

## 3次元仮想教室の構築\*

5Q-3

古平 宏光 Runhe Huang 上田 穰†  
会津大学‡

## 1 はじめに

インターネットが、教育、ビジネスや娯楽などいろんな場面で急速に使用されているが、これは、文字、絵、そして音などを容易に情報を送発信することができるからである。現在、インターネット上に3次元世界を構築するとき、VRML(Virtual Reality Modeling Language)という言語を用いて簡単に構築することができる。しかし、3次元のオブジェクトの情報量は、2次元のものより格段に多くなってしまう。それでは、ネットワーク上に負荷がかかり快適なネットサーフィンをすることはできない。また、CPUにも負担をかけるため、ある程度のマシンパワーが必要になる。そこで、ネットワーク上の負荷を減らし、比較的マシンパワーの低いものでも3次元の空間を楽しむように、VRMLで3次元仮想教室を作りVRMLでサポートしている機能を用いて快適に運用できるように検証してみる。

## 2 VRMLコンテンツの概要

## 2.1 3次元の英語教育のための仮想環境

生徒にとって、文字や音だけを提供するだけの既存の英語学習では、退屈なものになってしまいがちである。そこで、3次元の空間内で自由に生徒自身が学習できるシステムの構築をした。今回は、アメリカ風のbarとイギリス風のbarを用意した。そのbar内に人間のオブジェクトを配置し、それぞれの国の英語の発音を喋れるようにsound nodeを持たせた。これにより、生徒はアメリカ式発音とイギリス式の発音の違いを、生徒自身が人間のオブジェクトに近づくことで学ぶことができ、barからその国の雰囲気、特徴もつかむことができる。また、ブラウザは、その教室に訪問者があった場合、その訪問者を表示させることができる。これにより、ほかの訪問者(アバター)を認識することができる。

## 2.2 システム評価

はじめに、アメリカ風のbarを構築する際、ファイルサイズを減らす工夫を行わなかった。そのため、ファイルサイズも多くなり、コンテンツ自体も複雑なために、データのロード時間がかかり、操作するとき移動速度が遅く、勉強するには不向きであった。なぜなら、システムの操作性が悪いため、ユーザーがストレスを感じてしまうからである。その結果、こ



Figure 1: VRMLコンテンツの例

のシステムから遠ざかることになり、英語を学ぶ上で重要な重複学習がおろそかになってしまう。そこで、イギリス風のbarでは、VRMLでサポートしている機能を用いてファイルサイズを減らす工夫をした結果、移動速度が増し操作性は向上した。これにより、ユーザーはストレスをあまり感じずに勉強することができる。

## 2.3 アバターの組み込み

VRML97において、訪問者は、同一シーンに他の訪問者がいても、その存在を認識することはできない。しかし、ネットワークプログラムを用いることで、他の訪問者を認識することができる。今回は、Server-receiver、Server-sender、Client-receiver、Client-senderの4つのプログラムを書いた。これは、sonyによるCommunity Placeをもとにしている。例えば、アバターAが既に仮想教室にいるとする。アバターBが、その仮想教室を訪問したとき、Server-receiverは、アバターBのためにServer-senderを作る。アバターBのClient-senderは、Server-senderに、アバターBの位置と向き的情報を送る。Server-senderは、他のServer-senderとアバターBのClient-receiverにその情報を送る。最終的に、クライアントマシン上にアバターBを含んだ新たなシーンを表示する。このシステムでは、お互いに認識できるが、コミュニケーションをとることができない。しかし、次のVRMLの仕様では、マルチユーザーをサポートする予定なのでそれに期待したい。

## 3 快適な操作性のために

オブジェクトのポリゴン数が多いと、計算機に多くの負荷をかける。さらに、ファイルサイズが多いとネットワークに負荷がかかってしまう。このことをできる限り避けるために、ポリゴン数の削減とデー

\*Design of a 3D virtual environment for learning English

†Hiromitsu Kodaira Runhe Huang Minoru Ueda

‡The university of Aizu

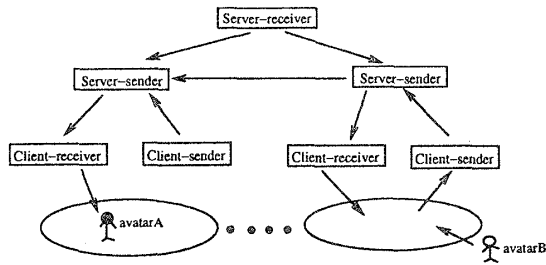


Figure 2: アバターの組み込みの仕組み

textureの有無	ファイルサイズ	ポリゴン数
use texture	35KB	89
no texture	78KB	426

Table 1: texture を用いた場合と用いない場合のポリゴン数の比較。

タサイズを抑えるために以下の方法で調べてみた。

### 3.1 ポリゴン数の削減

#### 3.1.1 texture の使用

オブジェクトを texture を張り付けることに変えた結果、表??のような結果になった。しかし、ファイルサイズとしては、texture の分増えてしまうのでこれを使いすぎた場合、ネットワーク上に負荷が掛り過ぎてしまう。

#### 3.1.2 node の使用

LOD(Level of Detail) node を用いて、オブジェクトが遠くにある場合、視覚的には変化が分からない程度に雑にオブジェクトをつくることにする。その結果は、表??の通り。この場合も、ポリゴン数を減らすことができるが、ファイルサイズが多くなってしまう。

### 3.2 ファイルサイズの抑制

#### 3.2.1 ファイル圧縮の利用

表??のように、ファイル圧縮によりオリジナルのファイルサイズからは、25-35%まで減らすことができる。ファイルサイズが減ることで、ネットワークからロードしてくる時間は短縮される。今後、ファイル形式が binary 形式をサポートするときは、さらに圧縮さ

LOD	ファイルサイズ	ポリゴン数
use LOD	153KB	364 ~ 1624
no LOD	74KB	1624

Table 2: LOD を用いた場合のポリゴン数の変化。詳細に表現した場合、2,648 個のポリゴン数があったが、オブジェクトと視点の距離が遠く離れているときに、粗くオブジェクトを表現した結果ポリゴン数を 432 個まで少なくすることができる。

Object	圧縮前	圧縮後	圧縮率
American Pub	1254.7KB	298.2KB	76.2%
British Pub	321.8KB	82.0KB	74.6%
Human (man)	450.0KB	119.9KB	73.3%
Human (woman)	344.8KB	107.8KB	68.8%
Furniture	11.4KB	4.0KB	64.6%

Table 3: ZIP 形式による圧縮の割合を表に表す。この表から、オブジェクトの複雑なものの方が圧縮されている。冗長な情報が多くふくまれているからだろう。

れるだろう。これにより、ロード時間によるストレスもあまり感じられなくなるだろう。

#### 3.2.2 node の使用

node を使用しても、ファイルサイズを押さえることができる。家具などの同じオブジェクトを使うときに USE、PROTO の node を使う。これにより、ノードの再利用を図ることができ、ファイルサイズを押さえることができる。また、Inline node を用いて、ファイルを分散してしまうのも方法の一つである。これにより、1つのファイルのサイズは小さくなるので、ロード時間は短縮される。しかし、多用するとネットワークからデータをロードする回数が増え、ネットワークトラフィックが増加する。

## 4 まとめ

VRML のファイルを快適に扱うために、以上のような方法で検証してみた。結果としては、ほとんどの場合、ポリゴン数の削減とファイルサイズの軽減を達成できた。しかし、どの方法においても、使いすぎると CPU にかかる負荷が増大したり、ネットワークトラフィックが増えたりするのでバランスをとる必要がある。また現段階では、VRML がマルチユーザーに対応していないため、アバター同士がコミュニケーションをとることができない。これは、VRML がサポートすることで解決できるだろう。また、今後の課題として今回行った方法を用いても大容量のファイルの場合、それに対する有効なネットワークからの効率的なロード方法について研究していく。

## References

- [1] Roder Lea, JAVA for 3D and VRML worlds, プレンティスホール出版, 1997
- [2] 三浦憲二郎, VRML2.0, 朝倉書店, 1996
- [3] Sony virtual Society Home-page, <http://vs.sony.co.jp/>