

I-009

## 立体モデルの局所変形 Local transformation of solid model

直江 尚範<sup>†</sup> 伊藤 誠<sup>†</sup> 古川 進<sup>‡</sup>  
Hisanori Naoe Makoto Ito Susumu Furukawa

### 1. まえがき

近年、様々な3Dモデリングソフトが数多く開発され、非常に高品質で多種多様なモデリングが可能となってきた。これらは、主に直線や曲線や平面、曲面などを自由に生成・変形を行うことで成り立っている。しかし、その一方で、3Dモデルのインタラクティブな局所変形の手法については、まだ検討の余地があると言える。

本研究では、指や手のひらなどの人体の一部や、やすりやナイフなどの人が身近で使用している道具をツールとして、直感的に平面や曲面を編集可能にすることを目的とした局所変形の一手法の提案を行う。

また、このようなモデリングを行う際、試行錯誤が何度も必要であると言える。そこで、点ごとにグラフ構造でデータ管理する『クリスタル』システムを用いて、履歴管理におけるメモリ消費量の削減を行った。

さらに、より一層直感的な局所変形を行うために、マウスやキーボード以外のユーザインターフェースの使用や製作の提案を行う。

### 2. 各ツールにおける変形の定義

人が身近で使用している道具や人体の一部をツールとして局所変形を行う場合、それによる変形の定義せねばならない。

身近で使用している道具として挙げられる「やすり」や「ナイフ」においては、「物体を削る変形」となり、人体の一部として挙げられる「指」や「手のひら」においては、押し込む動作で「物体を凹ませる変形」となる。

また、「物体を削る変形」を『削り変形』、「物体を凹ませる変形」を『凹み変形』と呼ぶことにする。

### 3. 削り変形

一言で削り変形と言っても、物体を削ることができる道具はいくつもある。ここでは、例として「やすり」と「ナイフ」を取り上げる。これらはそれぞれに合った変形処理が必要であると考えられる。

#### 3.1 やすりにおける削り変形

やすりは、面をこすりつけることで物体を削り取るため、該当部分を均等に削る処理になる。そして、やすりにもいくつか種類があるが、削り方という点では共通しているので、形に沿って削りさえすれば、同様の手口で処理を行えると言える。

選択したやすり目の粗さによって削る量は変わるが、削る量さえ決めてしまえば、それとやすりの形によって得られる立体モデルとの差集合をとることで実現できる。

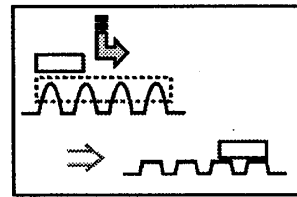


図1. やすりにおける削り変形

#### 3.2 ナイフにおける削り変形

ナイフは、2種類の変形手法が考えられる。

1つ目は、物体を両断する変形である。これは、やすりと似た手法であるが、ナイフの方向により得られる面の片側を削除することで実現できる。

2つ目は、表面を削り取る変形である。これは、一回の変形で削りの深さが変動する為、刃の角度から計算した立体モデルとの差集合をとることで実現できる。

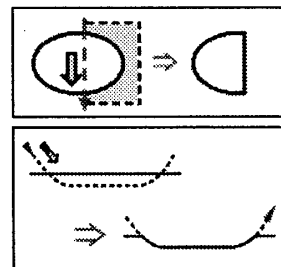


図3. ナイフにおける削り変形

### 4. 凹み変形

凹み変形とは、平面あるいは曲面を選択したツールの形で凹ませたかのように頂点および制御点を構成しなおすことである。

まず、凹み変形を行う対象とは別に、指や手のひらといった凹み変形のツールそのもののメッシュを作成しておく。そして、押し込む方向および押し込む量により得た座標における差集合をとることで実現できる。

ただ、物体の材質によっては、この処理で削られた部分の端にフィレットをかけるなど、元々の形に馴染ませる必要がある。

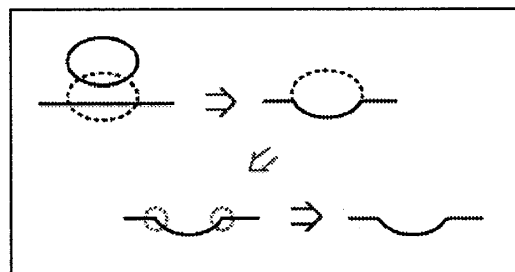


図3. 凹み変形

<sup>†</sup> 中京大学 Chukyo University

<sup>‡</sup> 山梨大学 University of Yamanashi

## 5. クリスタル構造について

### 5.1 クリスタル構造の概略

本研究で使用している『クリスタル』システムの構造は、3次元構造をグラフで表現する手法である。各頂点に隣接する点の情報を与えるだけのもので、平面・曲面分が混在する場合であっても、立体を全く同一のデータ構造で記述できる特徴を持つ[1]。これにより変形処理が柔軟に行える。

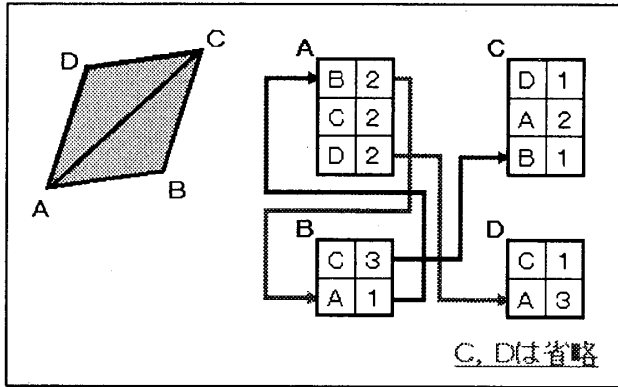


図4. 隣接点表の例

### 5.2 履歴操作

一般的な履歴操作は、1回およびそれ以上前における全ての点の情報を覚えている。それでは、多くの処理を行うと、容量が非常に大きくなってしまふ。

このことは、アンドゥ、リドゥなどの試行錯誤を繰り返しながら作業を行う『局所変形』にとっては、適切であるとは言えない。

それに対して、このクリスタル構造における履歴操作は点ごとに履歴データを管理している為、変化のない点の情報など必要のない情報を削減できるので極めて少ない容量で済む[2]。

このことは、試行錯誤を繰り返しながら作業を行う『局所変形』にとって適切であると考えられる。そこで、今回、このクリスタル構造を利用することにした。

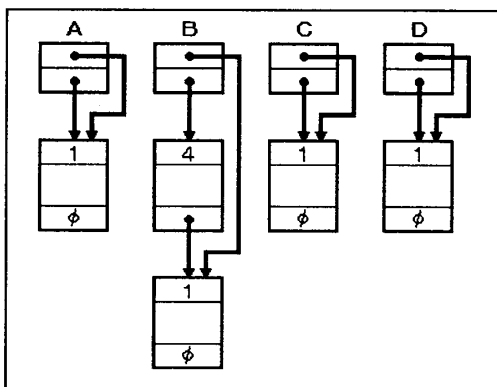


図5. 形状履歴の記憶・管理するデータ構造

## 6. 実行例

ここでは、前述に述べた処理の実行例として凹み変形処理およびナイフにおける両断する変形処理を取り上げる。

左図が凹み変形で、右図がナイフによる変形を行った実行例である。

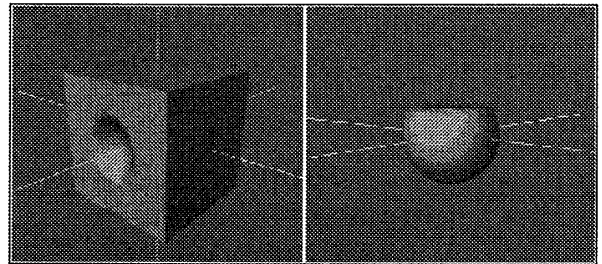


図6. 各変形の実行例

## 7. 専用ユーザインターフェースの模索

本システムにおいて、より直感的に局所変形を行うことを考えるとユーザインターフェースはマウスとキーボードだけでは力不足と言える。

ここでは、一例として特殊なセンサーを用いたナイフ型ユーザインターフェースを提案する。

このナイフ型ユーザインターフェースには、刃の3軸の傾斜を得る為に加速度センサーを、対象の物体を削り取る量を得る為に圧力センサーを、ナイフの位置を取得する為に2つの超音波センサーを、フォースフィードバックとして振動モーターを利用する。

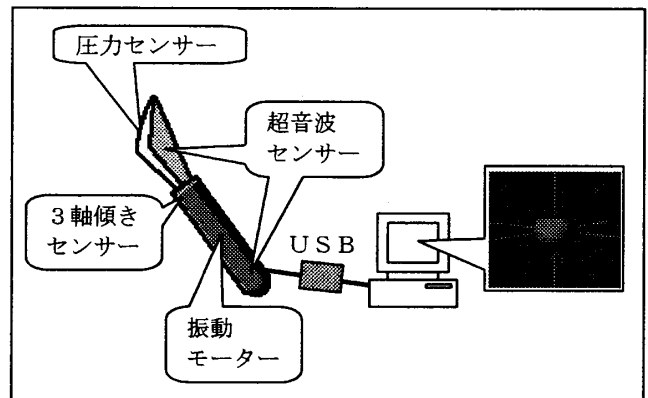


図7. ナイフ型ユーザインターフェース

## 8. まとめ

人が身近で使用している道具や人体の一部をツールとした局所変形の一手法の提案をし、ナイフ型ユーザインターフェースの提案・設計を行った。また、クリスタル構造を用いることで履歴操作における容量の削減を行った。

今後の課題としては、ツールの拡大、そして、専用ユーザインターフェースの制作とそれに合わせたソフトウェアの調整である。

### 謝辞

クリスタル構造のライブラリの提供をいただいた清水誠司様に心より感謝を申し上げます。

### 参考文献

- [1] 清水誠司, 向井伸治, 古川進. "多様体ソリッドモデル記述用の新しいデータ構造について", 日本設計工学会誌, 第42巻, 第1号, 2007
- [2] 清水誠司, 向井伸治, 古川進. "二次元・三次元図形の統一的データ表現と形状変形履歴の操作", 電子情報通信学会論文誌, VOL.J90-A, NO.10, 2007