

教育用 3 次元ソリッドモデラーの開発

Development of 3D-Solid Modeler for Education

新津 靖
Yasushi NIITSU

東京電機大学 情報環境学部 (〒270-1382 千葉県印西市武西学園台 2-1200,
E-mail: yniitsu@sie.dendai.ac.jp)

ABSTRACT. An 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 37 kinds of primitive models, the solid model is constructed. The developed software has several kinds of display modes more than 140, such as the edge-type, the surface-type, the perspective and parallel views, three faces and so on. The executing speed of the model construction and the display on graphics is higher than the other CAD systems.

1. 背景

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。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ない。

一方、初等中等教育に目を向けて見ると、近年の IT 化が教育の分野にも普及しはじめ、文部科学省は 2005 年を目途に全国の小中学校にコンピュータを導入しようとしている。高等学校に関しては、さらに、コンピュータを援用した授業の展開や情報教育の推進を計画している。このような状況下でありながら、通常の教育用ソフトウェアは「シナリオ型」と呼ばれるイベント対応による知識の向上を目指すものがほとんどであり、「Programming 型」の創造性を養うようなアプリケーションソフトウェアは十分にあるとはいえない。特に初等中等教育用のソフトウェアは十分ではない。このままの状況を放置した場合には、コンピュータを導入しても十分に教育に生かされないケースが多くでてくるものと考えられる。この問題を解決するためにも、創造性を養うタイプのソフトウェアを開発する必要があり、また、このようなソフトウェアに対する需要は世界的に見ても、かなりあるものと期待できる。

2. 目的

そこで本開発では、使いやすく教育的にも配慮された図学教育やコンピュータグラフィックス教育などに応用できる 3 次元ソリッドモデラーを開発した。開発したソフトウェアは、図学教育に援用することを考慮し、3 次元 CAD に近い機能を付加した。すなわち、集合演算と陰線処理を充実させた。さらに現在、表面積計算や体積計算、重心計算などの CAD 的機能の充実を図るため開発を進めている。

開発したソフトウェアは「Solid Interpreter」と名づけ販売を開始している。本ソフトウェアが、ソリッドモデルをテキスト形式のコマンド行を順次実行して生成・表示すること、そして、通常のコンピュータ言語のように、変数や制御構造（繰返しや条件分岐）を記述できることから、このような名前を採用した。

3. 教育用 3 次元ソリッドモデラー

開発した教育用 3 次元ソリッドモデラー「Solid Interpreter」は、テキスト形式の独自のスクリプト言語を解釈してモデルの構築、演算、表示を行うソフトウェアである。このソフトは、基本立体と呼ばれる 13 種類（内部的には 37 種類）の立体に移動や回転、アフィン変換、集合演算^[3]などを施して希望とする 3 次元立体を構築することができる。ソフトウェアの開発で留意した点は、初心者でも楽しく使えることと、図学教育が目的とする「空間把握能力」を高める効果が得られることである。

3.1 3 次元ソリッドモデラーのデータ構造

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

立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。CGS 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]}

開発したソフトウェアは 3 次元立体の生成、表示、管理を行うもので、ソフトウェアが扱うデータには、名前を付けて管理するものと、名前を付けずにポイント操作で管理するものの 2 種類がある。名前を付けて管理するデータ構造としては、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^[2]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。このため、非常に多様な表示モードが指定可能であり、その数は 140 種類以上にもなる。表示モードは 4 つのパラメータで表現される。

グループ構造は立体や子グループをまとめて扱うためのものであり、コンピュータのファイル構造に似せて設計した。たとえば、2 つのグループを作り、その中に同じ名前の立体を作ることができる。あるグループを表示コマンドで表示する場合、グループ内部のすべての立体とすべての子グループさらに孫、ひ孫グループが表示される。

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。たとえば、飛行機のプロペラを考えて見ると、プロペラは飛行機に接続して回転するが、このような動作を軸構造により表現することができる。

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ている。^{[3],[4],[5]} 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の位相関係を表現する方法である。立体を構成する要素である「頂点」、「稜線」、「面」、「ループ」は名前を持たないデータ構造である。高速処理を実現するため、位相データの検索・操作はポイント操作により実現している。

3.2 3 次元立体の形状表現

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、ポイントのリスト構造により管理されている。それらの主なデータ構造を表 2 に示す。ここで、面プロパティと稜線プロパティは面と稜線が平面か曲面か、稜線が連続面上にあるか峰のような状態かなどの多くの情報を持っている。

コンピュータ内での立体の表現方法は図 1 に示すような構造体の双方向リスト構造を取っている。Body (立体) は最初の Face (面) へのポイントとすべての Face を結ぶ双方向リストで記述されており、Face は最初(外側)の Loop へのポイントとすべての Loop を結ぶ双方向リストを持っている。Loop は最初の Edge へのポイントを持っており、Edge (稜線) は左右の Face と Loop へのポイントおよび始点・終点の頂点へのポイントを持っている。^{[2],[4],[5]} これにより、1 つの Edge を 2 つの Face と Loop で共有することができる構造になっている。このように、Face が複数の Loop (親 Loop と複数の子 Loop) を持つことができ、この位相構造により穴の空いた面を表

現することができる。

3.3 プリミティブ(基本)立体の種類と形状

B-Reps 法による立体表現は、コンピュータにとっては処理しやすい方法であるが、人間が直接データを入力するのは不可能である。このため、図 2 に示すように、6 種類 (直線、多角柱、多角錐、2 種類の回転体、螺旋体) の立体生成コードを用いて生成できる 13 種類 (直線、立方体、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、2 種類の螺旋体) の基本立体を考え、12 の立体生成コマンドとパラメータ指定により簡単に基本立体が生成できる方法を取った。基本立体のうち、直線は面を持たない特殊な立体である。図 2 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。実際にはこれに加えて正 12 面体、正 20 面体、32 面体、80 面体など、合計で 37 種類の基本立体を図 2 の 13 種類のコマンドとオプション指定で生成することができる。

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、表示効果
グループ	グループ名、子グループリスト、子立体リスト、内包軸ポイント
軸	軸名、関連立体/グループ名、軸方向ベクトル、移動量、上下リミット
立体	立体名、面色、稜線色、親グループ、境界面・稜線・頂点 Class 等へのポイント、面数・稜線数・頂点数等

表 2 面・稜線・ループ・頂点のデータ構造 (Class)

データ構造	データ構造が有する主な情報
面構造体	稜線ループポイント、法線ベクトル、面色、面プロパティ
稜線構造体	始点・終点頂点ポイント、左右構成面、構成ループ、2 面角、稜線色、稜線プロパティ
稜線ループ構造体	ループ番号、親ループポイント、子ループポイント、稜線ポイント
頂点構造体	頂点番号、頂点座標、法線ベクトル、頂点プロパティ

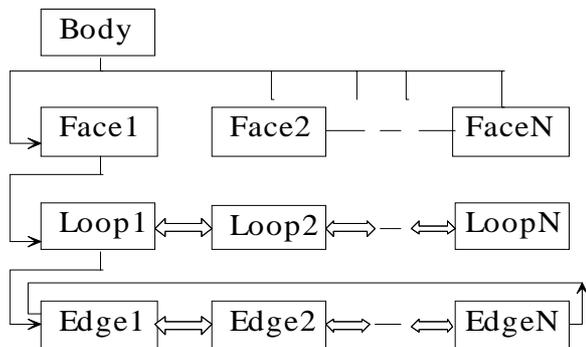


図 1 立体のデータ構造

3.4 表示モードと表示効果

「Solid Interpreter」の特徴はその表示機能にある。CADや図学教育を意識して開発されたため、陰線処理や3面図表示、リアルな面表示など140種類以上の多彩な表示方法を有している。図3は、トーラスを集合演算により組合せた立体の表示例であるが、陰線処理を施したものと、サーフェスモデルによる簡易陰線処理の違いが示されている。図4は3面図表示と陰線処理の例である。本ソフトウェアの開発においては、図学や設計の教育での応用を念頭に考えているため、陰線表示と3面図表示は充実している。表示モードと表示効果は、視線情報を生成するときに4つのパラメータで指定する。それぞれのパラメータの分類を下記に示す。

- (1) 表示効果 : 全稜線表示, 簡易陰線処理, 陰線処理, 面表示(3タイプ)
- (2) 表示モード: 透視投影, 平行投影, 3面図(2タイプ)
3面図と透視投影・平行投影の組合せ
- (3) 陰線表示 : 表示 ON・OFF 制御
- (4) 曲面上稜線: 表示 ON・OFF 制御

これらのパラメータの組み合わせにより原理的には192種類のパターンが設定可能であるが、面表示と陰線表示のように機能しないものもある。図5に種々の表示効果で円柱と円錐および角柱を表示した例を示す。

グラフィックス表示はWindowsのAPIを使用して実現しており、OpenGLやDirectXは使用していない。その理由はOpenGL等では陰線処理ができないためである。しかし、本ソフトウェアの表示エンジンでは、面表示をZソート法で行っており、正確な面表示や滑らかな曲面表示ができない。このため、将来的にはOpenGL等を利用した表示モードを追加する予定である。立体のデータ構造はOpenGLに十分対応できるように設計されており、技術的には面表示のOpenGL化は比較的容易である。図6に本ソフトウェアで作成した立体モデルのOpenGL表示例を示す。滑らかな面が表示できていることがわかる。

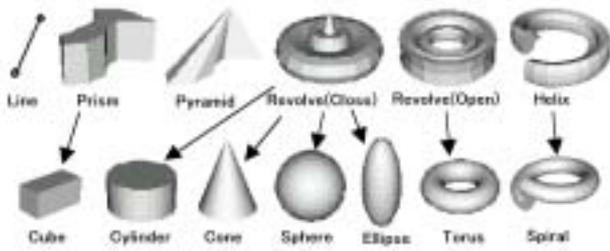
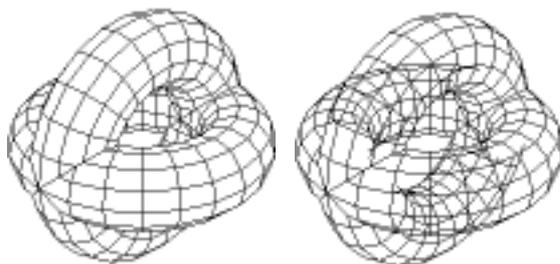


図2 基本立体の種類と形状



(a) 完全陰線処理 (b) 簡易陰線処理
図3 完全陰線処理と簡易陰線処理

4. コマンド群と3Dプログラミング

4.1 ソリッドインタープリタのコマンド群

ソリッドモデルの生成と表示はコマンドスクリプトを実行することで実現させることができる。コマンドは47種類あり、変数や繰返し処理、条件分岐処理など言語的な機能も備えている。CAD機能として、立体のグループ処理や立体の動作表現も可能である。表3~5に、47種類のコマンドの内、代表的な26種類のコマンドの機能を示す。コマンド群は大きく分けて以下の7種類に分類できる。

- (1) 立体生成コマンド : 基本立体の生成
- (2) 立体操作コマンド : コピー, 移動, 回転, 集合演算
- (3) 表示機能コマンド : 視線情報生成, 表示制御
- (4) 実行制御コマンド : 繰り返し・条件分岐, 変数・定数
- (5) ファイル操作コマンド : ファイル実行, DXF, VRML
- (6) グループ・軸コマンド : グループ化, 動作設定

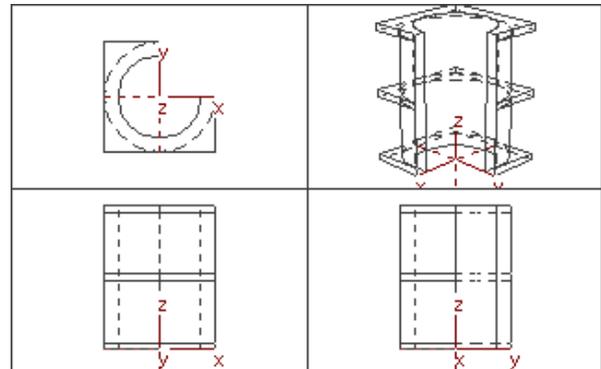


図4 3面図表示と陰線処理

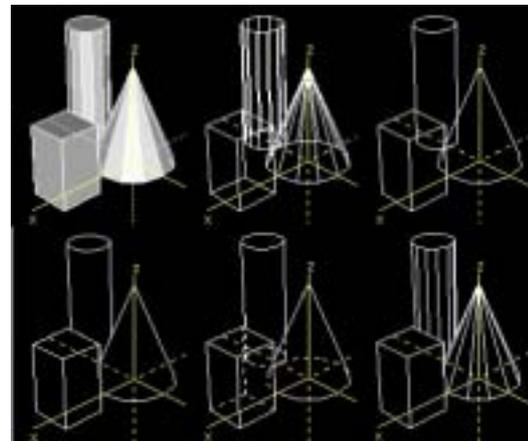


図5 陰線処理と立体の表示例

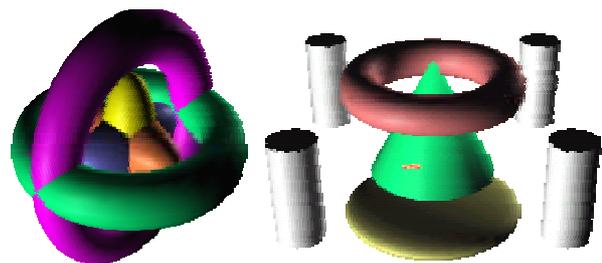


図6 OpenGLを用いた表示例(将来組み込む予定)

(7) その他のコマンド : 初期化,実行停止,コメント

表 3, 4, 5 にそれぞれ, 立体生成コマンド, 立体操作コマンドおよび表示機能コマンドの名前と書式を示す. これらのコマンドの機能の詳細と, その他のコマンドの機能と書式については CD-ROM 内のマニュアルを参照していただきたい. なお, ほとんどのコマンドはオプション指定によりその機能を変化させることができ, たとえば, 表 4 の Rotate コマンドは立体やグループを回転させるだけでなく, オプション指定により「視点」や「注視点」を回転させることもできるように設計されている.

表 3 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
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	立体名, 高さ, 座標点列
多角錐	Pyramid	立体名, 高さ, 座標点列
回転体	Revolve	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
螺旋体 1	Spiral	立体名, 大半径, 小半径, ピッチ
螺旋体 2	Helix	立体名, 座標点列, ピッチ

表 4 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
平行移動	Move	立体名, 相対変位
複数コピー	Ncopy	立体名, コピー方向, 数
回転移動	Rotate	立体名, 回転中心, 軸方向
カラー設定	Xcolor	立体名, 表示色(R.G.B.)

表 5 表示機能コマンド

機能	コマンド	書式または主なパラメータ
視線設定	Eye	視線名, 視点, 注視点, 拡大率, 表示モード(4Parameter)
画面消去	Clear	なし
座標軸表示	Dispcord	<op>座標軸長さ, 表示色
立体表示	Display	視線名, 立体/グループ
Window 設定	Window	表示枠指定
時間待ち	Mwait	待ち時間数(ms 単位)

4.2 3D プログラミング

簡単な例として, 以下に円柱と円柱の集合演算からできる立体のプログラム例 (13 行) と実行例 (図 7) を示す.

```
New # 初期化
Eye eye0 100,100,82, 0,0,0, 3, 1, 4, 0, 0 # 視線生成
Cylinder y0 32, 10, 30 # 円柱 y0 の生成
Move y0 0, 0, -15 # 円柱の移動
y1 = y0 # 円柱のコピー
Rotate y1 0,0,0, 1,0,0, 90 # 円柱 y1 の回転
Xcolor y0 255, 255, 255 # 着色 (白)
Xcolor y1 255, 128, 0 # 着色 (青)
```

```
XObj1 = y0 + y1 # 集合演算(和)
XObj2 = y0 - y1; Move XObj2 -25,-25,0 # 集合演算(差)
XObj3 = y0 * y1; Move XObj3 -25, 30,0 # 集合演算(積)
Display eye0 XObj1,XObj2,XObj3 # 表示
```

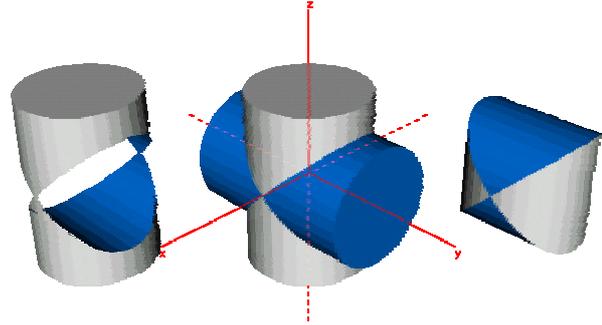


図 7 集合演算の例 (和, 差, 積集合)

Solid Interpreter のコマンドには, 繰り返しや条件分岐などの実行制御をコントロールすることができるコマンドが用意されている. さらに, 変数が使用できるため, プログラミングにより立体のパラメータが変更できる「拡張パラメトリック表現」が可能である. たとえば, 次の 3D プログラムのように, 変数により, 羽根の枚数を指定してプロペラを生成することができる. 通常のパラメトリック表記は, 立体のサイズを数値として指定するものであり, 基本的な形態を変化させるものではない. 次の 3D プログラムの実行により生成される立体像を図 8 に示す.

```
Eye eye0 100, 100, 82, 0, 0, 0, 3, 1, 4, 0, 0
N = 5 # Variable N is the number of wings
R = 360 / N # Variable R is the angle of wings
Group Propeller {
  Ellipse e0 12, 8, 20
  Cube q0 30, 20, 20
  Move q0 -15.00, -10.00, -24.0
  e0 = e0 - q0
  delete q0
  Ellipse e1 12, 5, 25
  Affine e1 0, 0, 0; 0.20, 0.00, 0.00;
  0.00, 1.00, 0.00; 0.00, 0.00, 1.00
  Move e1 0 0 20
  Rotate e1 0 0 0, 0 1 0 90
  Rotate e1 0 0 0, 1 0 0 -15
  Ncopy -s e1 0, 0, 0, 0, 0, 1, R, 0, N
  delete e1
}
clear; display eye0 Propeller
Dispcord -a eye0 20, 0x80ffff
```

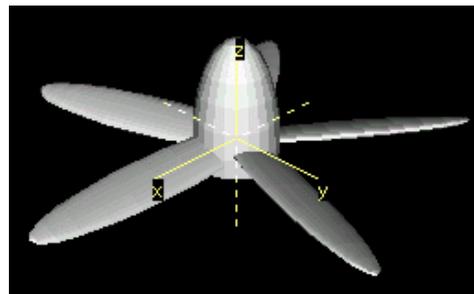


図 8 拡張パラメトリック表記によるプロペラの生成

5. 3DプログラミングのためのGUI環境

開発したソフトウェア「Solid Interpreter」は、立体生成やコマンド生成をサポートする次の5種類のツールを備えている。

- (1) **メイン Window :**
3Dプログラミング用のマルチテキストエディタを持ち、実行制御、ファイル管理を行う。
- (2) **Primitive 生成 Window :**
基本立体生成コマンドをGUI環境で自動的に生成するツールで、他のツールを呼び出す機能も持つ。
- (3) **Command 生成 Window :**
基本立体生成以外のコマンドを生成するためのツールで、色設定などビジュアルに指定することができる。
- (4) **状態表示 Window(Object Inspector) :**
実行時のグループ関係を Tree 表示や、メモリの使用状況を調べたりするためのツールである。
- (5) **グラフィック Window :**
立体を表示するだけでなく、ビットマップ画像の保存やクリップボードへの転送機能を持っており、表示画像の再利用を支援する機能を有する。(この報告書のグラフィックス画像はこの機能を利用してワープに貼り付けている。)

これら以外にも、初期設定を行うためのフォームや数値パラメータを入力するためのフォームなど多くのプログラム機能を有している。

これらのツールを使用することで、GUI環境下でビジュアルに立体を生成したり、コマンドを生成することができる。図9に基本立体を生成するためのツールである「プリミティブ生成 Window」(任意対称断面の螺旋体生成時の表示例)を示す。このツールにより37種類の基本立体を確認しながら生成ことができ、さらに、パラメータが多い視線情報も確認しながら設定することができる。

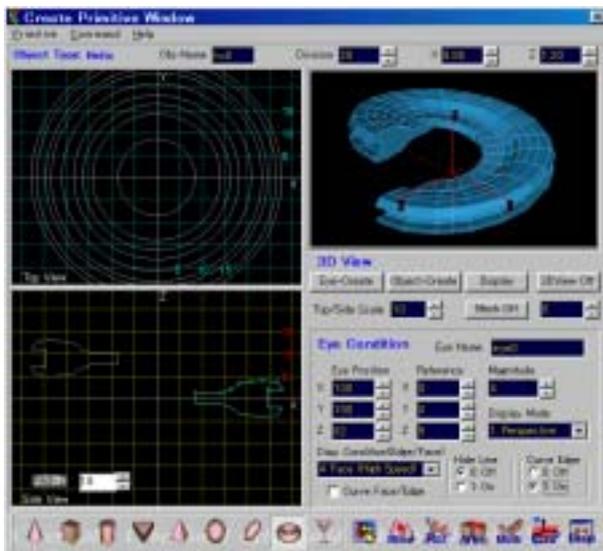


図9 プリミティブ生成 Window

マウスイベント操作とタイマースレッドを利用した高度なソフトウェアである。このツールだけでもC++のソースは280kbytes以上にもなる。

図10に生成した立体の構造を調べるための「Object Inspector」(グループ・軸・立体構造 Tree 表示例)を示す。このツールは表1の名前付きオブジェクトである「視線情報」、「グループ」、「軸要素」および「立体」の情報を調べるためのものである。特にグループ・軸と立体の主従関係は、立体を動作させる上で重要な情報である。これを図10のようにTree表示することで、3Dプログラムのデバックを容易にできる。他のタグを選択すると、個々のオブジェクトの情報を見ることができる。

6. 立体モデルの製作例

図11に、大学生たちが製作したソリッドモデルの例を示す。これらのモデルは動作するようにプログラミング

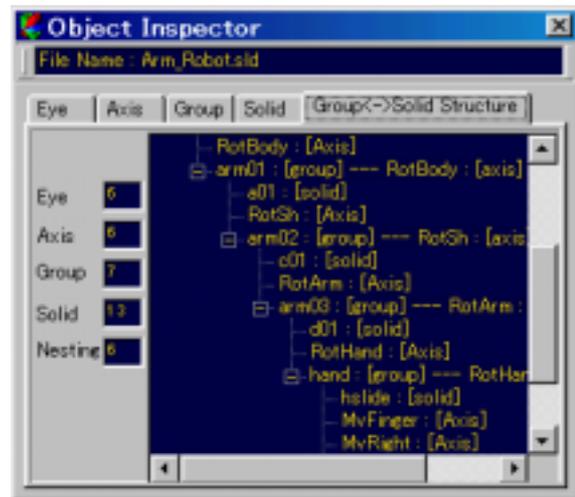


図10 オブジェクトインスペクター

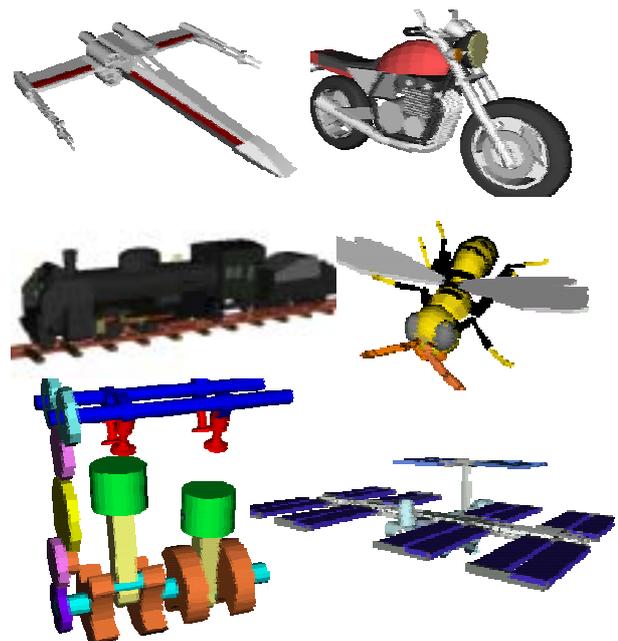


図11 3次元立体の製作例

されている。このようなモデルを週2時限，4～5週で作れるようになることがわかった。東京電機大学の機械工学科では1999年前期より1年次生のコンピュータ教育で開発したソフトウェア(Version2.0)を使用しており教育効果を上げている。また，拓殖大学工学部，山梨大学工学部でも授業支援に使用しており教育効果を上げている。

7. 結言

2000年度の「未踏ソフトウェア創造事業」により，モデリングおよび表示エンジンの強化とGUI環境の刷新が実現できた。これにより，高校や高専，大学で使用してもらえる環境が整ったと認識している。さらに，将来の機能拡張や応用分野の足がかりになるソフトおよびハードの開発ができた。

本ソフトウェア「Solid Interpreter」は，2000年度「未踏ソフトウェア創造事業」終了時点でVersion3.2が完成し，現在は，Windows-Meに対応したVersion3.3が最新版として完成している。また，Macintosh用にはVersion3.2がほぼ完成している。

一方，Solid Interpreterから制御できる入出力用ハードウェアの開発も同時に進めてきたが，このハードウェアについては，まだ完成しておらず，今後完成を目指して開発を進める予定である。

音楽機能については調査研究を進めており，ハードウェアと同様に，Solid Interpreterへの組込む作業を進めていく予定である。

今後は，(1): 作図機能，寸法線表示，曲線表示機能を充実させた簡易3次元CADの開発，(2): STEP,IGES等のファイル入出力機能の強化，(3): GUI環境を整備した小中学生でも直感的に使用できるソリッドモデラーの開

発。(4): 数式表示機能・数式処理機能の取り込み^[5]，(5): 音楽機能の取り込み，(6): 外部入出力装置と入出力機能の開発，(7): 3次元立体表示装置との連携などを進める予定であり，将来的には教育分野だけでなく，設計や計測，エンターテインメントなどより多くの分野で利用できるソフトウェアに発展させていきたい。なお，開発されたソフトウェアは <http://www.cyber-solid.com/> または <http://www.n-plus.co.jp/> からダウンロードすることができる。多くの教育機関で是非教育に利用していただきたい。

8. 参加企業および機関

- (1) 東京電機大学工学部，情報環境学部
- (2) ネプラス株式会社（サポート組織）

9. 参考文献

- [1] 山口富士夫：コンピュータディスプレイのよる図形処理工学，日刊工業新聞社，pp.6-45 (1981)
- [2] 千代倉弘明：ソリッドモデリング，工業調査会，pp.12-22 (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，pp.16-142 (1991)
- [4] 新津 靖：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，私立大学情報教育協会論文集，Vol.2, No.1, pp.7-12 (1999)
- [5] Yasushi Niitsu: "Solid Modeling with Polygon Data of Mathematica", Proc. of 4th Int. Mathematica Symp., pp.369-374 (2001).