

図学およびコンピュータグラフィックス教育のための 3次元立体生成ソフトウェアの開発

A Development of 3D Solid Modeling Software for Education of Descriptive Geometry and Computer Graphics

新津 靖*

東京電機大学工学部

〒101-8457 東京都千代田区神田錦町2-2

TEL 03-5280-3379 FAX 03-5280-3569

E-mail: yniitsu@cck.dendai.ac.jp

Abstract: A 3D solid modeling software has been developed for the education of the descriptive geometry and computer graphics. The boundary representation and the winged-edges method have been adopted as a description method of solid model. The developed solid modeler can create a 3D solid model by executing the command lines of text format. Combining the ten kinds of primitive solid models, the solid model is constructed. The developed software has several kinds of display modes such as the edge-type and the surface-type display modes. It is possible to advance the production of the solid model by confirming the graphic display.

This software was applied to the education of the descriptive geometry for the students in the first year of Tokyo Denki University. The result of the questionnaire after the class suggests that the three-dimensional understanding appeared to be improved.

Keywords: 3D solid modeler, descriptive geometry, computer graphics

1. はじめに

大学の初等教育において、機械系学科や土木・建築系学科では図学教育が行われているが、3次元の意識能力には個人差がある上、2次元投影図と3次元物体の関係の理解には多くの勉強時間と努力を要する。このため、3次元意識能力が将来技術者として働く上で非常に重要な能力であるにもかかわらず、学生から敬遠され、十分な実力をつけられていないのが実情である。また、コンピュータの進歩に伴い、図学の後にくる製図教育では、多くの大学でコンピュータを使用したCAD教育に移行しており、図学教育もコンピュータを援用した教育に移行する必要性が叫ばれている。そして、いくつかの研究機関および大学において、多くの資金を投入し設計製図や図学教育用を狙った、2次元図形および3次元立体生成ソフトウェアの開発が試みられてきた。しかし、広く一般の大学で使用される製図教育用あるいは図学教育用のソフトウェアは、いまだ完成していないのが現状である。3次元立体を扱えるソフトウェアについては、フリーソフトウェアの中にも、ポプレイなどレイトレーシングも可能なソフトウェアがあるが、コマンドや操作方法が複雑な上、デザイン的要素が強く、立体認識能力を養うための教育用には必ずしも適切なソフトウェアとはいえない。特に図学教育では、陰線処理や切断面、立体の相貫などを教育する必要がある、このような教育に対応できるフリーソフトウェアあるいは安価な市販ソフトウェアはない。

東京電機大学工学部機械工学科では、10年前、1年次生のコンピュータ教育の充実を機に、1年次の図学授業を廃止し、コンピュータ授業の中で図学教育を行う方向でコンピュータ教育のカリキュラムを組んだ。この中では前期にPascalプログラミングの習得を行い、後期に2次元および3次元グラフィックスのプログラムならびにデータ作成を

行った。3次元グラフィックスでは投影法、ワイヤフレームモデル、サーフェイスモデル、陰線処理、相貫線表示を行っていた。しかし、教育システムの環境がMS-DOSからWindowsに変わるのに伴い、ワープロや表計算、数式処理などのリテラシー教育に多くの時間が割かれるようになり、プログラミング教育に割かれる時間が減少してきた。そして、3次元グラフィックスについても最近ではワイヤフレームモデルまでしかできなくなってきた。このような現状から、図学教育を意識したコンピュータ教育を1年次に行うには、初心者でも操作できる図学教育用ソフトウェアを開発および導入する必要があった。

このように、使いやすく教育的にも配慮された図学教育用ソフトウェアの必要性から、また、東京電機大学の1年次生のための図学教育とコンピュータ教育の充実のため、本研究では、初心者でも使用可能な、図学教育を支援できる機能を備えた高機能3次元立体生成ソフトウェアを開発した。さらに、開発したソフトウェアを機械工学科の1年次に配布し、週3時間×5回のコンピュータ教室での授業を行った。

2. 3次元ソリッドモデラーの開発

3次元立体生成ソフトウェアの開発は6年前から始められた。1995年にMS-DOSで動作する3次元ソリッドモデリングソフトウェアを作成し、1年次生のコンピュータ教育で使用した。しかし、操作性の悪さと実行速度面で使用に耐えられるものではなかった。このため、一時そのソフトウェアの使用を中断し、2年前よりWindowsNTおよびWindows95で実行できる高機能3次元立体生成ソフトウェアの開発を始めた。ソフトウェア開発にあたり、従来のプログラムの欠点を洗い直し、開発環境をCからC++に変更した。また、立体のデータ構造には、有限要素解析への応用や曲面表示を可能にする機能を積極的に取り入れ、次の機能強化を図った。

*Yasushi Niitsu

Tokyo Denki University

- 立体，面，グループなどのデータ構造の改良とクラス化および機能強化
- 立体生成アルゴリズム，投影処理アルゴリズムの見直しと高速化
- コマンドの見直しとコマンド解析の効率化および分散処理化
- GUI環境による自動立体生成機能の追加および強化

現在までに開発されている高機能3次元立体生成ソフトウェア（Solid Interpreter）は十分に使用に耐えるものであり、初期の目的であった図学教育の補助的ツールの枠組みを超える機能を持たせることに成功したと考えている。すなわち、3次元グラフィックス教育やWindows95などのアプリケーションソフトウェアの標準的操作方法の習得などにも応用できるものと考えられる。

3. 3次元立体の形状表現とプログラミング

(1) 内部形状モデル

コンピュータ内部のデータとして、3次元立体を表現する方法には2種類の方法がある。一つは立体の境界条件である面や稜線、頂点を用いて立体を表すB-Reps法（Boundary Representation）であり、もう一つは、円柱や直方体など直感的にわかりやすい基本的な立体の組み合わせで目的の立体を表現するCSG法（Constructive Solid Geometry）である。CSG表現は、直感的に理解しやすい表現方法であるが、集合演算や投影表示の容易さや、市販のほとんどの3次元CADと有限要素解析用プリポストプロセッサがB-Reps法を採用していることを考慮し、本ソフトウェアも内部形状表現にB-Reps法を採用した。^{[1][2][3][4]}

立体は、基本的に多面体として表現され、そのデータ構造にはWinged-edgeデータ構造を採用した。^{[2][3][4]} すなわち、立体の境界を表す面の構造体と稜線の構造体のリスト構造により、立体を表現する方法である。プログラムで扱うデータには、名前を付けて管理するものとして、表1に示す4種類のものがある。

視線情報は、立体を表示するときの条件を設定するためのものである。表示モードには、面表示^[5]と稜線表示モードがあり、陰線の表示・非表示を指定することができる^[6]。また、3面図表示により、通常の2次元設計図面に対応した表示を行うことができるように設計した。軸構造体は、立体やグループに動作を定義するためのものである。グループは立体や子グループをまとめて扱うためのものである。

立体は内部にその立体を構成する面構造体のリストを情報としてもっており、面を順次表示することで、立体を表

表1 名前を持つObjectのデータ構造

Object	Objectが有する主な情報
視線情報	視線名，視点座標，注視点座標，表示モード
軸	軸名，関連立体/グループ名，軸方向ベクトル，移動量
グループ	グループ名，子グループリスト，子立体リスト
立体	立体名，面色，稜線色，親グループ，境界面へのポインタ，面数，稜線数

示させることができる。面，稜線，頂点は立体を表現するものであるが、これらは名前で管理せず、ポインタのリスト構造により管理されている。

(2) 3次元立体モデルの生成方法

本研究で開発した3次元立体生成ソフトウェア（Solid Interpreter）は、立体の生成や変形などをすべてテキスト形式のコマンドで行う方法を取っている。そして、基本立体生成コマンド，立体操作コマンドおよびその他のコマンドとして分類される31種類のコマンドで記述されたテキストファイルを実行することで、立体の生成と表示を行う。これらのコマンドの機能と書式を表2，表3，表4に示す。立体の生成や変形などの操作はすべてテキスト形式で記述されるため、その過程を調べたり変更することが容易となり、テキストファイルの修正と実行を繰り返しながら、目的の立体を生成することができる。

立体モデルの作成は、図1に示すような、5種類（直線，多角柱，多角錐，2種類の回転体）の立体生成コードを用いて生成できる10種類（直線，直方体，円柱，円錐，球，回転楕円体，トーラス，多角柱，多角錐，2種類の回転体）の基本立体を組み合わせて行うことができる。図1の矢印は、上の基本立体と同じ生成プロセスで下の基本立体を生成していることを示している。

生成した基本立体に、移動，回転，集合演算^{[4][5][6]}変形などを施して目的とする立体を生成することができる。たとえば、二つの立体の加算（和集合），減算（差集合），積（積集合）を作る集合演算の例を図2に示す。図2は球と円柱の集合演算によって生成される立体を示しているが、任意の立体間で図2のような集合演算が可能である。

表2 基本立体の生成コマンド

生成立体	コマンド	主なパラメータ
円錐	Cone	立体名，半径，高さ
直方体	Cube	立体名，X,Y,Z長さ
円柱	Cylinder	立体名，半径，高さ
回転楕円体	Ellipse	立体名，XY半径，XZ半径
直線	Line	立体名，始点座標，終点座標
多角柱	Prism	<op>立体名，高さ，座標点列
多角錐	Pyramid	<op>立体名，高さ，座標点列
回転体	Revolve	<op>立体名，座標点列
球	Sphere	立体名，半径
円環体	Torus	立体名，大半径，小半径

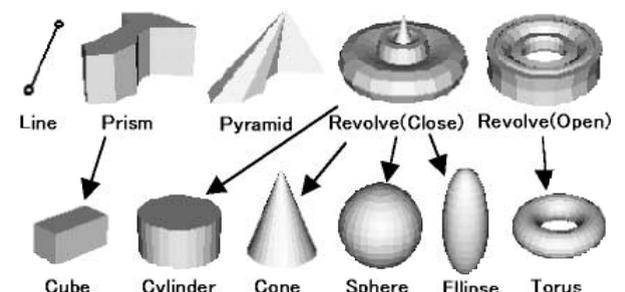


図1 基本立体の種類と生成方法

(3) 立体の表示

立体をグラフィック表示するためのコマンドがEyeコマンドとDisplayコマンドである。通常の3次元表示に必要な視点や注視点座標に加えて表示モードを指定することができる。この表示モードはグラフィック画面の表示効果を表すもので、その種類は75通りにもなる。図3にいくつかの代表的な表示モードによるグラフィック出力の例を示す。

表示モードには大きく分けて稜線表示と面表示がある。稜線表示には、さらに陰線の表示・非表示と曲面上の稜線

の表示・非表示の選択ができる。また、通常の透視投影に加えて、軸測投影と3面図表示モードを選択することもできる。

(4) 3Dプログラミングの支援ツール

立体の生成や表示コマンドをGUI環境下で入力したり、エラーの発生箇所やエラーの内容を表示するプログラミング支援ツールを用意した。これらの支援ツールを利用することにより、基本的な立体の生成方法を2～3時間の短時間で習得することができる。

表3 立体操作コマンド

機能	コマンド	書式または主なパラメータ
立体のコピー	=	複写先立体 = 複写元立体
集合演算	=, +, -, *	立体3 = 立体1 op 立体2
アフィン変換	Affine	立体名, 中心, 変換行列
消去	Delete	立体/グループ名
軸設定	Axis	<op>軸名, パラメータ
グループ設定	Group	<op>グループ名, パラメータ
平行移動	Move	立体名, 相対変位
複数コピー	Ncopy	<op>立体名, コピー方向, 数
名称の変更	Rename	旧立体名, 新立体名
回転移動	Rotate	立体名, 回転中心, 軸方向
カラー設定	Color	立体名, 表示色(16色)
面カラー設定	Fcolor	立体名, 表面色(R.G.B.)
カラー設定	Xcolor	立体名, 表示色(R.G.B.)

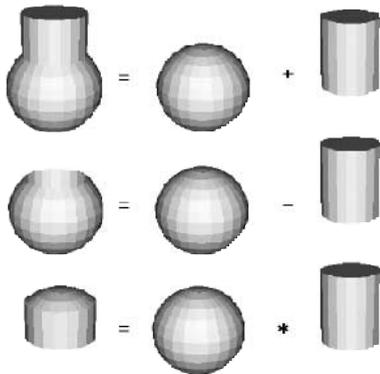


図2 集合演算によって生成される立体の例

表4 その他のコマンド

機能	コマンド	書式または主なパラメータ
コメント	#	# ~ 改行までがコメント
視線設定	Eye	<op>視線名, 視点, 注視点, 拡大率, 表示モード
画面消去	Clear	なし
座標軸表示	Discord	<op>座標軸長さ, 表示色
立体表示	Display	視線名, 立体/グループリスト
実行終了	End	なし
ファイル実行	Include	プログラムテキストファイル名
CADファイル出力	SaveCAD	DXFファイル名, 立体/グループ名リスト
Window設定	Window	表示枠指定

プログラミング用エディタ

コマンド行を入力するためのマルチドキュメントタイプのテキストエディタを付けた。テキストのカット&ペーストや検索・置換、Undoなど操作方法はワープロや他のエディタと同じに設計した。入力したプログラムはその場で実行してチェックすることが可能で、複数のファイルを開いて、コピー&ペーストによりコマンドを移動できるなど、操作性をよくした。

プリミティブ生成Windowとコマンド生成Window

Solid Interpreterのプログラミングはコマンド行をキーボードから直接入力して行うことができるが、マウスとグラフィックスを使ったコマンド行の生成ツールを作り、より視覚的に立体を作れるようにした。そのためのツールが図4に示すプリミティブ生成Windowとコマ

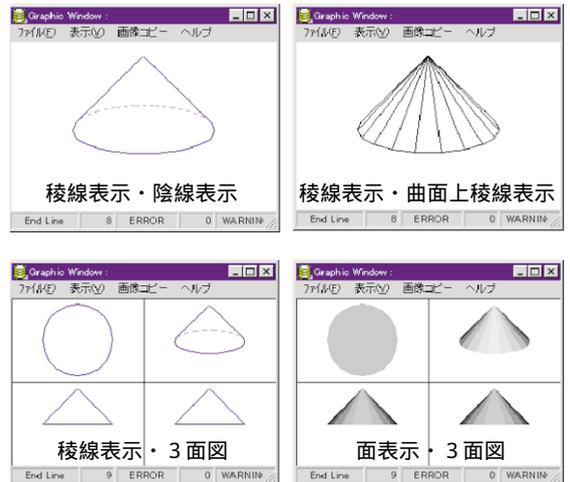


図3 各種表示モードによるグラフィック表示例

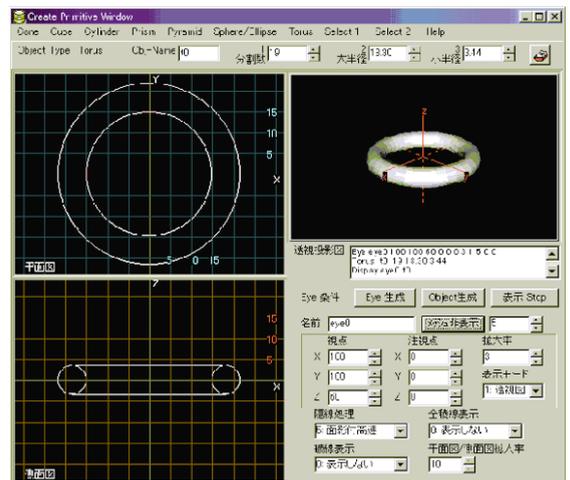


図4 プリミティブ生成Window (トーラス生成時)

ド生成Windowである。プリミティブ生成Windowでは、基本立体を視覚により確認しながら、簡単に生成することができる。同時に、視線情報も確認しながら生成することができるように設計した。移動や回転、軸指定やグループ生成、色指定やAffine変換などのコマンドはプリミティブ生成Windowからコマンド生成Windowを呼び出して生成することができる。

メッセージWindow

コマンド実行時のエラーや警告を表示するためのWindowがメッセージWindowである。メッセージをクリックすることで、メッセージが出されたコマンド行の内容とファイル名が分かり、ダブルクリックでその行に飛ぶことができる。

状態変数Window

立体やグループ、軸、視線情報のパラメータの状態を調べるためのWindowが状態変数Windowである。グループと軸と立体の関係もファイル構造のように調べて表示することができる。

(5) 3Dプログラミングの例

表2, 3, 4で示したコマンドを組み合わせて3次元立体モデルの生成と表示を行うことができる。コマンド行はコマンド名の後にパラメータをスペースかカンマで区切って記述する。次に二つの簡単なプログラム例を示す。図5および図6はそれぞれ、の実行結果である。

トーラスと円錐の生成と表示 (実行例1)

```
Eye eye0 100, 100, 60.0, 0, 0.3 1 5 0 0 #視線設定
Torus t0 18 17.90, 2.40 #トーラスの生成
Move t0 0, 0, 15 #トーラスの移動
Cone c0 18 9.31, 27.20 #円錐の生成
Display eye0 t0 c0 #t0,c0の表示
```

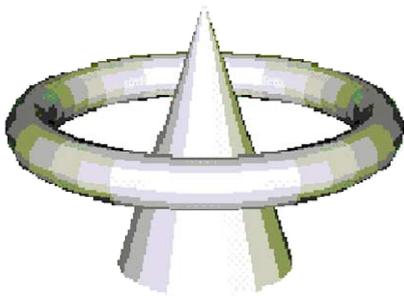


図5 実行例1

円柱同士の集合演算と表示 (実行例2)

```
Eye eye0 100, 100, 60.0, 0, 0.3 4 2 0 0 #視線設定
Cylinder y0 16 10.0, 30.0 #円柱y0を生成
Cylinder y1 16 5.0, 30.0 #円柱y1を生成
Rotate y0 0.0, 15 1.0, 0 90 #y0をx軸90度回転
Add = y0 + y1 #y0+y1の生成
Sub = y0 - y1 #y0-y1の生成
Display eye0 Add #Addの表示
Wait 2; Clear #2秒待ち, 画面消去
Display eye0 Sub #Subの表示
```

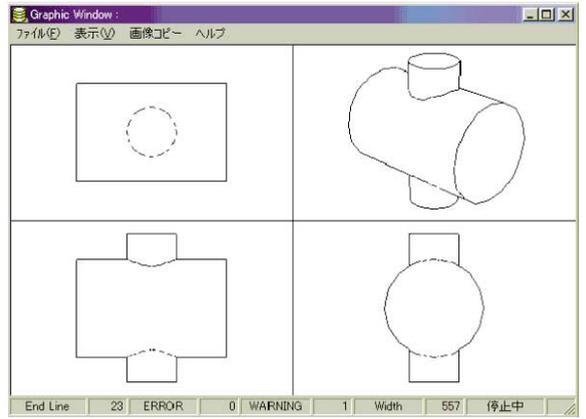


図6 実行例2

4. 教育事例

(1) 授業計画と授業内容

開発したソフトウェアは、東京電機大学工学部機械工学科1年次生約130名を対象にした「コンピュータ基礎および演習」(90分2コマ12~14週)において教育に使用した。この授業は新生が最初に受けるコンピュータの授業であり、ワープロ、表計算、電子メール、数式処理、ホームページ作成などを実習を通して習得するためのものである。この授業の後半の第7週から第11週の5週(90分2コマ×5)を使用して、開発したソフトウェアの使用方法を習得するとともに、三角図法による立体の表現方法と投影図法⁷⁾について学習した。

この実験授業の目的は、通常の図学と同様に、3次元構造物の認識能力を高めることにある。単にソフトウェアの使い方の習得に終わらないように授業計画を立て、演習問題と課題を作成した。学生には方眼紙と定規を持参させ、立体の透視図や3面図を描く作業も行わせた。授業は、前半の90分を幾何学の説明やソフトウェアの使い方とコマンドの説明に当て、主に教員の指示に従って操作させた。そして、後半の90分を課題作成などの実習に当てた。

主担当教員は著者の新津で、講師の竜田氏、助手の若井氏に加えて3名の院生のティーチングアシスタント(TA)が支援する態勢で授業は進められた。なお、4月15日の最初の授業でデモプログラムを見せ、5月中旬に学生にプログラムを配布したときに約1時間、基本的な使用方法を説明している。実際の各週の授業および実習内容を以下に示す。

第7週 6月3日

(基本立体の生成, 3面図, 投影図, 回転ベクトル)

[講義内容]

- ア. B5判12ページのマニュアルの配布
- イ. 基本立体の作り方と表示方法・表示モードの説明
- ウ. 平面図, 正面図, 側面図と立体の関係および陰線の意味と例題による確認
- エ. 立体の平行移動と回転移動コマンドの使い方の習得

[実習内容]

- ア. 簡単な形状の立体透視図からの3面図の描画
- イ. 3面図から透視図の描画(フリーハンド)
- ウ. 複数の基本立体の生成と表示
- エ. 立体の平行移動と回転移動の実習

第8週 6月10日
(立体の作り方とコマンドの習得)

- [講義内容]
- ア．立体間の集合演算方法の習得と断面図・相貫線の例
 - イ．立体および背景の色指定
 - ウ．アフィン変換による立体の変形の習得
 - エ．アニメーション表示方法の習得
 - オ．グループコマンドの説明と簡単な例
 - カ．最終課題の説明
- [実習内容]
- ア．集合演算を用いた立体の生成と表示の課題
 - イ．3面図から透視図の描画(フリーハンド)
 - ウ．身の回りにある立体の生成と表示

第9週 6月17日(課題の実習および自習)
 主担当教員が海外出張のため、竜田講師によるMS-Excelの高度な利用方法の説明と実習、後半に立体生成課題の実習および最終課題の自習を行った。

第10週 6月24日
(MS-PowerPointの使い方、最終課題テーマ決定)

- [講義内容]
- ア．PowerPointの使い方
 - イ．画像のカット&ペーストの方法
 - ウ．グループコマンドと軸コマンドの使い方
- [実習内容]
- ア．PowerPointを使用した最終課題発表スライドの製作
 - イ．最終課題の決定と最終課題の立体作成の自習



図7 授業風景



図8 教卓からの教材提示方法(6月3日)

第11週 7月1日
 最終課題発表会と優秀な作品の表彰およびアンケート調査

図7に授業風景を示す。約180台のコンピュータを備えた教室の約3分の2を使用して授業は進められた。図8に示すように、書画カメラとコンピュータ画面を切り替えて表示しながら講義および実習を行った。

課題提出は方眼紙での提出が3件、プログラムファイルの提出が3件である。そのうちの図面とプログラムの計2件が最終課題である。最終課題は各自でテーマを決めた立体を作るというもので、第5週目に発表会を行った。

(2) 最終課題発表会と学生の作品

最終課題の発表は、各自1分間の持ち時間を与え、PowerPointのスライド1枚による説明と製作したモデルの表示を全学生に見せる形式で行った。学生の発表風景を図9に示す。優秀なプログラムを組んだ学生は表彰し、最後にもう一度発表してもらった。ほとんどの学生が立体の移動や変形と表示を繰り返すアニメーションを作るという、かなり高度なプログラムを完成させているのには驚かされた。図10に学生の作品の一部を示す。実際はこれらのモデルが動き回るようにプログラムされている。

(3) アンケート結果

発表会終了後にSolid Interpreterを使用した3Dグラフィックスの授業に関するアンケートを実施した。アンケートの質問内容は、

- 最終課題に費やした時間
- 最終課題作成を行った場所(複数回答)
- 最終課題のでき映えについての満足度
- 空間認識能力が向上したと思うか
- 自宅でコンピュータが使える環境にあるか
- ソフトウェア配布についての意見

3Dモデラーを使った授業についての感想である。133名の学生からアンケートを回収することができた。アンケートの集計結果を表5と図11に示す。表5の数値と図11の縦軸は回答数を示している。

約71%の学生が自宅でコンピュータが使える環境にあり、ソフトウェアの配布は学生から非常によい評価を得ていた。最終課題の制作に10時間以上の時間を費やした学生のほとんどが、自宅でも最終課題に取り組んでいた。

表5の結果から、時間のかかる最終課題は、自宅で作る学生が半数程度にのぼった。最終課題にかけた時間は4~10時間をもっとも多く、若干少ないように思われるが、これは学生がコンピュータに向かって作業している時間を書



図9 最終発表風景: PowerPointで説明後実行

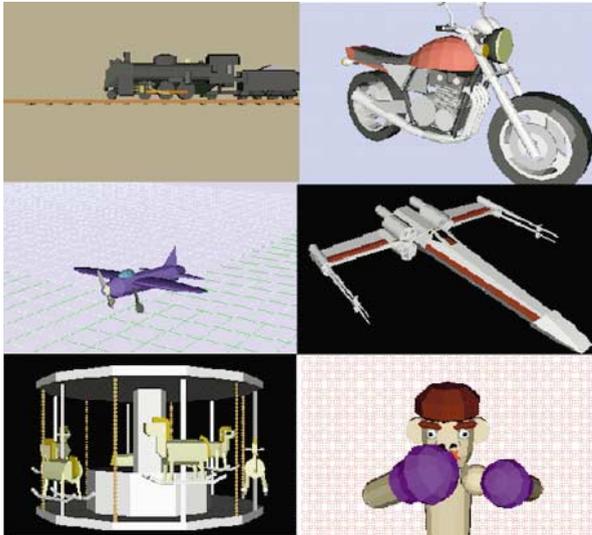


図10 最終課題で提出された学生の作品

表5 アンケート集計結果

最終課題に費やした時間		最終課題をどこで行ったか	
~2h	3	大学	90
2~4h	12	自宅	75
4~10h	60	その他	4
10~20h	36	自宅にコンピュータが	
20~30h	9	ある	94
30h以上	17	ない	39
最終課題の満足度		空間認識は向上したか	
満足	38	変わらないと思う	32
不満	67	良くなったと思う	73
わからない	24	わからない	28
その他	2	その他	0

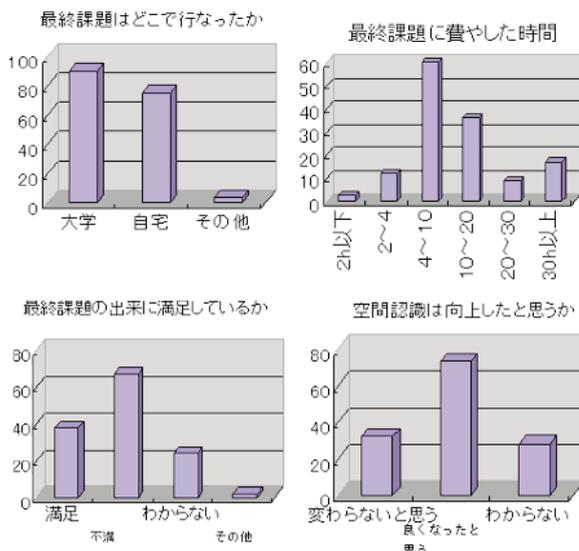


図11 アンケートの集計結果

いたためと考えられる。構想やスケッチなどを入れると、10日間前後の期間に、かなりの時間を費やしたものと思われる。以外なのは最終課題のでき映えに満足していない学生が多かったことである。教員側から見ると、8割程度の学生が満足してもよいと思われるでき映えであった。この理由は、3割程度の学生が非常に高度な内容のモデルを作っており、それらと比べて満足できない学生が多かったものと思われる。空間把握（認識）能力の向上については、半数以上の学生がはっきりと向上した（良くなったと思う）と答えており、主観的ではあるが、短期間の授業にもかかわらず教育効果があったものと考えられる。

5. おわりに

昨年度に提出していたシラバスでは4週をグラフィックスの実習に割り当てる計画であったが、Solid Interpreterの開発が予想よりも順調に進み、3次元モデルを簡単に作れるようになったため、発表会を最後に行う形態に変更し、期末試験を廃止した。その結果、学生は課題製作に集中でき、著者が予想していた以上の高度なプログラムを作ることができたものと考えられる。従来の提出課題ではコピーと思われるものが見られたが、Solid Interpreterを使った3次元モデル製作では、各自が非常にユニークな作品を作り、コピーと思われるものはまったくなかった。学生たちの自主性および独創性を養う効果もあったと自負している。そして、日ごろ学生たちの学力低下を嘆いていた著者も、彼らの能力と可能性を見直す結果となった。

反省点としては、教員がプログラムや教材の準備に追われ、ソフトウェアの使い方のTAへの説明が不十分で、実習時の個別指導で混乱することがあったことである。また、学生が比較的短期間に3Dプログラミングに慣れたのは、実行プログラムを配布したことも大きな要因であったと思われる。7割以上の学生が自宅や下宿でコンピュータを使える環境にあり、自宅学習ができた学生が多く学習効果があがったと考えられる。大学で開発されたという特徴を生かすためにも、今後、Solid Interpreterを使う教育機関では、授業を受ける学生に機能を限定した実行プログラムを無料配布できるようにしたい。

今後の開発については、他のCADソフトウェアとの連携を取るためのCAD用ファイル入出力機能の充実、インターネット上での標準3DフォーマットであるVRML形式のファイル入出力機能の追加、さらに、レンダリング機能やベジエ曲線による曲面表示、OpenGLへの対応などを計画している。

最後に、プログラムの配布と実行環境の整備、そしてSolid Interpreterの開発に協力してくださった、東京電機大学総合メディアセンターのスタッフの方々に感謝いたします。

参考文献

- [1] 山口富士夫：コンピュータディスプレイによる図形処理工学。日刊工業新聞社、1981。
- [2] 千代倉弘明：ソリッドモデリング。工業調査会、1985。
- [3] 鳥谷浩志・千代倉弘明：3次元CADの基礎と応用。共立出版、1991。
- [4] 遠山茂樹：ソリッドモデル入門。オーム社、1988。
- [5] 山岡祥：Turbo-Cによる3Dグラフィックス。森北出版、1997。
- [6] M. Mantyla: An Inversion Algorithm for Geometric Models, *Computer Graphics*, Vol.16 No.3, pp.51-59, 1982.
- [7] 磯田浩・鈴木賢次郎：図学入門（コンピュータグラフィックスの基礎）東京大学出版会、1986。