3D Freeform Design
－CyberGlove を用いたパペットのアニメーション－

佐藤 晶威† ウラジミール サブチェンコ†† 大渕 竜太郎†††
†, ††† 山梨大学 〒400-8510 山梨県甲府市武田 4-4-37
†† 法政大学 〒184-8584 東京都小金井市梶野町 3-7-2
E-mail: †g04mk012@ccn.yamanashi.ac.jp, ††vsavchen@k.hosei.ac.jp, †††ohbuchi@acm.org

あらまし 本論文は，キャラクターアニメーション等において有効な，直感的かつインタラクティブに 3 次元形状を変形させる手法について述べる．本論文で提案する手法では compactly supported radial basis function (CSRBF) を用いた高速な形状変形アルゴリズムと，CSRBF の入力装置として自由度の高い CyberGlove を組み合わせた．CSRBF を用いる事によって高速で合理的な形状変形を実現できるが，満足の行く変形を行うために CSRBF を決定するのは楽な作業ではない．そこで本手法では，両手に装着した CyberGlove の 10 本の指先からパラメータを取得し，CSRBF として適用した．結果，粘土細工をするような感覚で直感的に 3 次元形状を変形する事が可能となった．本手法で適用した変形アルゴリズムは物理的に正確な変形は生成できないが，計算の速さと妥当な形状変形を両立する事ができ，例えば，30000 ポリゴン (約 15000 頂点)の 3 次元モデルの変形を行った場合 30 フレーム/秒近くを確保する事ができた．

キーワード 3 次元形状変形，アニメーション，バーチャルリアリティ入力装置，radial basis function

3D Freeform Design
－Puppet Animation by the Use of CyberGlove－

Masatake SATO† Vladimir SAVCHENKO†† and Ryutarou OHBUCHI†††
†, †††Yamanashi University 4-4-37 Takeda, Kofu-shi, Yamanashi, 400-8510 Japan
†† Hosei University 3-7-2 Kajinocho, Koganei-shi, Tokyo, 184-8584 Japan
E-mail: †g04mk012@ccn.yamanashi.ac.jp, ††vsavchen@k.hosei.ac.jp, †††ohbuchi@acm.org

Abstract This paper presents an approach to realize a tool for interactive and intuitive deformation of 3D shapes, to be used, for example as a character animation tool. The approach combines a fast algorithm for 3D free form deformation that employs compactly supported radial basis function (CSRBF) with a high degrees-of-freedom input device of hand joint angles, namely, the CyberGlove. While shape deformation algorithm by using CSRBF allows for fast and reasonable shape deformations, it is not a simple task to position the CSRBF so that desired shape deformations can be created. In our approach, the 10 CSRBF are attached to the tip of the fingers of the CyberGlove worn on both hands of the users. This allows the user an intuitive manipulation of 3D shapes in a manner somewhat similar to clay modeling. While the shape deformation algorithm does not produce physically accurate deformation, the deformation is fast, plausible and quite useful. For example, using a 32k polygon mesh, a deformation involving all the 15k vertices can be produced at about 30 frames-per-second.

Keyword 3D shape deformation and animation, virtual reality input device, radial basis functions

1. 序章

3 次元形状の生成と編集は，現実世界で行うか (彫刻など)，仮想世界で行うか (形状CADソフトなど)を問わず，スキルと手間のかかる複雑な作業である．近年，現実の物体の形状をレーザースキャナ等で取り込む事が多いが，大抵の場合取り込んだデータの編集に手間がかかる．

近年，形状CAD等の仮想モデリングツールとバーチャルリアリティ (VR)によるユーザーインターフェースの統合が試みられている．例えば[1]では，指の関節角と手の空間上での位置情報を獲得する装置であるデータグローブを用い，3 次元形状をリアルタイムで操作する．同システムではニューラルネットワークを用いて手のジェスチャーを認識する事でモデリングのための命令を入力する．しかし，定義されたジェスチャーが人間にとって不自然なため，オペレータの訓練が必要である．

別の解決策として，必要な操作個々に対して「タンジブル(触る事のできる)な」実態をインターフェースとして使うアプローチがある．例えば Schkhole らのシステム[1]では，データグローブの様なデバイスに加え，トング，ボール等の実体を持ち，かつ操作に直接的に対応する道具を用いた．タンジブルなインターフェースは，物に触れたりつかんだりする感覚を提示する．しかし物の形は固定されているため，例えば，形状変形をする工具を触る事ができても，変形対象となる形状そのものを触る事はできない．また，変形や彫刻に伴うはずの触覚や力覚も表現できない．

作業に伴う力を触覚や力覚に提示するアプローチも考え

られる. 例えば機械のアームを用い, 仮想空間上における物体と 1 点との接触を力として表示できるデバイス Phantom[3]がある. FreeFormConcept [4]は Phantom を効果的に用い, 造形する 3 次元形状を「触れ」つつ 3 次元形状モデリングのできるシステムである. また, Baxter らの 2 次元絵画システム[5]では, 仮想空間上のキャンパスに油絵の具で絵を描いた際に生ずるであろう絵筆の変形や絵の具の粘性などの力を物理シミュレーションし, これを Phantom で表示した. しかしこれらの力覚表示デバイスはある 1 点での接触による力の表示しかできない. 例えば, 多点での接触, さらには線や面のような広がりのある接触で生ずる力を表示する事は, 今日の力覚表示装置では大変困難である.

別の解決策として, 物理的な変形モデルを基に現実的な形状変形をシミュレーションする手法がある. 単に見かけの形だけから物理シミュレーションを行っても「それらしい」変形はできない. 例えば顔の表情のアニメーションでは, 解剖学的, 生理学的に正しい筋肉, 脂肪, 頭蓋骨, 等のモデルが必要である. このアプローチでは, 計算時間がかかる事が多いものの,「自然な」変形が得られる. [6, 7, 8]が典型的な例である. 筋肉, 質点のバネモデル, 有限要素法[9]は, モデルを基にした変形アニメーションにおいて最も多く利用される手法である. この手法の主な問題点は, モデルやその要素に対して妥当なパラメータを見つける事が難しい点である. 例として, 筋肉の弾力性を再現する場合, 個人差や疲労による違いを考慮したパラメータを見つける事は困難である. また, 写実的な変形を行うには実時間を越える計算時間がかかる場合が多いのも問題である.

写実的な形状と変形のモデルはそれ独自の長所を備えてはいるが, 1 方で物理的には正確でない形状やその変形のモデルも重要で, 現に広く用いられている. これは例えば, それなりのアーティストが操作すれば, ゴムでできた人形の「非現実的」な変形をする顔からも生き生きとして魅力的な表情が生れる事からも分かるであろう. 2 次元セルアニメーションの場合もそうであるように, ある意味で, 変形が「非現実的」であるがゆえに魅力的な, 迫力のある, あるいは愛らしい動きや表情が創り出される事も多いのである. こうした非現実的変形には, 写実的モデルに比べて計算量が少ない, という現実的な利点もある. この利点は, プレイステーション等のゲームコンソールのように計算能力が限られており, かつ実時間の形状変形応答が要求されている場合に非常に重要となる.

本論文では VR デバイスとノンリアリスティックな形状変形アルゴリズムとを統合した形状変形の手法を提案し, その結果について考察する. 本研究で用いた形状変形アルゴリズムは CSRBF を用いてスペースマッピングを高速に行うもので, 物理的な正しさよりも実行速度に重点をおいている. また VR デバイスには CyberGlove[12] を用いている. CyberGlove は仮想空間上の物体と多点での接触が行えるため, 例えば人形の頬が膨れるような, 複雑な形状変形を容易に生成できる. 形状変形の際に CSRBF が必要なパラメータは, CyberGlove から取得したデータを加工して生成する. 本論文で提案する手法の長所は以下の 2 点である.

- CyberGlove を用いる事で多点接触による形状変形操作を実現し, 直感的かつインタラクティブな 3 次元形状の操作を可能にした.
- 有限の台を持つ Radial Basis Functions (CSRBF)による高速かつシンプルな形状変形アルゴリズムを利用した. 形状変形の際の変位量の場を決定するため, CSRBF を両手指の先端に割り当てた.

本論文では, CyberGlove のデータから求まる指先の 1 データから形状変形への対応付けの方法を 2 種類提案する. この対応は, 直感的, 自然な変形操作を実現するために重要である.

本章以降の各章では, 2 章で開発したシステムの概要, 形状変形のアルゴリズム, CyberGlove を用いた形状変形の操作を, 3 章ではシステムの実装, 実行例, 性能評価の結果を, 第 4 章では結論と今後の課題について述べる.

2. アルゴリズム

2.1. 概要

人間の手指の位置とその変化から変形の向きや大きさを求め, これを CSRBF を用いた高速な形状変形で実時間で実現するのが本論文で提案する形状アニメーションの手法である. 本論文で提案するシステムでは, 操作者は, 指の角度を測定する機能を持つ CyberGlove を両手に装着し, 仮想空間内の物体の変形を行う. 刻々計測される手の位置の時間差分から形状変形の向きや大きさを計算するが, この方法として, 本論文では 2 種類を提案する. 以後, この 2 種類の計算法を Type-1, Type-2 と呼ぶ (2.3.1 と 2.3.2 を参照). 仮想空間上において人間の手の動きを再現する必要があったため, 本研究では独自に簡単化した手の骨格である VR Hand を作成した. VR Hand は, CyberGlove から関節角のデータを取り込み, 手の各指関節の座標データを算出して出力する (2.2 を参照).

Type-1 は, CSRBF に与えるパラメータであるベクトル (コントロールベクトル)の始点を, 形状変形の最初に手の姿勢の指先座標から取り込む. 終点については手の姿勢が変化する度に更新する. 1 方 Type-2 は, 現時点とその直前の手の姿勢から指先の変位量を求め, コントロールベクトルとする. CyberGlove からは新たな関節角のデータが 1 定時間毎に出力されるため, コントロールベクトルの始点, 終点は手の動きに合わせて動的に変化する. Type-1, Type-2 ともに CSRBF を用いた形状変形アルゴリズムを基にしている. 形状変形アルゴリズムは下記の 5 段階の処理によって構成

される. (2.4 を参照)

- コントロールベクトルの始点群の定義とソート
- 各コントロールベクトルの終点を決定し，変位量を計算
- 変位量を基にコントロールベクトルごとに RBF 連立方程式を構築
- 連立方程式を解いて RBF の式の未知係数 α_i を決定
- 変位量の評価

2.2. VR Hand による変位量の生成

本章では，CyberGlove，VR Hand について述べる．VR Hand は CyberGlove からの入力データを基に手の姿勢データを生成して出力する．

2.2.1. CyberGlove

CyberGlove は手袋状の VR デバイスで，1 定時間毎に手の各関節の曲げ角を計測する．本研究では手の姿勢情報の取得に Virtual Hand Suite 2000[13，14]を使用した．Virtual Hand Suite 2000 に含まれる Virtual Hand Toolkit (VHT)は，C++言語のライブラリであり，CyberGlove を用いたアプリケーションを開発する際に必要な，様々な関数を提供する．

2.2.2. VR Hand

VHT は，CyberGlove から取得したデータを基に仮想空間内の手を動かす機能を実装しているが，この機能は OpenGL を利用して開発されている．特定のグラフィックスライブラリに依存した機能は，他のグラフィックスライブラリからの利用が困難になる可能性がある．そこで本研究では，特定のグラフィックスライブラリに依存しない VR Hand を作成した．作成した VR Hand は，手のスケルトンデータ，VHT を用いて CyberGlove から指の曲げ角情報を取得する関数，曲げ角を適用したスケルトンデータから各関節の座標値を計算し，出力する関数の 3 つの要素から構成されている．

VR Hand は C++言語のみを使用して作成し，また出力するデータも数値のみとした．従って様々なグラフィックスライブラリを用いて出力結果を可視化することができる (Figure1 (a)に例を示す)．なお，本研究で開発したシステムでは，”Visualization Tool Kit” (VTK)を用いて VR Hand からの出力データを可視化した．

作成した VR Hand は手のスケルトンをデータとして予め持っている．スケルトンデータはリスト構造で定義し，3 種類のノードを連結して構成した (Figure 1 (b)参照)．1 種類目は手首の部分 (Parent)，2 種類目は手のひらの付け根の間接 (PalmJoint)，そして 3 種類目はそれ以外の指の関節および指先 (Joint)である．VR Hand では，手首から指先方向にかけて手の姿勢データを計算する．これは親指の PalmJoint および Joint (ID: 3，7，11，15)が 2 自由度となっているためである．また，手の姿勢はスケルトンの初期形状に CyberGlove からの角度データを適用して計算するため，スケルトンデータは初期形状のまま不変である．従って長時間 VR Hand を稼動させたり，極端にフレームレートが落ち込んだりした場合でも手の姿勢と出力される姿勢データの同期を取る事ができる．

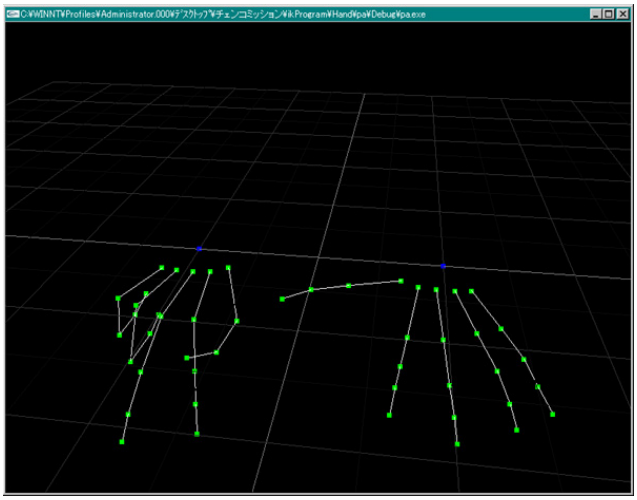

Figure 1 (a): VR Hand からの出力データを，OpenGL を使って可視化した例

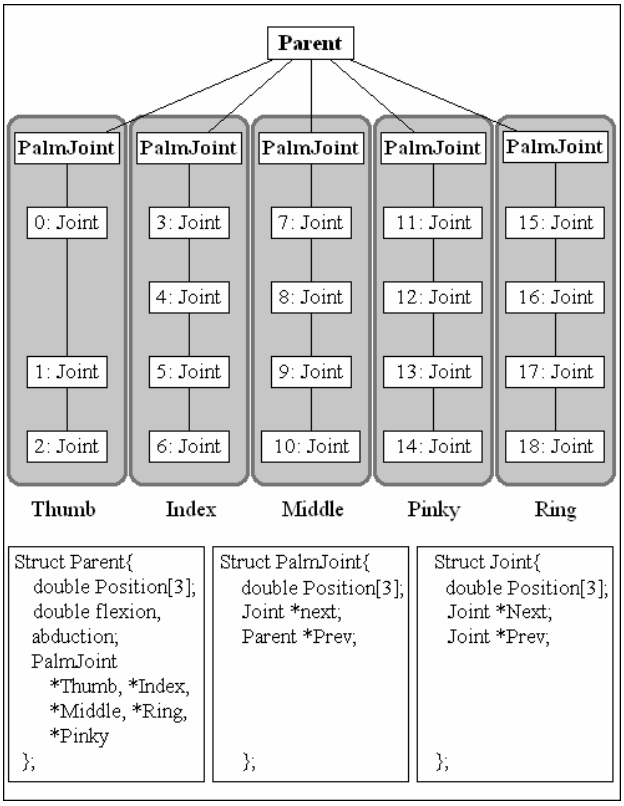

Figure 1 (b): スケルトンモデルのデータ構造

2.3. 2 種類の形状変形

直感的な形状変形を実現するためには，CyberGlove からのデータと形状変形の対応付けが重要である．本研究では Type-1，Type-2 の 2 種類の形状変形の操作法を作成した．Type-1，Type-2 共に CSRBF 形状変形アルゴリズム (2．2 参照)を基にしているが，コントロールベクトルの生成において手法が異なる．

2.3.1. Type-1 形状変形

Type-1 は CSRBF 形状変形アルゴリズムと同様，ベクトルに沿った形状の変形を生成するモジュールである．Type-1 では基本的にコントロールベクトルの始点 $\vec{s_i}$ は固定のまま，終点 $\vec{d_i}$ のみが VR Hand からの座標データの更新によって動く．$\vec{d_i}$ が変化する度にある初期形状に対して CSRBF を用い，形状の変形を生成する．始点 $\vec{s_i}$ はオペレータが任意の手の姿勢を決めた時点で自由に固定することができ，またモデルの初期形状もオペレータが任意の変形形状を決定した時点で更新することができる．Type-1 の利点は以下の 3 点にある．1 点目は始点ベクトルを更新しない限りソートした始点を使いまわせるため，CSRBF 形状変形アルゴリズムの第 1 段階を省略できる点，2 点目は連立方程式の左辺の疎行列部分を使いまわせる点，そして 3 点目は初期形状の更新頻度が低い点である．終点 $\vec{d_i}$ の生成法以外の点では，Type-1 は CSRBF 形状変形アルゴリズムと同等である．以下の 4 段階の処理を経て変形を生成する．

- 各 RBF について連立方程式の右辺を再構築．但し始点を決定した直後の場合，右辺，左辺全てを再構築
- 未知係数について連立方程式を解く
- 変位量の評価
- 初期形状を更新して第 1 段階で連立方程式全てを構築し直す，または終点のみを更新して第 1 段階へ

2.3.2. Type-2 形状変形

Type-1 は非常に安定した形状変形を実現できるが，1 方で，線分に沿って形状を変形するため，複雑な変形を行うには不向きである．そこで，線分ではなく任意の曲線に沿った形状変形を近似的に行う Type-2 を作成した．Type-1 モジュールと同様，Type-2 も CSRBF 形状変形アルゴリズムを基にして作成した．

Type-2 は以下の要領に沿って開発した．初めに，始点 $\vec{S_i}$ と終点 $\vec{D_i}$ を決定し，それらを通る任意の曲線を定義する．次に 1 定の距離間隔で曲線上から点を取り出す．そしてこれらの点を始点から終点方向に隣同士つなぎ合わせ．最後に各線分をコントロールベクトルと見なし，$\vec{S_i}$ から $\vec{D_i}$ にかけて CSRBF 形状変形と初期形状の更新を繰り返す．Type-2 ではコントロールベクトルを鎖のようにつなぎ合わせる事で，任意の曲線に沿った形状変形を近似的に行っている．形状変形の際は $\vec{S_i}$ から始まり，$\vec{D_i}$ に辿り着くまで以下に示す 5 段階の処理を繰り返す．

- コントロールベクトルの始点，終点を更新
- 各 RBF について連立方程式を構築
- 未知係数について連立方程式を解く
- 変位量の評価
- 初期形状を更新し，第 1 段階へ

2.4. CSRBF 形状変形アルゴリズム

本研究では Kojekine らによって確立された形状変形のアルゴリズム[10]を利用した．このアルゴリズムは空間を歪める事で，その内部に存在する物体の形状も変形させる．CSRBF の有効範囲は変形する空間の範囲を，CSRBF の中心からの距離は変位の重みを決定する．

CSRBF による多変量補間を利用する事で，$\mathbf{R}^3 \rightarrow \mathbf{R}^3$ への写像が可能となる．例えば，離散的な点 (コントロールポイント) $X = \{\vec{x_1},....,\vec{x_N}\} \subseteq \mathbf{R}^3$ と対応する変位量 g_1，…g_N が与えられたとすると，これら変位量を基にしてコントロールポイントの周囲の領域を滑らかに補間する関数が与えられる．

$$Sg, X(\vec{x}) = \sum_{i=1}^{N} \alpha_i \phi(\left\| \vec{x} - \vec{x_i} \right\|) + p(\vec{x}) \tag{1}$$

上式で，||･||は $\mathbf{R}^3$ におけるユークリッド距離を表す．また $p(\vec{x})$ は ϕ に含まれない低次項である．係数 α_i および低次項は以下の条件によって決定される．

$$Sg, X(\vec{x_i}) = g_i, \quad 1 \le i \le N, \tag{2}$$

$$\sum_{i=1}^{N} \alpha_i q(\vec{x_j}) = 0 \tag{3}$$

[10]では，正定値かつ有限の台を持つ radial function[9]が用いられている．

$$\phi(r) = \begin{cases} \psi(r), & 0 \le r \le 1 \\ 0, & r > 1 \end{cases} \tag{4}$$

Ψ は 1 変数の関数で，r は台の大きさ r （有効半径)を表す．

ここで，$\mathbf{R}^3$ において始点群 $\vec{s_i} = \{x_s^i, y_s^i, z_s^i\}$ とそれに対応する終点群 $\vec{d_i} = \{x_d^i, y_d^i, z_d^i\}$ を定義し，$\vec{s_i}$ を始点 $\vec{d_i}$ を終点としたベクトル（コントロールベクトル)を生成する．さらに低次項 $p(\vec{x})$ を 1 次式 $p(\vec{x}) = \beta_0 + \beta_1 x + \beta_2 y + \beta_3 z$ とおくと，コントロールベクトル，式 (1)と (4)，条件 (2)と (3)を用いて未知係数 α_i, β に関する連立方程式を構築する事ができる．Φ は限定された領域内のみで 0 以上の値を持つ RBF なので，連立方程式の左辺には疎行列部分が現れる．解 α_i, β を決定する事により，空間上の任意の点で CSBRF を用いて変位量を求める事ができる．形状変形を計算するアルゴリズムは下記の 5 段階から成る．

- コントロールベクトルの始点群の定義とソート
- 各コントロールベクトルの終点を決定し，変位量を計算
- 変位量を基に各コントロールベクトルについて RBF 連立方程式を構築
- 連立方程式を解いて RBF の式の未知係数 α_i を決定
- 変位量の評価

ソートの段階では，Octree 構造[15]を用いてコントロールベクトルの始点同士の隣接関係を調べ，始点群を整列し直

したリストを作成する. 行列 A を構築する際, 作成したリストの順序に従って頂点を並べる事で対角行列状の疎行列を構築する. 連立方程式の構築段階では左辺の 1 部として疎行列 A を使用するが, A が対角行列状の疎行列である事には 2 つの利点がある. 1 点目は, 行列を保存するためのメモリの使用量を抑えられること, そして 2 点目は部分ガウス消去法とコレスキー分解[14]を組み合わせる事で高速に解の計算ができる事である. 変位量の評価の段階では, 解 $\alpha_i \beta$ を基に, 空間上の任意の点での変位量を求める. 上に示した 5 段階の処理を CSRBF 形状変形と呼ぶ事とする.

3. 実装

開発したシステムの構造を Figure. 2 に示す. 実装には VHT, VTK, CSRBF 形状変形アルゴリズム[10]を用いた.

3.1. システムの構造

本研究で開発したシステムは CSRBF 形状変形アルゴリズムを核として, VR Hand, Type-1, Type-2 を統合したものである. Figure 2 に示すとおり, Type-1, Type-2 共に CSRBF 形状変形アルゴリズムを部分的, または全体を反復的に利用している. VR Hand は CyberGlove から角度データを取得し, スケルトンデータの姿勢を計算して出力する. Type-1, Type-2 は指先 10 本分の座標値を VR Hand の出力から取り出し, コントロールベクトルの定義に利用する. Type-1 では任意のタイミングでモデルの初期形状を更新できるが, Type-2 ではコントロールベクトルの更新に合わせて初期形状を自動的に更新する.

3.2. 実行例と実行速度

Figure 3 に人間の顔のモデル (1355 ポリゴン, 689 頂点) の変形例を, Figure5 と Figure 6 にモアイのモデルを用いた Type-1, Type-2 の変形例を示す. 顔およびモアイのモデルでは, 両手指の動きから変位量を生成し, それらをコントロールベクトルとして 10 の CSRBF に割り当てた. 有用な形状変形を生成するには CSRBF の半径の最適化だけでなく, オペレータの技術と訓練が必要である.

また, Type-1, Type-2 による形状変形の実行速度も計測した. 各辺 50 の長さの平面モデルを用い, 平面上に 10 個の CSRBF を配置した. 各 CSRBF に対してはコンピュータ内部で動的に生成したコントロールベクトルを割り当てた. CyberGlove からのデータ入力は行わなかったが, これには 2 つ理由がある. 1 つ目は人間の手の動作では計測の度に挙動が微妙に違ってしまう事, そして 2 つ目は CyberGlove の転送速度が遅い事である. CyberGlove は RS-232C インターフェースによってホストコンピュータと接続されているため, ホストコンピュータが高速な場合計測結果を悪くする要因となる. なお, 計測は CPU に Athlon64 3400+, グラフィックスボードに Nvidia Quadro4 380 xgl を用いて行った.

平面モデルのポリゴン数 (頂点数)と CSRBF の有効半径を変化させて計測した結果を Table 1 に示す. この結果から, 32768 ポリゴンのモデルで, CSRBF の有効半径を 80 (全てのコントロールベクトルがモデルの全ての頂点を含む)という条件でも 28 フレーム/秒を保てる事が分かった.

Figure 4 には変位量の計算回数と 1 フレーム当たりの実行時間の関係を示す. 表より, 有効半径が同じ場合, Type-2 の方が Type-1 よりも実行時間がかかっている事が分かる. これは頂点の変位量の計算よりも連立方程式の構築に時間がかかるため,連立方程式の構築が頻発する Type-2 の方が実行時間がかかるためである.

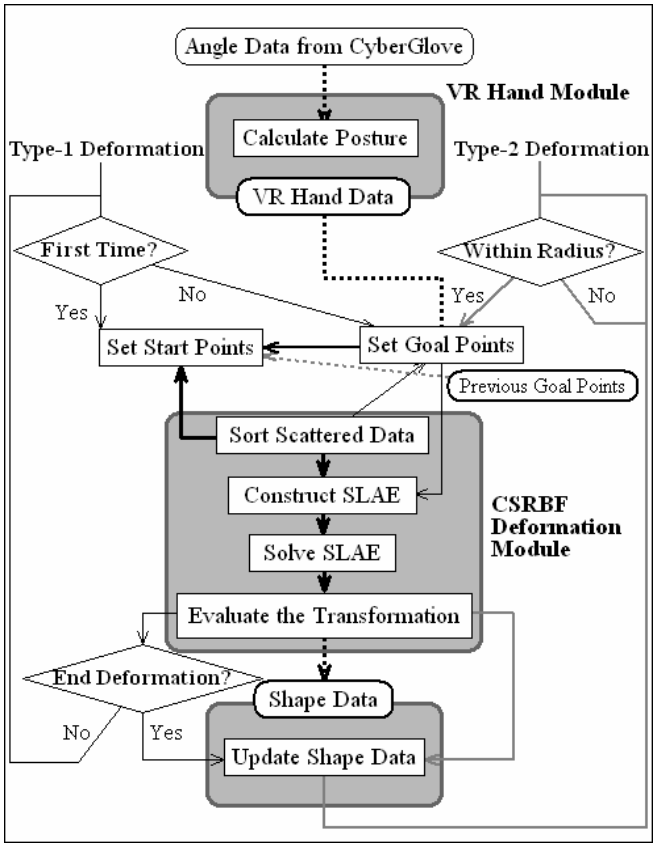

Figure 2: 開発したソフトウェアの構造. 実線は処理の流れを, 点線はデータの流れを表す.

4. 結論と今後の課題

本研究では多点接点の VR デバイスと高速な形状変形アルゴリズムを組み合わせ, 3 次元形状変形を直感的かつインタラクティブに操作する手法を提案した. 指先の位置の時間差分からコントロールベクトルを生成し, コントロールベクトルから CSRBF を用いて形状の変形を行う.

本研究では多点接触型の VR デバイスを用いる事で, 粘土細工をするような感覚で直感的に, 3 次元形状の変形を操作できる事を示した. また, 非常に高速な変形アルゴリズムを利用した事で, ポリゴン数の多い複雑なモデルをリアルタイムで変形させる事ができた. 以上の 2 点より, 本手法はゲーム等のリアルタイム性を重視したアプリケーションでの利用が可能であろう.

しかし改良すべき点も見受けられる. 1 点目は, 形状変形に両手の指先のみを使用するため, 手の動きの自由度を操

作性の向上に活かしきれていない事である．より多くの CSRBF を追加し，指先だけでなく手全体で形状変形を操作できるようになれば，手の動きをより直接的に仮想空間上の物体に伝えられ，操作者と物体との結び付きを強められるだろう．2 点目は，本システムが形状表面の変形だけしか行えない事である．同じシステム内で物体の切断やつなぎ合わせといった形状自体の編集もできた方が便利である．これらの機能を追加する事で，モデリングツールとしての機能が向上するだろう．3 点目は，全ての指で CSRBF の有効半径が同じため，詳細な形状の操作が難しい事である．複数の有効半径の CSRBF を組み合わせる事で，より精密な形状変形が可能になると考えられる．4 点目は，ポリゴン同士の衝突判定を一切行わないため，変形操作の仕方によってはメッシュの自己交差が発生する事である．アニメーションツールとして自己交差が発生しないような方法を考える必要がある．5 点目は，現在のシステムにインバースキネマティクスやキーフレームアニメーション等の手法を追加し，単なる形状の変形だけでなくアニメーションの編集にも利用できるシステムとして発展させる事である．

上記の改良を加える事で，pose space deformation[17]とは全く異なる理念の基，アニメーションやモデリングのためのシステムとして活用できるものと考える．

文献

[1] S.He.Y.Kawamura, K.Tanaka, N.Abe, J.Zheng, and H.Taki, A Virtual Reality System for Exterior Design, In proceedings The 17th International Conference on Artificial Reality and Tele-existence, Dec.3-5, Tokyo, 1997, 193-199

[2] Steven Schkolne, Michael Pruett, Peter Schröder, Surface Drawing: Creating Organic 3D Shapes with the Hand and Tangible Tools, Proc.CHI 2001.

[3] Phantom, SensAble Technologies.

[4] FreeForm Concept system, SensAble Technologies.

[5] W.Baxter, V.Scheib, M.Lin, and D.Manocha, DAB: Interactive Haptic Painting with 3D Virtual Brushes, Proc.ACM SIGGRAPH 01, August 2001, pp.461-468.

[6] N.Magnenat-Thalmann, N.E.Primeau, and D.Thalmann, Abstract muscle actions procedures for human face animation.The Visual Computer, 3, 1988, 290-297.

[7] Y.Lee, D.Terzopoulos, and K.Waters, Realistic modeling for facial animation.Computer Graphics, 29, (1995, 55-62.

[8] D.Terzopoulos and; K.Fleischer, Modeling inelastic deformation: Viscoelasticity, plasticity, fracture, Computer Graphics, 22 (4), Proc.ACM SIGGRAPH'88 Conference, Atlanta, GA, August, 1988, 269-278

[9] D.C.Zienkiewicz and K.L.Taylor, The Finite Element Method, Butterworth-Heinemann, 2000.

[10] N.Kojekine, V.Savchenko, M.Senin, I.Hagiwara, Real-time 3D Deformations by Means of Compactly Supported Radial Basis Functions, Short papers proceedings of Eurographics EG2002, ISSN 1017-4565, 2002, 35-43.

[11] H.Wendland, Piecewise Polynomial, Positive Defined And Compactly Supported Radial Functions of Minimal Degree.AICM, 4:389–396, 1995

[12] Immersion Corporation, 2004, www.immersion.com

[13] Virtual Technologies, VirtualHand Suite V1.0 User's guide, 1999.

[14] Virtual Technologies, VirtualHand Suite V1.0 Programmer's guide, 1999.

[15] H.Samet, The Design and Analysis of Spatial Data Structures, Addison-Wesley Pub Co., 1986.

[16] A.George, J.W.H.Liu, Computer Solution of Large Sparse Positive Definite Systems, Prentice-Hall, Englewood Cliffs, NJ, 1981.

[17] J.P.Lewis, M.Cordner, N.Fong, Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-Driven Deformation, Proc.of SIGGRAPH 2000, 165-172, 2000

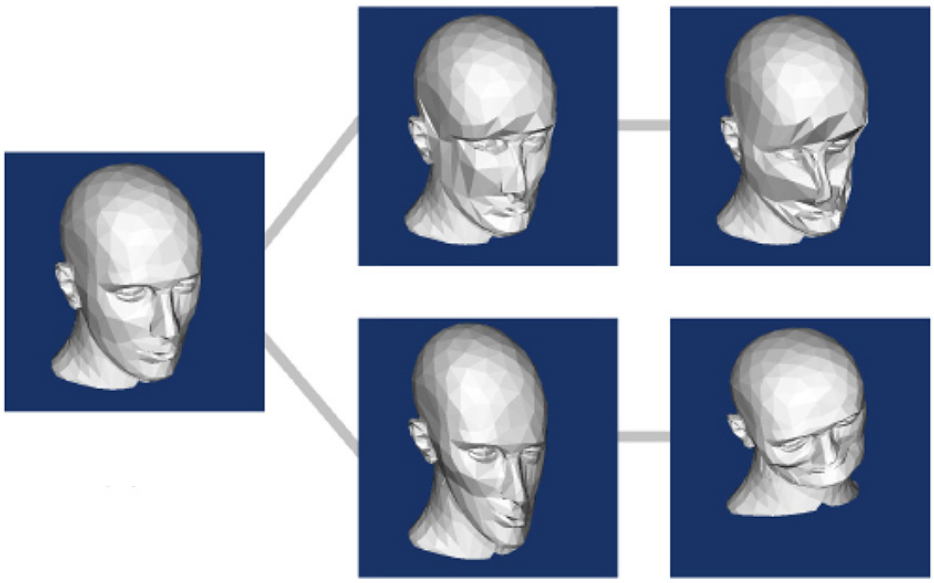

Figure 3: 顔のモデルにおける変形終了後の形状．上の段は Type-1，下の段は Type-2 による形状変形の結果である．

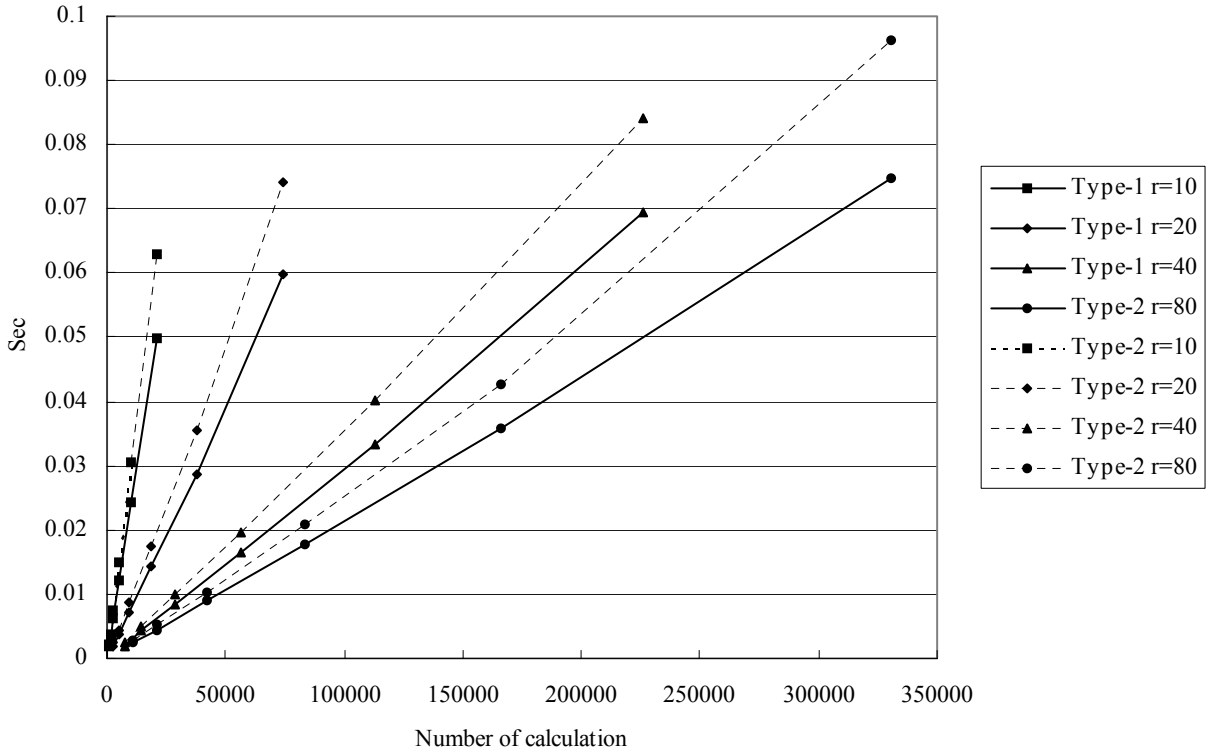

Figure 4: Type-1, Type-2 における 1 フレーム当たりの実行時間と計算回数の関係を, 半径 r が 10, 20, 40, 80 の場合についてそれぞれグラフ化した (詳細な結果は Table 1 を参照).

r / Models	Type-1 deformation					Type-2 deformation				
	10	20	40	60	80	10	20	40	60	80
N_p 2048 N_v 1089	496.3 (698)	502.1 (2414)	499.2 (7342)	427.4 (10124)	423.6 (10890)	446.3 -	427.6 -	380.2 -	367.0 -	364.0 -
N_p 8192 N_v 4225	162.6 (2646)	138.0 (9526)	119.7 (28686)	113.1 (39476)	112.0 (42250)	132.6 -	114.4 -	101.5 -	97.5 -	96.1 -
N_p 32768 N_v 16641	41.3 (10510)	34.8 (37562)	30.0 (113282)	28.2 (115812)	28.0 (166410)	32.9 -	28.2 -	24.9 -	23.9 -	23.4 -
N_p 65536 N_v 33025	20.0 (20790)	16.7 (74482)	14.4 (225714)	13.6 (309476)	13.4 (330250)	15.9 -	13.5 -	11.9 -	11.4 -	10.9 -

Table 1: *r*は CSRBF の有効半径, N_p はモデルのポリゴン数, N_v は頂点数を表す. 1 辺 50 の平面モデルを, 10 個のコントロールベクトルによって変形させた場合のベンチマーク結果. 有効半径を 10 から 80 まで変化させて計測を行った. 各セルの上の値はフレーム/秒を, 下の数字はモデル中の頂点に対して変位を計算した回数を示す (変位した頂点の数ではない事に注意). 例えばポリゴンモデルの 1 つの頂点が m 個のコントロールベクトルの有効半径内にあるとすると, 各コントロールベクトルからの変位量を計算することになり, この頂点に対しては m 回変位を計算する.

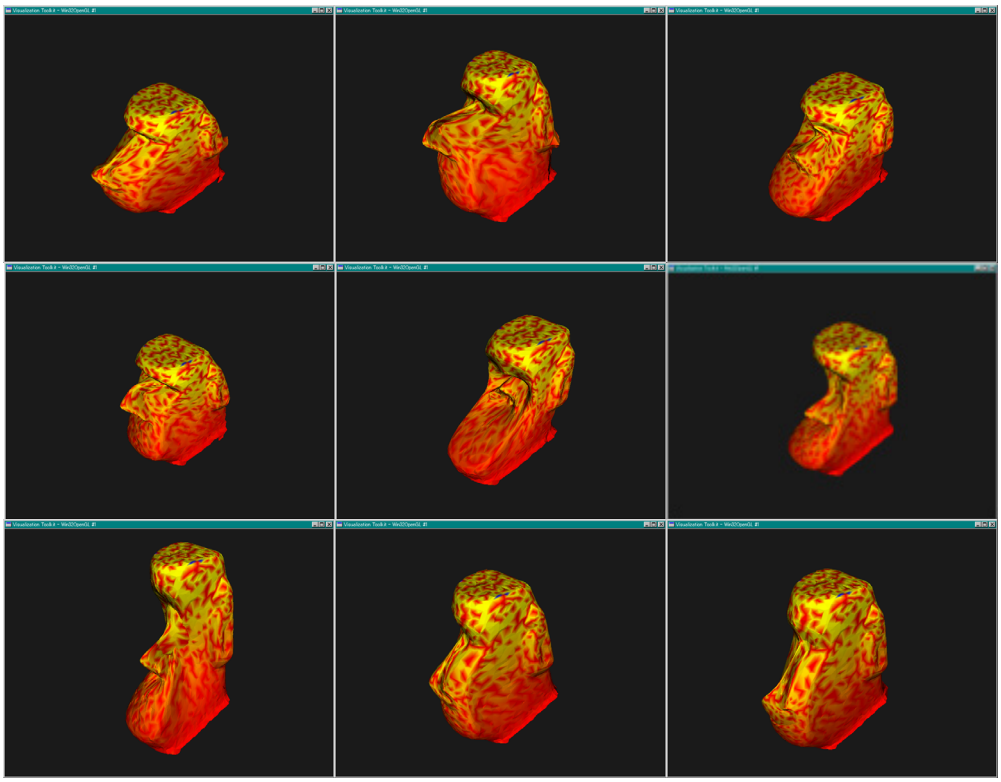

Figure 5: モアイのモデルを用いた Type-1 形状変形の例（7152 ポリゴン）.

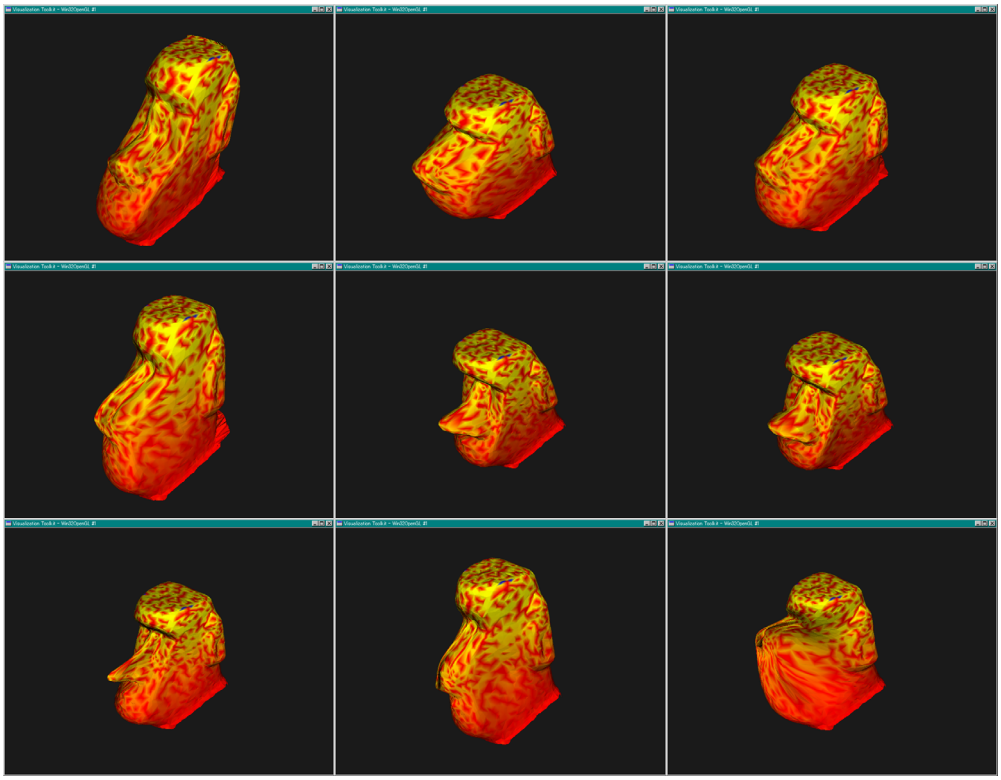

Figure 6: モアイのモデルを用いた Type-2 形状変形の例（7152 ポリゴン）.