

# 3D 仮想空間上でのリアルタイムアニメーションとそのネットワーク拡張に関する研究

伊沢 昂<sup>†</sup> 鈴木 智也<sup>†</sup> 井関 文一<sup>‡</sup>

東京情報大学総合情報学部情報システム学科<sup>†</sup> 東京情報大学総合情報学部情報文化学科<sup>‡</sup>

## 1. 概要

現在、メタバースと呼ばれる3D仮想空間サービスの内、最も代表的なものは米 Linden Lab社の提供する Second Life[1]、およびそれと互換性のある OpenSimulator[2] (以後 OpenSim)である。

Second Life/OpenSim 内ではアバターに任意のアニメーションを実行させることが可能であるが、それらのアニメーションデータは予めサーバにアップロードして置かなければならない。そのため、アニメーションを利用する場合の自由度はそれ程高くない。またアニメーションデータはアップロードの際、Second Life/OpenSimの内部形式に変換されるため、元のアニメーションを忠実に再現できない場合がある。

もし、アニメーションをリアルタイムに実行することが可能であれば、アニメーションを利用する際の自由度が増し、またその場での試行錯誤によるアニメーション(マシナマ等)の作成も可能となる。さらに、仮想空間内でのアバター間のコミュニケーション手段に幅を持たせることも可能となる。

これまでの研究で、我々はローカルな Viewerのみと言う制限はあるが、共有メモリを用いて、Second Life/OpenSim 上でのリアルタイムアニメーションの実現に成功している[3]。今回我々は、さらに、ローカルな環境に留まらず、ネットワーク上でリアルタイムにアニメーションを実行するシステム(SLKinect2[4])の開発を行った。

## 2. Viewerの修正と共有メモリ

Viewer本来のアニメーション実行の仕組みでは、本質的にリアルタイムのアニメーションを実行することは不可能である。そこで、外部からのリアルタイムのアニメーションデータに

対応させるために Viewerの修正を行った。

Fig.1に修正を行ったのルーチンの簡易的なフローチャートを示す。図中で灰色の箇所が Viewerの修正部分である。

以前のローカルなシステム[3]では、リアルタイムに実行を行うアニメーションの UUIDの識別機能はループの外側に存在したが、今回のネットワークに対応したシステムでは、UUIDが動的に変化する可能性があるため、逐次 UUIDの識別が行えるようにループの内側に配置されている。またこれに伴い、UUIDを識別するためのデータは、共有メモリによるアニメーション UUIDのインデックスとして与えられている。

Fig.2に共有メモリの構造を示す。共有メモリは、チャンネルとアニメーション UUIDの対応と取るインデックス領域と、ジョイントのデータそのものを格納するデータ領域に分かれる。データ領域はジョイントの数だけ存在する。

チャンネルはアニメーション毎のデータの位置を表す指標で、今回作成したシステム(SLKInect2)では、デフォルトでチャンネル0~21が存在する。チャンネル0はローカルな

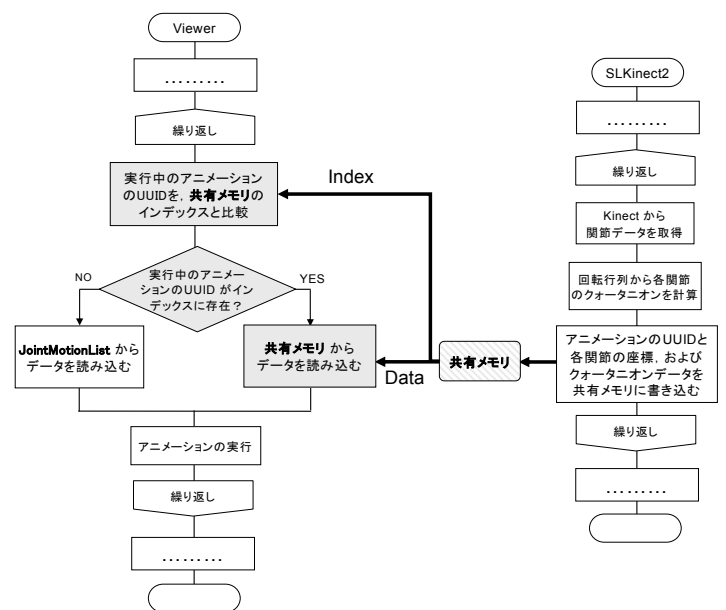


Fig.1 システムのフローチャート  
Viewerの灰色の箇所が修正部分

Research of the Real Time Animation and Network Extension in 3D Virtual Space.

Akira Izawa<sup>†</sup>, Tomoya Suzuki<sup>†</sup> and Fumikazu Iseki<sup>‡</sup>  
Tokyo Univ. of Info. Sci. Faculty of Informatics.

<sup>†</sup>Dept. of Information Systems

<sup>‡</sup>Dept. of Media and Cultural Studies

データ用であり、1~21 がネットワーク経由で受信するデータ用である。21 個のネットワーク用チャンネルの内、1 つには自分自身のデータを格納するので、実質的にはネットワーク上で 20 個の外部ノードをサポートする。

### 3. Kinect (OpenNI) とジョイントデータ

本研究では、外部データ用の機器として Microsoft 社の Kinect を使用している。また制御用にオープンソースの OpenNI (v1.3) と NITE (v1.4) [5]を用いている。

Kinect ではジョイントの回転データとして、絶対座標系に対する回転行列が得られる。Second Life/OpenSim では、回転データは親のジョイントに対する相対的なクォータニオンとして与えなければならないため、回転行列からオイラー角を計算し、さらに相対的なクォータニオンに変換しなければならない。

また、Second Life/OpenSim では、ジョイントの基点となるルートジョイントが存在するが、Kinect のデータでは全てのジョイントは独立しているため、ルートジョイントは存在しない。今回は、Kinect の左右の HIP の座標の中間点を仮想的なルートジョイントと見なして処理を行っている。

現時点で我々の開発したシステムでは、Kinect から得られたデータを補正せず、ほぼそのまま Viewer に転送している。そのため、しば

しばアバターの動きが不自然、またはあり得ない動きになる場合があり、これらは今後の課題である。

### 4. アニメーションデータの中継

ローカルで取得したデータをネットワーク上で同期させるために、Fig. 3 に示すようなアニメーション中継サーバを作成した。

アニメーション中継サーバはグループ化機能を有し、各クライアントソフト (SLKinect2) のローカルなデータを UDP データとして受信し、そのグループ内の全てのクライアントに転送する。これにより、グループ内でのアニメーションの同期を可能にする。

ただし、現時点ではネットワークの遅延や転送フレームレートの効率に関する評価は、まだ行っていない。また、セキュリティに関しても、十分に高いとは言えない。

### 5. まとめ

本研究では、Second Life/OpenSim の Viewer に若干の修正を加え、さらにアニメーション中継サーバを構築することにより、ネットワーク上でリアルタイムアニメーションが可能な事を示した。

本研究でのリアルタイムアニメーションの実行に関しては、システムの QOS の評価などがまだ行われていない等の問題点が存在するが、Second Life/OpenSim などの 3D 仮想空間上でリアルタイムアニメーションが容易に可能になれば、マシナマの作成やアバター間のコミュニケーション手段の広がりにも有益であると期待される。また、3D 仮想空間を使用したゲームなどにも応用が可能であると考えられる。

また、今後の課題としては以下の項目が挙げられる。

- ・サーバの QOS の評価.
- ・セキュリティの考慮.
- ・人間の骨格構造を考慮したジョイントデータの補正.

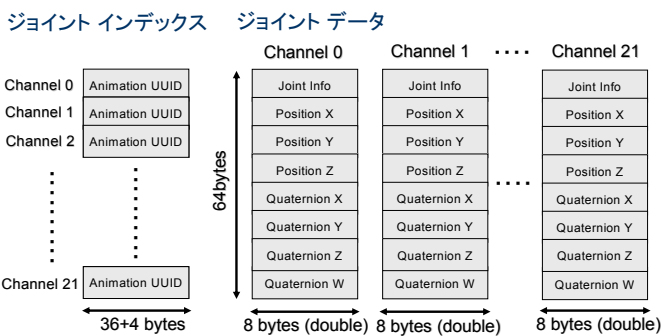


Fig.2 共有メモリの構造

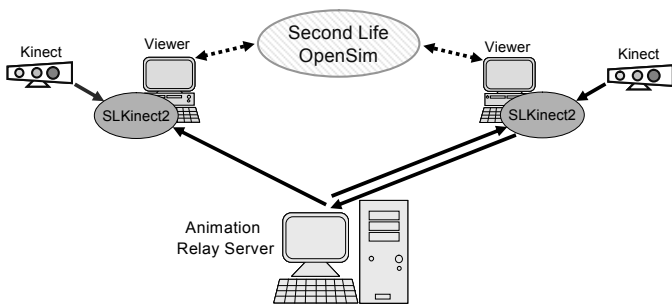


Fig.3 アニメーション中継サーバ

### 参考文献・URL

[1] <http://secondlife.com/>  
 [2] <http://opensimulator.org/>  
 [3] 井関文一, 伊沢昂, 鈴木智也, 「3D 仮想空間上でのリアルタイムアニメーションに関する研究」, 『ゲーム学会第 10 回全国大会講演論文集』, pp. 31-34, Dec. 2011.  
 [4] <http://www.nsl.tuis.ac.jp/xoops/modules/xpwiki/?SLKinect2>  
 [5] <http://openni.org/>