

仮想彫刻における6軸モーションセンサを用いた切削手法 The Carving Way with 6-axis motion sensors on Virtual Sculpting

蛭間 宏明[†]
Hiroaki Hiruma

武田 正之[‡]
Masayuki Takeda

1. はじめに

近年、ビデオゲームをはじめとしたエンタテインメントの分野では、加速度センサやジャイロセンサ、赤外線センサ、圧力センサといった様々なセンサ類を用いたコントローラが新たなユーザーインターフェースとして注目を集めている。Wii リモコン(任天堂)[1]の普及により一般的な家庭にもそうした新たな遊び方が広まり、2010年現在ではカメラを用いたモーションセンサコントローラとしてXbox360のNatal(Microsoft社)[2]のようなインターフェースも研究・開発されており、PS3でも加速度センサやジャイロセンサを搭載した6軸モーションセンサコントローラの開発が進んでいる。このようにエンタテインメントの分野ではユーザーインターフェースに着目した技術の需要が高まっているといえる。

こうしたエンタテインメントの分野においては従来慣性航法で使われていたような高価なセンサを用いることは困難である。そのため例えば加速度センサであれば分解能の低いセンサを用いなければならず、出力された値を2階積分して3次元的な位置を割り出すには誤差が増加していくために困難である。そのため、分解能の低いセンサを用いたときでもユーザーにとって利用しやすく直感的な操作を実現し得る技術が必要となる。

一方でCG(Computer Graphics)レンダリングの分野では絵画のように見えるノンフォトリアリスティックレンダリング(Non Photorealistic Rendering:NPR)の研究がされている。NPRの基本戦略には作品の外観の特徴を模倣すればそのレンダリング手法は特に問題としない目視概観主体駆動(appearance-based rendering ;ABR)と、作品の制作工程を再現し、各工程における物理的な現象をモデル化する物理モデル駆動(physics-based rendering ;PBR)に大別される。PBR研究の1つとして仮想彫刻・仮想版画が研究されている。これは彫刻の制作行程を模倣することで、3DCGモデリングの過程をユーザーにとって直感的なものにするという試みである。しかし現実の彫刻を完全に再現するような実装を行うわけではなく、彫刻の切削動作を通してユーザーに分かりやすい直感的な操作方法を提供することを重視する。これはキーボードやマウスといった従来のデバイスに依らないユーザーインターフェースを用いることで実現し得るものも多い。その具体的な例としてTeddy[8]やSPIDAR[9][10]が挙げられる。また、コンピュータから揭示される最終的な結果は2次元上のディスプレイであるが、現実には3次元の空間である。

そこで、本研究では3次元上での入力を行うユーザーインターフェースである3軸加速度センサと3軸ジャイロセンサを用いた直感的な彫刻刀操作の実装を目的とする。

2. 関連研究

2.1 仮想彫刻

NPRでは水彩画、毛筆、ステンドグラス、鉛筆画、油絵、トゥーンレンダリングなどの表現手法が研究されており、彫刻や版画の表現手法は”仮想彫刻”、”仮想版画”として研究されている。仮想彫刻の研究においては切削対象となる彫刻素材の研究が主になされてきた。彫刻素材の形状表現に半空間と三値論理代数系に基づく多面体表現[3]が提案された。これは彫刻素材と彫刻刀に対応する部分の部分集合差演算を用いることで切削を表現しているものである。この仮想彫刻システムによって版画生成を目的とした版木の切削を試みている。会話型システムによる切削や入力された画像の特徴量抽出に基づく自動切削の試み[4]、版画を使った浮世絵[5]などの研究もなされた。また、ユーザーがドラッグした部分に対応した部分の法線マップを動的に書き換えることで疑似的に凹凸表現を行う手法[6]も研究されている。その後ユーザーインターフェースからのアプローチとしては、ペンタブレットの筆圧の強さに応じてCSG形状表現された3Dオブジェクトに対して、インタラクティブに切削を行う彫刻・木版画システム[7]が研究されている。

2.2 ユーザーインターフェース

ユーザーインターフェース研究について述べる。ユーザーからの2次元的な入力により3次元形状を生成するTeddy[8]がある。従来の3DCGのモデリングツールは、ツールを扱うための多くの知識が必要であり初心者には扱いづらい点があったが、ユーザーからの簡単な入力からユーザーの意図を読み取り曖昧さを解消することで、少ない情報から3次元形状を生成できるアルゴリズムを実装している。また、対象をぬいぐるみに限定することでぬいぐるみ製作に特化したユーザーインターフェースを実装し、直感的な操作でユーザーの意図を的確に読み取ることに成功している。

3次元空間で入力可能なユーザーインターフェースにはSPIDAR[9][10]がある。中央に置かれた球をワイヤーで固定しその球をワイヤーで張られた空間内で操作する。張力を利用することでユーザーへの力覚揭示を行う。

3. 提案手法

本研究では彫刻刀の直感的な操作を実現すると共に、3次元上での入力デバイスによりユーザーの意図を正確に読み取ることが出来るユーザーインターフェースの実装手法を提案する。3軸加速度センサと3軸ジャイロセンサを併せて用い、「位置決め」、「角度決め」、「切削」の3つのフェーズにわけて彫刻切削の実装を行う。加速度は1階の積分で速度、2階の積分により位置を求めるが、積分による誤差を抑えるため、位置と角度は角速度の値から求めることとし、加速度は切削動作の判別のために用いる。

[†]東京理科大学大学院理工学研究科情報科学専攻

Dept. of Information Sciences, Graduate School of Science and Technology Tokyo University of Science
hirhir@mt.is.noda.tus.ac.jp

[‡]東京理科大学理工学部情報科学科

Dept. of Information Sciences, Tokyo University of Science

3.1 位置決めフェーズ

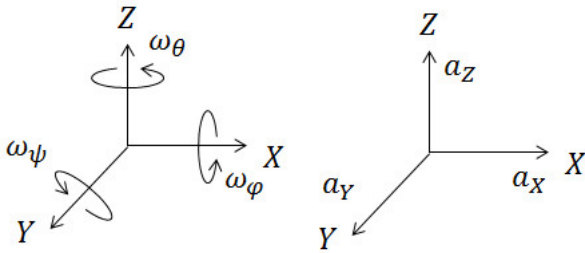


図 1: 3軸ジャイロセンサと3軸加速度センサからの情報

位置決めフェーズでは彫刻刀の位置を決定する。図1で示す3軸ジャイロセンサの角速度 $\omega_\phi, \omega_\theta, \omega_\psi$ と3軸加速度センサの加速度 a_X, a_Y, a_Z を用いる。図2で示されるベクトル \mathbf{F} はデバイスの正面方向を表すベクトルであり、角速度の値を用いてクォータニオン(四元数)による方法[11]で向きを計算する。ここでの回転クォータニオン Q は実部 t , 虚部 x, y, z を用いて

$$Q = (t \mid x, y, z)$$

で表現する。

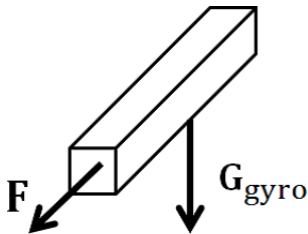


図 2: 正面方向のベクトルと重力方向のベクトル

クォータニオンにより回転軸 (α, β, γ) (ただし $\alpha^2 + \beta^2 + \gamma^2 = 1$) に対して、回転量 θ の回転を行う。まず次のような w, a, b, c を定義する。

$$\begin{aligned} w &= \cos \frac{\theta}{2} \\ a &= \alpha \sin \frac{\theta}{2} \\ b &= \beta \sin \frac{\theta}{2} \\ c &= \gamma \sin \frac{\theta}{2} \end{aligned}$$

この回転により3次元空間上の座標 (x, y, z) が (x', y', z') に移される場合、具体的には次で計算される。(R は Q の共役クォータニオン)

$$\begin{aligned} P &= (0 \mid x, y, z) \\ Q &= (w \mid a, b, c) \\ R &= (w \mid -a, -b, -c) \\ RPQ &= (0 \mid x', y', z') \end{aligned}$$

連続した回転は回転行列に変換された行列の積で表され、画面に向かう方向を初期値として \mathbf{F} を回転させていく。ある時点で検出された角速度ベクトル ω を用いて回転軸には ω , 回転量 θ は $k \|\omega\|$ (k は実数, ジャイロセンサの返してくる値に応じて重みを変える) である。

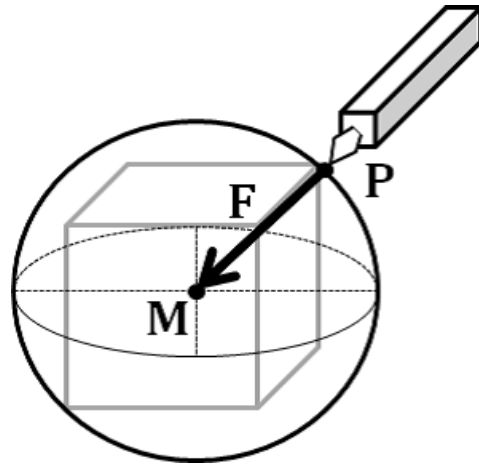


図 3: 彫刻素材上の仮想的な球体

彫刻素材は全く切削されていない状態は直方体の形であるものとし、彫刻素材の対角線を直径とする半径 r の仮想的な球体を考える(図3)。 \mathbf{F} の向きにより彫刻刀の先端位置 \mathbf{P} を定める。彫刻素材の中心 \mathbf{M} から $-\mathbf{F}$ 方向へベクトルを伸ばし仮想的な球体と重なる位置を彫刻刀の先端位置 \mathbf{P} と決める。このとき彫刻刀は \mathbf{F} 方向を向くように姿勢制御をする。すなわち彫刻刀の先端位置ベクトル \mathbf{P} は次で計算される。

$$\mathbf{P} = \mathbf{M} - \frac{r}{\|\mathbf{F}\|} \mathbf{F}$$

また角速度の1階の積分により彫刻刀の位置と角度を求めているため積分による誤差が生じてくる。加速度センサからの出力ベクトル \mathbf{a} を用いクォータニオンによる方法で誤差を修正する。加速度センサの出力する加速度は重力加速度や、移動により発生した加速度、また回転の遠心力などが加わった値である。加速度センサは静止状態であれば重力加速度を出力するのでこれを用いて \mathbf{G}_{gyro} を修正する。静止している状態は、角速度ベクトル ω の各成分の値がほぼ0であるときとする。

角速度により計算された重力方向ベクトル \mathbf{G}_{gyro} を加速度ベクトル \mathbf{a} を用いて修正するには、修正に使う回転クォータニオンを求めるため回転軸と回転量を計算する。

回転軸のベクトルは外積により求める。

$$\mathbf{a} \times \mathbf{G}_{gyro}$$

回転量 θ は \mathbf{a} と \mathbf{G}_{gyro} のなす角である。

$$\theta = \frac{\langle \mathbf{a}, \mathbf{G}_{gyro} \rangle}{\|\mathbf{G}_{gyro}\| \cdot \|\mathbf{a}\|}$$

この方法により Pitch 角と Roll 角の修正が行える。

3.2 角度決めフェーズ

位置決めフェーズで利用した手法で、彫刻刀の位置は \mathbf{P} で固定したまま彫刻刀の正面方向ベクトル \mathbf{F} だけを変える。画面上では彫刻刀から \mathbf{F} 方向に直線を出力することで切削する場所の目安を出力する。

3.3 切削フェーズ

切削のフェーズでは加速度センサの出力でユーザーが切削しようとしているかを判断する。具体的にはデバイスの正面方向あるいはその逆方向 (\mathbf{F} または $-\mathbf{F}$) の一定値以上の加速度を検出した場合、切削したと判断する。ここでは、検出される加速度ベクトル \mathbf{a} と \mathbf{G}_{gyro} の差をとることで重力加速度をキャンセルする (静止状態での出力値がほぼ 0)。すなわち $\mathbf{a} - \mathbf{G}_{\text{gyro}}$ の出力値を切削の判断に用いる。また彫刻刀が彫刻素材にぶつかったときにはその部分を切削し、デバイスへのフィードバックとして振動を行うようにする。

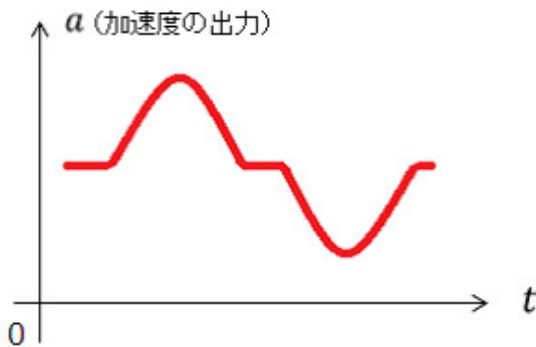


図 4: 動かして止めるという動作を行った際に検出されるべき波形

入力デバイスの正面方向だけでなく、逆方向も検出できるようにした理由を述べる。図 4 は”動かして止める”という動作を行うときに検出される理想的な波形である。”動きだしたとき”と、それとは逆の”止まったとき”の加速度の波形が検出される。これらを積分した値が 0 になるのが理想的であるが、通常は積分による誤差が生じ厳密には 0 にならない。さらに分解能が低い場合小さな動きをその出力値から読み取ることが困難であり、”動かして止める”という動作は図 5 で示す 3 種類の波形を出力する可能性がある。それぞれの波形を説明する。

● 上の波形

”止まったとき”を検出できなかった場合である。速度を 0 に戻すための逆方向の加速度が検出できない場合なので、これは速度ベクトルを毎フレーム一定値で割り、速度を 0 に近づけていくことで解決する。

● 真ん中の波形

”動きだしたとき”を検出できなかった場合である。これをそのまま積分して位置を求めた場合、意図した方向とは逆に動いてしまい、加速度の値のみからユーザーの意図を読み取るのは困難である。

● 下の波形

両方共検出できなかった場合である。これはそもそも移動させなければよい。

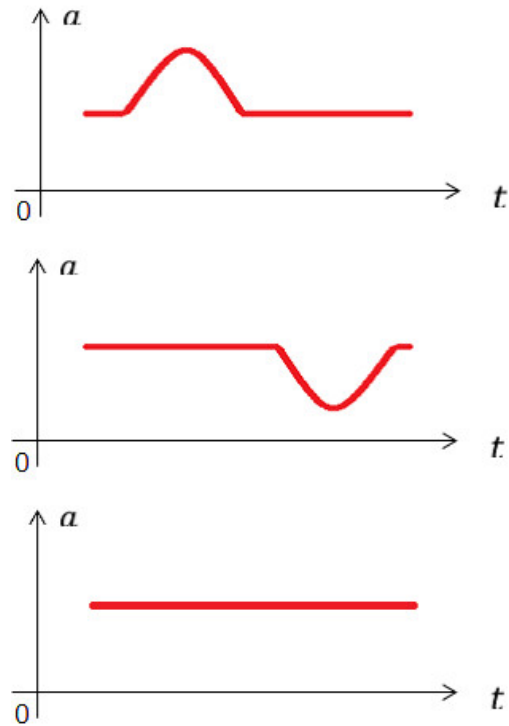


図 5: 分解能の低い加速度センサで出力される可能性がある波形

加速度センサからの出力値を積分して速度を求めた場合に次のような問題が生じる。動き出した時の加速度が検出できなかった場合、その積分値は動かした方向と正負が逆になってしまうため、加速度の値のみからユーザーの動かしたい方向を知るのは困難である。本研究の提案手法では位置と角度を事前に定めているため、ユーザーの動かしたい方向は既にわかっている。よってこの問題を解消することができ、これにより加速度センサの分解能が低い場合でもユーザーの意図を読み取ることができる。

4. 実装

提案した切削手法を実装した仮想彫刻システムについて述べる。3 軸加速度センサとして Wii リモコン、3 軸ジャイロセンサとして Wii モーションプラスを用い (C++ のライブラリである WiiYourself![12] を用いる)、切削対象となる彫刻素材及び彫刻刀の 3DCG モデルは DirectX 9.0 により表現する。C++ による実装を行う。またフェーズの切り替えのため Wii リモコン表側にある A ボタンと裏側にある B ボタンを用いる。操作方法を次に示す。

● 何も押していない状態 (位置決めフェーズ)

Wii リモコンの回転から位置を決定する。

● A ボタンと B ボタンのいずれかを押す (角度決めフェーズ)

位置が固定され Wii リモコンの回転から角度だけが変化する。

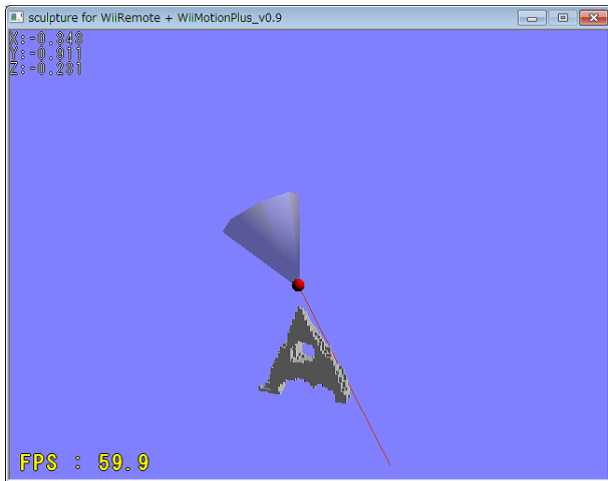


図 6: アプリケーションの実行

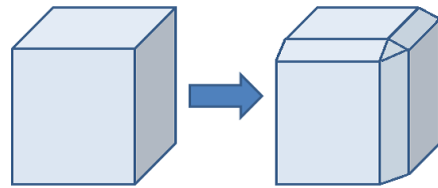


図 8: ユーザーテストで切削してもらう立体

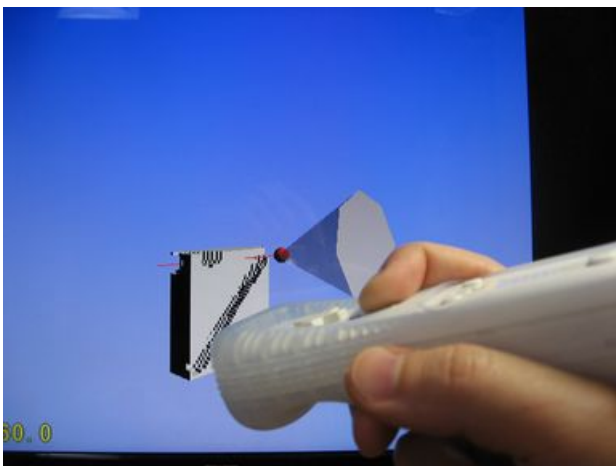


図 7: Wii リモコンによる操作

- A ボタンと B ボタンの両方を押す (切削フェーズ)

Wii リモコンを先端方向へ振ると彫刻刀が進む。彫刻素材とぶつかると切削が行われる。

彫刻素材の表現には 3 次元のボクセルを用い、そこから彫刻素材を表現するポリゴンメッシュを構成する。レンダリングの際には表面のみを描画する。アプリケーションの実行画面を図 6,7 に示す。

5. ユーザーテストによる評価

本システムで実装した提案手法の評価のため次のユーザーテストを行う。

被験者 6 名に対して簡単に操作方法の説明を行いながら 2, 3 分ほど実際に触ってもらうことで操作にある程度慣れてもらう。その後図 8 に示した立体の切削をしてもらう。被験者 6 名はいずれも彫刻の経験に関しては初心者であり、本研究で実装したシステムに触れるのも初めてである。図 8 右側の立体は図 8 左側の立体をそれぞれ縦、横、奥行き方向に対して一辺

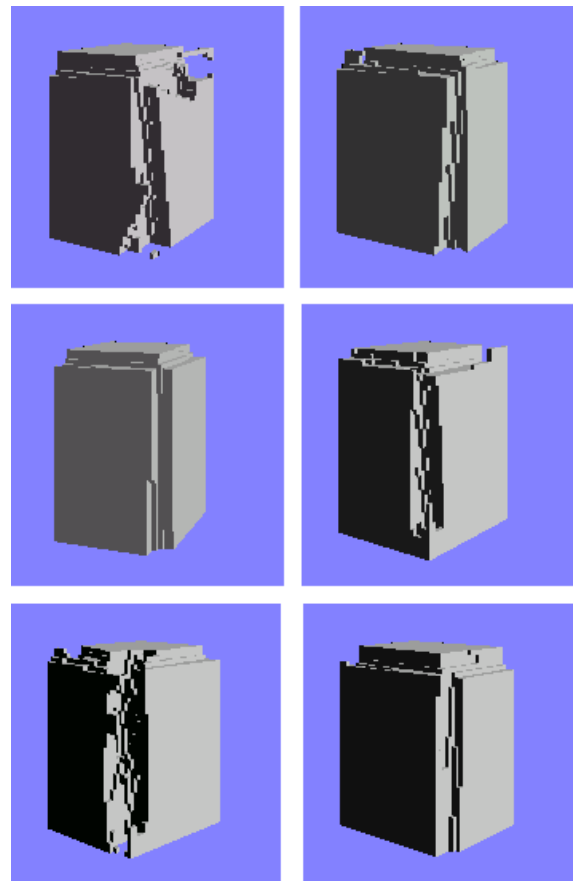


図 9: ユーザーテストにより切削した仮想彫刻

ずつの角を切削することで得られる立体である。これにより3次元空間での入力での彫刻の切削が行いやすいかを調べる、これにより得られた被験者の切削結果を図9に示す。

ユーザーテストによる結果、被験者はいずれも5分以内に切削を完了させ、意図する切削を行うことができていた。被験者からは次のような意見も出た。

- 奥行きが把握しにくい。
- 位置決めフェーズで操作が分かりにくい。
- 手首の曲がる角度に限界があり、ある程度以上の位置に持っていけない。
- 「柔らかい物を削っている」ような感覚である。

「柔らかい物を削っている」ような感覚であるというのは、雪を削っているようなイメージであると考えられる。

6. 考察

6.1 ユーザーテストと今後の展望

ユーザーテストでは全ての被験者が短時間で切削を完了した。これにより初心者でも簡単に仮想彫刻を行うことができることを示した。奥行きが分かりにくいという問題の解消方法としては、 $x-z$ 平面に対して影を落とすという手法が考えられる。また他にも切削フェーズでしか実装していないデバイスへの振動のフィードバックを位置決めにも用い、彫刻素材の角にぶつかったら振動をさせ視覚以外の方法でユーザーに位置を示す、などの方法も考えられる。6軸の情報ではYaw角の修正が行えないためこれを修正する手法の実装も検討する必要がある。また彫刻素材の回転について、本研究での実装ではキーボードのみで行っている。彫刻素材自体の回転も別のデバイスで行える方が幅が広がると考えられるので、これの実装も検討していきたい。具体的には彫刻素材の方にも3軸のジャイロセンサと加速度センサを用い6軸の情報により回転させる手法が考えられる。彫刻刀操作に関しても今後は他のセンサにも着目し、ユーザーの意図を的確に読み取り簡単に楽しく仮想彫刻を行うことができるユーザーインターフェースを考える必要がある。

6.2 他のユーザーインターフェースとの比較

3次元空間上での入力を行うユーザーインターフェースとしてSPIDAR[9][10]がある。SPIDARはワイヤーにより張られた空間の中央に球があり、それを移動させることでワイヤーの状態を変化させ、そこから3次元空間上での位置を計算する。同時にワイヤーの張力を利用しユーザーへの力覚揭示も行う。切削手法の実装はSPIDARを用いることも考えられる。これと比べたときの6軸モーションセンサの利点は場所を選ばずどこでも使用できることである。SPIDARはワイヤーの張られた空間内でしか球を動かせないのに対して、6軸モーションセンサであればそれを搭載したデバイス1つのみで完結しているため自由に歩き回りながら使うようなことも可能ではある。筆圧デバイス[7]を用いた場合も同様に、使用できる場所が2次元のディスプレイ上に制限されてしまい、自由にどこでも使うことができるわけではない。ここで挙げた他のユーザーインターフェースと本研究での6軸モーションセンサを比較すると場所を選ばず使えることに利点がある。

6.3 コンテンツへの応用

まずはボクセルモデリングとしてのツールへの応用が挙げられる。直感的な操作を利用することで初心者でも簡単にモデリングを行うことができるツールとなり得る。また本研究によるシステムではユーザーテストで「柔らかい物を削っている」という意見から、雪像を制作するようなコンテンツも考えられる。また内部の状態を記述することができるボクセル表現を生かすことを考えれば、透けている素材の表現、例えば、仮想水刻切削のようなコンテンツが考えられる。こうした仮想彫刻システムを遊びのためのシステムに応用したコンテンツの実装も検討する。

7. おわりに

以上、仮想彫刻における6軸モーションセンサによる切削手法を示した。モーションセンサとして3軸加速度センサと3軸ジャイロセンサを用いた。本研究はコンピュータによるエンタテインメントの研究として仮想彫刻のユーザーインターフェースに着目し、彫刻の3次元上での切削動作に特化した手法を提案した。フェーズを3つに分け、位置決め・角度決めフェーズではジャイロセンサの角速度のみを用い積分による誤差を抑え、切削フェーズでは加速度を用いてユーザーの切削動作を検知するようにした。また、誤差は完全に抑えることができないため、ジャイロセンサによる角速度の積分の誤差は加速度センサの重力加速度を利用して修正を行うこととした。ユーザーテストの結果では短い時間で切削動作を習得し、切削も問題なく行っていた。これにより本研究で実装したシステムは直感的な操作でコンピュータ上の仮想彫刻を行えることを示した。また、考察としてSPIDARや筆圧デバイスとの比較、どのようなコンテンツへの応用が考えられるかを述べた。

参考文献

- [1] Wii リモコン
http://www.nintendo.co.jp/wii/features/wii_remote.html
- [2] Xbox360 Project Natal
<http://www.xbox.com/en-US/live/projectnatal/>
- [3] 水野 慎士, 岡田 稔, 横井 茂樹, 鳥脇 純一郎, CSG を用いた仮想彫刻, 情報処理学会研究報告, グラフィクスと CAD 研究会報告, Vol. 95, No. 18, pp49-56, 1995.
- [4] 水野 慎士, 大河内 俊雄, 岡田 稔, 鳥脇 純一郎, 仮想版画における濃淡画像および三次元モデルに基づく仮想版木の自動生成法, 情報処理学会論文誌, Vol.40, No.9, pp3483-3492, 1999.
- [5] 岡田 稔, 水野 慎士, 鳥脇 純一郎, モデル駆動による仮想彫刻と仮想木版画, 芸術科学会論文誌, Vol.1, No.2, pp.74-84, 2002.
- [6] 脇田 航, 井門 俊, 仮想彫刻に基づくリアルタイム法線マップ生成システム, 情報処理学会論文誌, Vol.48, No.12, pp3670-3679, 2007.
- [7] 小林 大吾, 水野 慎士, 岡田 稔, 鳥脇 純一郎, 山本 眞司, 筆圧に基づく操作の強さを考慮した仮想彫刻・版画システム, インタラクシオン 2005 論文集, pp43-44, 2005.
- [8] Takeo Igarashi, Satoshi Matsuoka, Hidehiko Tanaka, "Teddy: A Sketching Interface for 3D Freeform Design

- ”,ACM SIGGRAPH 99 (Impact Paper) Los Angels, August, 1999.
- [9] 佐藤 誠, 平田 幸広, 河原田 弘, 空間インターフェース装置 SPIDAR の提案, 電子情報通信学会論文誌 D, Vol J74 - D2, No . 7, pp.887-894, 1991.
- [10] 太田智彦, 鈴木悠一, 原田哲也, SPIDAR に基づく 6 自由度空間インタフェース, 日本バーチャルリアリティ学会第 8 回大会論文集, pp.25-28, 2003.
- [11] Kim Pallister, 加藤諒, 川西裕幸, 中本浩, ”Game Programming Gems 5 日本語版”, 株式会社ボーンデジタル, ISBN4-86246-015-1, pp231-233, 2006.
- [12] Wiiyourself! NATIVE C++ WIIMOTE
LIBRARY BY gl.tter
<http://wiiyourself.gl.tter.org/>