

実践報告

3次元コンピュータグラフィックス実習の取組み

東京都立つばさ総合高等学校 主幹教諭
横枕 雄一郎

1. はじめに

コンピュータの計算能力が上がってきたことで、最先端の3次元コンピュータグラフィックス実習ができるようになってきた。ディズニー・ピクサー映画で使われている技術を高校生でも理解できるように、授業実践を行っている。その取組みについて紹介する。様々なオープンソースソフトウェア、非商用利用ソフトウェア(64bit)を使うことで、どこでも実習として利用可能となる。

総合学科である本校情報科では、1年次に必修科目「社会と情報」(2単位)を設置している他に、選択科目として学校設定科目「情報演習」(各2単位)を設けている。文書処理演習、画像編集処理演習、ネットワーク演習、プログラミング演習、表計算演習、データベース、3DCG・動画演習と様々な課題に対応し解決していく力を身に付けて行けるように7つの実習が選択できる。

夏期講習では「フリーソフトウェア Blender」[3ds max 3DCG アニメーション][CG プログラミング]を開講し、ディズニー・ピクサー・アニメーションで利用されているCGアニメーション技術であるRenderManについて体験学習を行っている。オープンソース python cgkitを使って実験ができる。興味・関心がある生徒にとっては、進路選択のきっかけになるように紹介している。

2. 3次元コンピュータグラフィックスの基礎基本

3次元CGの基礎知識として、3次元空間や座標系について理解し、実際、マウス操作によってカメラ(視点)を変更する実習を行う。さらにモデリングとして立体図形の制作を行い、質感設定(マッピング)である色付けや、完成画像を出力するためのレンダリングを行う。アニメーション制作では、物体の動きを制御し、どのように動かすかを学習するこ

表 1

モデリング Modeling	形をつくる。 ポリゴンモデリング(Polygon)、3角形又は4角形の集合、ゲーム開発 NURBS モデリング(ナーブス)、工業デザインによる自由曲面
質感設定	色や模様(テクスチャ)を付ける。 ・モデルに陰影付け(シェーディングモデル) ・マッピング(Mapping) ・シェーダ(Shader) 記述言語を用いてコンピュータに計算させることで模様(テクスチャ)を自動生成する。
レンダリング Rendering	カメラでシーンを撮影し写真を撮るような作業 グローバル・イルミネーション、レイトレーシング、パストレーシング
アニメーション Animation	キーフレームアニメーション モーションパスアニメーション インバースキネマティクス(逆運動学)
映像制作におけるCG技術	コンポジット(既存映像と3Dの合成) Web3D技術(WebGL, three.js) ストーリーミング技術

とができる。様々な状況に応じて、試してみることで発見・理解・課題解決へとつながる。

まず、モデリングを行い、質感設定、カメラ、ライトの設定を行ってからレンダリングを行う。

3. Pixar RenderMan(非商用利用ソフトウェア)

ディズニー・ピクサーなどの3次元コンピュータグラフィックスを用いる主要な映画作成で用いられている「RenderMan」をコンピュータ教室に導入している。RenderMan・インターフェイスの「シーン記述言語」であるRIBを理解することで、フォトリアルな画像を計算(レンダリング)することができる。RenderManは、ゲームで使われているリアルタイムレンダリングの技術ではない。RenderMan

に関する情報は、非常に少なく、難しいと捉えられているところがある。Autodesk Maya や Blender 等と連携して用いる便利な方法がある。それらがなくてもエディタとレンダラを用いて操作することができ、サンプルをレンダリングし、色、質感、大きさ、位置など設定を変えてみることで3DCGの基礎基本となる部分を理解していく。プログラミング言語のPythonを使ってRenderManを使いやすくするcgkitと呼ばれるツールを使って、いろいろと実験することで理解が深まる。

(1) 使用ソフトRenderManの特徴

① エディタ EmEditor 等を利用する。
 ② 互換レンダラがある。Photorealistic RenderMan (prman), RenderMan 互換レンダラとして3Delight や Sitex Air などの商用ソフト、オープンソース Aqsis Renderer, Pixie Renderer, XRT renderer がある。

ピクサー社は2015年、RenderManの非商用利用版をリリースした。バージョン20では新しいレンダリングシステムRISが搭載された。RIS(Rix Integration Subsystem)は、自然界の光や質感を忠実に再現する物理ベースレンダリングを用いている。

③ Blender2.76b とアドオンのプラグインであるPRMan For Blender を用いて、高画質の画像を出力が可能である。RISでのレンダリングは、時間がかかる。光、色、質感の設定を変更していくことでフォトリアルな画像ができる。

物理ベースレンダリングRISを利用の場合はコマンドプロンプトから以下のように実行する。

```
> prman -d it xxxxxx.rib
```

RenderManの基本を学ぶには、ケッソン先生のサイト(<http://www.fundza.com/>) CG References & Tutorials が役に立つ。カメラは原点にあり、Z軸の+方向を見ている。



RenderMan を使って作成した CG

■球体を表示する RIB 記述例

#部分はレンダリングで無視される。日本語文字は記入不可

```
# sphere01.rib
Display "sphere01.tif" "file" "rgb"
Projection "perspective" "fov" 10
Format 640 480 1
# placecam 10 5 - 10 0 0
Rotate - 19.47 1.00 0.00 0.00
Rotate 45.00 0.00 1.00 0.00
Translate - 10.00 - 5.00 10.00
WorldBegin
LightSource "pointlight" 1 "from" [- 10 10 - 10] "intensity" 300
LightSource "pointlight" 2 "from" [10 10 - 10] "intensity" 300
Color 1 0 0
Surface "plastic"
Sphere 1 - 1 1 360
WorldEnd
```

- ・色設定、RGBで0から1までの数値を利用する。
- ・ライト設定、各物体に質感Surfaceを設定しておかないとライトはうまく働かない。

```
LightSource "shadowdistant" 4 "from" [ 1 1 - 1]
"to" [0 0 0] "intensity" 1.5 "string shadowname"
["raytrace"]
```

で影を設定し、レイトレーシングが可能になる。

- ・質感Surfaceの設定、シェーダという。(インストールされたレンダラのshadersフォルダにある)プラスチックな質感はSurface "plastic"と記述する。自作可能であり、インターネットで開発したものを公開している方もいる。

4. Blender(オープンソースソフトウェア)

(1) すぐに使える統合3DCGソフトウェア

Blenderは、世界中で使われている無料3DCGソフトウェアである(気に入ったら財団へ寄付)。3次元コンピュータグラフィックスを行うための統合ソフトで、インターネットからダウンロードできる。3次元モデル作成からアニメーション作成、ビデオ編集、3Dプリンタ出力データ作成が可能である。また、プログラミング言語Pythonを同梱しており、カスタマイズすることができる。市販の高価格ソフトと同等の機能を持っている。教育機関での利用、個人での利用、CG制作会社での映像制作等、幅広く使われている。



Blender に用意されている CG のスザンヌ

高校体験授業としてこれまで多くの中学生に Blender を体験してもらっている。付属のスザンヌ (monkey) といっしょにすぐに楽しむことができる。

(2) インストール

各 OS の圧縮ファイル、自動インストーラがある。コンピュータ環境にあった圧縮ファイルを解凍して使用する。<http://www.blender.org/> から Windows 版 64 ビット、blender-2.76b-windows64.zip 圧縮ファイルをダウンロードする。コンピュータの好きなところでファイルを解凍し、中の blender.exe をダブルクリックすると起動する。フォルダ名が日本語になっているところの下に置くとうまく動かないことがある。なるべく C ドライブ、D ドライブ等の直下に置くとうまい。

(3) Blender の基本操作

Blender のマウス操作は、真ん中ボタンで行う。物体の選択は右クリックになる。以下、基本操作をまとめた。

■画面操作

- ・マウス真ん中ボタンをスクロール(回転)→画面が拡大・縮小(近くなったり遠くなったり)
- ・マウス真ん中ボタンを押しながら左右にドラッグ→画面が回転(ぐるぐる景色が回る)
- ・Shift キーを押しながらマウス真ん中ボタンを押しながら左右にドラッグ→画面が移動(見ている位置をずらす)

■物体の選択→右クリック

■見たい物体を画面の真ん中に近づける→キーボードのテンキーにある.(ピリオド)キー

■3Dview とカメラ画面の切り替え→キーボードのテンキーにある0。再び0を押すと3Dviewに戻る。マウス真ん中ボタンをスクロールで枠調整(キーボード上の0を押すと物体が全部消える!!) Ctrl + Z を押して元に戻る。

■Z キーを押す→ワイヤーフレーム表示になる。再びZを押すと元に戻る。

■Shift + D →物体の複製。たくさんスザンヌ(おさる)を作成することができる。

■レンダリング→キーボードのファンクションキー F12 を押す。画像がレンダリングされる。前の画面に戻るには、Esc キーを押す。戻らない時は、マウスの白い矢印を、できた画像がある場所の中に入れる。

■好きな方向からカメラをセットするのは→Ctrl + Alt + テンキーの0

■Ctrl + 0 →選択した物体がカメラ位置となるので注意、カメラを選択してから Ctrl + 0 とする。カメラ同士の切り替え。

■カメラ画面の微調整は、カメラ枠を出しておいて、Shift + F →ゲームのようなフライビューになる。

■画面配置がおかしくなって、デフォルトに戻せない。一回適当な名前で保存し終了し、Blender を起動しメニュー File → Open で開き、画面左下の Load UI の□のレ点(チェック)をはずしてから、ファイルを開く。

■4画面表示の切り替え→Ctrl + Alt + Q

■物体の回転 3Dview 画面下にある円弧のマークをクリック。赤、青、緑の線をドラッグすると物体が回転する。変な向きに動いたら Ctrl + Z で元に戻る。回転のリセットは Alt + R。Intel グラフィックカードによっては、回転がうまくできないことがある。キーボード R を使う。

X 軸回転→R を押し、次に X を押す。

Y 軸回転→R を押し、次に Y を押す。

Z 軸回転→R を押し、次に Z を押す。

5. RenderMan と Blender の連携 (PRMan For Blender の取り付け)

(1) ダウンロードと取り付け

<https://community.renderman.pixar.com/article/906/introduction-to-prman-for-blender.html> 「blender renderman」で検索すると上記のピクサーの紹介サイトが出てくる。

<https://github.com/bsavery/PRMan-for-Blender> から、Download ZIP をクリックしてダウンロードする。最新は v0.8.0 03/22/2016 になる。解凍し、PRMan-for-Blender-master のフォルダごと、blender の以下のフォルダにコピーする。例えば C ドライブならば、

C : ¥blender-2.76b-windows64 → 2.76 → scripts
→ addons

(2) blender での addon の呼び出し

① blender を起動し、メニュー File → User Preferences. を出す。

② Add-ons タブをクリックし、左側の Categories から Render をクリック。

③ Render : PRMan Render Engine の右側のチェックボックスをクリック。Save User Settings をクリックして完了。

(3) Blender で RenderMan を使ってレンダリング

画面の上、中央にあるレンダリング設定を Blender Render から PRMan Render に切り替える。Blender でモデルを配置し、シーンを作る。デフォルトのポイントライトは暗いので RenderMan のライトに変更する。質感も RenderMan シェーダ bxdxf に変更する。

6. 授業実践概要(授業案)

「Python Cgkit を用いたプログラミングによる画像制作」の実習を行った。一回の実習は 100 分である。実習 1 CG の基本と流れについて / コマンドプロンプトの使い方 / RenderMan・バイトストリーム・インターフェイスについて / RIB による基本シーンの理解とレンダリング / ライティングと様々なシェーダ。実習 2 RIB について基本図形の移動・回転・拡大縮小 / レイトレーシングとグローバル・イルミネーション / Python cgkit によるシーン記述とカメラ設定 / 複数の球体の生成について / 複数の図形をつくる。実習 3 cgkit によるアニメーション(三角関数について) / ウォークスルー・アニメーション / ODE Dynamics を利用した物理シミュレーション / OBJ インポートとレンダリング

■生徒感想から

① いろんな質感がピピッと作れちゃうのが面白いです。② RIB 記述の後、Python Cgkit からはカメラワークの設定がしやすくなって良かったです。すごく有り難味がありました。なんて、いい子なんだへび！。③ 今回はなんとアニメーションです！楽しかったあ！ボールが走り去ったらドナドナの曲が頭に流れてきたり、カメラがシーンを通り過ぎるのではターミネーターが流れたり面白かったです。CG のいいところは色をきれいに見せることができ

るということでしょうか。これは、とてもステキですね。いいと思いました。④ 今日は作ったものを動かしたらどうなるかシミュレーションした。細かい設定もできるようでとても驚いた。また実際動いているところを見てとても感動した。自分が思ったようにシミュレーションできるように知識をつけたいと思った。とても楽しく Python を学ぶことができ、3D プログラミングに興味を持つことができた。⑤ 今日は初めて数学の実用性がわかった。sin, cos を利用したアニメーションを行ったが、普通に数学を学ぶより理解できたと思う。難しいことでも何かとつなぎ合わせて考えると簡単に学ぶことができることがわかった。この調子で「アハッ」体験をしたいと思う。⑥ 今日は球体を落下させたり、バウンドさせたり、障害物や床を傾けたりして、落としてみるものすごく滑稽で笑ってしまいました。⑦ カラフルなブロックがどんどん崩れたり、物がすべり落ちたりはねたりが楽しかったです。もう終わりなんて早いなーと思います。講習に出て CG の大変さがよくよくわかったので、それを踏まえてトイストーリーとかモンスターズインクをもう一度見てみたくなりました。難しかったけど、めったにできない貴重な体験ができたので、やれて良かったです。ちょっと先のところまでできるようになれたらいいなーと思いました。(毛を生やしたりとか！)

7. 今後の課題

- (1) ピクサー社の RenderMan を利用して実習内容をさらに高めていく。新たな教材開発(OSL 利用)を行う予定。
- (2) 無料モデル素材やアニメーションについて興味ある中学生・高校生に紹介できるような機会を設けたい。3DCG 体験は、高校の英語・数学・理科・芸術の勉強と関連していることを実習を通して拡大していく。

参考資料

- 1) Google で「Blender 入門」で検索すればたくさん資料が出てくる。注意点としてはバージョン 2.5 以降の解説を参考にすることである。2.49 以前はインターフェイスがかなり異なる。
- 2) RenderMan (非商用版) のインストール方法
<http://www.metaseq.net/jp/archives/227/>
- 3) Blender で OSL を使ってみる一歩一歩
<http://c5h12.hatenablog.com/entry/2014/11/17/034848>