

# 自然言語による幾何形状モデリングと画像合成の一手法

河合善之<sup>†</sup>, 岡田稔<sup>††</sup>

本論文では自然言語インタフェースに基づいた三次元コンピュータグラフィクス(3D CG)のための幾何形状モデリングと画像合成の一手法を提案する。現在数多く利用されているCG作成システムでは、肢体機能障害などでキーボード・マウスなどのハードウェアデバイスを使うことが困難な肢体不自由者のユーザなどがCGを作成することは容易ではない。一方、言語指示によるインタフェースはそのような状況においても有効な手段と考えられる。そこで、本研究ではモデリングのためのマン・マシンインタフェースを自然言語に基づく音声とすることにより、このようなユーザにもCGが作成できることなどを目的としている。英語のサブセットによる文字列を利用して本手法を実装した三次元グラフィクスシステム“Do”に基づく基礎実験により、三次元モデリングに基づく簡単なCGを効率良く作成できることが確かめられた。また、本システムを利用することにより、CGの専門的知識のないユーザや、遠隔会議などにおけるコンピュータから遠隔地にいるユーザにもCGを作成することが可能となると考えられる。

## A Method for Geometrical Shape Modeling and Image Synthesis Based on a Natural Language

YOSHIYUKI KAWAI<sup>†</sup> and MINORU OKADA<sup>††</sup>

In this paper a method of geometrical shape modeling and image synthesis for three-dimensional computer graphics (3D CG) based on a natural language is proposed. Physically disabled users often find it difficult to use conventional CG modeling systems. This is due mostly to the modeling system's reliance on hardware input devices such as mouse, tablet and keyboard. Instead, a verbal interface should help them against such the obstacles. Our method is aiming to create CG images by such users with natural language voice as a man-machine interface for modeling. Our study found that the users can model simple 3D shapes and create the CG images easily. The experiments were performed by 3D CG System “Do” based on the proposed method with text strings in a small subset of English. The experiments also suggest that users with no expert knowledge and remote users on an on-line conference should be able to create CG images.

### 1. はじめに

三次元コンピュータグラフィクス(3D CG)を作成する際にはモデリング・レンダリングを目的としたグラフィクスソフトウェアが利用されよう。ここでは、幾何モデリングをはじめとした“CGに関する専門的知識”が必要なことが多く、そのような知識のないユーザがCGを作成することは容易ではない。

一方、既存のグラフィクスソフトウェアの多くでは、

キーボードのようなキャラクタ入力デバイス、マウスのようなポインティングデバイスを利用することが必要である。そのため、肢体機能障害によりそれらのデバイスが利用困難な肢体不自由者にとっては、3D CGを作成することは容易ではない。肢体不自由者のためのウィンドウ操作やテキストリーディングなどのインタフェースは福祉を目的としたHCIの一環として開発されているが、その基本機構を利用して二次元的な描画は可能であっても、幾何形状モデリングや物体変形操作<sup>1)</sup>を含む、三次元情報を駆使した3D CGの制作は容易ではない。しかし、音声を用いた言語指示によるインタフェースの利用はそのような状況においても有効な手段と考えられる。

そこで、本論文では自然言語を利用して3Dモデルを作成、画像合成するための一手法について述べる。すなわち、自然言語パーサとニーモニックによる中間

<sup>†</sup> 名古屋大学大学院工学研究科電子情報学専攻  
Department of Information Electronics, Graduate  
School of Engineering, Nagoya University  
現在、株式会社 NTT コムウェア  
Presently with NTT Communicationware Corporation

<sup>††</sup> 中部大学工学部情報工学科  
Department of Computer Science, College of Engineering,  
Chubu University

表現に基づいたモデリング方法を提案し、それに基づいたモデリングシステム“Do”の構成について述べる<sup>2),3)</sup>。特に、自然言語として英語を設定し、音声入力を最終目的とするための基礎検討として、文字列入力による三次元幾何モデリング法を検討する。システムとのユーザインタフェースを音声とすることができれば、キーボード、マウスのようなデバイスは必要なく、肢体不自由者でもグラフィクスシステムを扱うことが可能である。また、遠隔会議などのような、ユーザがシステムから遠隔地にいるような状況で、マイクを利用しシステムとの通信を行えば、このようなユーザにも容易に3D CGを作成することが可能となる。

このような自然言語を利用した関連研究として、二次元(2D)では自然言語と手をポインタとして利用した2Dレイアウト技術Put-That-There<sup>4)</sup>をはじめとして、風景描写文から風景画像を生成するPicnyck<sup>5)</sup>などが報告されている。三次元(3D)では仮想室内における家具配置作業のような室内レイアウトを考慮した3Dレイアウト技術<sup>6)</sup>に関する研究がある。また、動詞 *put* と *hang* を使用し、オブジェクトを配置するスクリプトベースのシステム、“Put”<sup>7)</sup>も報告されている。Putでは、オブジェクトをバウンディングボックスとして考え、前置詞の解釈・扱いを容易にしている。Putは、英単語を利用したプログラミング言語のスクリプトを入力ファイルとした、オブジェクトの単なる配置を行うシステムであり、回転、変形といった操作には対応していない。しかし、前置詞(*in, on, at, ...*)の解釈に関する考察はきわめて重要である。このほか、自然言語による画像アクセスの可能性に関して多くの研究<sup>8)~12)</sup>があるが、本研究はこれらの研究成果をふまえて、自然言語により三次元幾何モデリングおよび画像合成を行うことを目的としている。

以下、2章では提案する方式の基本的概念、次いで自然言語による幾何形状モデリングについて述べる。3章では提案方式を実装した3DCGモデラ“Do”による、CGエキスパートと非エキスパートを被験者とした実験の結果について述べる。

## 2. 自然言語による3D CGモデリング

### 2.1 自然言語による3D CGモデリングの基本的概念

本研究の最終目標は音声指示による3D CGモデリングであるが、本論文の実験ではその基礎検討として自然言語として英語のサブセットによる文字テキストを入力とする。本システムのフロントエンドとして音声認識部を付加すれば本研究の最終目的が達成され

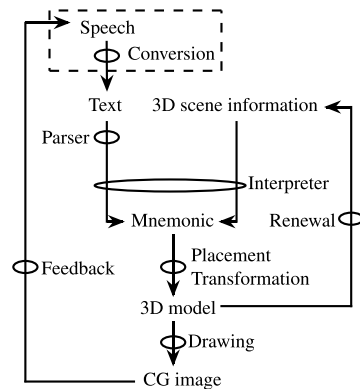


図1 自然言語によるCGモデリング処理の概要

Fig. 1 Overview of natural-language-based CG modeling process.

よう。

本論文で提案する自然言語による3D CGモデリング方式では、図1のような流れで処理を行う。

システムの基本的な流れは次のとおりである。

- (1) ユーザが入力した自然言語テキストは、2.3節で説明する構文解析部・インタプリタを通して、ニーモニックに変換される。
- (2) ニーモニックによりオブジェクトを配置・移動・変形などの操作を行い、3Dモデルを作成する。
- (3) スクリーン上に3Dモデルが描画される。

過去の3Dシーン情報、すなわち、スクリーン上に、今まで作成してきたオブジェクトの位置、サイズ、色などの情報を参照しながら構文解析部・インタプリタ(2.3節)によって、入力されたテキストは我々が定義するニーモニック(2.2節)に変換される。また、モデリングの際に、新しくオブジェクトを呼び出す必要がある場合は、システム付属のオブジェクト辞書からオブジェクトデータを読み出し、それを使用する。オブジェクト辞書にはオブジェクトを構成するCSG(Constructive Solid Geometry)のS式<sup>13)</sup>とS式を構成するプリミティブの情報が記述されている。以下、これらと付随情報を合わせて3Dモデルという。

このニーモニックによって、3Dオブジェクトを配置・移動・変形などの操作をしながら3Dモデルを作成・更新していき、生成された3Dシーンをスクリーンに描画する。この一連の操作を繰り返すことによって、ユーザは本システムを使用して、対話的に形状モデリングを行うことが可能である。本論文では、テキスト変換以降の処理(図1破線部の外)について述べる。

### 2.2 ニーモニック

本システムを作成するにあたって、CSGと自然言語

表 1 現在, “Do” で利用可能な英単語  
Table 1 Available words at “Do”.

Part of speech	Words	
Verb	<b>Mnemonic</b>	
	<b>PUT</b>	put, create, set, place, make
	<b>MOVE</b>	move, translate
	<b>ROTATE</b>	rotate, revolve
	<b>SCALE</b>	make ... smaller (larger, bigger), enlarge, shrink
	<b>PAINT</b>	paint, color, change the color of, alter the color of
	<b>DELETE</b>	delete, eliminate, erase, remove
	<b>UNITE</b>	merge, group, unite, take the union of
	<b>DISUNITE</b>	disunite, separate, ungroup
	<b>DIFFER</b>	cut, differ, take the difference of
	<b>INTERSECT</b>	intersect, take the intersection of
	<b>INVERT</b>	invert, take the inverse of
	<b>COPY</b>	copy, duplicate
	<b>NAME</b>	name, call
<b>REGISTER</b>	register	
<b>UNDO</b>	undo	
<b>QUIT</b>	bye, exit, quit	
Adjective	bigger, larger, smaller	
Adverb	right, left, up, down, in front, behind, back, center, little, more, to the right, to the left, to the behind, to the back, to the front, clockwise, anticlockwise, counterclockwise	
Preposition	above, behind, below, in front of, to the right of, to the left of, at, around	
Article	a, an, the	
Numeral	1, 2, 3, ...	
Conjunction	and	
Unit	dot(s), pixel(s), space(s) (スクリーン座標系) time(s), degree(s), mm, cm, m, km (ワールド座標系)	
Axis	x, y, z, x axis, y axis, z axis	
Color	red, green, blue, yellow, ...	
Object	primitive, newly created object, background, all	

表 2 Mnemonic  
Table 2 Mnemonic.

Mnemonic	意味
PUT <i>object direction (x, y, z)</i>	<i>object</i> を <i>direction</i> に配置
MOVE <i>object direction (x, y, z)</i>	スクリーン上の <i>object</i> を <i>direction</i> へ移動
ROTATE <i>object <math>\theta</math> (x, y, z)</i>	<i>object</i> を $\theta$ 度回転
SCALE <i>object k (x, y, z)</i>	<i>object</i> を $k$ 倍スケーリング
PAINT <i>object c (r, g, b)</i>	<i>object</i> を $c$ 色にペイント
DELETE <i>object</i>	<i>object</i> をスクリーンから削除
UNITE <i>object1 object2</i>	<i>object1</i> と <i>object2</i> をグループ化 (集合和演算)
DIFFER <i>object1 object2</i>	<i>object1</i> から <i>object2</i> を切削 (集合差演算)
INTERSECT <i>object1 object2</i>	<i>object1</i> と <i>object2</i> の集合積演算
INVERT <i>object</i>	<i>object</i> の集合否定演算
COPY <i>object</i>	<i>object</i> を複製
NAME <i>old-name new-name</i>	オブジェクト名を <i>old-name</i> から <i>new-name</i> に変更
REGISTER <i>object</i>	<i>object</i> をプリミティブ辞書に登録
UNDO	前回の操作をキャンセル
QUIT	システムの終了

を利用して 3D モデルを構築していく際に必要となる有益な英単語を集め, それらを分類した (表 1). そのうち, 動詞については特に詳しく分類を行った. その分類方法として, 同じような動作を示す動詞類 (たとえば *put*, *create*, *make*, *set*, *place* などはオブジェ

クトを配置する動作を示す動詞類) に分類し, 本システムにおける中間表現として表 2 に示すニーモニック (mnemonic) を定義した.

### 2.3 構文解析部

本研究ではその基礎検討として構文解析部は lex &

yacc<sup>14),15)</sup> を用いて作成した。現在、システムでサポートしている自然言語の書式を図 2 に BNF に準じて表記した。これらは英文法<sup>16)</sup> の第 1 文型, 第 3 文型, 第 5 文型に対応する。

現システムでは単文のみを解釈可能にしている。今後、より柔軟な会話,あるいは複雑な構文の解析が可能であるような高度な自然言語処理パーサをシステムに組み込む必要がある。

インタプリタでは、テキストを過去の 3D シーン情報を参照しながら二モーニックに変換する。図 2 の <sentence> はインタプリタによって、二モーニック(表 2)に変換される。

2.4 単位系の設定

システムが持つシーン情報とユーザが想定しているシーンには往々にして大きな隔りがある。その重要な一例として長さの単位系と画面の表示倍率の設定が

あげられる。システムは初期状態においてはユーザの想定しているサイズの程度を知ることはできず、ユーザが半径 10 cm の球の描画を指示しても、どのような大きさで表示すべきかは断定できない。そこで、システムは初期状態にあつては、長さの単位を無名数として扱い、ユーザが初めて単位を具体的に指示したときに、その単位を使用する。すなわち、初めて単位を指示する際には

- “Move the sphere 10 cm left.”
- “Move the sphere 10 km left.”

は画面に対しては同じ効果をもたらす。

単位系設定のための対処法はここで示した方法以外にも考えられるが、より詳細な検討が必要である。

2.5 度量・程度表現の解釈

インタプリタの設計の際に表 3 のようなユーザによる曖昧な表現の解釈方法が問題となる。

たとえば、“Move the sphere 1 m left.” のように、ユーザが位置を具体的な数値で指定した際は問題とならない。しかし、“Move the sphere left.” のような指示では移動距離が明示されていないため、システムは、ユーザが意図したとおりに移動させることは容易ではない。提案手法では、そのような曖昧表現の処理の基礎検討として、ある目標とする物体の位置 (z 地点) に関して、基準となる式 (1) で示す距離  $x$  を定め、 $x$  によって移動などの操作を行った (図 3)。

$$x = Y/10 \tag{1}$$

つまり、スクリーン上で一定の距離だけ移動するように設定した。したがって、物体が視点から遠くなる

```

<sentence> ::=
  < PUT >< obj. >
  | < PUT >< obj. >< adv. >
  | < PUT >< obj. >< prep. >< obj. >
  | < MOVE >< obj. > [< numeral >< unit > ]+
  < adv. >
  | < MOVE >< obj. >< prep. >< obj. >
  | < ROTATE >< obj. > [< numeral > degree(s) ]+
  [< direction > ]+ [around < Axis > ]+
  | < SCALE >< obj. > [< numeral > time(s) ]+
  [on < Axis > ]+
  | < PAINT >< obj. >< color >
  | < DELETE >< obj. > [< prep. >< obj. > ]+
  | < UNITE >< obj. > and < obj. >
  | < DISUNITE >< obj. >
  | < DIFFER >< obj. > from < obj. >
  | < INTERSECT >< obj. > and < obj. >
  | < INVERT >< obj. >
  | < COPY >< obj. >
  | < NAME >< obj. >< obj. >
  | < REGISTER >< obj. >
  | < UNDO >
  | < QUIT >
< obj. > ::= [< art. > ]< numeral > ]+ [< color > ]+
  < object >
< 大文字表記 > ::= (表 1: Verb (Mnemonic) 参照)
< adv. > ::= (表 1: Adverb 参照)
< adj. > ::= (表 1: Adjective 参照)
< prep. > ::= (表 1: Preposition 参照)
< numeral > ::= (表 1: Numeral 参照)
< Axis > ::= (表 1: Axis 参照)
< color > ::= (表 1: Color 参照)
< art. > ::= (表 1: Article 参照)
< object > ::= (表 1: Object 参照)
< unit > ::= dot(s) | pixel(s) | space(s) | mm | cm | m | km
< direction > ::= clockwise | anticlockwise
  | counterclockwise
    
```

図 2 Do で利用可能な文法 (BNF による)  
Fig. 2 Available grammar at Do (from BNF).

表 3 曖昧表現

Table 3 Ambiguous expressions.

曖昧表現	曖昧表現例	その問題点
位置	右に移動しろ	どれくらい右?
角度	右に回転しろ	どれくらい回転?
サイズ	大きくしろ	大きさは?
属性	球を置け	どんな球? 色は?
代名詞	それを動かせ	それって何?
前置詞	球の上に	上のどこ?

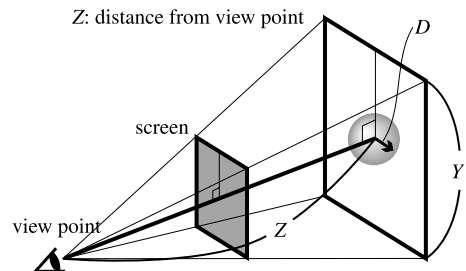


図 3 移動距離の基準  
Fig. 3 Criterion for moving amount.

と移動する距離も大きくなる．これに対して，“Move the sphere left little.” など，“少し” といった表現については移動距離を  $x/10$  とし，逆に “Move the sphere left more.” など，“もっと” といった表現には  $2x$  と解釈した（表 4）．また，サイズについても同様な方法で決定した（表 4）．しかし，これらの解釈方法は本システム的设计者の経験による主観に基づくものであるため，ユーザの意図を必ずしも反映していない．

前置詞，副詞の解釈でも，上記のような方法で位置決定した．そのため，前置詞が *above* であるのに，オブジェクトどうしが交差してしまう矛盾が生じうる．この問題に対しては，Put<sup>7)</sup> のようにオブジェクトにバウンディングボックスの情報を持たせ，オブジェクトどうしの干渉検知を行うことが必要である．また，本手法では前置詞 *on*，*in* については検討していないが，Put<sup>7)</sup> と同様に，*on* では接地を，*in* ではオブジェクトを包含させることを予定している．

このようなユーザの曖昧な表現をいかに解釈するかという問題はさらに詳細に検討する必要がある，今後の重要な課題である．

2.6 ニーモニック変換例

テキストがニーモニックに変換された例を表 5 に示す．この例で *object* とは，オブジェクトデータにリンクしている．“Make a red sphere above the cube.” ではオブジェクトは “a red sphere” であり，“Make

the sphere smaller.” では “the sphere” である．また “Make a red sphere above the cube.” の例では，インタプリタは “above the cube” に相当する 3D 空間の座標値を  $dir(x, y, z)$  として解釈する．

本研究では，第 2 文型・第 4 文型に関しては未対応とした．その理由として，まず，第 2 文型はオブジェクトを操作するモデリングに対して目的語 O のない V+C は重要度が低いと思われるからである．また，第 4 文型もモデリングには関連性の低い文型であると考えられる．しかし，より柔軟にテキストを解釈し，ユーザフレンドリなシステムを構築する際には，これらの文型も考慮する必要がある．

2.7 各操作の実現方法

本手法では操作対象のオブジェクトの情報として，オブジェクトの名前，形状，色 ( $(r, g, b)$  値)，位置 (3D 空間内の  $(x, y, z)$  座標値) 情報を持つ．

本研究ではモデリングプラットフォームとして CSG を使用しているため，以下のニーモニック表現でも CSG の考え方を利用している．

PUT, DELETE

たとえば，物体 A, B, C が 3D 空間に存在している様子を CSG 木で表現してみると，物体 A, B, C の和演算で表現できる．つまり，物体を置く操作 (PUT *object direction*) は CSG の和演算で表現できる．物体 C を 3D 空間から削除する操作 (DELETE C) は物体 A, B, C の CSG 木から葉 C を削除することにより実現できる．

UNITE, DISUNITE, INTERSECT, DIFFER, INVERT

これらは一般の CSG 木の變形により実現できる．物体 A, B の CSG 演算後に生成された物体を D とすれば，移動，變形などの操作は物体 D を操作すればよい．また，變形後に物体 A, B を操作することも可能である．

表 4 移動距離・サイズの解釈方法  
Table 4 Interpretation of distance and size.

	移動距離	サイズ	
		小さく	大きく
通常	$x$	0.8 倍	1.5 倍
		0.95 倍	1.1 倍
少し	$x/10$	0.5 倍	2 倍
		0.5 倍	2 倍
もっと	$2x$	0.5 倍	2 倍
		0.5 倍	2 倍

表 5 ニーモニック変換例  
Table 5 Examples of mnemonic.

文型	ユーザによる命令例	ニーモニック
第 1 文型 V	Undo. Bye.	UNDO QUIT
第 2 文型 V+C	未サポート	
第 3 文型 V+O	Make a red sphere above the cube. Rotate the cube around z axis.	PUT <i>object direction</i> ( $x, y, z$ ) ROTATE <i>object</i> $\theta$ ( $x, y, z$ )
第 4 文型 V+O+O	未サポート	
第 5 文型 V+O+C	Make the sphere smaller. Paint the sphere red.	SCALE <i>object</i> $k$ ( $x, y, z$ ) PAINT <i>object</i> $c$ ( $r, g, b$ )

## COPY

オブジェクトの複製を指示した際は、ニーモニック COPY が使われる。COPY は複製したい CSG 木と、同じ構造をした CSG 木を新たに作成し、シーンに配置する。

## MOVE, SCALE, ROTATE

オブジェクトの移動、スケーリング、回転操作はアフィン変換 (affine-transformation) により行った。

## PAINT, NAME

これらはオブジェクトの属性 (色, 名前) を変更する操作である。

## 3. 実験

### 3.1 実験方法と結果

提案手法を用いてどの程度の 3D モデルおよび画像を作成することができるのかを検証するために、提案手法によるグラフィクスシステム “Do” を実装して実験した。今回は簡易実験として行ったため、3D 対象シーン情報以外の設定、すなわち視点位置、画角、画像サイズ (アスペクト比) などの、観測者自身の設定 (カメラパラメータ) は固定値とした。また、プリミティブとして球と立方体のみを使用し、それらを配置・移動・変形し形状モデリングを行った。

その結果を図 4, 5, 6, 7, 8 に示す。図 4 の上部には、各々のシーンを作成する際に交わされたユーザとシステムとの会話を載せた。下部はその生成結果である。また、同様な方法でユーザはシステムと会話し、図 5, 6, 7 のようなモデルも作成できる。また、作成したオブジェクトはシステムのプリミティブ辞書に登録することができ、次回、同じオブジェクトを呼び出すことが可能である。辞書登録後、オブジェクトを呼び出しレイアウトしなおしたモデルが図 8 である。それぞれのモデルに対するユーザの会話に要した行数と作成時間を表 6 に示す。

また、本実験の被験者として、高度なコンピュータ利用者であり CG の非専門家に「雪ダルマ」を作成してもらい比較した (図 9, 10, 11)。ここでは、正確な位置・形状・寸法は要求せず、感覚的に作成することを意図とした。そして、他の方式のモデラとの比較を行った。その対象モデラとして Do, マウスのみを使ってモデリング可能な Extreme 3D II, POV-Ray のような入力をスクリプトとするモデラ (実際には CSG の S 式を記述) の 3 つで比較した (表 7)。

Do では、レンダリング方法として Z バッファ法を用いている。512 × 512 の画像サイズで、これらの画像をシステムと会話しながら生成するのに 6 ~ 40 分程

```
Input-> put an orange sphere.
Input-> put an orange cube.
Input-> move the cube down.
Input-> move the cube 10cm down.
Input-> put an orange cube.
Change the past object(cube) name. -> c1
Input-> move the cube up.
Input-> move the cube 3cm up.
Input-> cut the c1 from sphere.
Input a name for the newly created object. -> test1
Input-> cut the cube from test1.
Input a name for the newly created object. -> test2
Input-> make the test2 1.2 times larger.
Input-> put an orange sphere.
Change the past object(sphere) name. -> s1
Input-> cut the sphere from test2.
Input a name for the newly created object -> ring
Input-> put a chocolate sphere.
Change the past object(sphere) name. -> s2
Input-> make the sphere smaller.
Input-> move the ring down little.
Input-> group the ring and sphere.
Input a name for the newly created object. -> Saturn
Input-> make the saturn larger.
```

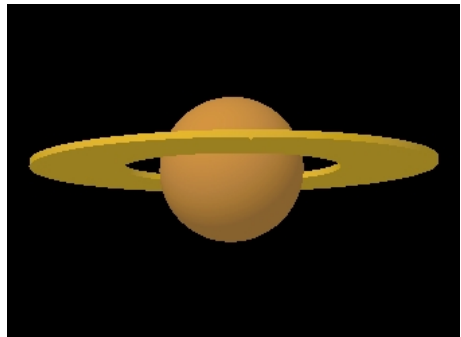


図 4 開発者が作成した土星とそのための会話の例  
Fig. 4 An example model and the conversation for 'Saturn' by the Do developer.

表 6 ユーザが入力に要したコスト  
Table 6 Cost for user's input.

3D モデル	ユーザ入力	生成時間 (約)
土星 (図 4)	24 行	6 分
イス (図 5)	62 行	15 分
コンピュータ (図 6)	126 行	30 分
デスク (図 7)	136 行	40 分
雪ダルマ (図 9)	91 行	20 分

度かかった (表 6)。使用機器は Sun Ultra 30 (Ultra SPARC II/300 MHz) である。

### 3.2 被験者による主観評価

2 名の被験者 A, B (コンピュータ専門家かつ CG 非専門家) を用いて、Do とマウス使用のモデラとの比較による主観評価を行った。Do の利点としては以下のような意見が出された。

- 事前知識をほとんど必要としない。

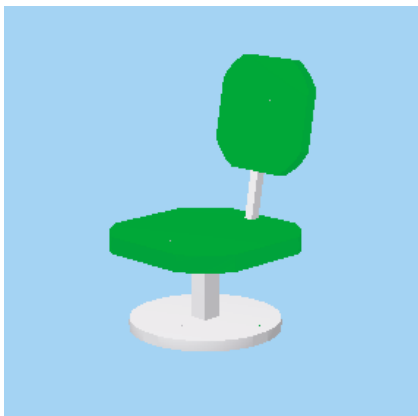


図 5 開発者が作成した“イス”  
Fig. 5 “A chair” by the Do developer.

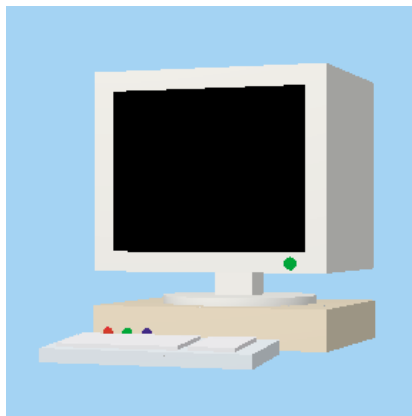


図 6 開発者が作成した“コンピュータ”  
Fig. 6 “A computer system” by the Do developer.

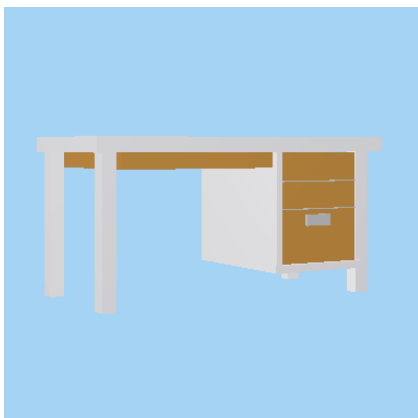


図 7 開発者が作成した“デスク”  
Fig. 7 “A desk” by the Do developer.



図 8 図 5, 6, 7 をレイアウト  
Fig. 8 A scene by the Do developer.



図 9 Do 開発者が作成した雪ダルマ  
Fig. 9 A snowman by the Do developer.



図 10 CG 非専門家 A が作成した雪ダルマ  
Fig. 10 A snowman by a subject A.



図 11 CG 非専門家 B が作成した雪ダルマ  
Fig. 11 A snowman by a subject B.

● オブジェクトの選択が容易である。

前者はマウス使用のモデラにおいてオペレーションメニューがどこにあるのかが分かりにくかったという意見である。Do では自然言語を志向したモデラであるため、その点が容易であったということである。また、オブジェクト選択が容易であるというのは、マウ

ス使用のモデラでは 3 次元でオブジェクトをクリックする際にオブジェクトどうしの重なりなどがあり、選択しにくいというものである。Do では、オブジェクトに名前を付けることによりオブジェクトの選択が容易である。しかし、今回の Do はすべてのオブジェクト名をユーザに命名させているためユーザフレンドリ

表 7 他のモデラとの生成時間の比較(雪ダルマ作成時)。ただし、Do の括弧内は会話に要した行数

Table 7 Generation time comparison for snowman.

対象モデラ	Do 開発者	CG 非専門家A	CG 非専門家B
Do	約 20 分 (91 行)	約 40 分 (138 行)	約 40 分 (113 行)
マウス使用のモデラ	約 10 分	約 40 分	約 45 分
スクリプトモデラ	約 110 分	約 150 分	-

であるとはいえない。

一方、Do の欠点としては以下のような意見が出された。

- 微調整がしにくい。
- スクリーン座標系・ワールド座標系の感覚がつかみにくい。

### 3.3 考察・検討

表 7 から分かるようにスクリプト入力のモデラではモデル作成にかなりの時間を費している。マウス使用のモデラでは Do 開発者と被験者 A, B との作成時間にかかなりの差がみられるが、これは被験者 A, B はこのモデラを使用するのが初めてであるためである。しかし、このことは Do でも同様であり、今回両者は Do でも初めての体験である。また、POV-Ray のようなスクリプトを入力とするモデラで、Do, マウス使用のモデラのように短時間、かつ、インタラクティブにモデリングすることは困難である。

また、実験結果のユーザとシステムの会話を見ると、オブジェクトの名前を変更している箇所がいくつか存在する。これは、指示語 (*a, an, the* など) をシステムが認識していないため、3.2 節でも述べたが、ユーザが名前を変更することによって、過去のオブジェクト (名前が同一のオブジェクト) との区別を図ったのである。今後、指示語、定冠詞やオブジェクトの属性データ (大きさ、色、位置など) の参照によって、たとえば、直前に操作した球、一番大きな球、赤色の球、一番左の球といった具合に、同じ種類のオブジェクトでも指示語によって区別できるシステムにする予定である。

ユーザの主観評価 (3.2 節) によると微調整がしにくいという意見が存在した。これは、事後の聞き取りによれば曖昧表現の解釈方法が不完全であるためである。今後 2.5 節でも述べた曖昧表現の詳細な考察を行うことは重要な課題である。

今回の手法の検討・使用・評価を通して得られた主要な問題点を以下にまとめる。

- ユーザによる度量、程度のあいまい表現の解釈法 (2.5 節)
- 数値指定されていない表現の解釈法 (2.4 節)
- 数値指定における単位系 (2.4 節)

- 反対の操作 (大きく ⇔ 小さく、など) による対称性 (2.5 節)
- オブジェクトの干渉検査 (2.5 節)
- 第 2, 4 文型への対応方法 (2.6 節)
- オブジェクトの選択・指示の方法 (固有名詞, 代名詞, 冠詞) (3.2 節)
- 形状・サイズなどの微調整 (3.2 節)
- カメラパラメータ (3.1 節)

## 4. ま と め

本論文では、肢体不自由者、CG の専門的知識のないユーザなどにも CG が作成できることを目標とした自然言語に基づく 3D CG モデリングの一手法を提案した。音声入力による最終目的の前段階として、テキスト入力による自然言語を利用してモデリングを行う際の問題を考察した。特に動詞類を核としてニーモニックを定義し、その中間表現を用いたインタプリタによる方式に基づいた。

本手法に基づくシステム Do による実験では、CG 非専門家の被験者により他のモデラに対する Do の主観評価を行った。実験結果からも分かるように、自然言語をユーザインタフェースとして用いても、ある程度の 3D モデルの作成は可能であることが分かった。

今後の課題として、3.3 節で述べたシステムの構築と問題点の検討を行うとともに、ユーザの意図をシステムが学習しながら理解していくシステムを作成し、文献 (17), (18) に見られるような非線形な変形操作もシステムに組み込むことがあげられる。また、実装したシステム “Do” は限定された集合演算、限定されたプリミティブ、固定されたカメラパラメータなど、幾何モデラとしてはかなり制約がある。幾何モデラとしての必須基本機能を持ちつつ、非専門家にも容易にその機能を扱えるようにすることが必要である。

なお、本研究は自然言語インタフェースを用いた 3D CG モデリングと画像合成の可能性の基礎検討を目的としており、自然言語処理そのものを目的とはしていない。このため、簡易なパーサを用いたが、高度な自然言語処理パーサの利用<sup>19)</sup>により、柔軟なインタフェースを構成することが可能であろう。英語以外の自然言語によるインタフェースの可能性の検討も重要課題で



ある。

### 参 考 文 献

- 1) Barr, A.H.: Global and Local Deformations of Solid Primitives, *Computer Graphics*, Vol.18, No.3, pp.21-30 (1984).
- 2) 河合善之, 岡田 稔: 自然言語による 3DCG モデル Do, 情報処理学会研究報告, GCAD 94-1, pp.1-6 (1999).
- 3) Kawai, Y., Higashiyama, Y., Koyama, K. and Okada, M.: A Fundamental Study on a Natural-Language-Based 3D CG Modeling, *Proc. SMC'99 - 1999 IEEE Systems, Man, and Cybernetics Conf.*, Vol.V, pp.714-719 (1999).
- 4) Bolt, R.A.: Put-That-There: Voice and Gesture at the Graphics Interface, *ACM-SIGGRAPH*, Vol.14, No.3, pp.262-270 (1980).
- 5) 西山晴彦, 大久保達真, 松下 温: Picnyck: 風景描写文から風景画像の創造, 情報処理学会論文誌, Vol.38, No.5, pp.997-1007 (1997).
- 6) 望月研二, 岸野文郎: 自然言語による 3 次元画像へのアクセス, 電子情報通信学会技術報告, NLC92-30・PRU92-44, pp.69-75 (1992).
- 7) Clay, S.R. and Wilhelms, J.: Put: Language-Based Interactive Manipulation of Objects, *IEEE Computer Graphics and Applications*, Vol.16, No.2, pp.31-39 (1996).
- 8) 山田 篤: 自然言語による視覚表現と対象間の位置関係の対応に関する考察, 電子情報通信学会技術報告, NLC95-34・PRU95-139, pp.27-32 (1995).
- 9) 原田 実, 野村佳秀, 山本幸二, 大野雅志, 田村浩樹, 高橋史郎: 自然言語要求仕様からオブジェクト指向設計図を自動生成するシステム CAMEO, 情報処理学会論文誌, Vol.38, No.10, pp.2031-2039 (1997).
- 10) 高橋友一, 島 則之, 岸野文郎: 位置情報を手がかりとする画像検索法, 情報処理学会論文誌, Vol.31, No.11, pp.1636-1643 (1990).
- 11) 横田将生: 言語および画像におけるメディア変換について, 電子情報通信学会技術報告, NLC95-35・PRU95-140, pp.33-38 (1995).
- 12) 北橋忠宏: メディア変換から見た言語と画像および認識・理解の役割, 電子情報通信学会技術報告, NLC95-39・PRU95-144, pp.57-62 (1995).
- 13) 岡田 稔, 金 環垠, 横井茂樹, 鳥脇純一郎: 半空間プリミティブと 3 値論理代数系に基づく多面体の記述, 情報処理学会論文誌, Vol.38, No.8, pp.1583-1592 (1997).
- 14) Levine, J.R., Mason, T. and Brown, D.: *lex & yacc*, アスキー出版局 (1996).
- 15) 五月女健治: *yacc/lex*, テクノプレス (1996).

- 16) 杉山忠一: 英文法詳解, 学習研究社 (1998).
- 17) 福嶋茂信, 美濃導彦, 池田克夫: 画像情報と言語的指示を用いた曲面モデルの作成, 画像の認識・理解シンポジウム (MIRU'96), Vol.1, pp.301-306 (1996).
- 18) 福嶋茂信, 美濃導彦, 池田克夫: 言語表現を利用する 3 次元形状モデルの作成, 情報処理学会研究報告, Vol.97, No.79, pp.55-60 (1997).
- 19) 特集号各論文, フィールドを広げる自然言語処理, 情報処理, Vol.40, No.4 (1999).

(平成 12 年 10 月 5 日受付)

(平成 13 年 3 月 9 日採録)



河合 善之 (学生会員)

昭和 47 年生・平成 9 年名古屋大学工学部電気系学科卒業。平成 11 年同大学大学院博士前期課程電子情報学専攻修了。同年 4 月 (株)NTT コミュニケーションウェア (現在, NTT コムウェア) 入社, 現在に至る。在学中, コンピュータ・グラフィクス, 特に半空間プリミティブに基づく幾何形状モデリング, 自然言語による幾何形状モデリングに関する研究に従事。



岡田 稔 (正会員)

昭和 59 年名古屋工业大学工学部電気工学科卒業。平成元年名古屋大学大学院博士課程後期課程情報工学専攻修了。工学博士。昭和 63・平成元年度日本学術振興会特別研究員。平成 2 年名古屋大学情報処理教育センター助手。平成 5 年同大学助教授。平成 10 年同大学院工学研究科情報工学専攻助教授。平成 11 年中部大学工学部工業物理学科教授を経て, 平成 12 年 4 月同情報工学科教授, 現在に至る。平成 7 年より 1 年間, 米国ミシガン大学客員准教授。画像とパターンの認識・理解, 計算幾何学, 画像合成, マルチメディア情報通信の基礎理論と応用に関する研究に従事。電子情報通信学会, 画像電子学会, IEEE 各会員。平成 4 年度情報処理学会研究賞, 平成 6 年度市村賞学術貢献賞, 平成 12 年度情報処理学会優秀教育賞各受賞。著書「C によるプログラミング演習」(近代科学社), 「情報科学基礎論」(朝倉書店), 「インターネット時代のコンピュータ活用法」(コロナ社) 等。