

Nara Women's University

Personal LINKSによるCG利用法

メタデータ	言語: Japanese 出版者: 奈良女子大学情報処理センター 公開日: 2014-09-19 キーワード (Ja): キーワード (En): 作成者: 今岡,春樹 メールアドレス: 所属:
URL	http://hdl.handle.net/10935/3800

Personal LINKS による CG 利用法

家政学部 助教授 今 岡 春 樹

1. はじめに

情報処理センターにCG（コンピュータ・グラフィックス）用のソフトウェアが導入された。CGは、数値データを目でみることのできる図形や画像で表現するためのコンピュータ技術の総称である。表現技術として考えると、図形処理に特化したものとしてCADがあり、画像処理に特化したものとしてペイントシステムがある。ドロー系とペイント系として区別されている。導入されたCGシステムは3次元のシステムで、3次元上の物体を表現する目的で用いられる。3次元CGには様々な種類のソフトウェアが提供されているが、通常、物体を記述するモデラーと物体を描画するレンダラーとが2本柱となっている。本システム“Personal LINKS”もその一つであり、他のシステムと共通な部分とユニークな部分がある。

ここでは、Personal LINKSの概略を紹介することにする。

2. ハードウェア

ソフトウェアはSUN4/2 GX (nara3) にインストールされている。周辺機器として、モニター(PVM2044Q)とフレームバッファ(PFB-2)が画像出力のために接続されている。画像の入力装置としてカラーキャナー(UP-5100)があり、画像の記録用にビデオプリンター(JX-220)が用意されている。

ソフトウェアの使用については、nara3のSUN4にログインし、オープンウィンドウを開き、コマンドツールの中でコマンドを入力することで実行できる。作成したファイルは自分のディレクトリにセーブすることになる。

3. 物体が目で見えること

ソフトウェアの説明に先立って、3次元CGの基本的な考え方を述べる。ここで野球のボール（以下ボールとする）をディスプレイに表現することを考える。ボールの写真を撮ってきてキャナーで読み込み、そのまま表現する方法が考えられる。しかしこの方法は3次元CGの方法ではない。まず3次元というときには、遠近も含めてあらゆる方向から見るのが可能でなければならない。すべての方向からの写真を用意することは不可能なので、上記の方法ではなく、ボールを3次元の数値データとして与えることを考える。現実世界にボールが存在するように、CGの世界でボールを存在させる。次に現実世界での目に相当する視点を与える。視点を変化させることであらゆる方向から見る事が出来るよう

になる。最近の用語で人工（仮想）現実感というのがあるが、これはCGの世界に現実の人間が参加することを意味している。ここでいうCGの世界は、現実を模擬したもので、コンピュータの世界であるので、数値データを用いて現実を模擬しようとしていることになる。物体が見えるということを経験で模擬するのがCGの技術であると言い替えてもよい。現実の世界では物体はすでに存在しているし、それが見えるということは常識以前のことである。そのからくりを分析し、法則を発見すれば、統一的に模擬が行える。幸い、形については幾何学によって数量化できるし、色については色彩学によって数量化できる。また、見えるための媒介となる光の性質は光学を用いることが出来る。これらの基礎知識とコンピュータ技術との連携によってCGは成り立っている。

実は以上述べた知識だけではボールをボールらしく表現することはできない。野球のボールはサッカーのボールとは異なっている。表面の模様が異なっているのだが、色だけではない微妙な質感（テクスチャ）が異なっている。光の反射特性や微妙な表面形状の差異がテクスチャを決定している。さらに光源の問題もある。

ともかく、野球のボールを表現するためには、まず球状の形を決め、表面の色とテクスチャを与え、光源の位置と性質、視点の位置を決めることによってやっとそれらしい画像を生成できる。これらの作業をいちいち汎用プログラムで実行するのは大変なので、このような作業を支援するソフトウェアが活用されている。ただし、現実世界に存在するすべての物体がCGの技術で表現できるかというところではない。したがって多くの研究者が今まで表現できなかった物体を表現すべく努力を続けている。ガラスのような透明物体、雲や水、木や山、人体、衣服、髪の毛など多くの課題が残されている。

4. ソフトウェア

3次元CG作成の標準的手順は、1)個々の物体の形を作成する。2)個々の物体の配置や構造（肩の先に腕があり、その先に手があるなど）を決める。3)個々の物体の質感を決定する。4)光源や視点（このソフトではカメラという）を決定する。5)どの物体にどの物体が反射したり影を落としたりするかの関係を与える。6)画像を生成する。となっている。

まずモデラーとしてL/Shapeがある（実行時のファイル名はlshape）。このソフトウェアを用いて様々な物体の形状を作成することが出来る。使用頻度の高い表面だけのサーフェースモデルが用いられ、ポリゴン（多角形）によって表面を記述する。汎用的な3次元CADとしても利用できる。

L/Manageはシステム全体を統括するデータマネージャで、上記の2)から5)までの部分を担当する（実行時のファイル名はannie）。また6)を実行する場合にもannie上で指示する。

2)の部分を担当するのがannieのサブシステムであるvtreeと呼ばれるソフトウェアである。個々の物体の配置や構造を木による階層構造で表現する。木構造はパスとクラスタの2つの要素から構成されている。クラスタはモノすなわち存在する物体を定義する。パスはコトすなわち物体間の関係や質感や光源などを定義する。階層構造とは、クラスタでいえば上位の物体名は下位の物体の集合全体を意味し、パスでいえば下位にあるすべてを制御する。ただし、質感（アトリビュート）のように合成できない性質については直近上位の質感が与えられ、位置関係のように合成できる性質については根元（ルート）まで合成され、絶対位置が得られる。パスとクラスタで定義される木構造は、単なる入れ物で実態を与えるためには、モデラーで作成した形状をクラスタに、質感をパスにリンクする必要がある。

3)から5)までの部分が、このようにデータをリンクするという操作で実行される。質感の制御は

複雑であるので特別に Surface Editor と呼ばれる質感編集ソフトが用意されている。さらに質感の事例集である Material Library も用意されている。視点や表示装置の制御は対象物体とは異なった位置付けがなされている。実際にレンダラーを動かす時には、これらの具体的な値を指定する必要がある。この例のような定型的な指定に対しては、テンプレートエディタが用意されている。

6) の部分が具体的なレンダリング部分で、L/Image (実行名は limage) を利用する。この他、質感の元絵を作成する 2 次元ペイントシステム Neozoic Paint や、アニメーション作成支援ツール Animation Tool が利用できる。アニメーションについては、先ほどの木構造が威力を発揮する。例えば腕を肩の部分で動かせば自動的に腕の先の手も連動する。このようにしてある程度ラフなポーズを時系列的に指定すると、中間のポーズを自動的に生成するのが基本となっている。

5. 基本操作

ここでは、必要最低限の操作法について説明する。

5.1 使用前に

まず SUN の nara3 にログインし、オープンウィンドウを開き、コマンドツールをセレクトする。

5.2 形状の作成

nara3% というプロンプトの後に、lshape& と入力する。& はバックグラウンドジョブを意味する。L/Shape が起動され、以下は L/Shape のメニューを使って形状を生成する。オープンするのに少々時間がかかるが、一度アイコンにしておけば以後はすぐオープンできる。作成したデータはファイル名に .ppd を付けるが、ppd ファイルのセーブは FILE Ppd-Save メニューを利用する。L/Shape から出るには FILE EXIT をメニューからセレクトする。

5.3 木構造の作成

nara3% というプロンプトの後に、annie と入力する。L/Manage が起動され、annie-(1) というプロンプトが表示される。L/Manage から出るには、exit と入力するか CNTL+c を使うかのどちらでもよい。annie の中で、vtree と入力すると木構造を作成するためのエディタが起動される。木構造の要素であるパスとクラスタを作成するのであるが、特殊なキコマンドを用いる。自動的にパスの root とクラスタの [root] が生成される。編集対象はカーソルで示される。編集対象のパス・クラスタ間での移動はスペースキーで行う。根元パス root に対して a と入力すると子供を作る準備ができる。o と入力すると兄弟を作成する準備ができる。準備ができたところで、パス名クラスタ名を入力し RETURN を 2 回押す。なお vtree から出るには CNTL+c を使う。

木構造ができ上がれば、次に様々な具体例をリンクすることになる。形状データのリンクは 2 段階の手続きをとる。まず、クラスタを対象とし、ESCshift+l と入力し、リンカを起動させる。clus/クラスタ名が表示されるので、それに対して o と入力し、shape/クラスタ名を入力する。ここで形状データをリンクする入れ物 shape/クラスタ名が出来上がる。これに対して ee (テンプレートエディタの起動) と入力し、name の欄に i (インサート) し形状データファイル名を入力する。最後に CNTL+c で確定しながら抜ける。キャンセルしたいときは ESC+c で抜ける。

5.4 質感の設定

木構造のパスを対象とし、es と入力しサーフェスエディタを起動する。サーフェスエディタはす

べてメニュー方式であり、抜けるにはメニューの Exit Accept (終了) か Suspend Accept (一時終了) をセレクトする。質感ライブラリを利用するには Edit Get sur Library と順にセレクトする。

5.5 レンダリングの準備と実行

annie のレベルで、まずレンダリングの準備を行う。デバイス、スイッチ、カメラ、ライトの設定を行う。それぞれ de de、sw sw、ca ca、li li でテンプレートエディタに入り、CNTL+c で確定しながら抜ける。

レンダラの起動は run limage で行う。実行は view で、中止は rquit、プロセス中止は rkill というコマンドを入力する。なお、レンダリングの方式はレイトレーシング法によっている。

以上の操作によって形成されたファイルは、形状データファイルが .ppd であったように .dm を付けた dm ファイルとしてセーブしたりロードしたりできる。save all dm ファイル名によってセーブし、load dm ファイル名でロードする。

6. 特殊な形状データとその応用

形状データは L/Shape を使用して作成するのが標準的である。これはサーフェースモデルによるものであるが、半ソリッドモデルとでも言うべきメタボール (濃度球) も用意されている。メタボールは 3次元空間に濃度を定義するもので、中心が濃く、中心から離れると薄くなり、ある程度離れると濃度がなくなる。2つのメタボールを定義すると、空間の濃度はその和で定義される。このようなもやもやした物体を視覚化するために、ある濃度以上の点は存在すると考える。実際はこのある濃度の点が表面となるので、その表面を描画することになる。レンダリングの方式がレイトレーシング法であるので、この描画が可能となっている。なお、メタボールを利用するには、L/Shape ではなく、L/Manage の中で定義する。

ここで、図1、図2に人体のモデリングの例を示す。図2はテクスチャマッピングを行ったものである。およそ 100 個のメタボールを使用している。

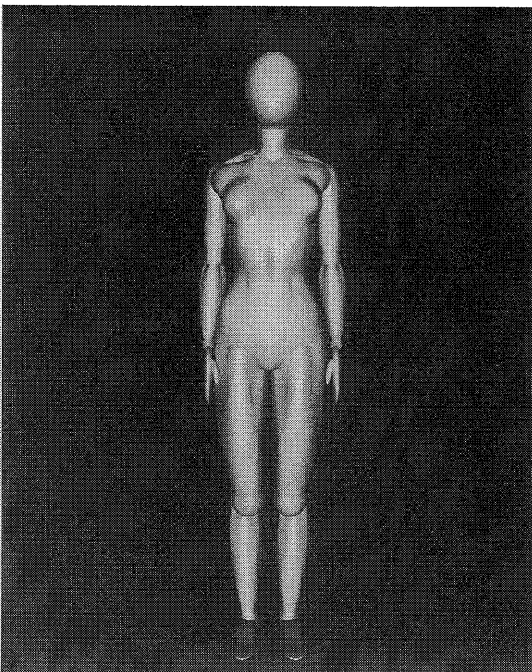


図 1

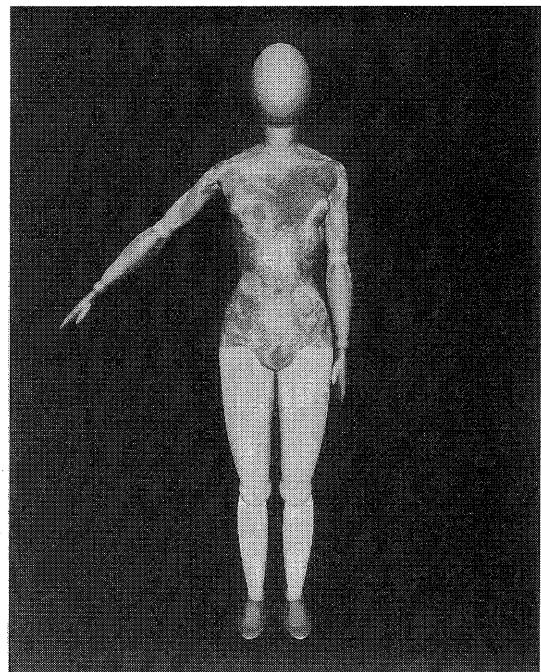


図 2

7. データ形式の公開

物体の形状をモデラを使って手作りで作成する場合もあるが、何か他の計算機を利用して形状を生成する場合もある。このようにレンダラだけを利用したいという要求がある場合、ppd ファイルのデータ形式が公開されている必要がある。本ソフトウェアでは、このデータ形式が公開され、しかも unix マシンで標準的なアスキーファイルであるので、コンバータの作成が容易である。

8. データと構造とリンク

以上のまとめとして、Personal LINKS での物体表現法をデータ情報の立場から見ることにする。本ソフトウェアで管理するデータは、基本的には dm ファイルの中に収められている。ppd ファイルは重要であるがむしろ例外的なものと考えた方が分かりやすい。マッピング素材ファイルである pic ファイルも同様に例外と考える。

管理されている dm ファイルの個々のデータにはその種類が決まっている。この種類をクラスと呼ぶ。クラスごとに必要なデータが定められている。必要不可欠のデータは定型的であるので編集しやすいようにテンプレートエディタが用意されている。クラスは表1のようになっている。

構造	path cluster
形状	quad meta triangle beam ring shape*
質感	attribute
マッピング	mapp mapv texture* volume*
カメラ	camera
ライティング	light
反射・屈折・影の制御	rgroup tgroup sgroup
表示装置・サイズ	device
レンダリングパラメータ	parameter switch
アニメーション	channel track record
レンダラー	renderer

表1 クラス名 *印は実体である下位のファイル名を記述するもの

個々のクラスには特有の意味があり、データ名を指示するときには、クラス名/データ名が正式な名前となる。これらの具体的なデータのことをインスタンスという。各インスタンスは単なる集合として管理されてはいない。木構造とリンクという構造をもつ集合として管理されている。

木構造はパスとクラスタの2種類のクラスのインスタンスから構成される。集合の部分集合の一部を木という構造で明示することで操作性を向上させている。例えば、パスには移動と回転のパラメータを必要とするが、このパラメータを変化させるとそのパスにより指示される部分集合の全てが同じ動きをする。

リンクは通常名前に実体を与えることを意味する。例えば、クラスタのインスタンスには shape や meta などのインスタンスがリンクされる。meta のインスタンスは中心の位置や半径などのパラメータを持つ最下層部のデータでプリミティブと呼ばれている。一方 shape のインスタンスはさらに下層

の ppd ファイル名を指示する役割をもつ。shape や meta のインスタンスはパスのインスタンスにリンクされることはなく、リンクできるクラスの組み合わせは決まっている。

テクスチャマッピングや反射・影などの詳細は省略するが、そのようなクラスが用意されているので、インスタンスを作成し、リンクするという手順に従えばよい。

9. おわりに

CG のソフトウェア Personal LINKS について、その概要を説明してきた。その内容は相当複雑であり、操作に慣れるのも大変である。CG の基礎知識がないと、マニュアルのどこを調べていいかわからない。初心者は例題を数回実行して見ることをお勧めする。最後にマニュアルのリストを記載しておく。

L/Shape Manual 1-3

L/Shape Video Manual

L/Shape Surface Editor

Material library

L/Manage Annie Manual 1-2

L/Manage Data Manual 1-2

L/Manage Animation Editor

Utility Manual

Neozoic Paint