

3次元ユーザインタフェースによる仮想物体の形状操作

上野 大輔[†] 杉本 晴季[†] 中島 将之[†] 濱川 礼[†]

[†]中京大学 情報理工学部 情報システム工学科

1. 概要

本論文では、Kinect センサー[1] (以下 Kinect とする) の機能である赤外線による深度情報、RGB 画像、骨格トラッキングシステムからデータを抽出し、実際に物を操作する要領で 3D モデリングを行う手法について提案する。

2. はじめに

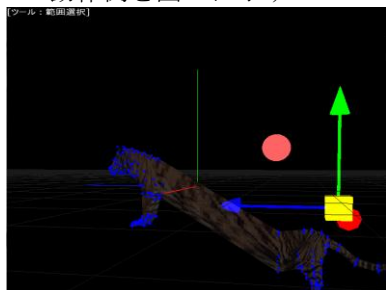
近年、コンピュータ技術の発達に伴い、三次元コンピュータグラフィックス (以下 3D-CG とする) を用いたゲームやアニメーション等のコンテンツが増加の傾向に見られ、趣味でコンテンツを作成する一般ユーザも増えている。また携帯端末に拡張現実を用いたアプリケーションなども増えている[2]。

現在、3D-CG を作る際、仮想三次元空間上に物体を成形する 3D-CG モデリング(以下モデリングとする)を行うツールはマウスを用いたものが主流である。しかし、マウスは二次元入力装置であり、二次元の操作を処理の中で三次元的な入力を行うことは感覚的なずれが生じ、ユーザは直観的な操作を行うことができない。本研究室内の 11 名に既存のモデリングシステムが使用しやすいかどうかのアンケートを取ったところ、8 割以上が使いづらいという回答を得た。

本研究では、処理速度の要求されるゲームでも実用されている Kinect の骨格トラッキングシステムを用いることで、ユーザの動きと連動した操作が行えるのではないかと考えた。

本研究で開発したシステムでは、直感的な操作により、既存のモデリングシステムの操作が困難、または不慣れたユーザであっても簡単に操作が行え、3D-CG の作成が行うことが可能となる。

システムの動作例を図 1 に示す。



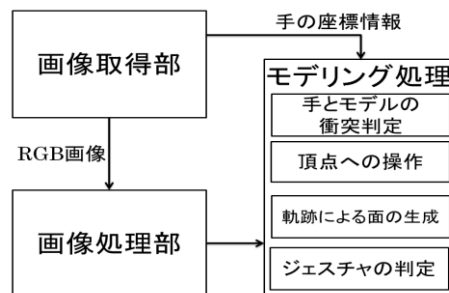
【図 1 システム動作例】

Manipulating virtual object by three-dimensional user interface
Daisuke Ueno[†], Haruki Sugimoto[†], Masayuki Nakashima[†]
and Rei Hamakawa[†]

[†] Department of information system technology and Faculty of information science, Chukyo University.

3. システムの概要

システムは、図 2 のような構成となっている。



【図 2 システム全体の構成】

(1)画像取得

Kinect の RGB 画像、深度センサーから得た情報を用いてユーザの検出、ユーザの手の位置の座標の取得を行っている。

手の位置座標は Kinect の近赤外線カメラから入力された情報を使って身体の関節部位を推定し骨格構造を構築する機能を利用し取得している。Kinect を操作するにあたって OpenNI[3]を使用している。

(2)画像処理

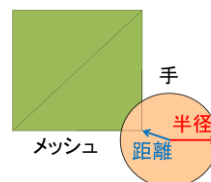
Kinect から取得した画像データに色の検出を行い、手だけを検出する処理を行う。これにより手以外の服の袖などを消す。色の検出を行ったデータにラベリング処理[4]を施し重心座標を取得する。この重心座標をポイントとして、物体を操作するときの目安とする。その際、ラベリングで得たものの中で面積が最大のものだけを取り出す。この処理によって手の部分と認識させることができ、ノイズを減らすことができた。

(3)モデリング処理

本システムのモデリング処理について以下に説明する。

(3-1)手とモデルとの衝突判定

手を球体と仮定し、モデルを頂点、または面単位に分解する。描画空間内の手の位置とそれぞれの距離を算出し、図 3 のように距離が球体の半径より小さければ手とモデルは衝突していると判定する。



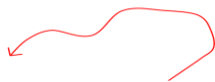
【図 3 手とモデルの衝突判定】

(3-2)頂点への操作

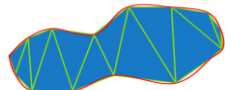
本システムでは、球に内包する頂点、球が接触しているモデルの頂点、任意の二点で作られる直方体に内包する頂点、および面に対し三次元方向への自由な操作を行うことができる。また、前述した衝突判定を用いているので頂点を押すような操作も行うことができる。

(3-3)軌跡による面の生成

フレーム毎に座標を保存し、移動値が一定値を超えた時、軌跡を通る点として保存する。最初に保存した軌跡と最後に作成した軌跡が、一定値より近づいた時、面として生成する。この時、面は3つの頂点からなり、面がユーザから見えるように法線の向きの調節を行う。軌跡が図4の赤線のような場合、面は図5のような青色部分のような形になる。また、立方体や球の配置を行うことができるため、空間に対して絵を描くような感覚でモデリングが行うことができる。



【図4 軌跡】



【図5 面の生成】

(3-4)ジェスチャの判定

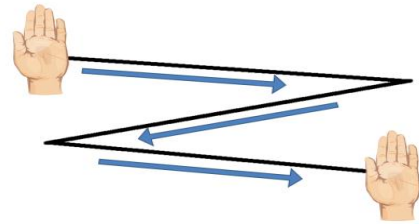
本システムでは、Kinectを用いた操作を実現しているため、入力フラグにマウスやキーボードでの入力を行わず、ジェスチャの判定による決定や取り消しを行う。

つかむや決定のジェスチャの判定方法は、図4のように画像処理により得られた手の面積の大きさの判定により行う。手をグーにした時、手の面積は小さくなるので、その状態の時を決定フラグと判定する。取り消しのジェスチャは、図6のような日常生活において断るときに行う手を横に振るような動作を仮想した。またモデリング操作を右手からの取得情報で行うため、左手の取得情報を用いてソフト内カメラの操作を行う。

判定方法は一定フレーム単位で手が一定距離以上移動した時、座標を保存し、その軌跡のベクトルとスクリーンの横方向に平行なベクトルの為す角度が鋭角になった時のベクトルを軌跡として保存し、それが図7のようになった時、取り消しフラグと判定する。



【図6 つかむ・決定の判定】



【図7 取り消しの判定】

4. マウスを用いたモデリングシステムとの比較

(4-1)頂点・面との当たり判定

マウスを用いたモデリングシステムでは、ポインタの位置から画面の法線ベクトル上に伸びた線を用いて、メッシュとの衝突判定を行う。従って、画面から見て前面にある面のみにはしか衝突判定を行うことができず、背面に配置された面と衝突判定させるためには、ソフト内のカメラを移動、または、回転させなければならない。

しかし、本システムでは、任意の位置座標を指定することができるため、カメラを移動させる手間を軽減することができる。

(4-2)カメラ操作

マウスを使用するシステムでは、逐次カメラ操作を行わなければならないが、手の位置座標は任意の値で行うことができるので、二次元入力のモデリングシステムと比べて、ソフト内のカメラを動かす手間が少なくすることができる。

4. 評価

既存の3Dモデリングシステムと本システムとの比較評価を行っている。評価項目は、手の動作とシステム上の動きが同期しているか、マウスとは異なり、直観的な動作が行えているか、ジェスチャの認識について、本システムのモデリング機能が目的の形状作成に対して十分かである。

5. 今後の展望

評価の結果を反映させシステムとしての機能の充実と操作性の向上させたい。

手の検出の処理速度の向上と、光によって検出精度が影響されるので改善を図りたい。

6. 参考文献

[1] KINECT
<http://www.xbox.com/ja-JP/kinect>
 [2] 今村 弘樹, ARTToolKitを用いた拡張現実による3D-CGモデリングツール構築のための一検討, 創価大学工学部情報システム工学科, 映像情報メディア学会技術報告 33(44), 35-42, 2009-10-26
 [3] OpenNI
<http://75.98.78.94/default.aspx>
 [4] Labeling.h
 Copyright (c) 2010, IMURA Masataka