

# 両手マウスを利用した2次元形状の操作

五十嵐 健夫<sup>1,3</sup>

Tomer Moscovich<sup>2</sup>

John F. Hughes<sup>2</sup>

<sup>1</sup> 東京大学

<sup>2</sup> ブラウン大学

<sup>3</sup> 科学技術振興機構さきがけ

## 1. はじめに

コンピュータグラフィックスの分野では、形状を編集するためのインターフェースがいろいろと研究されてきている。しかし、そのほとんどはマウス入力によって制御点を一つずつ動かすといったものであり[2][5]、実世界において両手で物を掴んで曲げたり伸ばしたりするような感覚で形状を操作することは難しかった。一部、両手を使った物体の変形手法も提案されている[7]が、あくまでも形状そのものでなく形状が埋め込まれた空間を歪ませるといったアルゴリズム[3][8][9]なので、実世界の物体を掴んで操作するのとは大きく異なる操作感となっている。

本インタラクティブ発表では、著者らが開発してきた物体の堅さを表現した2次元形状の変形アルゴリズム[6]を利用した、直感的な物体操作インターフェースのデモンストレーションを行う。特に、入力デバイスとして、両手で扱うことができ水平面上での回転も得ることのできる市販マウス（ワコム製）を利用することで、従来のマウスでの操作では得られなかった非常に直感的な操作を実現する（図1）。なお、デモソフトウェアおよびビデオは <http://www-ui.is.s.u-tokyo.ac.jp/~takeo> から入手可能である。

## 2. インタフェースの概要

本システムでは、ユーザは左右の手それぞれに位置と姿勢を得ることのできるマウスを持って、画面上の画像を操作する。画像としては、ベクトルグラフィックスで表現されたドローデータあるいは

---

As-Rigid-As-Possible Shape Manipulation using Two-handed Mice. Takeo Igarashi, Tomer Moscovich, John F. Hughes.

はビットマップで表現されたイメージデータが利用できる。画面上には、各マウスの位置と姿勢に対応したカーソルが表示されている。ユーザは、まずこのカーソルを操作して、形状の上で掴みたい場所へもっていく。次に、その場でマウスボタンをクリックすることで、物体を「掴む」ことができる。掴んだあとは、マウスの動きに追従して、物体の移動および変形が生じる。マウスのボタンをもう一度クリックすることで、物体を「離す」ことができる。なお、片手で掴んだだけの場合は、単に変形を伴わない移動と回転がおきるだけであるが、両手で掴むことで伸ばしたり曲げたりといったことが可能となる。

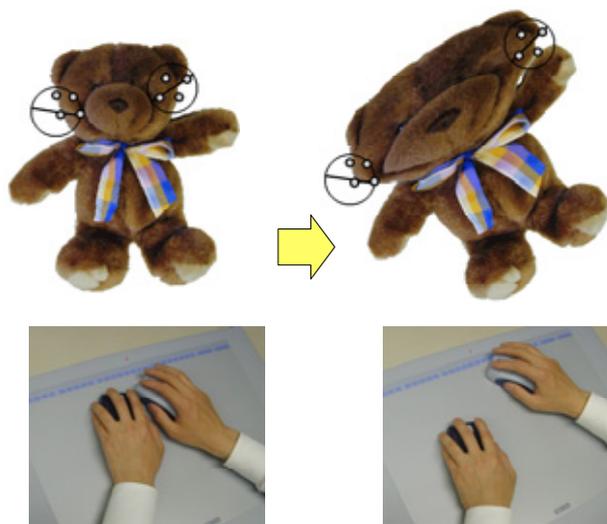


図1:ワコムの両手マウスによる操作

なお、本発表ではデモンストレーションは行わないが、両手の指の位置を直接検知することのできるデバイスを用いることで、より直接的に形状を掴んで操作することが可能となる。一例として、スマートスキン[10]を利用した実験を行っており、初心者でも様々な動き簡単に表現できることを確認している。

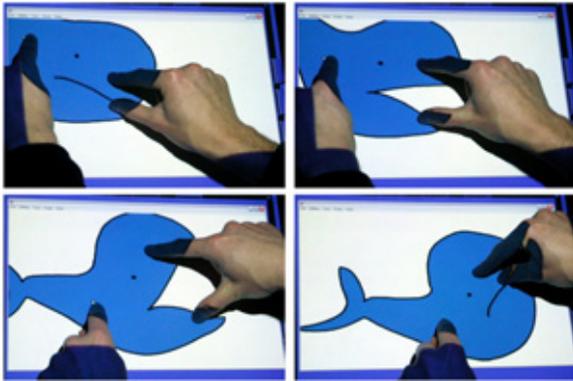


図 2: スマートスキンを利用した操作例。両手で 2 次元図形を掴んで自由に動かすことができる。

### 3. アルゴリズムの概要

操作対象である画像は 2 次元のメッシュとして表現されている。最終的に画面に表示されるのは、メッシュの変形に応じて変形された画像データである。ドローデータの場合には直接座標を計算、ビットマップデータの場合にはテクスチャマップを利用している。一方、個々のマウスカーソルは一定の半径を持った円領域として表現されている。ユーザがマウスボタンをクリックすると、そのマウスカーソルの円内にあるメッシュの頂点群が制御点群として登録される。以後、ユーザがマウスを操作すると、それに応じて制御点群の位置が決定され、最後に物体の歪みを最小にするように残りのメッシュの形状が決定される。

形状計算処理アルゴリズムは、入力として制御点の座標を受け取り、出力として制御点以外の自由頂点の座標を返す。その際に、変形前の形状と、変形後の形状の間の局所的な差異がなるべく小さくなるように計算を行う。より具体的には、三角形メッシュを構成するそれぞれの三角形の歪みの合計が小さくなるように最小化計算を行う[6]。

なお、本形状計算アルゴリズムは、従来のように空間座標の変換として形状変形を表現するのではなく、頂点の座標からなる連立方程式を解くことで形状を決定するという、近年盛んに用いられているアプローチ[1][11][12]を組み合わせたものとなっている。

### 参考文献

- [1] Alexa, M., Cohen-Or, D., And Levin, D. 2000. As-Rigid-As-Possible Shape Interpolation. In Proceedings of ACM SIGGRAPH 2000, 157-164.
- [2] Barrett, W. A., And Cheney, A. S. 2002. Object-Based Image Editing. ACM Transactions on Graphics, 21, 3, 777-784.
- [3] Beier, T. And Neely, S. 1992 Feature-Based Image Metamorphosis, In Computer Graphics (Proceedings of SIGGRAPH 92), 26, 2, 35-42.
- [4] Davis, T. A. 2003. Umfpack Version 4.1 User Guide. Technical Report Tr-03-008, University of Florida.
- [5] James, D. L., And Pai, D. K. 1999. Artdefo Accurate Real Time Deformable Objects, In Proceedings of ACM SIGGRAPH 1999, 65-72.
- [6] Igarashi, T., Moscovich, T., Hughes, J.F. As-Rigid-As-Possible Shape Manipulation. ACM SIGGRAPH 2005.
- [7] Llamas, I., Kim, B., Gargus, J., Rossignac, J., And Shaw, C. D. 2003. Twister: A Space-Warp Operator For The Two-Handed Editing of 3d Shapes. ACM Transactions on Graphics, 22, 3, 663-668.
- [8] Maccracken, R., And Joy, K.I. 1996. Free-Form Deformations With Lattices of Arbitrary Topology. In Proceedings of ACM SIGGRAPH 1996, 181-188.
- [9] Milliron, T., Jensen, R., Barzel, R. And Finkelstein, A. 2002. A Framework For Geometric Warps And Deformations. ACM Transactions on Graphics, 21, 1, 20-51.
- [10] Rekimoto, J. 2002. Smartskin: An Infrastructure For Freehand Manipulations on Interactive Surfaces. In Proceedings of Chi'02, 113-120.
- [11] Sorkine, O., Cohen-Or, D., Lipman, Y., Alexa, M., Rossl, C., And Seidel, H. 2004. Laplacian Surface Editing. In Proceedings of Geometry Processing, 179-188.
- [12] Sumner, R. W., And Popovic, J. 2004. Deformation Transfer For Triangle Meshes, ACM Transaction on Graphics, 23, 3, 399-405.