

「空間図形における3次元ソフトの利用について」

愛媛県立松山西中等教育学校 高田 修和

1 はじめに

3次元空間における図形感覚は、かなり難しいものがある。現在は小学校・中学校で、「空間図形の展開と切断」、「空間図形における直線・平面の位置関係」、「空間図形の表面積・体積」を学習している。これらの内容は生徒にとっては興味深く学習できる反面、理解できにくい生徒が多い傾向にある。

高校では数学Bの「空間ベクトル」において、空間的なイメージが必要になる。上記の理由等により、高校においても、3次元空間の基本問題を理解することが難しい生徒が多い。そこで、空間図形の学習において、視覚的により効果のあるソフトについて研究することにした。

2 研究概要

(1) Mathematica

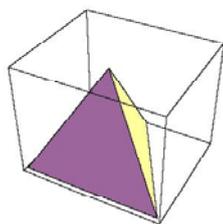
Mathematicaとは、Wolfram Reserchが提供する数学に関するソフトで、誰しもがその価値を認める。特に、3次元グラフィックスではその機能をフルに発揮する。

(ア) 作成例（正多面体の描画）

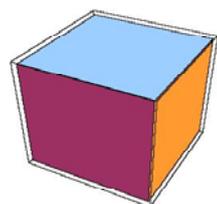
正多面体を描画するにはパッケージGraphics`Polyhedra`に含まれる関数を用いる。

```
Needs["Graphics`Polyhedra`"];  
• Polyhedron[ 多面体 ]  
• Polyhedron[ 多面体 , { x , y , z } , r ]  
  ←r倍に拡大・縮小, 中心{x, y, z}
```

ア 正六面体 (Show[Polyhedron[Tetrahedron]])



イ 正六面体 (Show[Polyhedron[Cube]])



(2) VRML

VRML(Virtual Reality Modeling Language)とは、HTMLの3次元版であり、インターネット上でサイバースペースを構築するためのシーン記述言語である。

VRMLのイメージは、まず一つの風景(シーン)があり、その中に幾つものオブジェクト(ノード)が配置される。

他のプログラミング言語のようにコンパイルされ、実行されるわけではない。VRMLファイルはコンピュータに理解可能なオブジェクトの集合(パーツ)に変換され画面上に表示される。コンピュータグラフィックスに向けており、豊富なモデルを容易に生成することが可能で、組み込み機能を多く備えている。何よりテキストベースで作成し、読みとり可能なブラウザがあれば、すぐに表示できるところが一番の利点である。

(ア) 作成例（基本幾何図形の作成）

VRMLで基本幾何図形を形成するときはShapeノードを用いる。

```
Shape {  
  geometry NULL      ← 外観ノードの指定  
  appearance NULL   ← 幾何ノードの指定  
}  
  
ア 直方体の描画 Box  
geometry Box {size 2 2 2} ← 幅、高さ、奥行き
```

(イ) 作成例（点、線、面の作成）

任意の点集合や折れ線、多角形、多面体を作成するときはCoordinateノードを用いて、まず点集合を定義する必要がある。点座標の定義はpointフィールドを用いる。

```
coord Coordinate {  
  point [           ← 点座標の指定  
    x0 y0 z0       ← 頂点番号0  
    x1 y1 z1       ← 頂点番号1  
    .....  
  ]  
}
```

```
ア 任意の面の描画 IndexedFaceSet  
geometry IndexedFaceSet {  
  coord NULL      ← 点集合
```

```

color NULL          ← 色リスト
coordIndex []      ← 描画点の指定
colorIndex []      ← 面の割り当て色順
colorPerVertex TRUE ← 点・面色割り当て
}

```

(3) Live Graphics3D

Live Graphics3Dとは、Mathematicaで作成した3D図形を、Web上で表示するためのJava Appletである。Mathematicaは静的なものであるが、このソフトは動的なものであり、より3次元の世界を身近なものに導いてくれる。

このclassを用いることで、Java Appletの知識を持たなくても、引数を指定するだけでAppletを実行させることができる。VRMLよりもLive Graphics3Dの方が操作が簡単な上、光立体図形にあたる光の強さや方向などの指定をしなくてもMathematicaの図形を忠実に実現することができる。

Webページ上でLive Graphic3Dを使用するには、次の5つのApplet Classをインストールすることが必要である。これらのclassを使用するHTMLファイルと同じフォルダに置いて使用する。

- 1 Live.class
- 2 Graphics3D.class
- 3 Parser.class
- 4 Primitive3D.class
- 5 Quaternion.class

(ア) 作成例 (直方体)

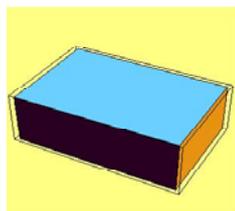
LiveGraphics3DのAppletを用いてHTMLファイルに埋め込むためには、Appletタグを用いる。

```

<APPLET CODE="Live.class" WIDTH=200 HEIGHT=200>
<PARAM NAME=BGCOLOR VALUE=#FFFF80>
<PARAM NAME=INPUT VALUE="
Graphics3D[{
  Polygon[{{0,0,0},{0,0,1},{0,2,1},{0,2,0}}],
  Polygon[{{3,0,0},{0,0,0},{0,2,0},{3,2,0}}],
  Polygon[{{3,0,0},{3,0,1},{0,0,1},{0,0,0}}],
  Polygon[{{3,0,0},{3,0,1},{3,2,1},{3,2,0}}],
  Polygon[{{3,0,1},{0,0,1},{0,2,1},{3,2,1}}],
  Polygon[{{3,2,0},{3,2,1},{0,2,1},{0,2,0}}]
}]
">
</APPLET>

```

これで作成されたHTMLファイルをブラウザで見ると次の様になる。



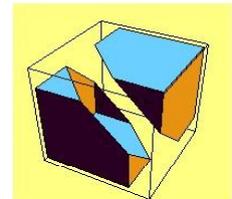
(イ) 作成例 (直方体の切断面)

Mathematicaで作成し、LiveForm関数で変換する。任意の図形の切断面を作成する関数knife[]を用いる。

```

Get["quadrics`"]
LiveForm [g_] := NumberForm[InputForm[N[g]], 4];
cube=Graphics3D[Cube[]];
gr=knife[cube, {1,1,1,0}, {0,0,0}, {.3,.3,.3}];
Show[gr]
LiveForm [gr]

```



これをLiveGraphics3DのAppletタグにおけるINPUTパラメータのVALUEに用いると次の様になる。

(ウ) 特徴

- ・回転、ズームイン・アウト、焦点距離の変更、部品除去、ステレオ画像作成が可能であり、Web上ではマウスを操作するだけである。
- ・自分の作った面が間違っている場合に、どこが間違っているかの確認には部品除去が有効である。また、平面間の位置関係などもイメージできる。

(4) POV-Ray

POV-Ray (Persistence of Vision Ray Tracer) とは、リアルな3次元グラフィックスを作り出すためのソフトである。3次元図形をレイトレーシング法によって、光の影や透明感などを演出する。

標準インクルードファイルで、リアル感を持たせるための様々な素材が数多く用意されており、シーンファイルの記述が簡単に作成することができる。

(ア) 作成例 (インクルードファイル)

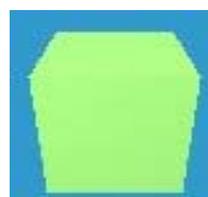
これを拡大・縮小、回転、平行移動して配置していく。

```

#include "shapes2.inc" ← 形状に関するインクルードファイル
... ← カメラ、背景等の設定
object { ← オブジェクトの設定
  形状の名前
  texture { ... } ← テクスチャ指定
}

```

UnitBox (Cube)



Pyramid



(5) MathGL3D

MathGL3D (OpenGL Viewer for Mathematica) とは、Mathematica で作成した 3 次元図形をインタラクティブに再現する Viewer を作る Mathematica のパッケージである。また、このパッケージを用いることにより、Mathematica で作成された画像を様々なタイプの画像 (POV-Ray、VRML) に変換可能である。

(ア) Mathematica への画像の貼り付け

GL Viewer の画像をもう一度 Mathematica へ貼り付ける事も可能であり、そのためには MVPasteGraphics を用いる。

(イ) 作成例 (立体の切断)

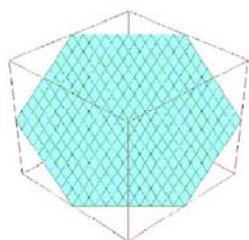
```
Get["Graphics`Polyhedra`"]
Get["Graphics`Shapes`"]
Get["knife`"]
cube=Graphics3D[Cube[]];
gr=knife[cube, {1, 1, 1, 0}, {0, 0, 0}, {1.3, .3, .3}];
MVClear [];
MVShow3D [ gr]
```



(6) Function View

Function View とは、すべてマウスの操作により、図形の表示が可能でソフトである。点のドラッグに対応、入力・訂正パネルによる簡単なインターフェースを実現している。多数の補助プログラムも準備されている。また、3D では陽関数、図形、媒介変数曲線も表示でき、様々な空間の図形の表示が可能である。画像の連続保存により、アニメーションの作成も支援している。背景に写真等を張り付けて、グラフと重ねて表示もできる。Basic 準拠のマクロ機能搭載により自動実行が可能である。

(ア) 作成例 (立方体の断面)



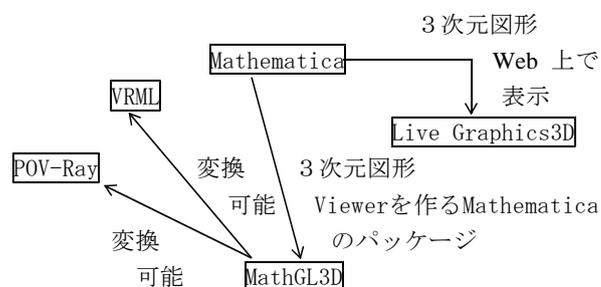
3 今後の課題

Mathematica、VRML、POV-Ray、MathGL3D は静的なものであり、LiveGraphics3D と Function View は動的なものである。実際に、回転や拡大・縮小などを実感させる場合には後者を、3次元グラフィックスの鮮やかさを実感させる場合には前者が向いているようである。いずれにせよ、視覚的な効果もさることながら、空間図形に興味・関心を持つことが一番である。

3次元ソフトの使用においては、3次元感覚とグラフィックスのどちらがメインになるかに注意し、最終的には、3次元感覚とならなければならない。

参考

1 関連図



2 リンク先一覧

(1) Mathematica

「3DGraphics In Mathematica」(M.Sanaee)

http://www.nikonet.or.jp/spring/sanae/3D_Math/3D_Math.htm

(2) VRML

「VRML2.0 TUTORIAL」(M.Sanaee)

<http://www.nikonet.or.jp/spring/sanae/VRML2/Vrml2.htm>

(3) Live Graphics3D

「LiveGraphics3D Homepage」(Martin Kraus)

<http://theorie3.physik.uni-erlangen.de/~mkraus/Live.html>

(4) POV-Ray

「POV-Ray Homepage」(Martin Kraus)

<http://www.povray.org/>

(5) MathGL3D

「OpenGL Viewer for Mathematica」(Jene-Peer Kuska)

<http://www.mpae.gwdg.de/~kuska/mview3d.html>

(6) Function View

「fv555a.exe」

<http://hp.vector.co.jp/authors/VA017172/>