 3 次元位置センサが取り付けられており、頭部と手の位置および姿勢を計測する。システムを使用している様子を図 1 に、システム構成を図 2 に示す。なお、計算機として SGI 社 ONYX2 (MIPS R10000, 195 MHz, 16 CPU)を用い、HMD としてオリンパス社 Mediamask を、3 次元位置センサとして Polhemus 社 3SPACE Fastrak を、手形状入力装置として日商エレクトロニクス社 Super Glove をそれぞれ接続して使用した。

2.2 モデリング操作

本システムでは、形状に対して次の 2 つのモデリング操作を行うことができる。

(1) 形状を凹ませる

まず、手を図 3 のように人差し指だけを伸ばした形にする。そして、形状に対して人差し指を接触させ押し込んでいくことで、粘土のように、人差し指部分の

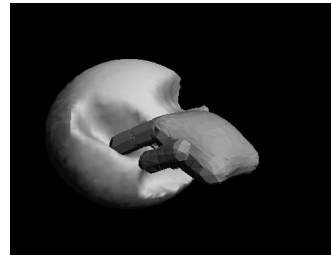


図 3 形状を凹ませた様子

Fig.3 Dent deformation.

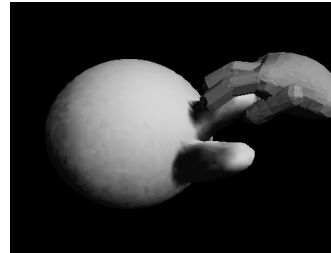


図 4 形状をつまみ出した様子

Fig.4 Pinching deformation.

形状を凹ませていく。このとき、接触部分の形状は指形状にあわせて変形する。また、陰関数曲面による形状はつねに連続性を保つため、凹みを加えた部分の形状も滑らかである。

(2) 形状をつまみ出す

まず、親指と人差し指、中指を伸ばし、手を図 4 のように形状をつまむ形にする。そして、手を移動させることで、形状をつまみ出す。実際には、手の動きに沿って 3 本の指の間に円柱状の形状が連続的に生成され、他の形状と接触する場合は滑らかにつながるが、これをユーザが意識することはない。そのため、手を形状内部から外部へ動かすと次々と形状が生成され、ちょうど手で粘土をつまみ出すように変形させることができる。

3. 仮想粘土細工システムの実装

本章では、システムの実装について述べる。まず、3.1 節で処理の概要を述べた後、3.2 節で本システムにおける仮想粘土の表現手法について述べる。そして、仮想粘土の変形手法について 3.3 節で述べ、最後に 3.4 節で画像提示手法について述べる。

3.1 概 要

システムのデータ処理の流れを図 5 に示す。まず、3 次元位置センサと手形状入力装置からの計測データを基に、変形処理部で、本システムで用いるスケルトラルサーフェスによる形状データであるスケルトンデー

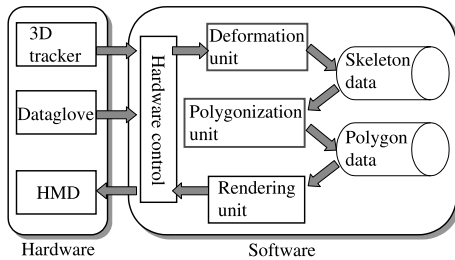


図 5 データ処理の流れ

Fig. 5 Data flow diagram of the system.

タを生成する．なお，スケルトンデータについては 3.3 節で詳しく述べる．次に，このスケルトンデータをポリゴン変換部で，頂点と稜線で構成されるポリゴンデータに変換する．そして，ポリゴンデータをグラフィクスライブラリの 1 つである OpenGL を用いてレンダリングし，HMD へ出力するという流れで処理が行われる．3.3 節，3.4 節では，この変形処理とポリゴン変換について詳述する．

3.2 仮想粘土の表現手法

現実空間での粘土細工では，手で粘土を凹ませる際に，粘土は手の形状に沿って凹む．また，つまみ出す際には，つまんでいる指と指の間に粘土がつまみ出される．さらに，このような変形を加えても，粘土形状は滑らかである．本研究では自由形状の表現手法として，陰関数曲面の中でもスケルトン (Skeletal Element) と呼ばれる形状の骨格要素を基にして形状を生成するスケルトルサーフェス (Skeletal Implicit Surfaces)^{(10),(11)} を用いる．スケルトルサーフェスは，一般に知られている，Blobby Molecule⁽¹²⁾ やメタボール⁽¹³⁾，Soft Objects⁽¹⁴⁾ と類似のものであり，これらの違いは，その定義に用いられている関数の違いである．スケルトルサーフェスでは，この関数の定義が広く，概形が同じである様々な関数が用いられている．そこで，このスケルトルサーフェスで仮想の粘土と手を表現し，適切な距離場関数とスケルトンの生成手法を開発することで，先述のような仮想粘土を表現する．まず，本節ではスケルトルサーフェスについて述べる．

スケルトルサーフェスを用いた陰関数曲面では，自由形状は式 (1) の陰関数多項式で表現され，このとき自由形状はスケルトン S と重み w ，距離場関数 (Field Function) F によって決定される． $f(\mathbf{P}) < c$ となる点 \mathbf{P} は形状の外部の点， $f(\mathbf{P}) > c$ となる点 \mathbf{P} は形状の内部の点， $f(\mathbf{P}) = c$ となる点 \mathbf{P} は形状の表面上の点であることを表す．ここで， f を陰関数値と呼ぶ．また，スケルトルサーフェスによる形状はつねに C^1 連続な曲面となり，式の偏微分により容易に曲面

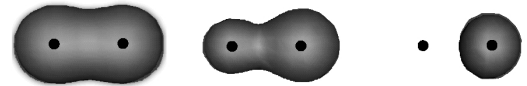
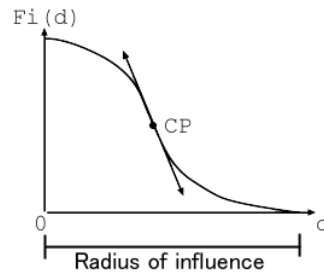
(a) $w_1=1.0$ $w_2=1.0$ (b) $w_1=0.5$ $w_2=1.0$ (c) $w_1=-1.0$ $w_2=1.0$ 図 6 重み w による形状の変化Fig. 6 Effects of changing the weight w in skeletal surfaces.

図 7 距離場関数の例

Fig. 7 An example of a field function.

上の任意の点での法線が求まる．

$$\{\mathbf{P} \in R^3 | f(\mathbf{P}) = c\}$$

$$\text{where } f(\mathbf{P}) = \sum_{i=1}^n w_i F_i(d(\mathbf{P}, S_i)) \quad (1)$$

スケルトン S は形状の骨格となるもので，点や線分，平面などを用いることができる．重み w はスカラー値であり，正負の値をとることができる．また， $d(\mathbf{P}, S_i)$ は点 \mathbf{P} とスケルトン S_i 上の最も近い点との距離を表す．2 つの点スケルトンからなる形状の，重み w による変化の例を図 6 に示す．図 6 (a)，(b) に示すように，重み w が正の値どうしの形状を組み合わせた場合，一方の形状が他方の形状に付加される．また，図 6 (c) に示すように，重み w が正の値と負の値の形状を組み合わせた場合，正の値の形状が負の値の形状に切削される．次に，距離場関数 F は典型的には図 7 に示されるような形状となる関数が使われる．制御点 CP における傾きと影響範囲となる幅を制御することで形状の変形具合を制御することができる．制御点 CP の傾きによる形状の変化を図 8 に示す．傾きの絶対値が小さい場合，図 8 (a) のようにつなぎ目が緩やかになり，傾きの絶対値が大きい場合，図 8 (b) のようにつなぎ目が急になる．

3.3 仮想粘土の変形手法

前節で述べたスケルトルサーフェスの特徴を利用して，仮想粘土の変形を行う．本節では 2.2 節で述べた 2 つのモデリング操作について，それぞれで行われる

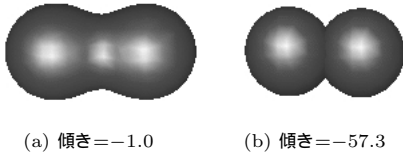


図 8 制御点 CP における距離場関数の傾きの影響

Fig. 8 Effects of changing the slope of field function at control point CP.

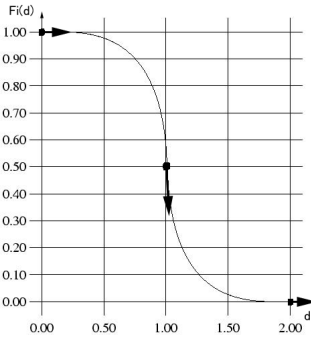


図 9 エルミート曲線による距離場関数

Fig. 9 Hermite curve as a field function.

変形処理について述べる。

まず、本研究では前節で述べたスケルトルサーフェスの表現式 (1) を次の式 (2) のように改良して用いる。

$$f(\mathbf{P}) = \sum_{i=1}^n w_i F_i(\sqrt{d(\mathbf{P}, S_i)}) \quad (2)$$

このように式変形するのは、経験的に、表現式 (1) のままでは指形状に沿ってプリミティブを凹ませることができず、指形状の周囲が広範囲にわたって凹んでしまうためである。形状をつまみ出す際にも、つまみ出した形状が広範囲に広がることを防ぐ。また、同様の理由から、距離場関数 F として図 9 に示すエルミート (Hermite) 曲線による関数を一貫して用いる。

次に、プリミティブ形状が 3 つの点スケルトンからなる形状を例に、2.2 節で述べた 2 つのモデリング操作による形状変形処理について説明する。プリミティブ形状 $f_1(\mathbf{P})$ は式 (2) から、点スケルトンを PS_i ($i = 1 \dots 3$)、重みを $w_1 = w_2 = w_3 = 1.0$ とし、次の式 (3) で表される。

$$f_1(\mathbf{P}) = F(\sqrt{d(\mathbf{P}, PS_1)}) + F(\sqrt{d(\mathbf{P}, PS_2)}) + F(\sqrt{d(\mathbf{P}, PS_3)}) \quad (3)$$

3.3.1 形状を凹ませる

人差し指形状 $f_2(\mathbf{P})$ は線スケルトンで表現し、線スケルトンを LS として、式 (4) で表される。

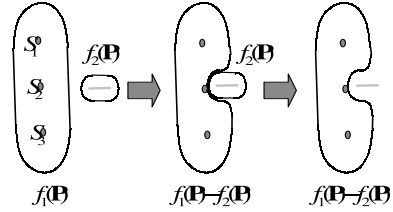


図 10 形状の凹みの表現

Fig. 10 A sequence of dent deformation.

$$f_2(\mathbf{P}) = w_1 F(\sqrt{d(\mathbf{P}, LS)}) \quad (4)$$

指形状 $f_2(\mathbf{P})$ によるプリミティブ形状 $f_1(\mathbf{P})$ の凹みは、図 10 に示すように、 $f_1(\mathbf{P}) - f_2(\mathbf{P})$ という減算によって表現することができ、変形後の形状 $f(\mathbf{P})$ は、指形状の重みを $w_1 = -1.0$ として式 (5) のようになる。

$$f(\mathbf{P}) = f_1(\mathbf{P}) - f_2(\mathbf{P}) \quad (5)$$

$$= F(\sqrt{d(\mathbf{P}, PS_1)}) + F(\sqrt{d(\mathbf{P}, PS_2)}) + F(\sqrt{d(\mathbf{P}, PS_3)}) - F(\sqrt{d(\mathbf{P}, LS)})$$

ここで、手が一定距離移動するごとに、指のスケルトンの形や位置といった幾何情報と、減算という属性をスケルトンデータとして連続的に取得していく。スケルトンのサンプリング間隔は、経験的に、指のスケルトンの長さの 100 分の 1 にした。この間隔であれば、連続的に生成しても、表現式 (2) と図 9 に示す距離場関数によって、指形状に沿った「形状を凹ませる」表現ができる。

3.3.2 形状をつまみ出す

つまみ出される (付加) 形状 $f_2(\mathbf{P})$ は、凹ませる場合と同様に線スケルトンで表現し、式 (4) で表される。2.2 節で述べたように、付加形状は 3 本の指の間に生成される。プリミティブ形状 $f_1(\mathbf{P})$ への形状 $f_2(\mathbf{P})$ の付加は、図 11 に示すように、 $f_1(\mathbf{P}) + f_2(\mathbf{P})$ という加算によって表現することができ、変形後の形状 $f(\mathbf{P})$ は、付加形状の重みを $w_1 = 1.0$ とし式 (6) のようになる。

$$f(\mathbf{P}) = f_1(\mathbf{P}) + f_2(\mathbf{P}) \quad (6)$$

$$= F(\sqrt{d(\mathbf{P}, PS_1)}) + F(\sqrt{d(\mathbf{P}, PS_2)}) + F(\sqrt{d(\mathbf{P}, PS_3)}) + F(\sqrt{d(\mathbf{P}, LS)})$$

ここで、手が一定距離移動するごとに、付加形状のスケルトンの形や位置といった幾何情報と、加算という属性をスケルトンデータとして連続的に取得していく。スケルトンのサンプリング間隔は、3.3.1 項と同様、指のスケルトンの長さの 100 分の 1 にした。この間隔で

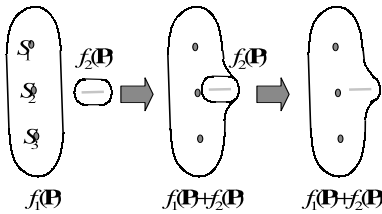


図 11 形状をつまみ出しの表現

Fig. 11 A sequence of pinching deformation.

あれば、連続的に生成しても、表現式 (2) と図 9 に示す距離場関数によって、細く滑らかにつながった「形状をつまみ出す」表現ができる。

次節では、以上のようにして生成されたスケルトンデータによって表される、スケルトルサーフェスを実時間で描画する手法について述べる。

3.4 画像提示手法

システムの実行時には、スケルトルサーフェスを用いた形状を実時間で描画してユーザに提示する必要がある。陰関数曲面では曲面上の任意の点での法線が容易に求まることから、その描画手法としてはレイトレーシング法を用いるのが一般的であるが^{15),16)}、レイトレーシング法は実時間処理には向かない。そこで、本研究においては陰関数曲面を用いた形状をポリゴンで近似し、ハードウェアによる高速なポリゴン描画手法を利用する。陰関数曲面を用いた形状をポリゴンに変換する手法としては、スケルトンから多方向に伸ばした直線と曲面との交点を頂点としてポリゴンを生成する Desbrun らの手法¹⁷⁾ や、関数や密度ボリュームデータの等値面をポリゴン変換する手法^{18),19)} を基にした、ボクセルと曲面との交点を頂点としてポリゴンを生成する Bloomenthal の手法²⁰⁾ などがある。しかし、これらの手法は実時間での対話的なモデリングを考慮したものではない。また、ボクセルを利用して、メタボールの実時間での対話的なモデリングを可能とした手法⁶⁾ もある。この手法では、濃度値(本論文では陰関数値)の計算時に参照するメタボールを限定することによって処理を軽減している。しかし、参照するメタボールを判別する処理はメタボールの総数に依存した処理量となっており、本システムのようにスケルトン(メタボールに相当)の数が多くなる場合には適用できない。そこで、Bloomenthal の手法を基に、実時間での対話的なモデリングを可能にするためのポリゴン変換手法を提案する。以下に具体的なアルゴリズムを述べる。

まず、前処理として以下の処理を行う。

- (1) 変換対象形状の周囲にポリゴン変換用ボクセル

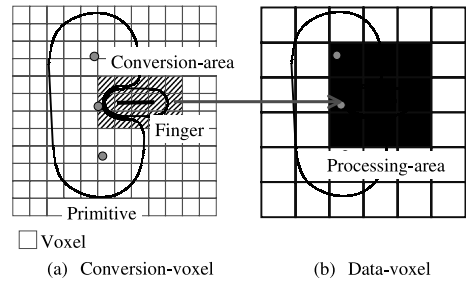


図 12 ボクセル空間

Fig. 12 Two voxel spaces: (a) conversion-voxel space and (b) data-voxel space.

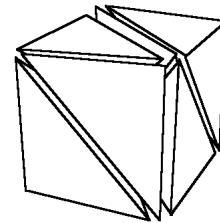


図 13 ボクセルの分割

Fig. 13 Decomposition of a conversion-voxel.

空間とスケルトンデータ用ボクセル空間を設定する。ポリゴン変換用ボクセル空間は曲面とボクセルの稜線との交差点を得るために使用する。また、スケルトンデータ用ボクセル空間はスケルトンデータの保持と陰関数値を求めるために使用する。2つのボクセル空間を簡単のため2次元で表したものを図 12 に示す。

- (2) 変換対象形状の表面と交差する全ポリゴン変換用ボクセルを用いてポリゴンに変換する。ボクセルを用いたポリゴン変換処理については後述する。

前処理の後、システムの実行時には以下の処理を実時間で行う。

- (1) 変形処理部において生成されるスケルトンデータを、その位置にあるスケルトンデータ用ボクセルに格納する。
- (2) 図 12 に示すような変換領域にあるポリゴン変換用ボクセルに対して、以下の処理を繰り返す。
 - (a) ボクセルの頂点を4点ずつ組み合わせて6個の4面体分割する。このとき図 13 に示すように、6個の4面体それぞれが互いに隣接するように分割する。
 - (b) 四面体の各頂点座標における陰関数値を求める。このとき、変換領域と重なるスケルトンデータ用ボクセルに格納されて

- いるスケルトンデータを用いて計算する．
- (c) 図 14 に示すように、求まった陰関数値からその頂点が形状の内部であるか外部であるかを判別し、それを属性として頂点の情報に付加する．
- (d) 両端点の属性が異なる稜線は必ず形状と交差するため、その稜線について形状との交点を求める．稜線と形状との交点を求めるということは、稜線の端点間に含まれた陰関数式の根を求めることであり、これを非線形方程式の根を求める手法である二分法²¹⁾によって求める．
- (e) 図 14 に示すように、求まった交点を基に、三角形ポリゴンを生成する．同時に、交点における法線を計算する．

以上のアルゴリズムに関して説明を加える．本システムでは変形する領域は付加および切削形状の近傍のみであるので、システムの初期化時に前処理として一度全ボクセルをポリゴン変換し、システムの実行時には変換領域のみをポリゴン変換している．これによって、処理を軽減することができる．

また、3.2 節で述べたように、スケルトルサーフェスでは、スケルトンごとに距離場関数の影響範囲があるため、変換領域が影響範囲内に入っているスケルトンデータのみを参照すればよい．そのためにポリゴン変換用のボクセル空間とは別に、スケルトンデータ用のボクセル空間を設定し、スケルトンデータ用ボクセルごとにスケルトンデータを記憶している．ポリゴン変換時には、変換領域と重なるスケルトンデータ用ボクセル内のスケルトンデータのみを参照することによって、処理を軽減するとともに、演算量の増加を抑えている．また、ボクセルを利用しているため、参照するスケルトンデータを判別する処理が必要ない．

さらに、スケルトンデータの無駄な増加を防ぐために、1つのスケルトンデータ用ボクセル内に記憶でき

るスケルトンデータの最大数を設ける．最大数以上のスケルトンデータは生成できないため、最大数に達したボクセルの領域には、それ以上の変形を加えることができなくなる．このため、本システムにおいては、1)ポリゴン変換用ボクセルの大きさ、2)スケルトンデータ用ボクセルの大きさ、3)1ボクセルあたりのスケルトンデータの最大記憶数、をパラメータとして、これをユーザの希望するモデリングの複雑さと、計算機の性能に応じて設定することとした．

なお、以上の処理はボクセルごとに独立した処理であるため、これらを複数の CPU で並列に処理することが可能である．

4. モデリング例

本システムによるモデリングの例として、ティーカップ、水槽の中の魚、たこ、鳥の頭を作成したものを図 15、図 16、図 17、図 18 にそれぞれ示す．図 15 のティーカップのモデリング手順について説明する．まず、図 19 のように球プリミティブが用意されている．図 20 で示すように「形状を凹ませる」を用いて

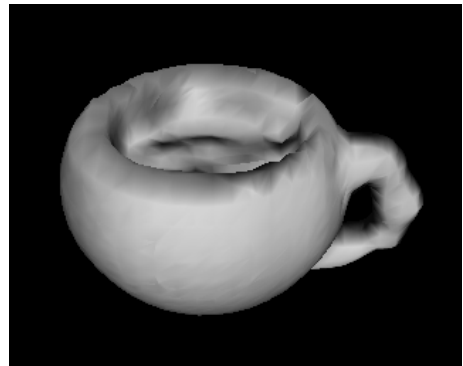


図 15 モデリング例：ティーカップ

Fig. 15 An example of modeling: A teacup.

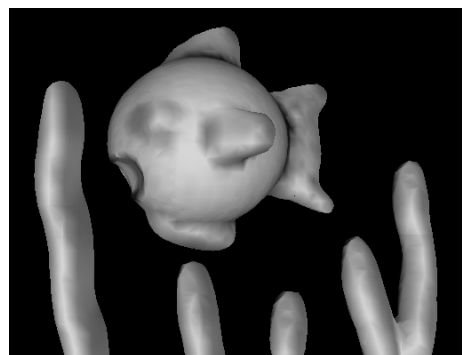


図 16 モデリング例：水槽の中の魚

Fig. 16 An example of modeling: A fish with water plants.

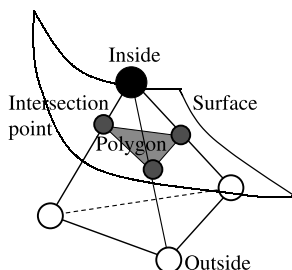


図 14 4 面体と形状との交点

Fig. 14 Intersection points between tetrahedra and a curved surface.

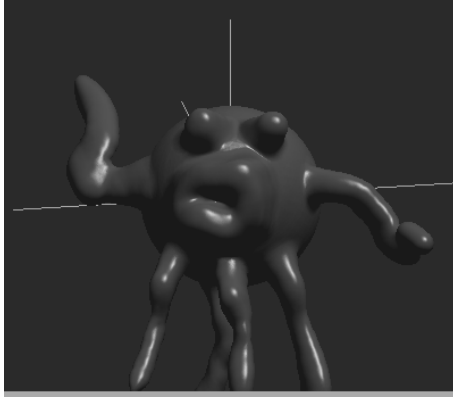


図 17 レイトレーシング法による画像：たこ

Fig. 17 An image rendered by using ray tracing: An octopus.

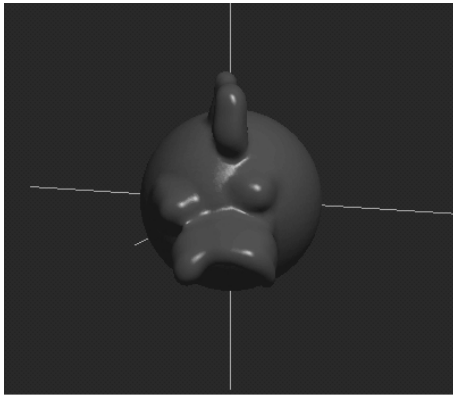


図 18 レイトレーシング法による画像：鳥の頭

Fig. 18 An image rendered by using ray tracing: A bird's head.

球プリミティブの上部を指で凹ませ、さらに飲み口の部分と中に入っている飲み物をモデリングした。その後、図 21 で示すように「形状をつまみ出す」で円弧状に手を動かし、取っ手の部分を付け加えた。さらに、細かな調整を加えて、図 15 のようなティーカップが完成した。図 16 の水槽の中の魚は、球プリミティブにつまみ出して目やひれを付け、凹ませて口を開けた。そして、つまみ出して海藻をモデリングした。他の形状も同様に作成し、いずれの形状も 5 分程度で作成できた。このように、本システムによって滑らかな曲面を持った自由形状を容易な操作で短時間にモデリングすることができる。

なお、モデリング作業時の画像更新レートは、CPU として MIPS R10000 (195 MHz) をポリゴン変換処理に 3 つ、その他の処理に 1 つ用いて、約 10 fps である。このとき、ポリゴン変換用ボクセルデータの大き

さは $120 \times 120 \times 120$ 、陰関数データ用ボクセルデータの大きさは $48 \times 48 \times 48$ 、1 ボクセルあたりの陰関数データの最大記憶量は 10 である。

ここで、図 15、図 16 はシステムの実行時の画面から切り出した画像である。本システムは陰関数曲面を直接レイトレーシング法によってレンダリングした画像や、DXF²²⁾ や STL²³⁾ といったポリゴンデータ形式で出力できる。図 17、図 18 はレイトレーシング法による画像である。また、図 22 はたこモデルを DXF で出力し、市販の 3 次元 CG ソフトウェアに利用した例である。図 23 は STL で出力し、光造形装置 (三井造船, COLAMM-300) によって実物体として生成した例である。

5. ま と め

本論文では、粘土細工を行うような感覚で直感的かつ容易に自由形状のモデリングが可能な仮想粘土細工システムについて述べた。本システムでは、現実空間での粘土細工のような操作と形状の表現を、仮想現実感技術と陰関数曲面による形状の表現を用いることで実現した。また、モデリング例を通して、実際に自由形状を短時間でモデリングすることができることを確認した。本システムでのモデリングは精密さや正確さに欠けるため、精密な工業製品などを正確にモデリングするような用途には向かないが、設計の初期モデルのモデリングや、4 章で示したような正確さが求められないデザイン作成などの用途に有用である。また、本システムは仮想空間を利用した新たな造形手法を提供する。

本システムの問題点として、一度凹みを加えた部分をつまみ出すことができないことがあげられる。これは、生成されたスケルトンデータの消去や更新といった処理を行っていないため、正と負のスケルトンが重なり、形状が削られてしまうためである。この問題点は、スケルトンのサンプリング点を格子状に限定し、消去や更新を行う方法で解決を試みている。また、現在の実装では、頭を移動させることでしか、作成中の形状を平行移動や回転させることができない。この操作も、現実空間同様に行える必要があるため、今後研究するとともに、システムに組み込みたいと考えている。

本システムでは、体積保存などのような粘土らしい変形過程に関しては考慮していない。また、操作における反力^{24)~26)}を表現していない。これらは粘土細工において重要な要素であるため、今後、粘土らしい変形過程の表現や反力の計算手法に関して研究する予

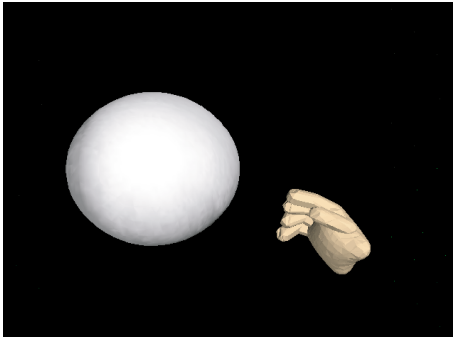


図 19 モデリング過程 1

Fig. 19 A sequence of modeling 1.

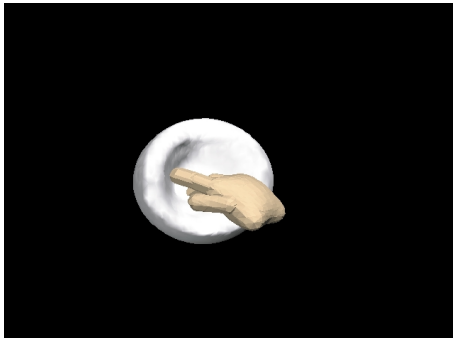


図 20 モデリング過程 2

Fig. 20 A sequence of modeling 2.

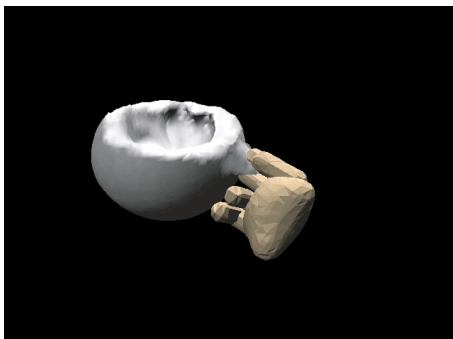


図 21 モデリング過程 3

Fig. 21 A sequence of modeling 3.

定である。

参考文献

- 1) Galyean, T.A. and Hughes, J.F.: Sculpting: An Interactive Volumetric Modeling Technique, *Proc. ACM SIGGRAPH '91*, pp.267-274 (1991).
- 2) 水野慎士, 岡田 稔, 鳥脇純一郎, 横井茂樹: 仮想彫刻—仮想空間における対話型形状生成の一手法, *情報処理学会論文誌*, Vol.38, No.12, pp.2509-2516 (1997).



図 22 市販ソフトウェアを利用して生成した画像

Fig. 22 An image created by using commercial software.

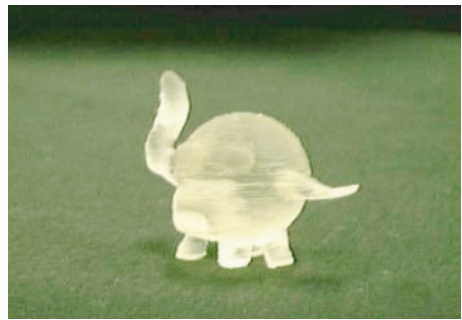


図 23 光造形装置を利用して生成した実物体

Fig. 23 An actual model created by using a stereolithography.

- 3) 荒田秀樹, 高井昌彰, 高井那美, 山本 強: 能動的ボクセル空間における仮想粘土モデリング—モデルの基本理念, *信学論 (D-II)*, Vol.J82-D-II, No.11, pp.2008-2016 (1999).
- 4) 小田泰行, 千葉則茂: 粒子ベースモデルによる粘土のビジュアルシミュレーション, *情報処理学会研究報告*, Vol.97, No.124, pp.25-30 (1997).
- 5) 前野 輝, 岡田 稔, 鳥脇純一郎: 形状関数を用いた双三次ベジエ曲面の直観的変形手法, *日本バーチャルリアリティ学会論文誌*, Vol.5, No.1, pp.811-817 (2000).
- 6) 梅村 隆, 岡田 稔: メタボールを用いた会話的モデラのためのパッチの一生成法, *画像電子学会誌*, Vol.26, No.4, pp.306-313 (1997).
- 7) 亀井克之, 中村泰明, 阿部 茂: エネルギー最小化による変形可能仮想ろくろモデラ, *信学論 (D-II)*, Vol.J76-D-II, No.8, pp.1772-1779 (1993).
- 8) Pimentel, K. and Teixeira, K.: *Virtual Reality*, Windcrest (1994).
- 9) Burdea, G. and Coiffet, P.: *Virtual Reality Technology*, John Wiley & Sons (1994).
- 10) Wyvill, B.: Animation and Special Effects, *Introduction to Implicit Surfaces*, Bloomenthal, J. (Ed.), pp.242-269, Morgan Kaufmann

- (1997).
- 11) Cani-Gascuel, M.P. and Desbrun, M.: Animation of Deformable Models Using Implicit Surfaces, *IEEE Trans. Visualization and Computer Graphics*, Vol.3, No.1, pp.39-50 (1997).
 - 12) Blinn, J.: A Generalization of Algebraic Surface Drawing, *ACM Trans. Graphics*, Vol.1, No.3, pp.135-256 (1982).
 - 13) 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, *信学論*, Vol.J68-D, No.4, pp.718-725 (1985).
 - 14) Wyvill, G., McPheeters, C. and Wyvill, B.: Data Structure for Soft Objects, *The Visual Computer*, Vol.2, No.4, pp.227-234 (1986).
 - 15) Hart, J.C.: Sphere Tracing: A Geometric Method for the Antialiased Ray Tracing of Implicit Surfaces, *The Visual Computer*, Vol.12, No.10, pp.527-545 (1996).
 - 16) Wyvill, G.: Ray Tracing Implicit Surfaces, *Introduction to Implicit Surfaces*, Bloomenthal, J. (Ed.), pp.166-195, Morgan Kaufmann (1997).
 - 17) Desbrun, M., Tsingos, N. and Gascuel, M.P.: Adaptive Sampling of Implicit Surfaces for Interactive Modeling and Animation, *Proc. Implicit Surfaces '95*, pp.171-185 (1995).
 - 18) 土井章男, 小出昭夫: 等関数値曲面生成のための4面体格子法, 第3回 NICOGRAPH 論文コンテスト論文集, pp.55-61 (1987).
 - 19) Lorensen, W.E. and Cline, H.E.: Marching Cubes: A High Resolution 3D Surface Construction Algorithm, *Computer Graphics*, Vol.21, No.4, pp.163-169 (1987).
 - 20) Bloomenthal, J.: An Implicit Surface Polygonizer, *Graphics Gems IV*, Heckbert, P. (Ed.), pp.324-349, Academic Press (1994).
 - 21) Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P.: *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed., Cambridge University Press (1993).
 - 22) Murray, J.D. and van Ryper, W.: *Encyclopedia of Graphics File Formats*, O'Reilly & Associates (1994).
 - 23) 3D Systems, Inc.: *Stereolithography Interface Specification* (1988).
 - 24) Thompson II, T.V., Johnson, D.E. and Cohen, E.: Direct Haptic Rendering of Sculptured Models, *Proc. ACM Interactive 3D Graphics*, pp.167-176 (1997).
 - 25) Popescu, V., Burdea, G. and Bouzit, M.: Virtual Reality Simulation Modeling for a Haptic

Glove, *Proc. IEEE Computer Animation '99*, pp.195-200 (1999).

- 26) McNeely, W.A., Puterbaugh, K.D. and Troy, J.J.: Six Degree-of-Freedom Haptic Rendering Using Voxel Sampling, *Proc. ACM SIGGRAPH '99*, pp.401-408 (1999).

(平成 12 年 10 月 17 日受付)

(平成 13 年 3 月 9 日採録)



松宮 雅俊(学生会員)

平成 9 年大阪工業大学工学部電子工学科卒業。平成 11 年奈良先端科学技術大学院大学情報科学研究科博士前期課程修了。同年同大学情報科学研究科博士後期課程入学, 現在に至る。形状モデリング, 仮想現実の研究に従事。平成 13 年情報処理学会関西支部学生奨励賞受賞。平成 13 年電気関係学会関西支部連合大会奨励賞受賞。ACM, 日本 VR 学会各会員。



竹村 治雄(正会員)

昭和 57 年大阪大学基礎工学部情報工学科卒業。昭和 62 年同大学大学院基礎工学研究科博士後期課程単位取得退学。同年(株)ATR 入社。3 次元ユーザインタフェース, CSCW, 仮想現実の研究に従事。平成 6 年奈良先端科学技術大学院大学情報科学研究科助教授。工学博士。電子情報通信学会, 映像情報メディア学会, IEEE, ACM, 日本 VR 学会各会員。



横矢 直和(正会員)

昭和 49 年大阪大学基礎工学部情報工学科卒業。昭和 54 年同大学大学院基礎工学研究科博士後期課程修了。同年電子技術総合研究所入所。以来, 画像処理ソフトウェア, 画像データベース, コンピュータビジョンの研究に従事。昭和 61~62 年マツギル大学知能機械研究センター客員教授。平成 4 年奈良先端科学技術大学院大学情報科学センター教授。現在, 同大学情報科学研究科教授。平成 1 年情報処理学会論文賞受賞。工学博士。電子情報通信学会, 人工知能学会, 日本認知科学会, 映像情報メディア学会, IEEE 各会員。