



東京大学大学院
工学系研究科

修士論文

赤外線画像を用いた視体積交差法による
把持形体の認識

Grasp Recognition with Volumetric Fitting
using Infrared Images

指導教官 池内 克史 教授

平成14年8月

提出者

所属	電子情報工学
学籍番号	37-06391
氏名	橋本 謙太郎

目次

第 1 章	序論	1
1.1	背景	1
1.2	目的	2
第 2 章	関連研究	5
2.1	人間の手動作	5
2.2	接触型・非接触型インタフェース	7
2.3	カメラシステム	7
2.4	本論文の位置づけ	8
第 3 章	手のモデル	10
3.1	骨格モデル	10
3.2	骨格モデルの運動	13
3.3	表面形状モデル	17
第 4 章	手の 3 次元形状計測	20
4.1	ビジョンベースの手形状計測	20
4.2	Octree による形状再構成法	21
4.3	Voxel モデルのメッシュ表現	22
4.4	メッシュの simplification	23
4.5	復元データ作成の流れ	24
第 5 章	手の形状推定	26
5.1	位置姿勢推定	26

目次

5.2	3DTMの原理	27
5.2.1	対応点の計算	27
5.2.2	位置姿勢の決定	27
5.2.3	ロバストな位置姿勢の決定	28
5.2.4	位置姿勢の最適化	30
5.3	提案手法の原理	30
第6章	手形状推定実験	34
6.1	形状推定の流れ	34
6.2	カメラキャリブレーション	36
6.3	モデルの位置姿勢を固定した場合の精度評価	38
6.3.1	繰り返し回数の違いによる推定精度の比較	38
6.3.2	Voxelモデル階層数の違いによる推定精度の比較	38
6.4	手の位置姿勢が未知の場合の推定	40
6.5	手のトラッキング	41
6.6	把持物体を考慮した手のトラッキング	50
6.7	実画像を用いた実験	50
6.7.1	実験システム	50
6.7.2	赤外線カメラ用カメラキャリブレーション	53
6.7.3	連続画像入力による手形状推定	54
第7章	考察	62
7.1	精度に対する考察	62
7.2	アプリケーションのツールとして考察	63
第8章	おわりに	65
	謝辞	67

目次

1.1	一般的なカメラと赤外線カメラの比較	3
2.1	手動作の分類	6
3.1	手の構造と皮線	11
3.2	手の骨格	12
3.3	手の関節運動	14
3.4	手の自由度配置	15
3.5	中指の関節軸	16
3.6	手の石膏モデル	18
3.7	手の表面形状モデル	19
4.1	Octree モデルとその構造	22
4.2	エッジ消去と頂点分割	23
4.3	silhouette 化された手の画像	24
4.4	Octree により構成された手の Volume	24
4.5	Volume のメッシュ化	25
4.6	メッシュの simplification	25
5.1	外部繰り返しによる位置姿勢・各関節形状の変化	33
6.1	シミュレータ上での手の撮影風景	35
6.2	撮影された画像	35
6.3	Voxel モデル	36

6.4	MP 関節・PIP 関節の角度の推定誤差 (1) Level: 5	40
6.5	MP 関節・PIP 関節の角度の推定誤差 (2) Level: 6	40
6.6	手の位置姿勢が未知の場合のフィッティング (1)	42
6.7	手の位置姿勢が未知の場合のフィッティング (2)	43
6.8	手の位置姿勢が未知の場合のフィッティング (3)	44
6.9	フィッティング後に Voxel モデルと観察モデルを交換 (1)	45
6.10	フィッティング後に Voxel モデルと観察モデルを交換 (2)	46
6.11	フィッティング後に Voxel モデルと観察モデルを交換 (3)	47
6.12	位置姿勢が未知の場合の手の Voxel モデル (1)	48
6.13	位置姿勢が未知の場合の手のトラッキング結果 (1)	49
6.14	手の位置姿勢が未知の場合のフィッティング (1)	51
6.15	手の位置姿勢が未知の場合のフィッティング (2)	52
6.16	実空間上での実験環境	53
6.17	赤外線カメラ専用キャリブレーションボックス	54
6.18	赤外線カメラで撮影したキャリブレーションボックス	54
6.19	実画像から作成した手の Voxel モデル	55
6.20	手の赤外線画像 (1)	56
6.21	復元データ (1)	57
6.22	手の赤外線画像 (2)	58
6.23	復元データ (2)	59
6.24	手の赤外線画像 (3)	60
6.25	復元データ (3)	61
8.1	使用しているロボット “ドットちゃん”	66

表目次

3.1	各指の自由度配置	15
3.2	手首の自由度配置	15
3.3	リンクパラメータ	16
6.1	繰り返し回数による推定 (1) Level: 5	39
6.2	繰り返し回数による推定 (1) Level: 6	39

第1章

序論

1.1 背景

新世紀に入った今、20世紀を振り返ってみると、20世紀は科学技術の時代といえることができるだろう。その中で、工場での大量生産を支えた自動機械は我々に豊かな生活をもたらした。さらに、家電製品として家庭にも浸透した。自動機械の一つの発展形として、産業用ロボットが50年程前に出現した。出現当初は珍しい機械として注目されたが、今では機械産業や電機産業などでごく当たり前に使われており、その動作のスピードも精度も着実に向上している。最初は人の手の代わりをするものに限られていたが、今はものを運んでくれる搬送ロボットが工場内を動き回るようになった。とくに人を嫌う半導体製造では欠かせない存在となっている。しかしながら、このような生産に用いられるロボットは、一般の人の目に触れるチャンスが少ない。

20世紀にはコンピュータも生まれ、工場の自動化や、事務作業の高速化などが始められ、新幹線の座席予約や銀行のATMのように我々が身近に感じるできるようになった。さらに、ワープロやゲームとして使われるようになると、個人にも普及し、さらにインターネットの利用が加速を大きく促進した。これは、ちょうど電話や自動車の発展と似ている。つまり、それがビジネスに使われている間はその利用が限られているが、個人の楽しみとして使われるようになると爆発的に普及する。

そして今世紀、ロボットはどうなるのであろうか。コンピュータのようにロボッ

第1章 序論

トが工場から出て、家庭に浸透していくだろうか。確かに家電製品の自動化が進んでおり、その自動化技術はロボット技術と区別することができない。しかし、全自動洗濯機をロボットという人は少ない。多様な作業ができるロボットが家庭に入るのはいつになるのであろうか。また、ロボットは個人的な娯楽に使われるようになるだろうか。すでに動物の形をしたペットロボットが商品化されているので、家庭への普及は当然のことと考えるかもしれない。しかし、おもちゃのロボットを除けば、ロボットがPCのような魅力を備えているとはまだいえない。その理由として、ロボットが人間のような意志・感情・感覚・コミュニケーション能力を今だ持ち合わせていないことがあげられる。そのため、これらのいわゆる‘人間らしさ’をロボットに与える多種多様な研究が行われている。その中でもロボットの‘視覚’も一つの大きな論点してあげられる。

人間に限らず、地球上の多くの生物は生活を営む上で、敵や獲物を発見する、障害物を避けて歩行する、道具を用いる、文字を読む、危険を回避する、色を判別するなど、視覚に大きく依存している。視覚とは、ただ単純に外界を眺めることを意味しているのではなく、外界の情報をキャッチし、その情報を的確に処理することを意味している。このことをロボットに置き換えると、ロボットに搭載したカメラで外界を眺めるだけでは視覚を持ったとはいえず、カメラ撮影した外界の情報を的確に処理する必要がある。

1.2 目的

人間のように行動することができるロボットは、大きく自律型と模倣型に分類される。

自律型ロボットとは、ロボット自身がどのように行動すべきかを考え意思決定する。それに対し、模倣型ロボットとは人間の行動を注意深く観察し、人間行った行動を理解した上で、人間が行ったのと同じように行動する。模倣型ロボットを扱う場合、人間の行動を観察し、理解する必要がある。例えば人間が物を扱うときには、

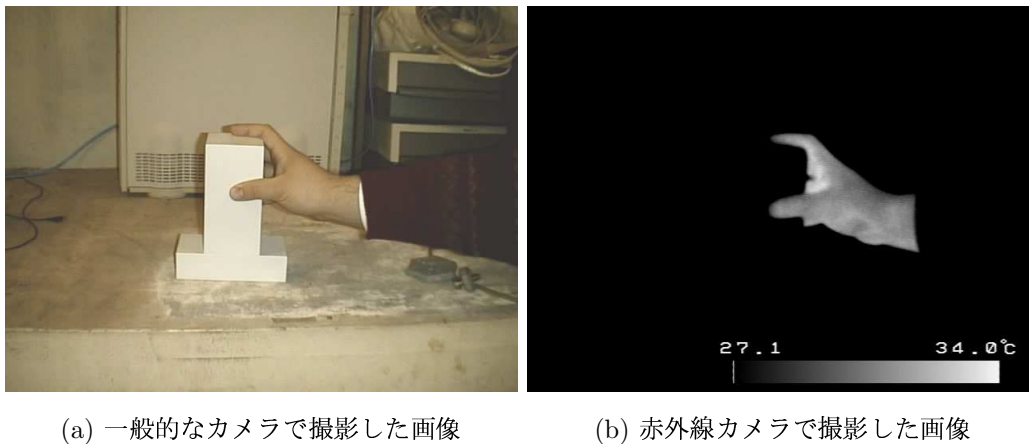


図 1.1: 一般的なカメラと赤外線カメラの比較

ほぼ例外なく手を使う。このことから、ロボットに人間の行動を模倣させる場合、人間の手に特に注目することで、物を扱う場合の人間の行動をより深く獲得し、模倣することができる。ここで、ロボットに作業をさせるためには適切な動作の指示が必要である。

そのため、これまで多くの手法が提案されてきた。これらの中で、視覚システムにより、対象物体および人間が行う作業動作を観察することにより、どのような作業が行われたのかを認識し、それをロボット用に展開することにより、人間と同じ作業を実行するシステムが提案されている (assembly plan from observation[1, 2] や teach-by-showing[3] 等)。

しかし、これら視覚システムを用いた手法の問題点は、カメラで撮影した画像中の手などによるオクルージョンに弱い点である。その打開策として、これまで、肌色抽出や、データグローブを用いる手法などが提案されてきた。しかし、肌色抽出ではオペレータの肌の色の情報を必要とすることや、データグローブを用いる手法ではオペレータがそれを手にはめなければならないことなど、簡単に誰でも操作できるものではなかった。

そこで、画像から手の位置を正確に把握することができれば、頑健に手の位置姿

第1章 序論

勢や形状を推定できると考えた。手の部分だけを抜き出す方法として、赤外線カメラを使用する。

赤外線カメラは物体の表面から発せられる赤外線を検知し、物体表面の温度を正確に計測することのできるデバイスである。これを用いることで、例えば30℃前後の物体だけを検知するようにすれば、それ以外の温度の背景などは全て無視され、手だけを正確に抜き出すことができる。一般的なカメラで撮影した画像と、赤外線カメラで撮影した画像をそれぞれ図1.1(a), 図1.1(b)に示す。

以上から本論文では人間が物を扱う場合の手動作に注目し、赤外線カメラを用いた非接触での任意の手形状推定を目的とし、実現のためのアプローチを提案している。

第2章

関連研究

人間の手動作を認識してインタフェースとして応用している従来の研究について概観していく。本論文の位置づけを明確にするため、

1. 人間の手動作
2. 接触型・非接触型インタフェース
3. カメラシステム

のような観点で関連研究を分類し、整理する。

2.1 人間の手動作

Pavlovicらはヒューマンコンピュータインタラクションの観点から手動作を図2.1のように分類している[4]。この分類では、手や腕の動きはジェスチャ (Gestures) と無意識の動き (Unintentional Movements) に分けられる。その中でもジェスチャは把持や形状形成などの操作的 (Manipulative) な動作と、対話的 (Communicative) な動作に分けられる。対話的な動作は、模倣 (Mimetic) や指示動作 (Deictic) などのようにその動き自身に直接的な意味を持つ動作 (Acts) と、言語や言葉の代わりとして用いられる動作 (Symbols) に分けられている。

この分類の上で手動作認識システムを考えると、対話的な動作の認識に関して現在まで多くの研究がされている。言語や言葉の代わりに用いられる動作に関しては、

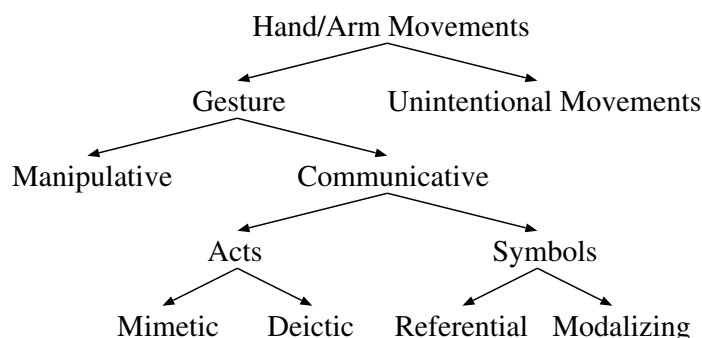


図 2.1: 手動作の分類

ハンドサインや手話などを認識するシステム [5, 6] があり，規定語の認識は実時間で可能である．また，動きそのものに意味を持つ動作を認識し，仮想物体などをインタフェースとする研究も数多くされている [7, 8, 9]．多くはテンプレートマッチングによって形状を決定し，手の動きベクトルにより操作量などを決めている．例えば，内海らの両手手振りによる仮想空間操作では多視点動画像を用い，手の形状によって決まる 8 種類のコマンドとスライド操作で仮想空間の物体の操作を行っている [10]．また，Maggioni らによる Gesture Computer [11] ではシルエット画像のモーメントと指先検出により手形状認識を行い，インタフェースとして用いている．しかし，これらのアプローチで認識できる手形状はあらかじめ決められた数種類の形状しか認識することができず，そのためインタフェースとしても従来のコマンドライクなものになってしまう．

ユーザの手動作を，VR 空間上での 3 次元形状を直接操作する動作として用いるためには限られたパターンの手形状を認識するのみでは不十分であり，任意の手形状を認識 (図 2.1 の Manipulative) することが必要である．任意の手の形状を認識するためには，システム内に 3 次元の手のモデルを持ち，関節角度や手首の姿勢などによって任意の手形状を表現する必要がある．

2.2 接触型・非接触型インタフェース

3次元形状操作などの入力インタフェースとして用いるために手形状を推定する場合、手の観測データを得るためには大きく分けて2種類の方法がある。一つは、位置センサや角度センサなどの機能を内蔵したデータグローブや、それに力覚提示の機能を加えた Cyber Grasp などを手に装着してその位置や関節角度情報を直接計測する接触型の方法である。これらの手法では、位置や関節角度の測定が高精度で行えるという利点がある。Ogawara らはデータグローブを装着して実空間の物体を把持するなどの操作を行い、ロボットに模倣させるインタフェースを構築している [12]。舟橋らはデータグローブを装着して仮想空間の物体を把持するなどの操作を行うインタフェースを構築している [13]。また Massie は指先の位置のみを計測し、物体と接触した際の力覚を提示する市販の PHANTOM haptic interface を用いて3次元形状を作成するインタフェースを構築している [14]。

それに対し非接触型は、ユーザは何も装着することなくカメラからの観察によって認識を行っている。ユーザにとっては接触型の方法よりも拘束感が少ないため、研究も盛んにされている。Shimada らは単眼視動画像を用い、緩やかな拘束条件のもとでの形状推定を行っている [15]。渡辺らはカラーグローブを使用することにより、ロボストに指文字を認識する手法を提案している [16]。

2.3 カメラシステム

非接触型のインタフェースの観測方法として最もよく用いられている手法は、カメラによる画像を入力とし、コンピュータビジョン技術によりその3次元形状を計測する方法である。

このような計測を行うためのシステムをカメラの台数によって分類してみると、単眼カメラシステム・ステレオカメラシステム・多視点カメラシステム (視体積交差 [17, 18, 19]) などのカメラシステムが考えられる。

ステレオカメラシステムとは、画像のテクスチャや特徴点などの情報の相関に基

第2章 関連研究

づいて対応付けを行い、幾何学的計算により奥行きを算出している。それに対し、多視点カメラシステム (視体積交差法) としては、Laurentini のシルエット制約による手法がある [20, 21]。シルエット制約とは2次元画像上のシルエットを3次元空間に逆投影し得られる錐体、視体積の内側に対象が含まれるということである。このことから、同時刻に得られた複数の2次元シルエットから各々視体積を求め、その共通部分を求めることで実際の対象の形状の Visual Hull を復元する。そこで、復元された Visual Hull を用いて手の形状推定を行う。

Kameda らは単眼視シルエット画像を用い、画像と関節物体モデルとの2次元モデルマッチングにより手形状推定を行っている [22]。また、Rehg らは単眼視で手のセルフオクルージョンを考慮し、関節形状の推定を行っている [23]。齋藤らは多視点カメラにより手の撮影を行い、複数の画像の中から1枚を選択し、2次元モデルマッチングにより手形状推定を行っている [24]。Delamarre らはステレオ画像から再構成した手の表面形状と手のモデルとの間に仮想的な力を発生させ、この力によって手形状を推定する手法を提案している [25]。高橋らは3眼視による人物姿勢推定を行っている [26]。さらにUtsumi らは6台のカメラを用い、最もオクルージョンの少ない視点を選び認識に用いて推定を行っている [10]。また大野らは複数のステレオカメラシステムを用いて手のトラッキングを行っている [27]。Ueda らは視体積交差法を用いた手形状推定法を提案している [28]。

2.4 本論文の位置づけ

接触型や力覚提示型デバイスを用いた接触型インタフェースを、実際にロボットに人間の行動を獲得させるとき、教示者が常に装着しておかなければならないという事は現実的でなく、非接触型のインタフェースが望ましい。また、ロボットが人間行動を獲得する際、物体を扱う手形状を認識する必要がある。この際の手形状はある決まった種類の手形状ではなく任意の手形状である。また、物体を扱う場合、把持物体によって手が隠れてしまう場合が多くある。しかし、従来研究では把持物体

2.4. 本論文の位置づけ

によって手が隠れてしまうような場合を考慮した手法はほとんど確立されていない。

本論文では、把持物体がある場合にも手の形状を認識するための任意の手形状推定を行う手法の提案を目的とする。

第3章

手のモデル

本論文では、物体を扱う場合などの人の手形状を計測し、ロボットに模倣させることを目指している。

これまでに提案されてきた手形状推定法では、大きく分けて2種類の方法がある。

1. モデルマッチングを用いた手法 [10, 11]
2. 手の関節情報を用いた手法 [15, 22, 23, 25, 28]

前者は手話のための手姿勢推定や、VR インタフェースのための手形状推定などを含んでいる。この手法ではリアルタイムでの手形状認識は可能であるが、あらかじめ決められた数種類の形状しか認識することができず、インタフェースとしての手の動作はコマンドとして使用に限られてしまう。

一方、後者は任意の手形状を推定対象としている。この手法はリアルタイムでの手形状認識は困難だが、物体を扱う場合などの手の形状は非常に多くのパターンが存在する。

以上のことから本論文では手の関節情報を用いた手形状推定を行う。

3.1 骨格モデル

“手”とは手首の関節より先の部分をいう (3.1)。これには手根と中手と指の3つの区分がある。手根と中手は外表からは区分がはっきりしないので、この2つを合わせた部分の前面、すなわち屈側を“手掌 (てのひら)”といい、反対側を“手背 (て



図 3.1: 手の構造と皮線

のこう)”という。ただし手掌はしばしば手掌と指の橈側面を合わせた区分について用いられることがある。手全体の母指側の縁は橈側縁，小指側の縁は尺側縁と呼ぶ。

日本語の指は5本の指の全てを指すが，英語では母指が thumb，その他の指が finger で表されることが多い。母指は形態的にも機能的にもその他の指と異なっている。母指以外の指は3つの区分をもち，つけ根の方から，基節，中節，末節と呼ばれる。母指は2区分しかなく，基節，末節のみである。

手の皮膚には運動によって生じたと思われる多数のしわがある。これを皮線という。掌側面では特に明瞭である。

母指球皮線は，示指基部の手掌の橈側縁に始まり，母指球の尺側縁を弧を描いて走り，手首の中央で終わる。遠位手掌皮線は，小指の基部の手掌の尺側縁に始まり，ゆるやかな曲線を描きつつ斜めに遠位へ走って，示指と中指の間に終わる。近位手

第3章 手のモデル

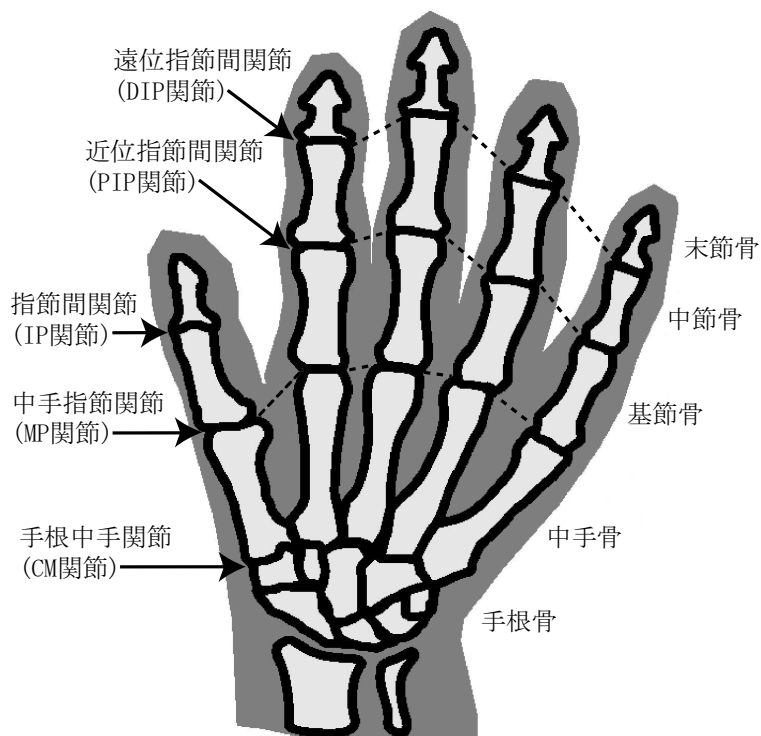


図 3.2: 手の骨格

掌皮線は母指球皮線と同じ場所から始まり，小指球のほぼ中央に向けて斜めにまっすぐ走る．指を伸ばしたままつけ根で曲げてみると，遠位手掌皮線と近位手掌皮線の橈側寄りの部分がいつそうはっきりと刻まれる．

図 3.2 は骨格標本を示すものである．母指の支柱としては 2 個，その他の指の支柱としては各々 3 個の指節骨がある．遠位より，末節骨，中節骨，基節骨ともいう（母指は末節骨，基節骨のみ）．中手は中手骨によって支えられる．これは手背から容易に触れることができる骨である．母指につながる中手骨を第 1 中手骨といい，以下橈側から順に，第 2 中手骨，第 3 中手骨，第 4 中手骨，第 5 中手骨と呼ぶ．各指ごとの指節骨と中手骨のひとつながりを指列といい，関節の名は先端より，遠位指節間関節 (distal interphalangeal joint, 以下 DIP 関節)，近位指節間関節 (proximal

3.2. 骨格モデルの運動

interphalangeal joint, 以下 PIP 関節), 中手指節関節 (metacarpophalangeal joint, 以下 MP 関節) という. DIP 関節は遠位指節間関節皮線よりもやや遠位に, PIP 関節は近位指節間皮線のほぼ真下に, MP 関節は中手指節皮線と遠位手掌皮線尺側部分および近位手掌皮線橈側部のちょうど間にある. いわゆる手首, すなわち手根部には 8 個の手根骨があって, 靭帯によって密に連結されている. 手根骨と中手骨がなす関節を手根中手関節 (carpo-metacarpal joint, 以下 CM 関節) という. 手には多数の関節があり, どのような複雑な手の動きも関節運動の組み合わせであらわすことができる. 母指とその他の指の関節運動は図 3.3 のようなものがある. 屈曲と伸展は全ての DIP, PIP, MP 関節において生じる. また, 外転と内転は母指の CM 関節と他の指の MP 関節で生じる運動である. 対立は複合運動とみなされる.

3.2 骨格モデルの運動

3.1 節で述べたような手の骨格構造と関節運動を考慮した上で, 本論文では手を手首に共通のベースを 5 つのマニピュレータの集合としてモデル化する. 各指は, リンク (骨) とジョイント (関節) の集合として図 3.4 のようにあらわされる. これにより手の動作はマニピュレータ解析の手法を用いて表現できるようになる. 図 3.3 に示す運動をあらわすために, 本論文での各リンクの自由度は図 3.4 の各ジョイントに示すように配置した. 各関節の自由度は表 3.1, 3.2 に示すとおりである. 手のモデル全体の自由度は手首の並進・回転自由度を含めて 31 である.

中指を例にとって, 指の運動学について概観する. 図 3.5 に示すように各関節の回転軸を設定し, 各関節の相対座標 Σ をリンクパラメータを用いて決定していく. 指の各自由度における相対座標系のリンクパラメータは表 3.3 のようになる.

このように各々のローカル座標系を決定した上で, 隣り合う回転座標系間の関係

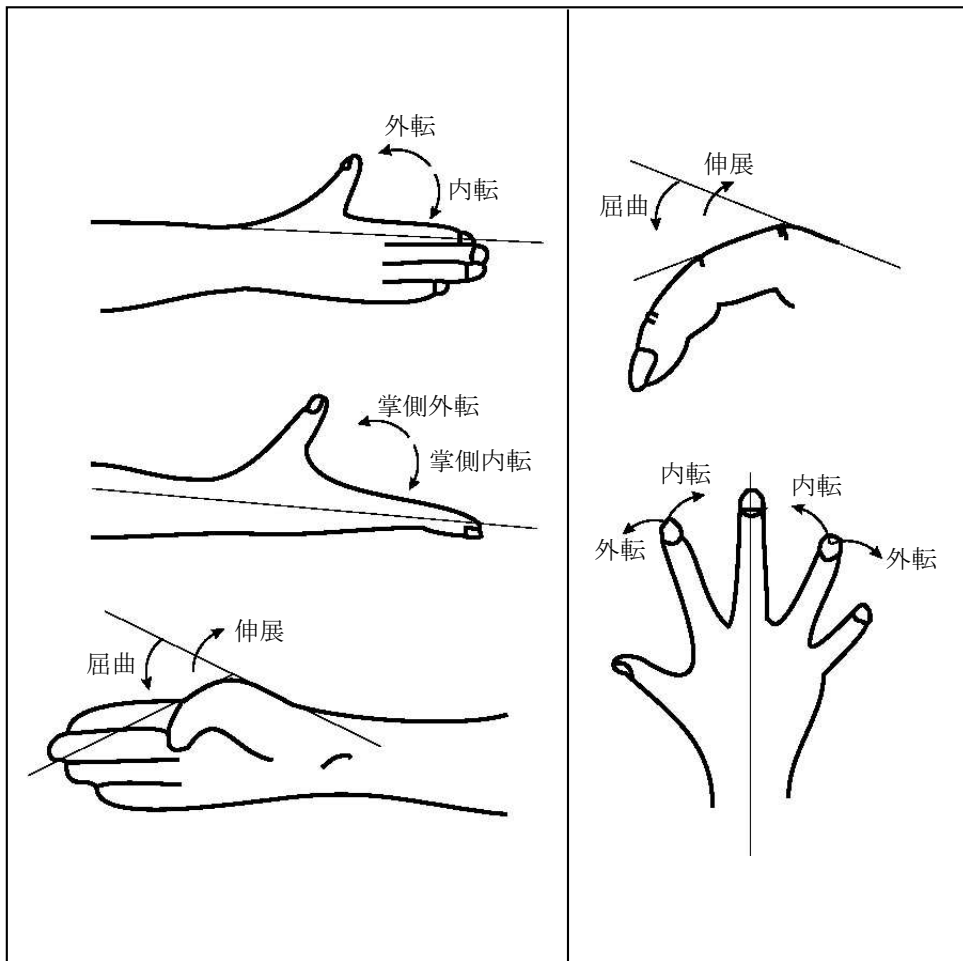


図 3.3: 手の関節運動

を示す同時変換行列 A_i は

$$A_i = \begin{pmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.1)$$

定義される. 例えば, 座標系 $\Sigma_1 \rightarrow \Sigma_0$ の変換行列 A_1 は

$$A_1 = \begin{pmatrix} \cos \theta_1 & -\sin \theta_1 \cos \left(-\frac{\pi}{2}\right) & \sin \theta_1 \sin \left(-\frac{\pi}{2}\right) & l_1 \cos \theta_1 \\ \sin \theta_1 & \cos \theta_1 \cos \left(-\frac{\pi}{2}\right) & -\cos \theta_1 \sin \left(-\frac{\pi}{2}\right) & l_1 \sin \theta_1 \\ 0 & \sin \left(-\frac{\pi}{2}\right) & \cos \left(-\frac{\pi}{2}\right) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.2)$$

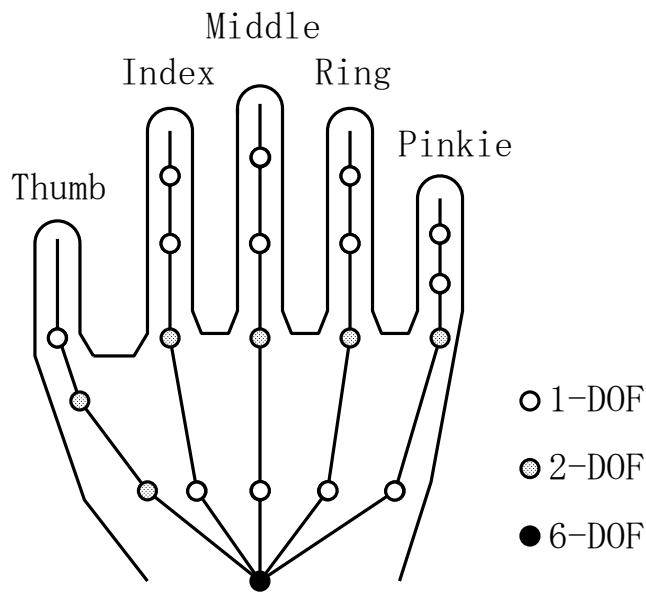


図 3.4: 手の自由度配置

表 3.1: 各指の自由度配置

	CM 関節	MP 関節	DIP 関節	PIP 関節	IP 関節	合計
母指	2	2	-	-	1	5
指	1	2	1	1	-	20 (4本分)

表 3.2: 手首の自由度配置

	回転	並進	合計
手首	3	3	6

$$= \begin{pmatrix} \cos \theta_1 & 0 & -\sin \theta_1 & l_1 \cos \theta_1 \\ \sin \theta_1 & 0 & \cos \theta_1 & l_1 \sin \theta_1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.3)$$

となる.

この行列を順次かけていくことで, 各関節の位置を一意に決定することができる.

第3章 手のモデル

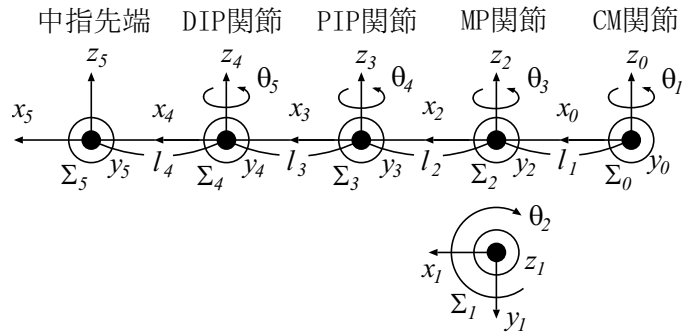


図 3.5: 中指の関節軸

表 3.3: リンクパラメータ

Joint	θ	d	α	a
1	θ_1	0	$-\pi/2$	l_1
2	θ_2	0	$\pi/2$	0
3	θ_3	0	0	l_2
4	θ_4	0	0	l_3
5	θ_5	0	0	l_4

例えば, PIP 関節の位置を求めるための変換行列 \mathbf{T} は

$$\mathbf{T} = \mathbf{A}_1 \mathbf{A}_2 \mathbf{A}_3 \quad (3.4)$$

$$= \begin{pmatrix} C_1 & 0 & -S_1 & l_1 C_1 \\ S_1 & 0 & C_1 & l_1 S_1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} C_2 & 0 & S_2 & 0 \\ S_2 & 0 & -C_2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} C_3 & -S_3 & 0 & l_2 C_3 \\ S_3 & C_3 & 0 & l_2 S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.5)$$

$$= \begin{pmatrix} C_1 C_2 C_3 - S_1 S_3 & -C_1 C_2 C_3 - S_1 C_3 & C_1 S_2 & l_2 (C_1 C_2 C_3 - S_1 S_3) + l_1 C_1 \\ S_1 C_2 C_3 - C_1 S_3 & -S_1 C_2 C_3 - C_1 C_3 & S_1 S_2 & l_2 (S_1 C_2 C_3 - C_1 S_3) + l_1 S_1 \\ -S_2 C_3 & S_2 S_3 & C_2 & -l_2 S_2 C_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.6)$$

の式で与えられる.

Σ_3 座標系の原点座標 $(0, 0, 0, 1)^T$ を \mathbf{T} にかけることで Σ_0 座標系での PIP 関節の位置が決まる. この場合は \mathbf{T} の第 4 列目が PIP 関節の位置を示している.

3.3 表面形状モデル

各関節位置が決まると骨格の姿勢が一意に定まる. そこで, 手の像をレンダリングするために手の表面形状をあらわす皮膚の形状データが必要となる. 手の表面形状は骨格姿勢の変化に応じて柔軟に変化できなければならない [30, 31]. 本論文では手の表面形状を多数の小さな 3 角形のポリゴンパッチで表現する. また各ポリゴンパッチの形状が骨格姿勢の変化に対応して変形できるように, ポリゴンパッチの各頂点データがどの骨リンクに対応しているのかを属性分けしている. また, 実際の手形状を直接レーザレンジファインダで 3 次元計測し, 多くの点を精密に計測している. 具体的には以下のような手順で作成される.

1. 開いた手の石膏モデル (図 3.6) を作成し, レーザレンジファインダを用いて複数の方向から撮影する.
2. 得られたデータをアラインメントする [32].
3. アラインメント後, Voxel データに変換し, 表面の点列を生成する [33].
4. 関節位置を指定し, 各関節をリンクする.
5. 手の表面の全ての点とその最近傍の骨リンクを対応付ける.
6. 関節形状の変形に合わせて, 手の表面の点も変形させる.
7. 点列を縫い合わせるように 3 角パッチを生成する.

このようにして骨格モデルの運動学をあらかじめ考慮に入れた滑らかな手の形状を作成する (図 3.7).

第3章 手のモデル



図 3.6: 手の石膏モデル

3.3. 表面形状モデル



(a) 関節の変形前



(b) 関節の変形後

図 3.7: 手の表面形状モデル

第4章

手の3次元形状計測

4.1 ビジョンベースの手形状計測

本論文では、物体を扱う場合などの人の手形状を計測し、ロボットに模倣させることを目指している。データグローブや力覚提示デバイスを用いた接触型インタフェースを採用することは、作業教示者にとって拘束感が非常に大きくなるため現実的ではない。このことから実世界では拘束感の少ない非接触型のインタフェースが望ましいと考えられる。

非接触型の方法により観測データを得るためのカメラシステムとして、第2章で示したように、単眼カメラシステム・ステレオカメラシステム・多視点カメラシステム(視体積交差)などのカメラシステムが考えられる。単眼カメラシステムを用いて手形状を計測する場合、キャプチャや計算に必要な時間が少なくリアルタイム性に富むという長所を持つ反面、画像から奥行き情報を得ることができないという欠点がある。またステレオカメラシステムでは奥行き情報を得ることができるため、単眼システムの欠点は除去することができる。しかし、手で物体を扱っている場合、把持物体によるオクルージョンの問題が発生する。それに対し、複数のカメラを用いた多視点カメラシステム(視体積交差)では、カメラの台数を増やすことでオクルージョンの影響はより小さくなる。さらに得られた画像群から3次元形状を再構成する方法を使えば、画像間のマッチングをとる必要がなくなり、カメラシステムに起因する精度の低下も避けることができる。またこの方法ではカメラ間の位置関係を知る必要はなく、3次元空間中の位置が各カメラ画像のどこに射影されるかを

知るだけでよい。そのためそれぞれのカメラは位置は自由度が非常に大きくなる。

本論文ではこれらの理由から多視点カメラシステム (視体積交差) を用いて手形状を計測する。

4.2 Octree による形状再構成法

本論文では多視点カメラシステムを用いて得られた複数の 2 次元画像から、空間分割表現によって再構成された 3 次元形状を手の観測データとして取り扱う。この観測データを手の Voxel モデルと呼ぶ。

コンピュータビジョンの分野において、複数の画像から 3 次元形状を復元する手法としては、画像の陰影情報を用いる Shape from shading, 画像の輪郭を用いる Shape from countour, 画像の動き情報を用いる Shape from motion, 画像のシルエット情報を用いる Shape from silhouette などの手法がある。今回採用した 3 次元形状再構成手法は、この中でも最も安定な画像の特徴量であるシルエット情報を用いた Shape from silhouette の手法である。

2 次元画像が Pixel の集合で自然な画像を表現するように、3 次元形状は Voxel と呼ぶ小さな立方体の集合体として表現することができる。複雑な形状を表現しやすい反面、大量のメモリと計算量が必要であるという欠点を持っている。この欠点を改良するために、3 次元形状を大きさの異なる立方体を組み合わせて表現する手法が Octree 表現である。Octree 表現で用いられる各サイズの立方体を Octant と呼ぶ。Octree は図 4.1 のように 8 文木による再帰的な構造になっている。Octant (Octree の各ノードにあたる) はそれぞれデータとの関係をあらわす属性を持っている。この属性には 3 種類あり、データの中に完全に内包されている立方体は “BLACK,” データの完全に外側にある立方体は “WHITE,” データの一部にかかっている立方体は “GRAY” と定義されている。“GRAY” と定義された立方体は、さらに 8 つの小さな立方体に分割され、それぞれに対してさらに属性を持つ。この 8 分木の階層数によって形状の表現精度を変化させることができる。

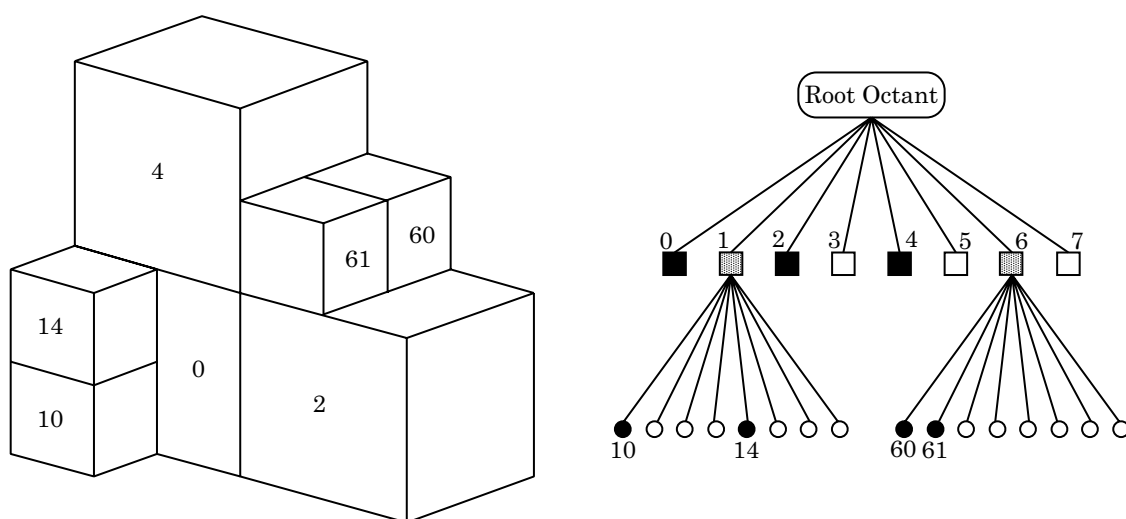


図 4.1: Octree モデルとその構造

4.3 Voxelモデルのメッシュ表現

実際に手の Volume データを扱うには、表面を 3 角形パッチのメッシュにする必要がある (第章参照). Volume データをメッシュ状にする手法はコンピュータ・グラフィックスにおいてよく使われている. 本論文では, 広く用いられている Lorensen らのマーチング・キューブ法 [34] を改良した, 剣持らの手法 [35] を利用している.

マーチング・キューブ法とは, 3次元の Volume データの各 Voxel を 3次元格子空間の格子点とみなし, ある Voxel の濃淡値 v より大きい濃淡値を持つ格子点を黒点, 小さい濃淡値を持つ点を白点とする. そして, 黒点と白点の間に v を通る面を補完することにより, 濃淡値 v の等値面を 3次元空間中に生成する. マーチング・キューブ法の特徴は, 8つの格子点を頂点とする単位立方領域において, 黒点と白点の各配置に対して, あらかじめ 3角形による面の補完の仕方を決めておくことである. 3角形の頂点は黒点と白点を端点とする単位立方体の辺上にあり, 辺上での位置は黒点と白点の濃淡値の線形補完により決まる. 続いて, 隣り合う単位立方領域における 3角形を順につなぎ合わせていき, 最終的に得られる多面体を等値面とする.

単位立方領域における黒点と白点の配置数は 256 通りあるが, Lorensen らの手法

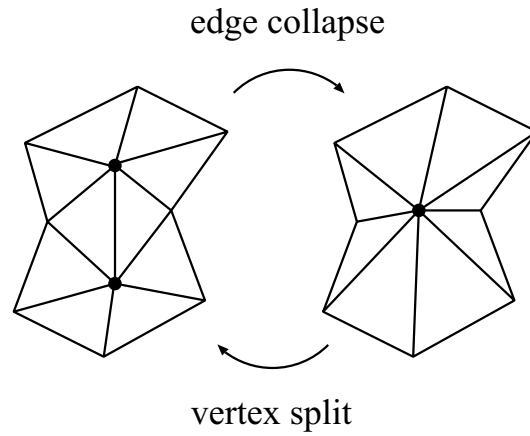


図 4.2: エッジ消去と頂点分割

では基本的な 15 通りを挙げ、その他の配置は黒点と白点の入れ換え、回転により得られるとしている。剣持らの手法は黒点と白点の入れ換えは行わず、回転のみを考慮し、基本的な配置は 23 通りとしている。

4.4 メッシュの simplification

メッシュによる表現は、3 角形パッチという非常に単純な要素のみで成り立っているため、生成や描画が容易である反面、冗長であり、計算コストが非常に多く必要となる。これを解決するために、形状の幾何学的特長を保ちつつ形状要素である三角形パッチの個数を減らす (simplification) という最適化の研究が近年盛んに行われている。

本論文では Hoppe らによる progressive mesh 法 (PM 法) を利用した [36]。この手法は、エッジ消去 (edge collapse) とその逆操作である頂点分割 (vertex split) の 2 つの操作を基本とし (図 4.2)、常に変形による誤差を検地しながら最適化を行っている。したがって、非常に精度の高い圧縮が可能となる。

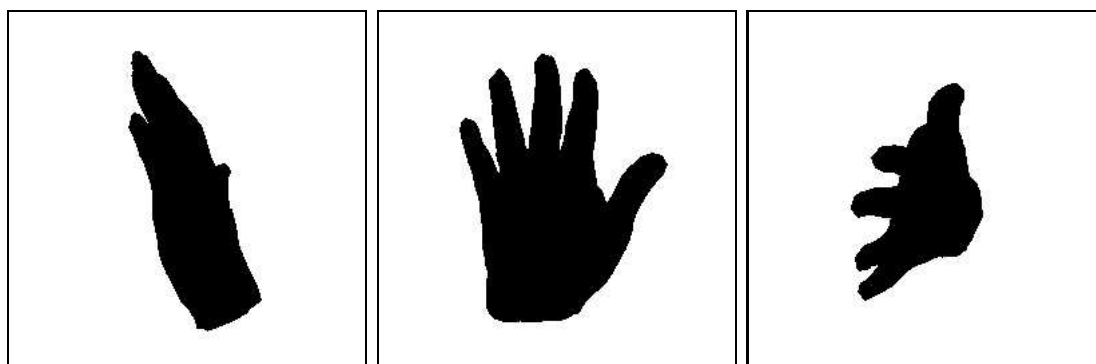


図 4.3: silhouette 化された手の画像

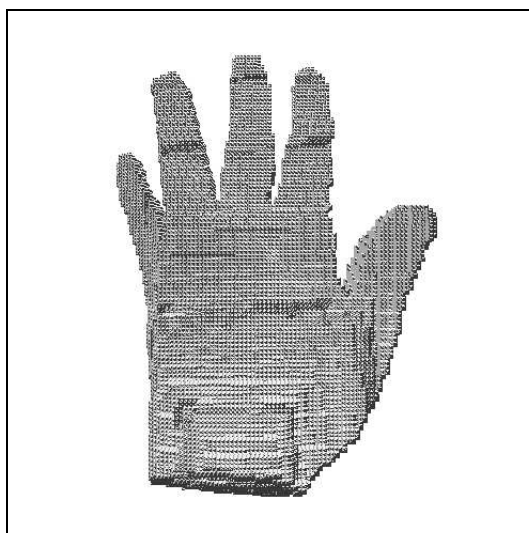


図 4.4: Octree により構成された手の Volume

4.5 復元データ作成の流れ

ここで、以上の流れを示す。

1. カメラにより撮影した画像の silhouette 化 (図 4.3).
2. Octree 表現による, Volume の作成 (図 4.4).
3. Volume のメッシュ化 (図 4.5).

4.5. 復元データ作成の流れ

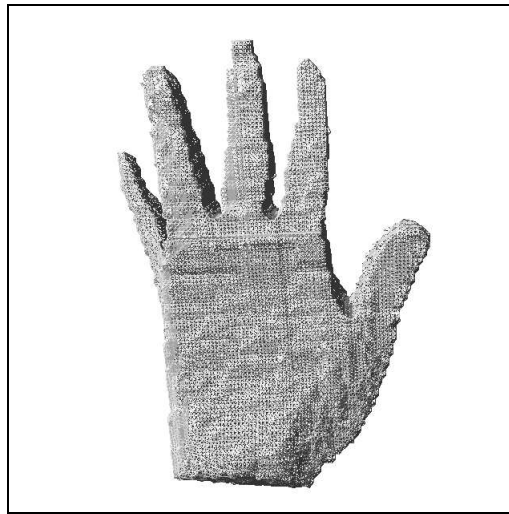


図 4.5: Volume のメッシュ化

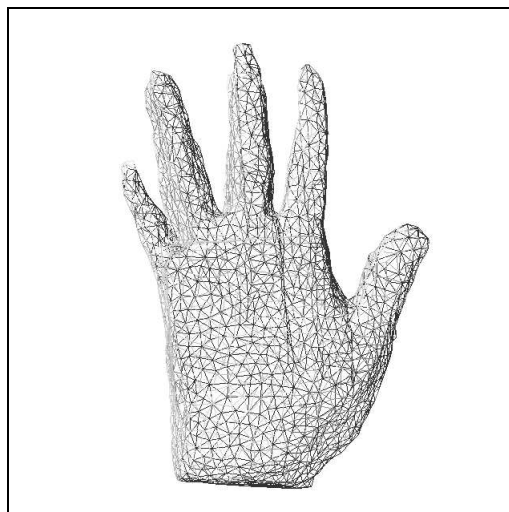


図 4.6: メッシュの simplification

4. メッシュの simplification (図 4.6).

以下, このように作成した復元された手を復元データと呼ぶ.

第5章

手の形状推定

5.1 位置姿勢推定

第4章の方法で得られた Voxel モデルと表面形状モデルのフィッティングによって、骨格モデルの各関節角度を推定する必要がある。

関節情報を含まない事前に作成した物体モデルと位置姿勢推定を行いたい物体の3次元レンジデータ間でのマッチングを行う手法はこれまでも数多く提案されている。Beslらはモデル中の点とレンジデータ中の点の最近傍点を繰り返しにより求め (iterative closest-point: ICP), 最小2乗法によりこれらの対応点からポーズの最適化を行った [37]. 高速な ICP アルゴリズムを用い, 効率的に位置姿勢推定を行ったが, モデルの初期の位置姿勢が良好なときのみ有効である。Zhangは, 動的なしきい値変化を利用した ICP を用いた [38]. 外乱に対し比較的ロバストであり, また二分木を多次元に拡張した k-d tree 法 [39] を用いて計算効率のよい位置姿勢推定を行ったが, モデルのポーズのずれが十分小さいときのみ有効である。Haralickらは反復重み付け法を用いた位置姿勢の計算を行った [40]. 大きな外乱下でも正しく位置姿勢を計算したが, 物体モデルとレンジデータ間の対応点が最初から決まっている場合にのみ適用できる。さらに位置姿勢推定の計算を高速化した方法があるが [41], モデルの初期ポーズが良好なときのみ有効であり, またノイズ (正規分布に従う計測誤差) やオクルージョンに弱い。これらの問題点を考慮し, モデルの初期位置姿勢にたいしてロバストで, かつ計算コストを抑えた位置姿勢推定法に Wheelerらによる 3D template matching (3DTM) がある [42]. 本章では, まず 3DTM のアルゴ

リズムについて述べ、続いて、並進・回転自由度のみを考慮する 3DTM に、指の各関節自由度をも考慮する手法を提案する。

5.2 3DTM の原理

5.2.1 対応点の計算

まず、ある物体モデル点に最も近いレンジデータ点をモデルとレンジデータの対応点と考える。すなわち、ある位置姿勢における i 番目のモデルパッチの重心点 m_i に対応するレンジデータ点は、

$$d_i = \|m_i - r\| \quad (5.1)$$

$$i = 1, \dots, k \quad (k : \text{モデルの可観測パッチ数})$$

で定義される距離 d_i を最小にする点 r とする。ただし、 $r \in D$ (D はレンジデータ点の集合、 r は各レンジデータに対応した 3次元データ) である。

一般に、レンジデータ点数は膨大であるため、最近傍点の探索には多くの計算時間を必要とする。したがって、効率的に対応点を探索する必要があり、これを k-d tree 法を用いて計算する。k-d tree 法は 2分木法を多次元に拡張したものである。

5.2.2 位置姿勢の決定

あるモデルパッチの重心点 m_i に対応するレンジデータ中の点 r_j は、ある 3×3 回転行列 R および 並進ベクトル t を用いて、

$$r_j = Rm_i + t \quad (5.2)$$

と表すことができる。一般に、測定にはノイズが伴うので、 r_j にはノイズ成分 β が含まれることになり、

$$r_j = Rm_i + t + \beta \quad (5.3)$$

と書き直される.

β が正規分布, すなわち確率密度が

$$P(\mathbf{n}) \propto \exp\left(-\frac{\mathbf{n}^T \mathbf{n}}{2\sigma^2}\right) \quad (5.4)$$

であると仮定すれば, 最適なモデル位置姿勢は最小2乗法を利用して求めることができる. すなわち,

$$f(\mathbf{R}, \mathbf{t}) = \sum_i \|\mathbf{R}\mathbf{m}_i + \mathbf{t} - \mathbf{r}_j\|^2 \quad (5.5)$$

を最小にする \mathbf{R} および, \mathbf{t} を求めることで, モデル点とレンジデータ点間の距離を最小にすることができる.

しかし, \mathbf{R} は, ある方向ベクトル $\mathbf{u} = (\alpha, \beta, \gamma)^T$ および, スカラ量 s によって構成される単位4元数,

$$\mathbf{b}_q = (\alpha, \beta, \gamma, s)^T \quad (5.6)$$

によって表現される. したがって, モデルの位置姿勢 \mathbf{p} は並進を加えて,

$$\mathbf{p} = (\mathbf{b}_q^T, \mathbf{t}^T)^T \quad (5.7)$$

で与えられる. また, 反対にベクトル \mathbf{p} から元の回転行列および, 並進ベクトルを計算することが可能である.

5.2.3 ロバストな位置姿勢の決定

モデル中の点とレンジデータ中の対応点との関係に誤りがなく, またレンジデータに含まれている誤差が正規分布をしている場合は, 式(5.5)を利用してモデルの位置姿勢を求めることができる. しかし, 一般にはこれらの対応関係の中で, どれが正しく, どれが正しくないかは知ることができない. また不正確な対応を起こしている場合やレンジデータがオクルージョンなどにより隠蔽されている場合などの距離の誤差は正規分布にはしたがっておらず, こうした問題を解決する方法が必要であ

る。そこで、ロバスト統計量に基づいた計算手法を用いてこの問題を解決する。ここでは、ノイズやオクルージョンに最もロバストであると考えられる M 推定法 [44] を用いる。

M 推定法とは、最小 2 乗法を一般化した最尤推定法である。重み関数を用いることで、外乱による影響を小さくすることができる。評価関数は局所解に落ちにくく、比較的滑らかであり、よりよい位置姿勢推定を行うことができる。

M 推定法では評価関数

$$E(e) = \sum_i \rho(e_i) \quad (5.8)$$

を最小にする。ここで $\rho(e_i)$ は i 番目のモデルパッチと、これに対応するレンジデータ点間の距離である。さらに e_i はモデルの位置姿勢により変化するので、 $e_i(\mathbf{p})$ と置くことができる。

したがって、 $E(e)$ の最小値は、 \mathbf{p} により偏微分をとり、

$$\frac{\partial E(e)}{\partial \mathbf{p}} = \sum_i \frac{\partial \rho}{\partial e_i(\mathbf{p})} \frac{\partial e_i(\mathbf{p})}{\partial \mathbf{p}} = \mathbf{0} \quad (5.9)$$

で求められる。

ここで、

$$w(e) = \frac{1}{e} \frac{\partial \rho}{\partial e} \quad (5.10)$$

とおくと、

$$\frac{\partial E(e)}{\partial \mathbf{p}} = \sum_i w(e_i) e_i \frac{\partial e_i}{\partial \mathbf{p}} \quad (5.11)$$

が得られる。 $w(e)$ は重み関数である。

$w(e)$ が e の関数でないと仮定すると、式 5.10 より、

$$\frac{\partial \rho}{\partial e} = w e \rho = \frac{1}{2} w e^2 \quad (5.12)$$

となり、これは重みつき最小 2 乗法と等価である。この場合、 w は誤差 e の総和への重みを与える。 $w(e) = 1$ とおくと、それぞれの誤差はその大きさに関係なく同一の重みを持つことになり、ICP で用いられているような純粋な最小 2 乗法となる。

5.2.4 位置姿勢の最適化

前節までで、M推定法を用いてモデルの位置姿勢 \mathbf{p} を求める一般的な方法について述べたが、本節では、より具体的な \mathbf{p} の計算方法を述べる。

モデルの位置姿勢が変化すると、可観測なモデルパッチ総数は変化する。したがって、パッチ総数の違いによる、評価関数への影響を少なくするために以下の正規化を行う。すなわち、 $V(\mathbf{p})$ をモデルの位置姿勢が \mathbf{p} であるときの可観測モデル点の集合 ($|V(\mathbf{p})|$ をその点数) とすれば、 \mathbf{p} は式 (5.8) より、

$$E(\mathbf{p}) = \frac{1}{|V(\mathbf{p})|} \sum_{i \in V(\mathbf{p})} \rho(e_i(\mathbf{p})) \quad (5.13)$$

を最小にすることで求められる。ここで $e_i(\mathbf{p})$ は、モデルの位置姿勢が \mathbf{p} であるときの i 番目のパッチの重心点 \mathbf{m}_i と、これに対応するレンジデータ点 \mathbf{r}_j 間の距離である。式 (5.2) と同形式で記述すると、

$$e_i(\mathbf{p}) = \|\mathbf{R}_q \mathbf{m}_i + \mathbf{t}_q - \mathbf{r}_j\| \quad (5.14)$$

となる。ただし、 \mathbf{R}_q , \mathbf{t}_q はそれぞれ、 \mathbf{p} の回転部分の 3×3 行列および、並進ベクトルである。

5.3 提案手法の原理

本来は並進・回転自由度のみを考慮していた3DTMを、手の関節自由度を含む合計32自由度(並進3, 回転4(4元数), 関節25)に拡張した手法を提案する。ただし、計算の簡略化のため、並進・回転と関節形状変化は独立に行う。

関節形状変化に注目する場合、ある手の表面形状モデルパッチの重心点 \mathbf{m}_i と、対応するレンジデータ中の点 \mathbf{r}_j は、式 (5.2) のように、例えば、中指のDIP関節より先にある場合は、図 3.5 より、

$$\begin{aligned} \mathbf{r}_j = & \mathbf{R}_{\theta_1} \mathbf{R}_{\theta_2} \mathbf{R}_{\theta_3} \mathbf{R}_{\theta_4} \mathbf{R}_{\theta_5} (\mathbf{m}_i - \mathbf{c}_4) + \mathbf{R}_{\theta_1} \mathbf{R}_{\theta_2} \mathbf{R}_{\theta_3} \mathbf{R}_{\theta_4} (\mathbf{c}_4 - \mathbf{c}_3) \\ & + \mathbf{R}_{\theta_1} \mathbf{R}_{\theta_2} \mathbf{R}_{\theta_3} (\mathbf{c}_3 - \mathbf{c}_2) + \mathbf{R}_{\theta_1} (\mathbf{c}_2 - \mathbf{c}_1) + \mathbf{c}_1 \end{aligned} \quad (5.15)$$

で示される。ただし、 \mathbf{R}_{θ_i} は θ_i による回転行列、 \mathbf{c}_i は各関節の位置ベクトルである。実際は 25 の関節があるので、パッチの重心がどの関節に連動するかによって、25 種類の式が存在する (以下、中指の DIP 関節より先の部分に関して論ずる)。

このとき、

$$\begin{aligned} f(\boldsymbol{\theta}) = & \mathbf{R}_{\theta_1} \mathbf{R}_{\theta_2} \mathbf{R}_{\theta_3} \mathbf{R}_{\theta_4} \mathbf{R}_{\theta_5} (\mathbf{m}_i - \mathbf{c}_4) + \mathbf{R}_{\theta_1} \mathbf{R}_{\theta_2} \mathbf{R}_{\theta_3} \mathbf{R}_{\theta_4} (\mathbf{c}_4 - \mathbf{c}_3) \\ & + \mathbf{R}_{\theta_1} \mathbf{R}_{\theta_2} \mathbf{R}_{\theta_3} (\mathbf{c}_3 - \mathbf{c}_2) + \mathbf{R}_{\theta_1} (\mathbf{c}_2 - \mathbf{c}_1) + \mathbf{c}_1 - \mathbf{r}_j \end{aligned} \quad (5.16)$$

とすると、評価関数 $E(\boldsymbol{\theta})$ は

$$E(\boldsymbol{\theta}) = \|f(\boldsymbol{\theta})\|^2 \quad (5.17)$$

で与えられる。

以上より、M 推定量での評価関数 $E(\boldsymbol{\theta})$ の最小値は

$$\frac{\partial E(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_i \frac{\partial \rho}{\partial e_i} \frac{\partial e_i}{\partial \boldsymbol{\theta}} = \mathbf{0} \quad (5.18)$$

より求めることができる。

ここで、式 (5.10) より式 (5.11) は

$$\frac{\partial E(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_i w(e_i) e_i \frac{\partial e_i}{\partial \boldsymbol{\theta}} \quad (5.19)$$

と変形される。本論分では、重み関数 $w(e)$ として Lorentz 関数を用いた。また、Voxel メッシュは視体積交差を用いているため、全ての点に関し可観測としている。そのため、 $|V(\mathbf{p})|$ は常に Voxel モデルのパッチ数と一致する。

以下にモデルの位置姿勢、関節形状の推定アルゴリズムの流れを示す。

入力：レンジデータ点の集合 D

入力：モデル V

入力：モデルの初期位置姿勢 \mathbf{p}

入力：モデルの初期関節形状 $\boldsymbol{\theta}$

第5章 手の形状推定

出力：最終位置姿勢 \mathbf{p}'

出力：最終関節形状 $\boldsymbol{\theta}'$

1. D に対して k-d tree を構築
2. **repeat**
3. \mathbf{p} に関する評価関数値の計算 $E_0(\mathbf{p}) \leftarrow E(\mathbf{p})$
4. \mathbf{p} に関する評価関数の勾配計算 $d\mathbf{p} \leftarrow -\nabla_{\mathbf{p}}E(\mathbf{p})$
5. \mathbf{p} に関する勾配方向の最小値計算 $\lambda_{\mathbf{p}} \leftarrow \min E(\mathbf{p} + \lambda_{\mathbf{p}}d\mathbf{p})$
6. 最適位置姿勢を決定 $\mathbf{p} \leftarrow \mathbf{p} + \lambda_{\mathbf{p}}d\mathbf{p}$
7. $\boldsymbol{\theta}$ に関する評価関数値の計算 $E_0(\boldsymbol{\theta}) \leftarrow E(\boldsymbol{\theta})$
8. $\boldsymbol{\theta}$ に関する評価関数の勾配計算 $d\boldsymbol{\theta} \leftarrow -\nabla_{\boldsymbol{\theta}}E(\boldsymbol{\theta})$
9. $\boldsymbol{\theta}$ に関する勾配方向の最小値計算 $\lambda_{\boldsymbol{\theta}} \leftarrow \min E(\boldsymbol{\theta} + \lambda_{\boldsymbol{\theta}}d\boldsymbol{\theta})$
10. 最適関節形状を決定 $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \lambda_{\boldsymbol{\theta}}d\boldsymbol{\theta}$
11. **until** $E_0(\mathbf{p}) - E(\mathbf{p}) < \varepsilon_{\mathbf{p}} \wedge E_0(\boldsymbol{\theta}) - E(\boldsymbol{\theta}) < \varepsilon_{\boldsymbol{\theta}}$
12. 最終位置姿勢 \mathbf{p}' を出力 $\mathbf{p}' \leftarrow \mathbf{p}$
13. 最終関節形状 $\boldsymbol{\theta}'$ を出力 $\boldsymbol{\theta}' \leftarrow \boldsymbol{\theta}$

本論文では、最適解を求めるために共役勾配法 [47] を用いている。また、繰り返しの行えば、より真値に近づき、1回の繰り返しによる変位量は少なくなるという仮定から、繰り返し数が進むにつれ、変位が小さくなるような重みをかけ、過大な変位を避けている。しかし、収束する前に変位量がほぼ0になることが多くなる。そのため、単純に11に到達した時点で収束が終了するわけではない。したがって、さらに上記の一連の作業で求めた \mathbf{p}' , $\boldsymbol{\theta}'$ を新たな入力として、同様の作業を繰り返

5.3. 提案手法の原理

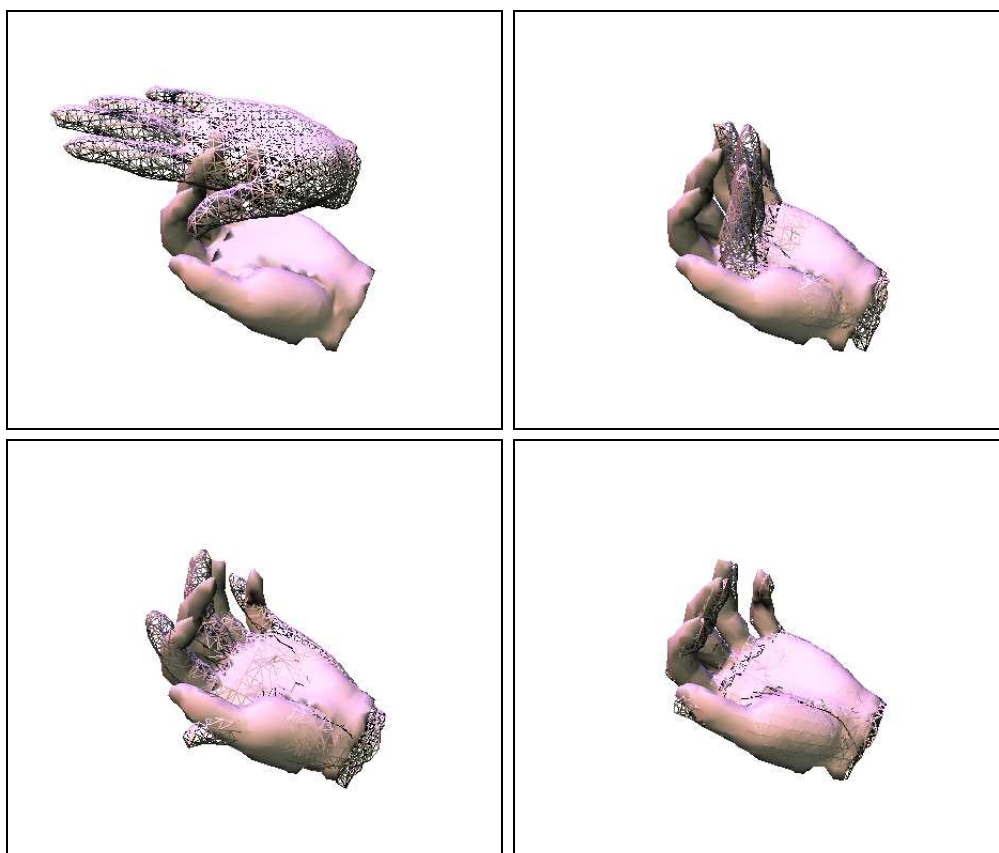


図 5.1: 外部繰り返しによる位置姿勢・各関節形状の変化

すことで、最適解に収束させる (図 5.1). 以下, 1回の作業の中での繰り返しを内部繰り返し, 一連の作業の繰り返しを外部繰り返しと呼ぶ.

第6章

手形状推定実験

前章までの手法などから実際に手形状推定を行う。

ここで、提案する手法は、視体積交差により、手を Voxel 表現し、関節形状の変化を考慮した手のモデルとのフィッティングを行うことで手の形状を復元するという観点では、Ueda らの手法 [28] に類似している。しかし、この手法では、

1. 動きの解析が難しい親指の形状変化は考慮しない
2. 手の位置は変化せず、関節のみの形状変化とする
3. オクルージョンは考慮しない

などの問題点がある。

本論文で提案する手法はこれらの問題点を克服した手法であることを本章の実験から示す。

本章は、前半はシミュレータを用いた実験について述べ、後半で実装した上での実験について述べる。

6.1 形状推定の流れ

骨格モデルの各関節角度を入力することによって手形状シミュレーションを行うことのできるシミュレータを用いて第5章で提案した手形状推定手法を用いた推定精度の評価と推定速度の評価を行った。本来、手の Voxel を作成する場合、無数の

6.1. 形状推定の流れ



図 6.1: シミュレータ上での手の撮影風景

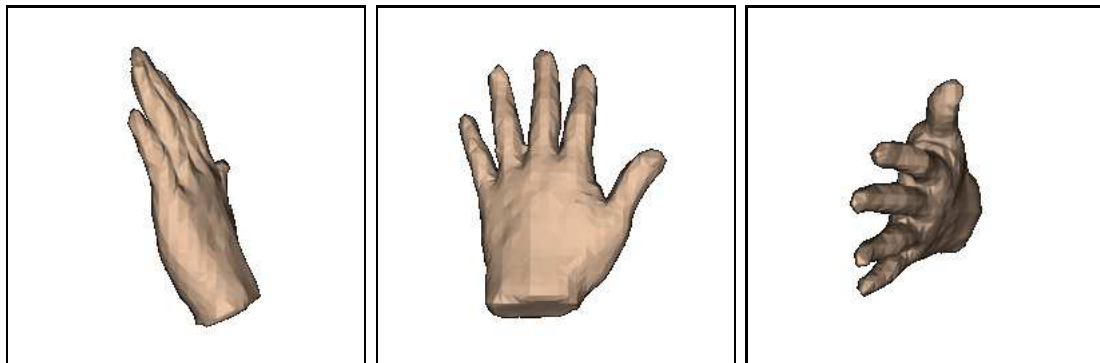


図 6.2: 撮影された画像

カメラがあれば手の表面形状をより正確に表現できる。しかし、実世界ではカメラの台数には空間的にも、コスト的にも制約が生じる。そこで、シミュレータ中でもカメラの台数を3台とした。



図 6.3: Voxel モデル

図 6.1 のようにカメラは正面，側面，上面の 3 点に設置し，それぞれの視点から手形状画像を撮影し，保存する (図 6.2).

オフラインでシルエット画像をそれぞれ読み込み，図 6.3 のような Voxel モデルを生成する．シミュレーションでは手の位置や関節形状は既知であるが，シミュレータ上で自由に動くことができる．

6.2 カメラキャリブレーション

手の Octree 表現によるモデルを作成するために，任意視点での 3 次元空間と 2 次元画像の対応する点を 1 つの中心射影行列によって関係付けるキャリブレーションが必要となる．ピンホールカメラモデルにおけるキャリブレーション法には大きく分けて 2 種類ある．1 つは外部変数と内部変数を独立に求める方法である．もう 1 つ

は世界座標系から画像座標系までの光学的・幾何学的な関係を1つの中心射影で表す方法である。本論文では、Octantの各頂点座標が2次元画面(撮像面)のどの位置に投影されるかを1つの中心射影で関係付ける。この中心射影行列は6つの3次元空間上の点と2次元画像上の点との対応付けから求めることができる。ここで、中心射影行列を \mathbf{P} とし、3次元空間中の点 $\mathbf{w} = (x, y, z)^T$ 、画像上の点を $\mathbf{m} = (u, v, 1)^T$ とすると、

$$s\mathbf{m} = \mathbf{P}\mathbf{w} \quad (6.1)$$

$$\mathbf{P} = \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{pmatrix} \quad (6.2)$$

の関係が成り立つ。キャリブレーションボックス(図6.17は実装時に使用したもの)をカメラで撮影し、キャリブレーションボックス上の交点座標がカメラ画像のどの位置に映っているかを指示する。キャリブレーションボックス上の同一平面上にない6組以上の3次元座標位置と2次元座標の組を指定し、中心射影行列を求める。

具体的には、6組以上の基準点から最小2乗法によってパラメータを同定する。 n 点の基準点の物体座標 $(X_i, Y_i, Z_i)^T$ と、それに対応するカメラ座標 (X_{ci}, Y_{ci}) が既知であれば、 $P_{34} = 1$ とすることにより、

$$\begin{pmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -X_1X_{c1} & -Y_1X_{c1} & -Z_1X_{c1} \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -X_1X_{c1} & -Y_1X_{c1} & -Z_1X_{c1} \\ & & & & & \vdots & & & & & \\ X_n & Y_n & Z_n & 1 & 0 & 0 & 0 & 0 & -X_nX_{cn} & -Y_nX_{cn} & -Z_nX_{cn} \\ 0 & 0 & 0 & 0 & X_n & Y_n & Z_n & 1 & -X_nX_{cn} & -Y_nX_{cn} & -Z_nX_{cn} \end{pmatrix} \begin{pmatrix} c_{11} \\ c_{12} \\ \vdots \\ c_{32} \\ c_{33} \end{pmatrix} = \begin{pmatrix} X_{c1} \\ Y_{c1} \\ \vdots \\ X_{cn} \\ Y_{cn} \end{pmatrix} \quad (6.3)$$

となる。これを、

$$\mathbf{A}\mathbf{c} = \mathbf{r} \quad (6.4)$$

と表せば、最小2乗法により

$$\mathbf{P} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{r} \quad (6.5)$$

で行列 \mathbf{P} を得ることができる。

6.3 モデルの位置姿勢を固定した場合の精度評価

はじめに、アルゴリズムの定量的な評価を行うため、手の位置姿勢を固定、各関節の屈曲もあまりないものとして、繰り返し回数による推定精度の比較、Voxelモデルの階層数の違いによる推定精度の比較を行った。

6.3.1 繰り返し回数の違いによる推定精度の比較

推定収束までの反復回数に伴う、関節角度の推定過程を表 6.1, 6.2 に示す。推定形状は中指の MP 関節を 10 度屈曲 (手掌側へ倒す動き) した手形状である。

表 6.1 は Voxel モデルの復元レベルを 5 にしたものである。このとき、Voxel モデルの最小 Octree サイズは 1 辺の長さが約 8mm である。各外部繰り返しごとに、少しずつ MP 関節は 10 度に近い値に収束している。しかし、全体として PIP 関節も前屈していると推定される。それに対し、表 6.2 は Voxel モデルの復元レベルを 6 にしたものである。このときの Voxel モデルの最小 Octant サイズは 1 辺の長さが約 4mm である。外部繰り返しを行うことで、最終的に MP 関節・PIP 関節とも非常に良い精度を示している。このことから、推定精度には繰り返し回数が非常に大きな影響を与えていることが分かる。

6.3.2 Voxel モデル階層数の違いによる推定精度の比較

観測データとして用いる Voxel モデルの復元レベルは、どの程度観測データを精密に表しているかを示している。Voxel モデルの最小 Octant サイズは 1 辺の長さはそれぞれレベル 5 では 8mm, レベル 6 では 4mm である。図 6.4, 6.5 は前節の中指の動きに関して、外部繰り返し回数に対する MP 関節, PIP 関節それぞれの関節角度誤差を示している。ここで、推定誤差 error は

$$\text{error} = |e_{\text{ang}} - t_{\text{ang}}| \quad (6.6)$$

6.3. モデルの位置姿勢を固定した場合の精度評価

表 6.1: 繰り返し回数による推定 (1) LEVEL: 5

外部繰り返し回数	内部繰り返し回数	MP 関節角度 [deg]	PIP 関節角度 [deg]
1	2	16.13	45.00
2	5	0.00	36.27
3	8	0.67	33.06
4	6	1.32	28.67
5	6	2.30	26.69
6	6	8.72	11.38
7	5	8.29	10.69
8	4	8.84	10.40

中指 MP 関節を 10 度前屈

表 6.2: 繰り返し回数による推定 (1) LEVEL: 6

外部繰り返し回数	内部繰り返し回数	MP 関節角度 [deg]	PIP 関節角度 [deg]
1	2	13.92	45.00
2	4	0.00	36.32
3	7	1.19	24.06
4	8	7.71	11.97
5	2	8.04	11.36
6	9	8.82	4.51
7	7	10.21	1.12
8	11	11.93	0.09

中指 MP 関節を 10 度前屈

と定義する。ただし、 e_ang は推定角度、 t_ang は設定した関節角度である。PIP 関節の初期角度は 45 度、MP 関節の初期角度は 0 度としている。

これらの図から復元レベルが大きい方が推定精度が上がるということが容易に読み取れる。特に PIP 関節角度の推定には復元レベルが大きな影響を与えている。以下、全ての実験は Octree レベルを 6 として行う。

第6章 手形状推定実験

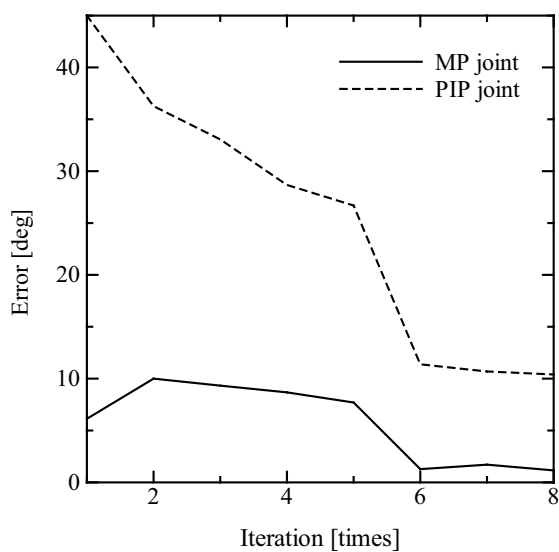


図 6.4: MP 関節・PIP 関節の角度の推定誤差 (1) Level: 5

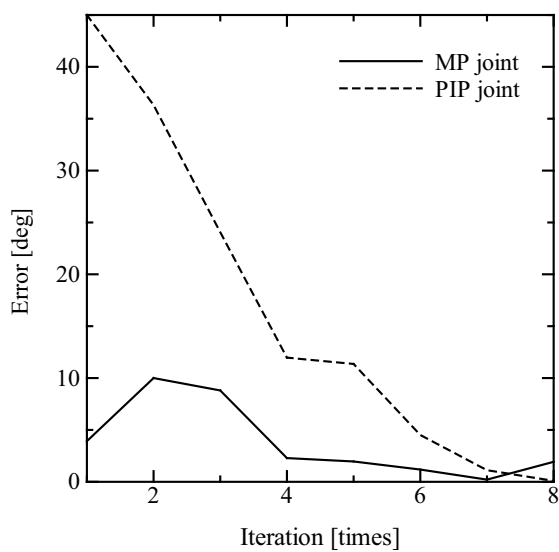


図 6.5: MP 関節・PIP 関節の角度の推定誤差 (2) Level: 6

6.4 手の位置姿勢が未知の場合の推定

5.3 節の要領で、手の Voxel から、手の位置姿勢と各関節角の推定を行った。

図 6.6, 6.7, 6.8 はその様子を表している。これに対し、図 6.9, 6.10, 6.11 は、図 6.6, 6.7, 6.8 各々の Voxel モデル作成前の観察された手の位置を元の座標系に変換し、Voxel モデルと交換したものである。これを見ると、よくフィッティングしているように見えていた図 6.7 など、実は誤差が生じていることが分かる。

一般的に、視体積交差による Visual hull[20, 21] の体積は、本来の体積より大きくなる。手の場合、特に手掌側がカメラからのセルフオクルージョンにより、体積が大きくなる。そのため、体積が大きくなることで広がった部分が手の表面形状モデルとフィッティングされることによるためと考えられる。そのため、各関節の変位が大きくなればなるほど、フィッティング時の表面形状モデルとの誤差が大きくなる。その意味では、精度が不十分であるという結果となった。

6.5 手のトラッキング

手のフィッティング時に 3DTM の特性により、初期位置姿勢を完全に未知として計算した場合、推定結果の精度が非常に低くなる。そのため、常に人間が手動で初期位置姿勢をある程度あわせる必要がある。しかし、これでは人への負担が非常に大きくなり、実用的とは言い難い。

また、実際に人間が作業を教示する場合、手の位置姿勢を時間とともに変化させることが多い。そこで、手の変位を時間の流れとともに観察する必要がある。一般的に、カメラで撮影した画像の変位は、フレーム間では微小なものとなる。そこで、初期フレームに関しては、人が手動で位置姿勢を設定し、その後のフレームは常に 1 つ前のフレームから算出した位置姿勢をそのフレームにおける初期位置姿勢とすることで、連続的に位置姿勢を推定し、人の手のトラッキングを行った。ただし、手の関節形状は時間変化に関係なく、位置姿勢のみが変化していると仮定する。

図 6.12 のようにシミュレータで手を移動させ、各フレームごとに位置姿勢・関節形状をそれぞれ算出した。図 6.13 ではその結果を示す。

第6章 手形状推定実験

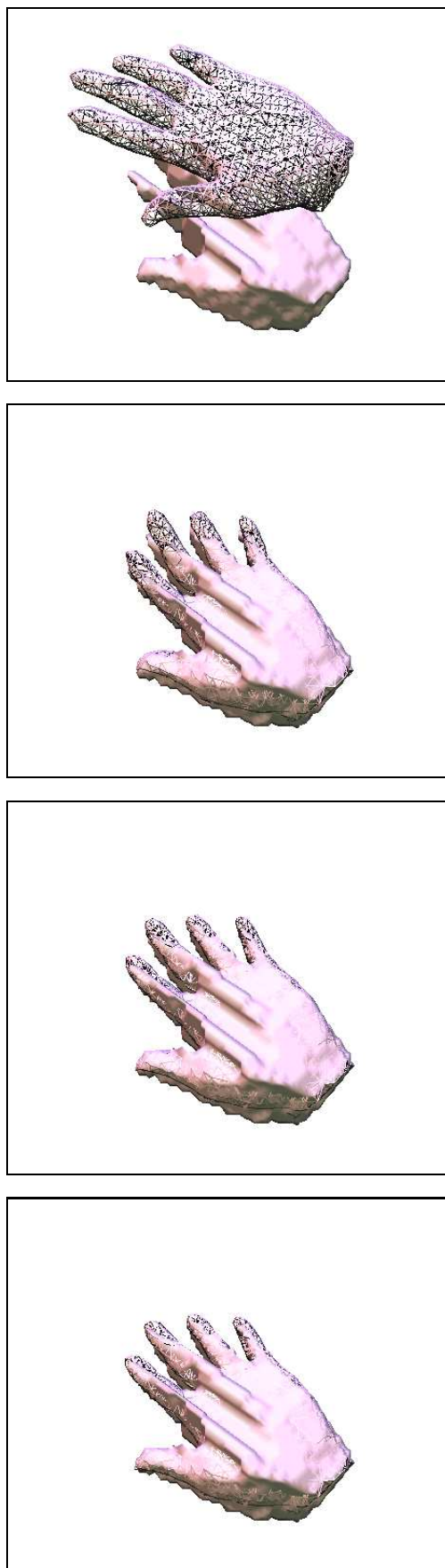


図 6.6: 手の位置姿勢が未知の場合のフィッティング (1)

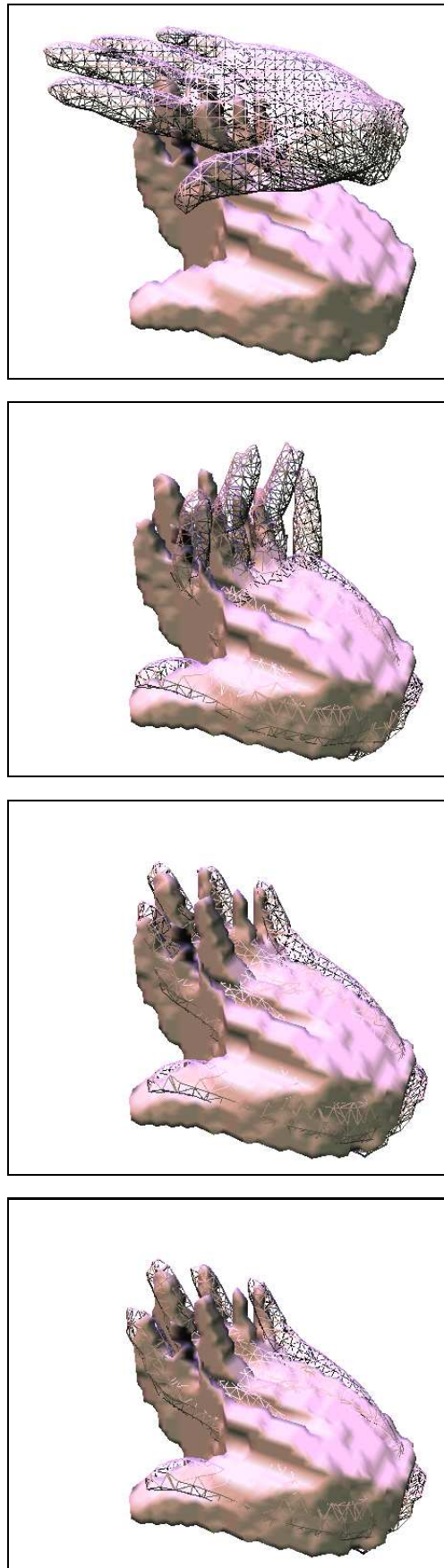


図 6.7: 手の位置姿勢が未知の場合のフィッティング (2)

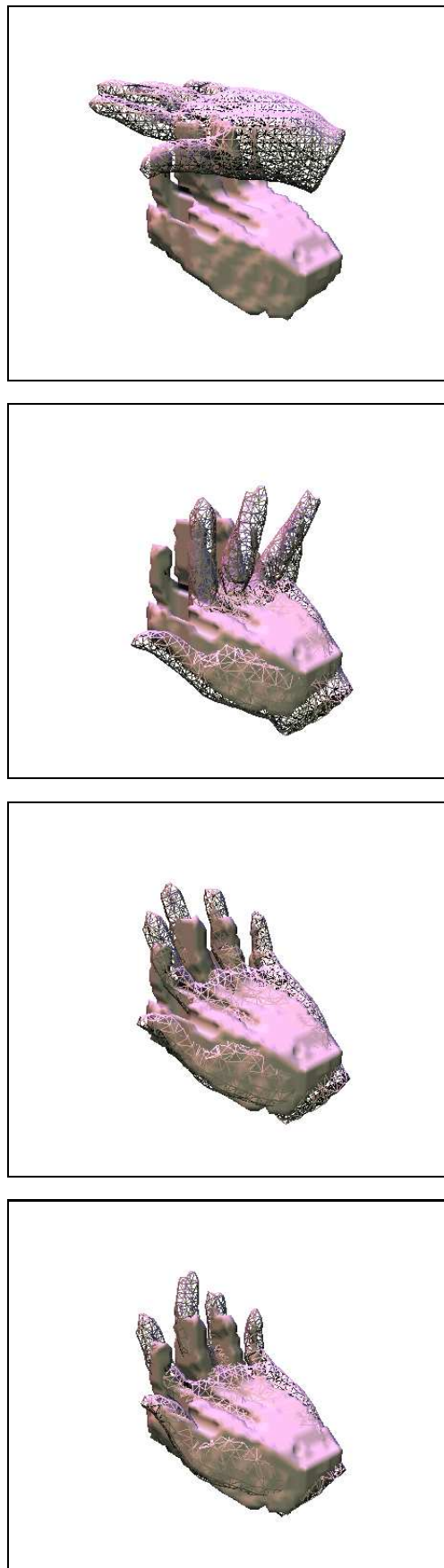


図 6.8: 手の位置姿勢が未知の場合のフィッティング (3)

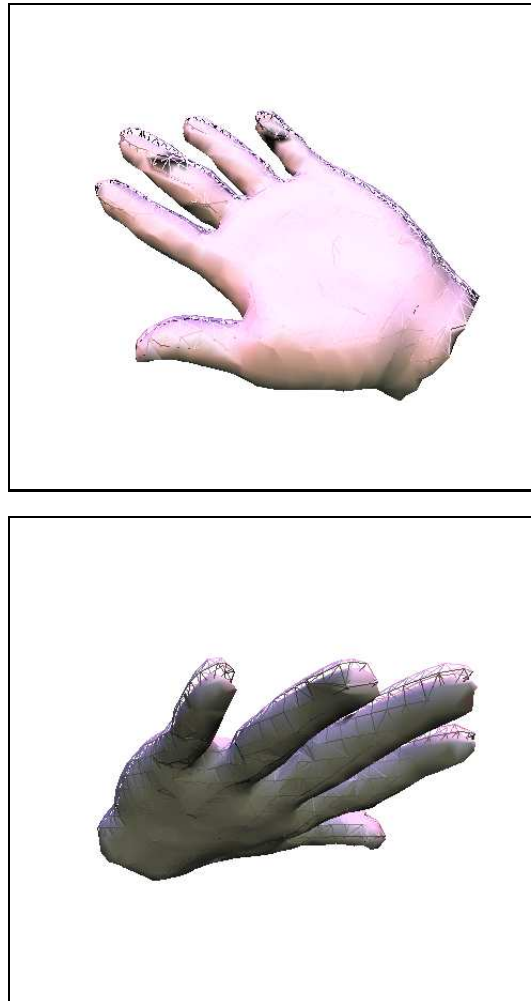


図 6.9: フィッティング後に Voxel モデルと観察モデルを交換 (1)

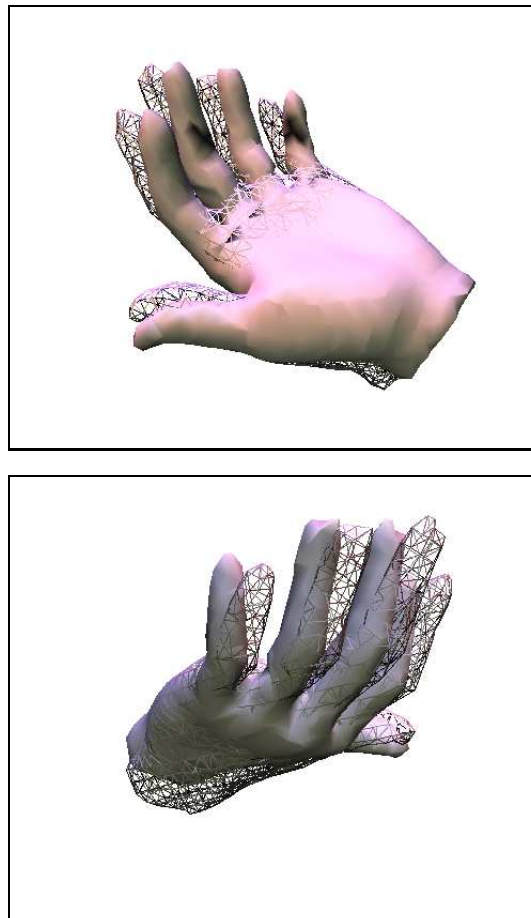


図 6.10: フィッティング後に Voxel モデルと観察モデルを交換 (2)

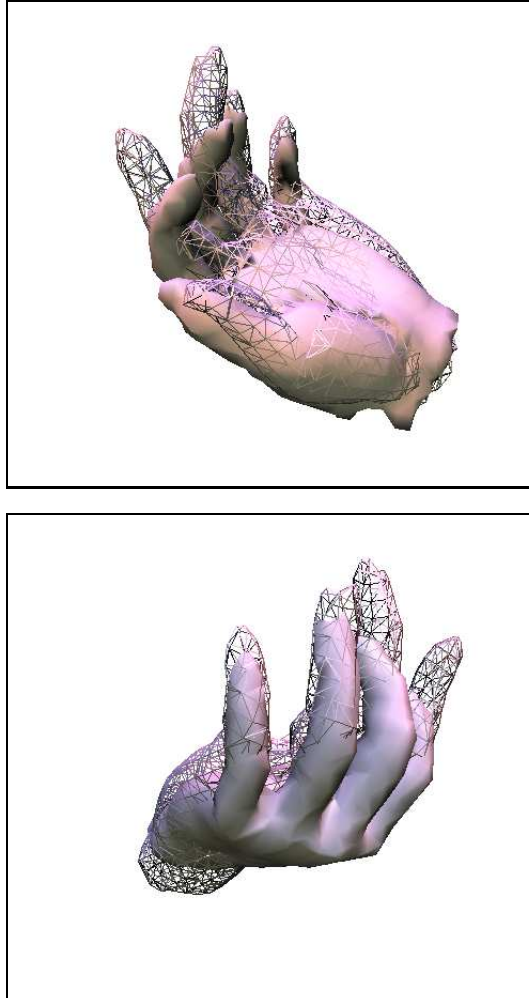


図 6.11: フィッティング後に Voxel モデルと観察モデルを交換 (3)

第6章 手形状推定実験

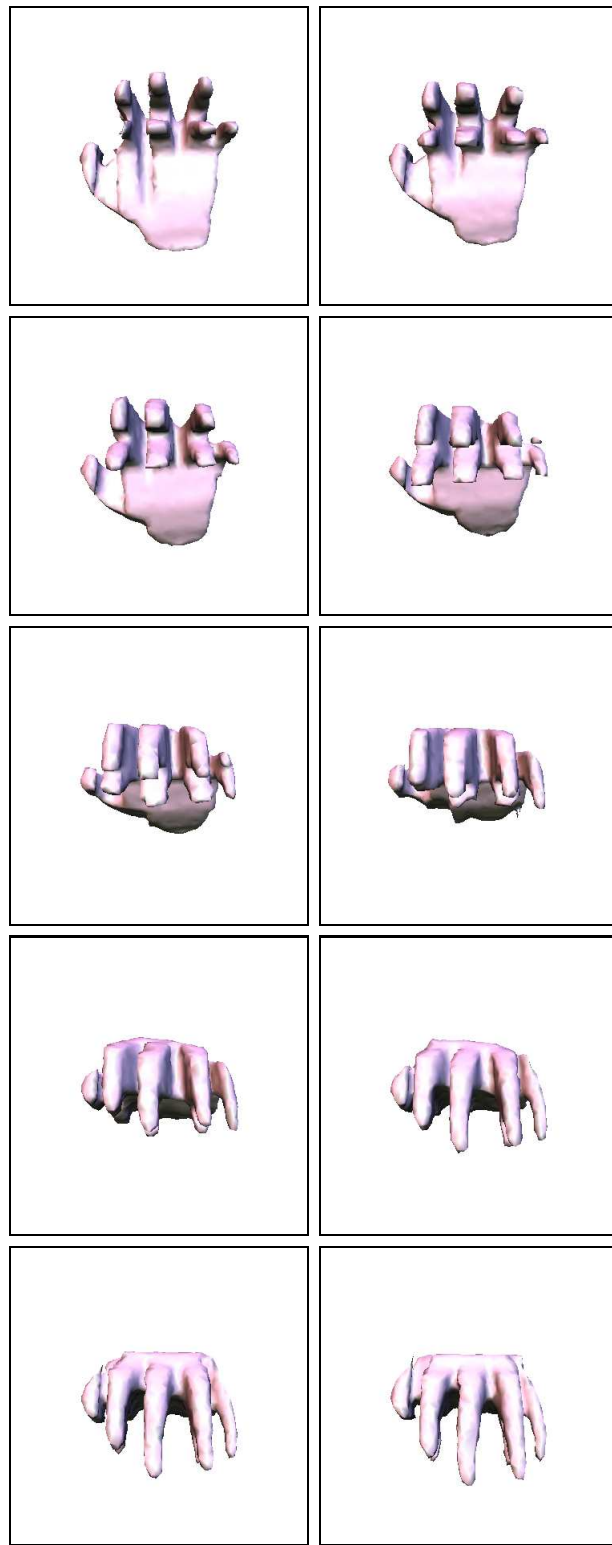


図 6.12: 位置姿勢が未知の場合の手の Voxel モデル (1)

6.5. 手のトラッキング

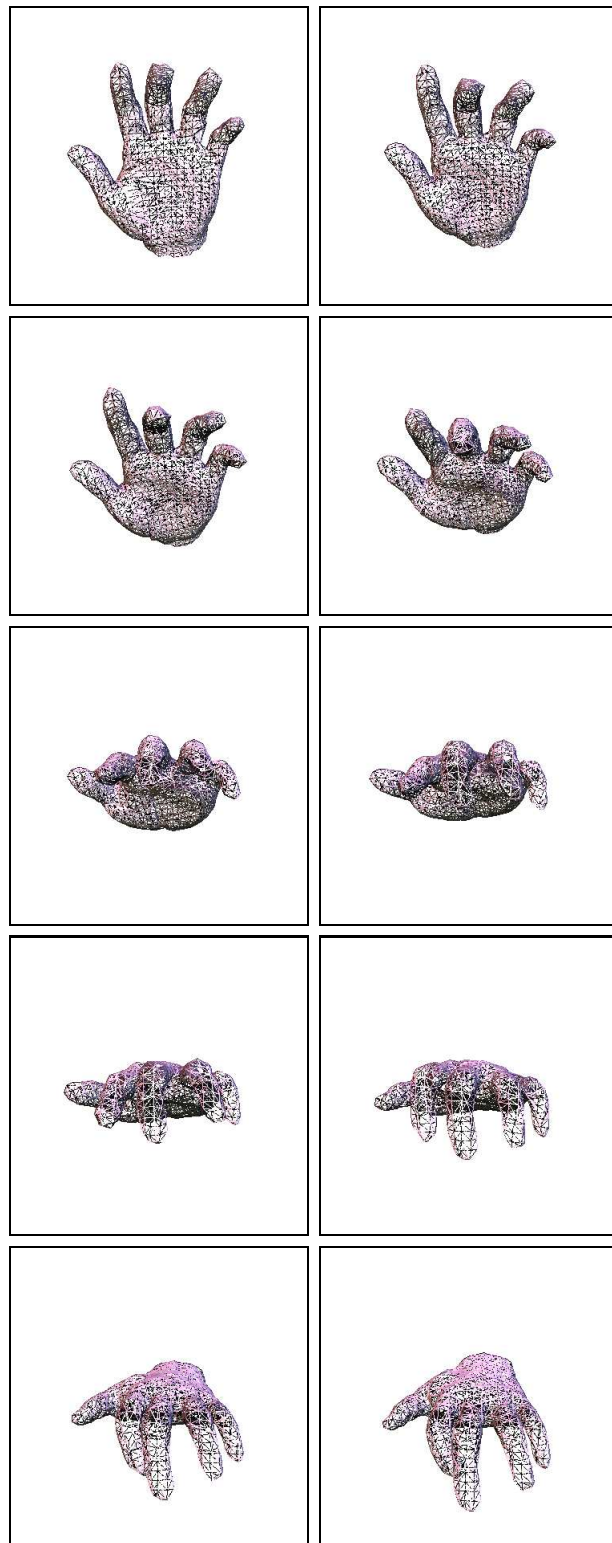


図 6.13: 位置姿勢が未知の場合の手のトラッキング結果 (1)

6.6 把持物体を考慮した手のトラッキング

本論文の目的は、人の手の動きをロボットに観察，理解させた上で，模倣させることである．人が作業を行う場合，物体を把持していることが多い．そのため，物体を把持している場合にも手の形状を認識する必要がある．しかし，カメラの位置によっては，把持物体によるオクルージョンが発生し，復元された手の Voxel モデルの一部が抜け落ちている場合があり，このような場合，形状を認識するのは非常に難しい．

6.5 節でも述べたように，人間が手の位置姿勢を固定したまま作業を行うことはあまり多くない．そのため，時間変化とともに撮影を行うことで，手の位置姿勢により，あるフレームでは把持物体によるオクルージョンが大きくても，別のフレームでは小さくなる場合や，あるフレームで観察可能であった部分が別のフレームでは観察できず，逆に観察を行えなかった部分が行えるような場合もある．したがって，時間変化とともに関節形状が変化しないと仮定すれば，時間変化における位置姿勢の変化から，手の関節形状と位置姿勢の軌跡を推定することができる．

6.7 実画像を用いた実験

6.7.1 実験システム

実際の手の赤外画像に対して提案手法を適用するために実カメラシステムを構築した．図 6.16 のように，縦 1500[mm]，横 1500[mm]，高さ 2000[mm] のフレームを構築し，赤外線カメラを 3 台設置し，手の画像を撮影する．カメラの位置は正面，側面，上面である．また，赤外線カメラは Nikon の ‘サーマルビジョン LAIRD-S270’ を使用している．

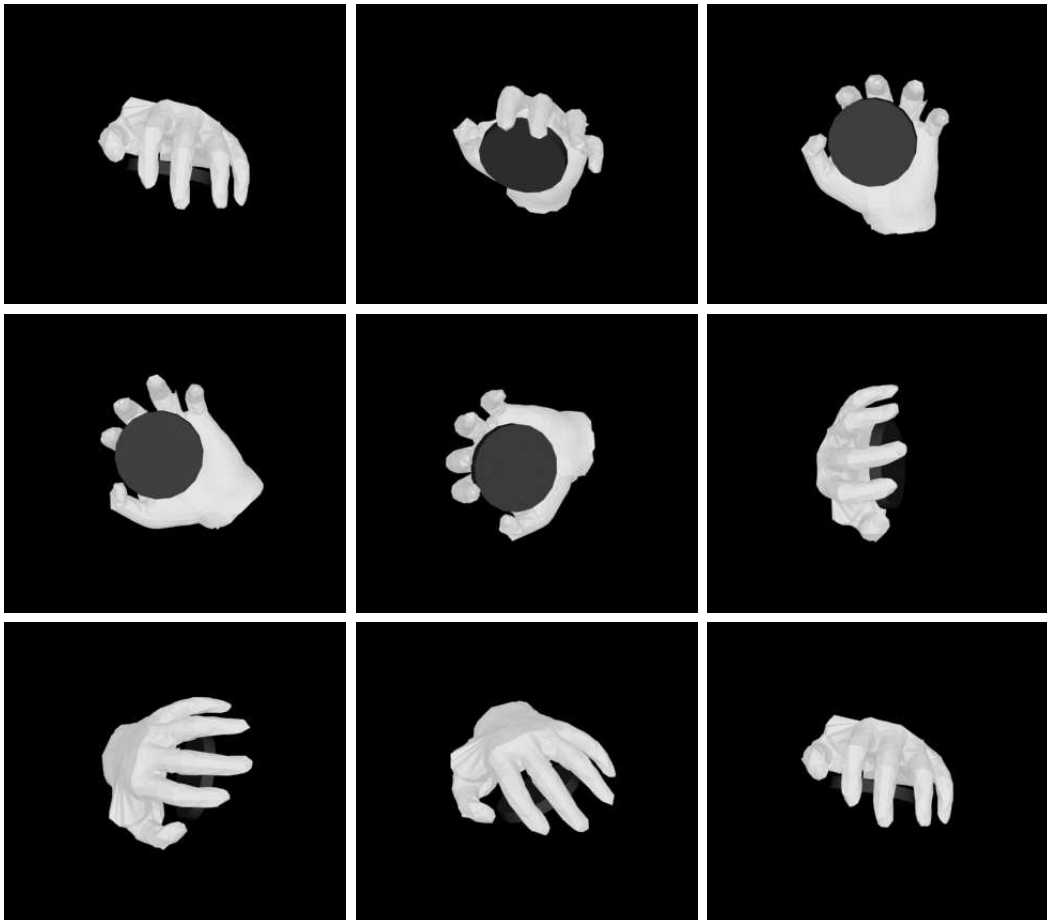


図 6.14: 手の位置姿勢が未知の場合のフィッティング (1)

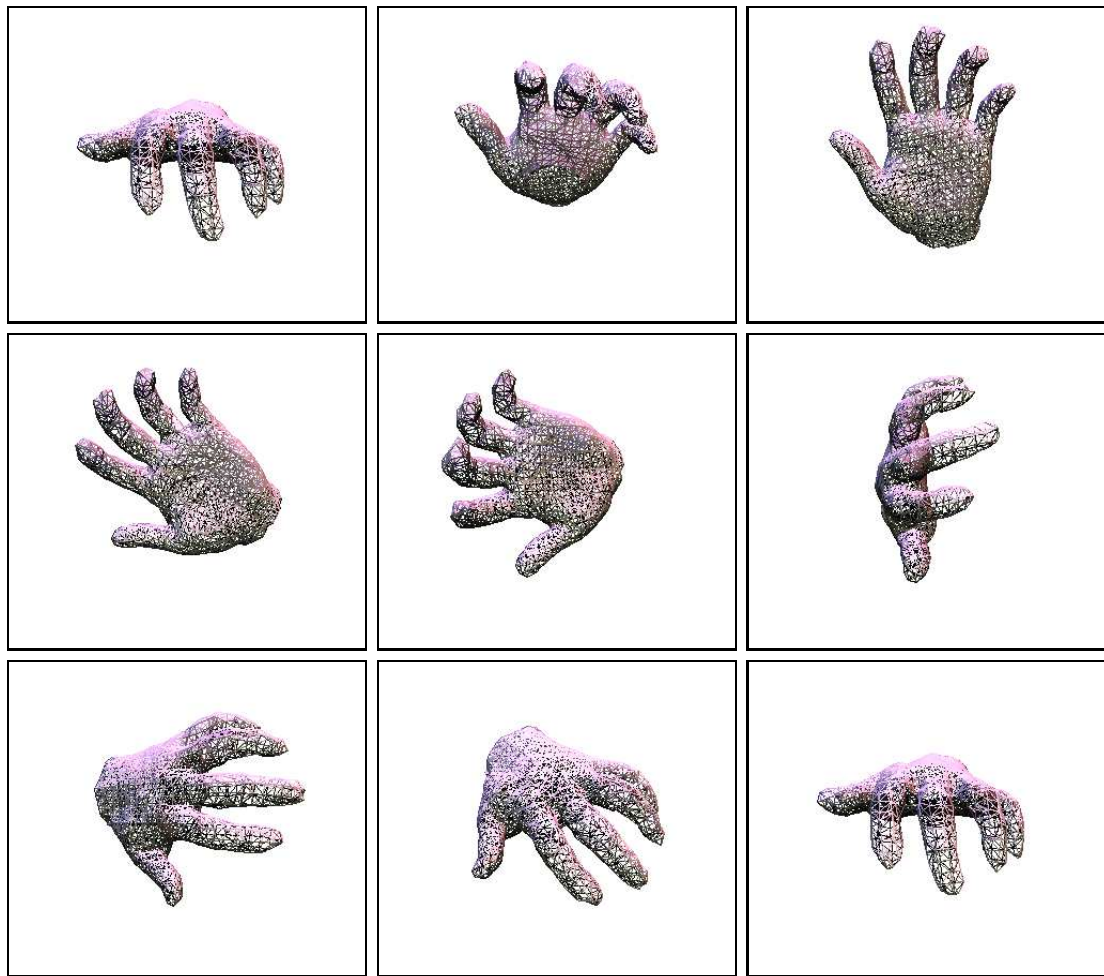


図 6.15: 手の位置姿勢が未知の場合のフィッティング (2)



図 6.16: 実空間上での実験環境

6.7.2 赤外線カメラ用カメラキャリブレーション

実際にキャリブレーションを行うとき、一般的にはチェスのマス目状の模様付きのボードをカメラで撮影し、格子点を検出することにより、キャリブレーションを行うことが多い。しかし、赤外線カメラは撮影した物体の表面温度を検出するデバイスのため、マス目ごとに温度の差がない限り、このような方法を適用することはできない。

そこで、図 6.17 のようなボックスを作成した。このボックスは、等間隔に、横 64 個、縦 4 個の豆電球が設置してある。豆電球を点けると、フェラメントが発熱するため、周囲よりも高温になる。したがって、このボードを赤外線カメラで撮影すると、図 6.18 に示すように、豆電球の部分を容易に検出できる。

第6章 手形状推定実験

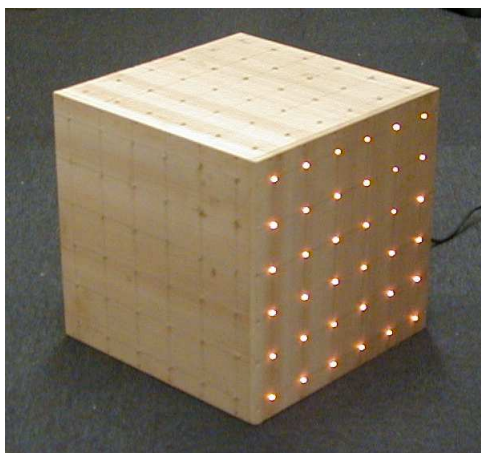


図 6.17: 赤外線カメラ専用キャリブレーションボックス

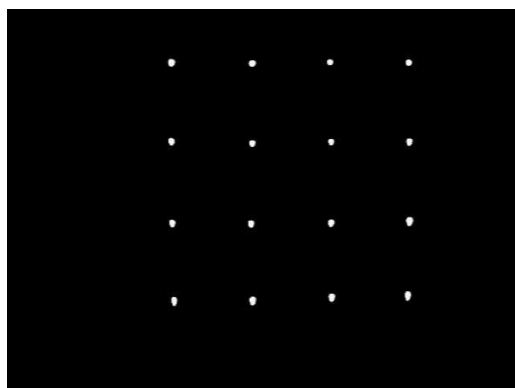


図 6.18: 赤外線カメラで撮影したキャリブレーションボックス

6.7.3 連続画像入力による手形状推定

実画像における実験では、赤外線カメラにより非接触で行っているため、シミュレータ実験のように定量的な位置姿勢・関節形状の真値との比較などを行うのは困難である。そのため、本節では入力画像と出力画像の比較は“見ため”で行う。

6.7. 実画像を用いた実験

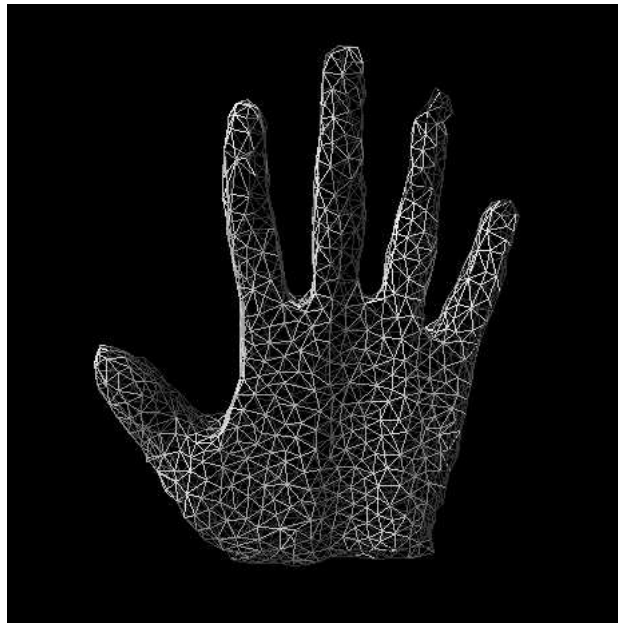


図 6.19: 実画像から作成した手の Voxel モデル

第6章 手形状推定実験

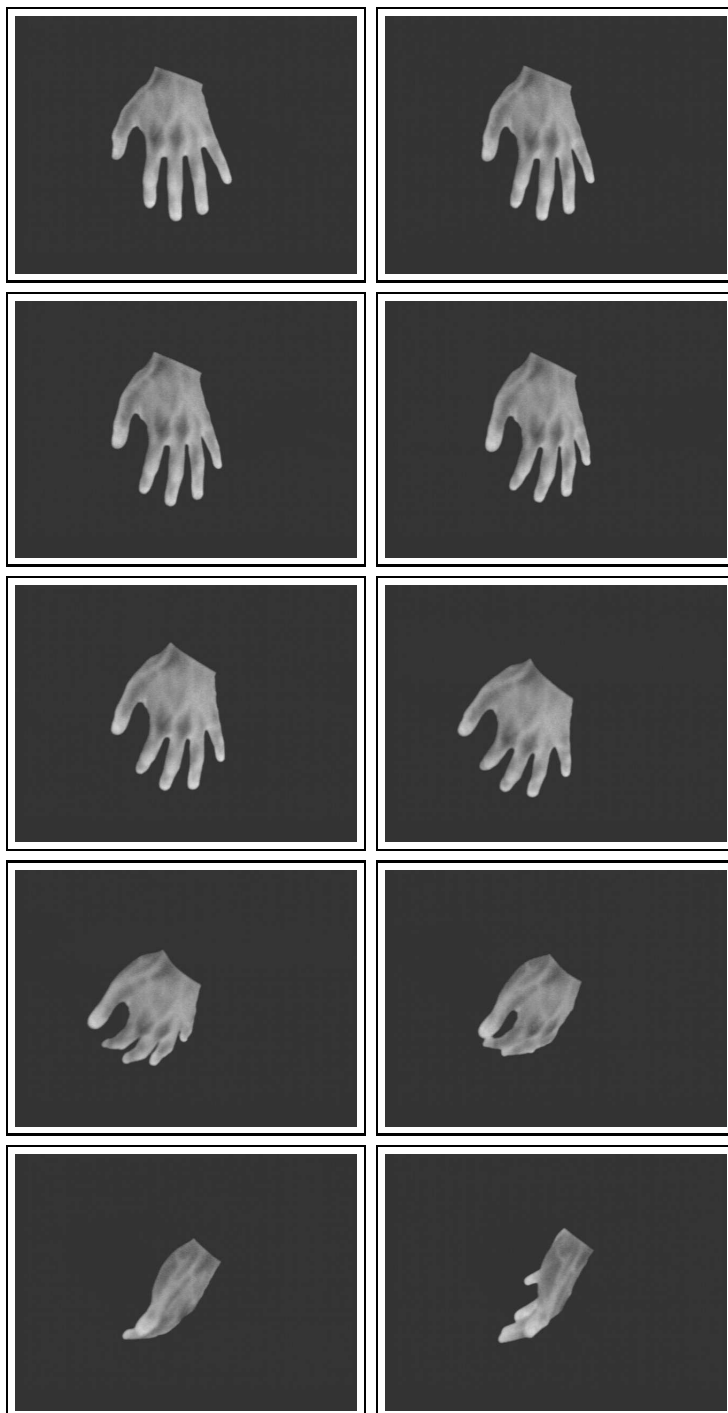


図 6.20: 手の赤外線画像 (1)

6.7. 実画像を用いた実験

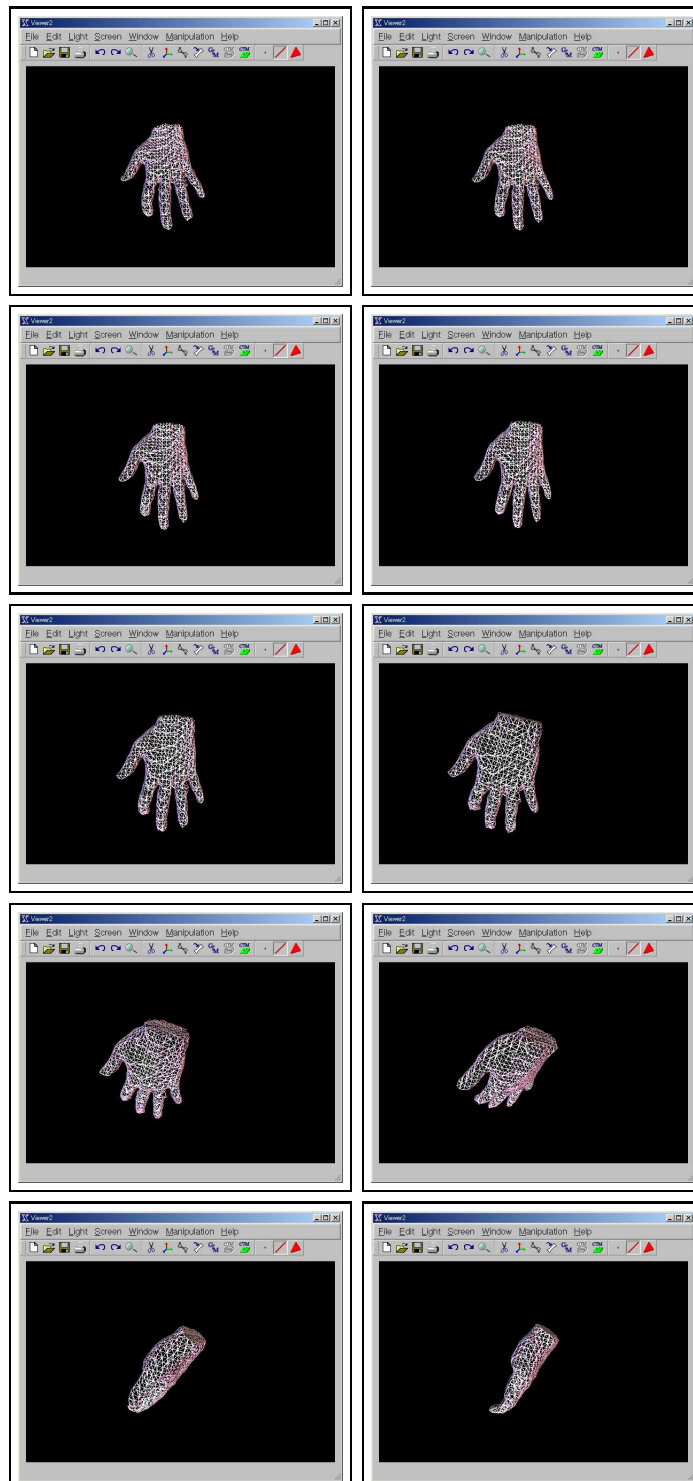


図 6.21: 復元データ (1)

第6章 手形状推定実験

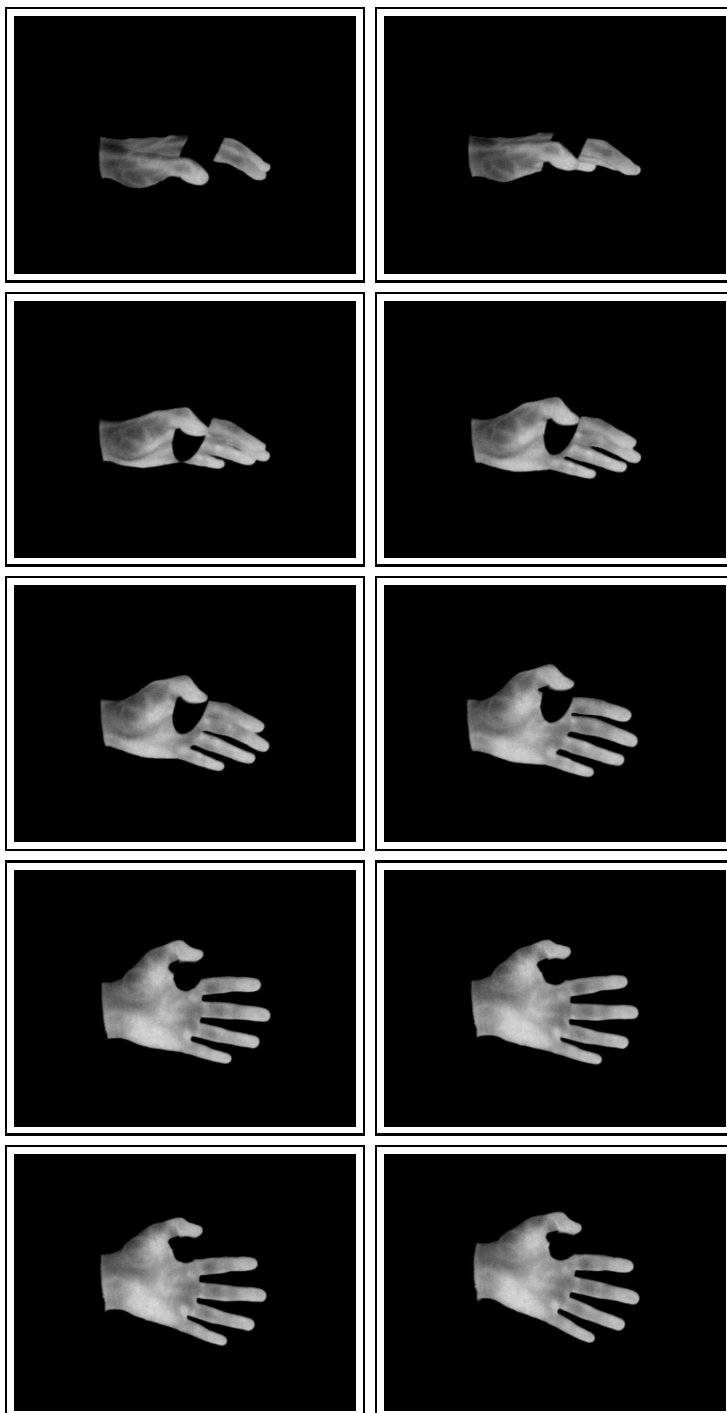


図 6.22: 手の赤外線画像 (2)

6.7. 実画像を用いた実験



図 6.23: 復元データ (2)

第6章 手形状推定実験

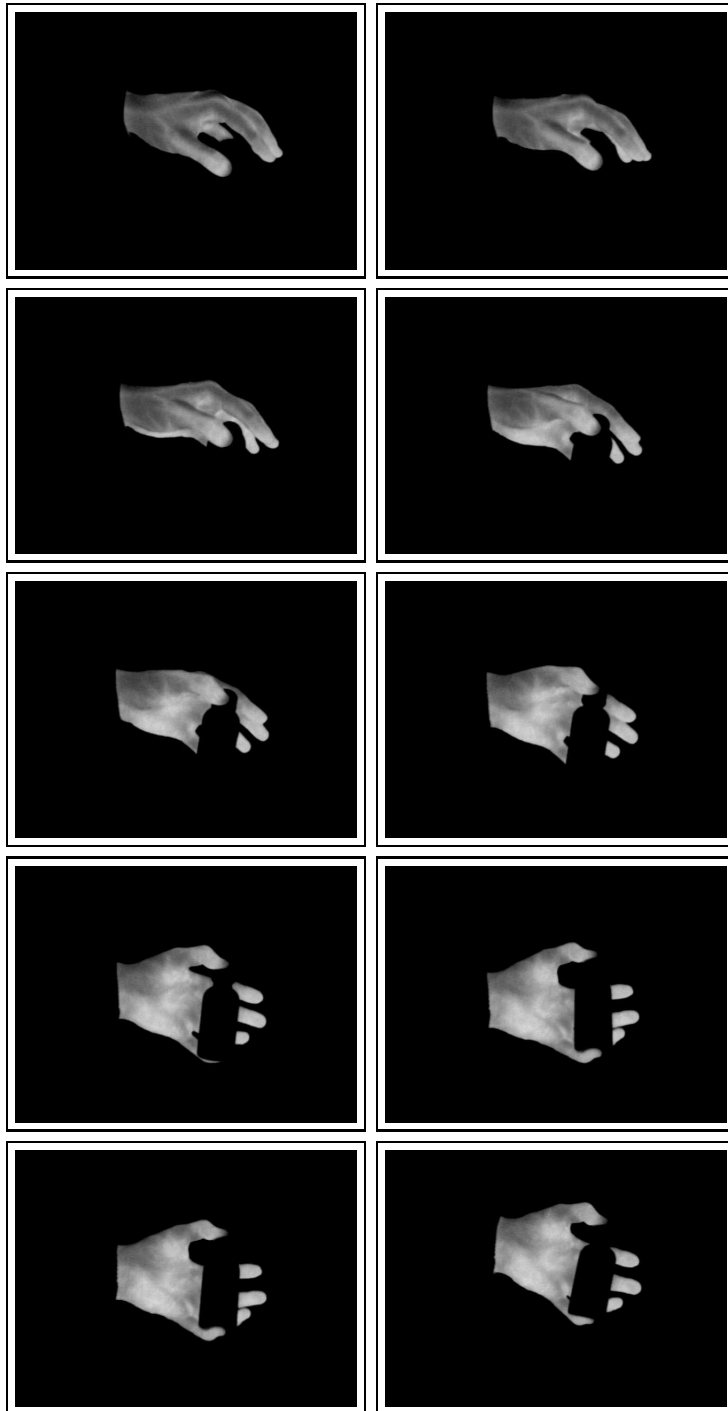


図 6.24: 手の赤外線画像 (3)

6.7. 実画像を用いた実験

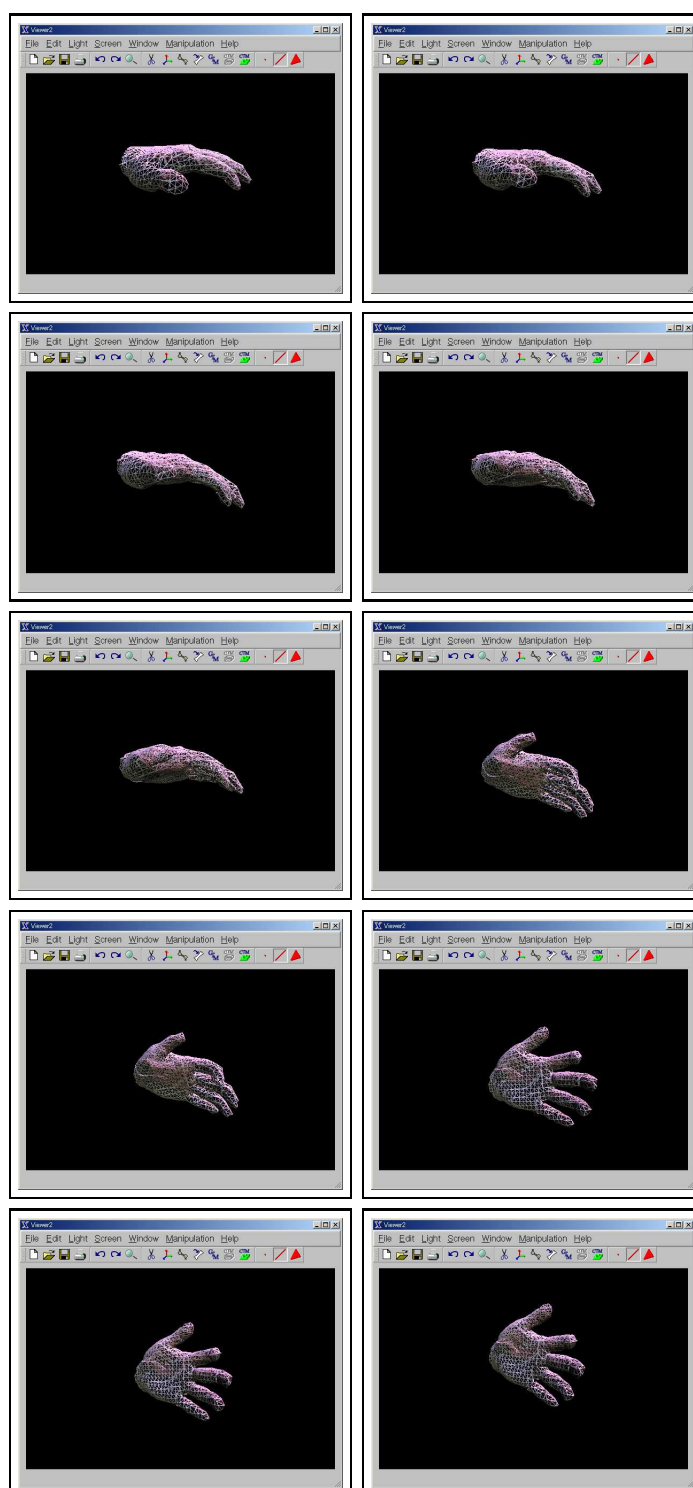


図 6.25: 復元データ (3)

第7章

考察

7.1 精度に対する考察

本提案手法では，計測した手のレンジデータを忠実に再現していればしているほど推定精度が向上するという性格を持つ．この問題点はシミュレータ上，実空間上には関わらない．

1つ目の解決方法として，撮影時のカメラの台数を増やすことが挙げられる．例えば，カメラが無数に存在すれば，より忠実な手の Visual hull の構築を行うことができる．しかし，本論文で使用した赤外線カメラなどは非常に高価であることや，空間的な制約により，容易に増やすことはできない．

2つ目の解決方法は，ある個人に特化した精密な手形状の3次元計測を行い，表面形状データを作成する方法がある．この方法では，シミュレーションのように推定精度は上がるが，モデル対象者以外の手形状推定が容易に行えないという欠点を持つ．

3つ目の解決方法は，階層数を大きくしたレンジデータから表面形状データを作成するアプローチが考えられる．このアプローチでは，より正確な手の表面形状データを作成できるが，フィッティング時の計算コストが非常に高くなり，推定時間を短縮するためには不都合である．そこで，頂点数を減らして (simplify) 計算することもできる．しかし，この場合もフィッティング時の計算コストは削減されるが，simplify 時の計算コストが高くなる．

また，母指関節の運動が他の指の関節運動に比べ，非常に複雑であることから，シ

7.2. アプリケーションのツールとして考察

ミュレータ、実画像とも精度が低い結果となった。今後は、母指関節運動の基本となる回転座標軸の高精度化を図り、母指の関節運動を忠実に再現しなければならない。

さらに、視体積交差法により作成した Voxel モデルの Visual hull は、本来、元々の観察対象よりも全体的に体積が大きくなってしまいうという問題点がある。また、手の関節を変化したとき、手掌側の形状は非常に精度が低くなってしまふ。そのため、関節形状の変化が大きければ大きいほどフィッティング時の誤差が大きくなり、推定の精度が低くなる。

把持物体が存在する場合、

7.2 アプリケーションのツールとして考察

本論文では、提案した手法により推定した手形状を、ロボットのための人間行動観察用入力インタフェースとして利用することを目的としている。人間が物体を把持し作業を行う“操作的”な手動作は、手話やハンドサインや踊りなどの“Communicative”な手動作に比べ、その動作速度や形状変化が遅いという特徴がある。また、各関節角度の精度より、空間中での手の位置、姿勢の精度の高さを要求される。インタフェースとして考えた場合、手と把持物体との位置関係や、接触姿勢などがより重要な情報となる。

本論文で提案した手法では、計測対象となる手を骨格モデルと Voxel モデルで表現している。そして、これらを用いることで、把持物体がある場合にも手の形状と姿勢を得ることができる。また、本来問題点であった把持物体により発生したオクルージョンから、把持物体の形状推定を行う可能性や、把持物体との接触状態を推定する可能性も有している。したがって、インタフェースとしての応用の可能性は十分実現が可能であると期待できる。

一方、片手での操作のみでは人間の持つ手動作の多様性を十分に生かしきることができないばかりか、非常に使いにくいインタフェースとなる危険性を含んでいる。インタフェースとして十分な機能を果たすためには、両手の位置姿勢・形状推定は

第7章 考察

避けては通れない。提案手法を両手に拡張するために必要となる事項には以下を挙げることができる。

1. 相互オクルージョン問題の処理
2. オクルージョンの少ないカメラ配置
3. 左右の手の判定
4. 両手を動かすことのできる自由度の大きい実験環境

しかし、提案手法を両手に拡張することができれば、複数人間が1つの対象を操作することができるインタフェースへの拡張も十分に考えられる。

第8章

おわりに

本論文では、人間の作業動作のロボットへの教示のためのインタフェースとして、把持物体がある場合にも利用可能なビジョンベースの手形状推定手法を提案した。提案手法では、手を骨格モデルと表面形状モデルで表現し、それらを統合した手のモデルを用いている。その上で、骨格モデルの関節角度を推定することにより、手全体の形状と位置姿勢を推定していくものである。

観測対象の手は、オクルージョンの影響がより少なく、複数のカメラの相互間の位置関係やキャプチャ画像間のマッチング処理などの必要ない多視点カメラシステムにより撮影される。実験では、互いの位置関係に制約のない3台の赤外線カメラを用いて撮影した。得られた画像より手の領域をシルエット化し、視体積交差法により空間分割表現 (Octree 表現) で統合し、1つの手の3次元形状を再構成する。本論文では、この再構成された手形状を Voxel モデルと呼び、空間中の位置や形状を表現するモデルとしてシステムで使用した。Voxel モデルの再構成の後、simplify 処理を行うことで、フィッティング時の計算速度の高速化を図った。

手の表面形状モデルを観測データである Voxel モデルにフィッティングする方法で関節角度と、位置姿勢を推定した。フィッティングには、オクルージョンなどにもロバストな3次元データの位置姿勢フィッティングアルゴリズム (3DTM) に関して、関節25自由度も扱えるよう拡張した手法を提案した。

シミュレータにより生成した各種の手形状に対し、提案手法を適用し、把持物体がある場合にも形状推定が行えることを確認し、さらに実画像を入力として提案手

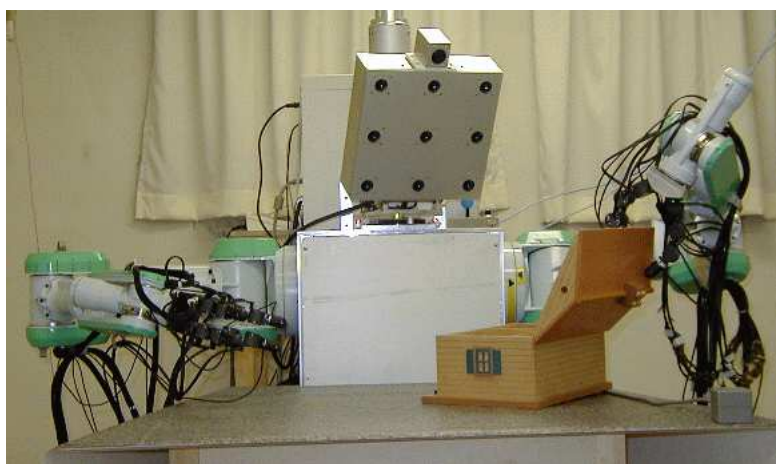


図 8.1: 使用しているロボット “ドットちゃん”

法の有効性を確認するための実験を行った。その結果、把持物体がある場合にも手の位置姿勢と形状を推定を行うことができた。また、画像を連続的にキャプチャし、推定を行うことで、手の動きを追従できることも確認した。

実験を通じて、本論文で提案した人間行動のロボットへの教示は、モデルフィッティングの精度や、Voxel モデル作成時の精度の向上により実現可能であるということがわかった。今後は、現時点での問題点を順次解決し、我々が使用しているロボット (図 8.1) に実際に人間行動の模倣の実用化を目指していく予定である。

謝辞

本研究を進めるにあたって、毎週打ち合せをして頂き、すばらしい研究環境と適切な助言・ご指導をいただいた池内克史教授に心から感謝いたします。研究だけでなく、生活、人生に関して心づかいをいただき、研究生活の支えになりました。

研究を遂行する上で、困ったとき、苦しいとき、大きな壁にぶち当たったとき、多くの助言や協力をしていただき、かつ、共同研究者として研究に関する様々なアドバイスをいただいた、電気通信大学 木村浩助教授、小松製作所 田貫富和さん、小川原光一博士研究員、博士2年 高松淳さん、佐藤啓宏さん、修士2年 森田拓磨さん、園田展人さん、修士1年 崎田健二さん、Keni Bernardin 研究生に深く感謝します。

研究室の生活を送るにあたりお世話になった、九州大学 倉爪亮助教授、影澤政隆助手、長谷川仁則技官、中澤篤志博士研究員、西野恒博士研究員、Supatana Auethavekiat 博士研究員、原健二協力研究員、幸村琢協力研究員、元木恵子秘書、松浦芳子秘書、斉木優子秘書、平山幸恵秘書、九州電力 河村憲太郎さん、Carnegie Mellon University 竹内秀人さん、楊進華研究生、松田真理研究生、Paul Reuter 研究生、大野一さん、佐川立昌さん、川崎洋さん、山崎俊太郎さん、Robby T. Tan さん、運天弘樹さん、平原清隆さん、吉田達哉さん、工藤俊亮さん、高橋徹さん、宮崎大輔さん、大石岳史さん、佐藤いまりさん、小野晋太郎さん、朱成華さん、Shirmila Mohottala さん、中岡慎一郎さん、増田智仁さん、小川希さん、吉藤伸幸さん、猪狩壮文さん、上原康彦さん、原田貴昭さん、白鳥貴亮さん、廣田祐一郎さん、高橋拓二さん、村尾真洋さん、三枝旭さん、稲熊伸昭さん、射場総司さん、ドットちゃんに感謝します。

坂内研究室や佐藤研究室の方々には研究の遂行や発表練習などにご協力いただき、大変お世話になりました。ここに感謝致します。

どんなときも精神的な支えとなり、暖かく見守ってくれた関山瑞恵さんをはじめとする多くの友人・知人たちに心より感謝します。

そして最後に、本当に長い間、金銭的、肉体的、そして精神的に学生生活を常に陰で支えてくれた、両親をはじめとする家族に、深く深く感謝します。

参考文献

- [1] S. B. Kang and K. Ikeuchi : “Toward automatic robot instruction from perception–temporal segmentation of tasks from human hand motion,” *IEEE Trans. Robot. Automat.*, Vol. 11, No. 5, pp. 670–681, Oct. 1995.
- [2] S. B. Kang and K. Ikeuchi : “Toward automatic robot instruction from perception–mapping human grasps to manipulator grasps,” *IEEE Trans. Robot. Automat.*, Vol. 13, No. 1, pp. 81–95, Feb. 1997.
- [3] Y. Kuniyoshi, M. Inaba, and H. Inoue : “Learning by watching: extracting reusable task knowledge from visual observation of human performance,” *IEEE Trans. Robot. Automat.*, Vol. 10, No. 6, pp. 799–822, Dec. 1994.
- [4] V. I. Pavlovic, R. Sharma, and T. S. Huang : “Visual interpretation of hand gestures for human-computer interaction: A review,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 677–695, Jul. 1997.
- [5] J. Weng and Y Cui : “Recognition of hand signs from complex backgrounds,” In R. Cipolla and A. Pentland, editors, *Computer Vision for Human-Machine Interaction*, pp. 235–265, Cambridge University Press, 1998.
- [6] 今川和幸, 呂山, 猪木誠二, 松尾英明 : “肌色抽出により隠れて見える場合を考慮した手話動画像からの手の実時間追跡,” 電子情報通信学会論文誌, Vol. J81-D-II, No. 8, pp. 1787–1795, Aug. 1998.
- [7] C. J. Taylor, A. Lanitis, T. F. Cootes, G. Edwards, and T. Ahmad : “Model-based interpretation of faces and hand gestures,” In R. Cipolla and A. Pentland,

- editors, *Computer Vision for Human-Machine Interaction*, pp. 217–233, Cambridge University Press, 1998.
- [8] M. Yachida and Y. Iwai : “Looking at human gestures,” In R. Cipolla and A. Pentland, editors, *Computer Vision for Human-Machine Interaction*, pp. 291–311, Cambridge University Press, 1998.
- [9] 宮里 勉 岸野文郎 : “臨場感通信会議における仮想空間を介した指差し指示動作知覚の評価,” 電子情報通信学会論文誌, Vol. J81-D-II, No. 5, pp. 1221–1230, May. 1997.
- [10] 内海章, 大谷淳, 中津良平 : “多数カメラを用いた両手手振りの検出,” 情報処理学会論文誌, Vol. 40, No. 8, pp. 3143–3154, Aug. 1998.
- [11] C. Maggioni and B. Kämmerer : “Gesture computer – history, design and applications,” In R. Cipolla and A. Pentland, editors, *Computer Vision for Human-Machine Interaction*, pp. 23–52, Cambridge University Press, 1998.
- [12] K. Ogawara, S. Iba, T. Tanuki, H. Kimura, and K. Ikeuchi : “Acquiring hand-action models by attention point analysis,” In *Proc. of International Conf. Robotics and Automations (ICRA) 2001*, Vol. 4, pp. 465–470, Seoul, Korea, May. 2001.
- [13] 舟橋健司, 安田孝美, 横井茂樹, 鳥脇純一郎 : “サイバースペースのためのグラフィックス, ディスプレイ,” 電子情報通信学会論文誌, Vol. J81-D-II, No. 5, pp. 822–831, May. 1998.
- [14] T. Massie : “A tangible goal for 3D modeling,” *IEEE Computer Graphics and Applications*, Vol. 18, No. 3, pp. 62–65, May./Jun. 1998.
- [15] N. Shimada, Y. Shirai, and Y. Kuno : “3-D pose estimation and model refinement of an articulated object from a monocular image sequence,” In *Proc. of*

参考文献

- The 3rd Asian Conf. on Computer Vision*, pp. 672–679, 1998.
- [16] 渡辺賢, 岩井儀雄, 八木康史, 谷内田正彦 : “カラーグローブを用いた指文字の認識,” 電子情報通信学会論文誌, Vol. J80-D-II, No. 10, pp. 2713–2722, Oct. 1997.
- [17] J. Veenstra and N. Ahuja : “Efficient octree generation from silhouettes,” In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 537–542, Miami Beach, Florida, Jun. 1986.
- [18] W. N. Martin and J. K. Aggarwal : “Volumetric descriptions of objects from multiple views,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 5, No. 2, pp. 150–158, Mar. 1987.
- [19] C. H. Chien and J. K. Aggarwal : “Model construction and shape recognition from occluding contours,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 11, No. 4, pp. 372–389, Apr. 1989.
- [20] A. Laurentini : “The visual hull concept for silhouette-based image understanding,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 16, No. 2, pp. 150–162, Feb. 1994.
- [21] A. Laurentini : “How far 3D shapes can be understood from 2D silhouettes,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 17, No. 2, pp. 188–195, Feb. 1995.
- [22] Y. Kameda, M. Minoh, and K. Ikeda : “Three dimensional pose estimation of an articulated object from its silhouette image,” In *Proc. of The Asian Conf. on Computer Vision*, pp. 612–615, 1993.
- [23] J. M. Rehg and T. Kanade : “Model-based tracking of self-occluding articulated objects,” In *Proc. of 5th International Conf. on Computer Vision*, pp. 612–617, Cambridge, UK, Jun. 1995.

- [24] 齋藤真希子, 佐藤洋一, 小池英樹 : “Perceptual glove: 多視点画像に基づく手形状・姿勢の実時間入力とその応用,” 情報処理学会論文誌, Vol. 43, No. 1, pp. 185–194, Jan. 2002.
- [25] Q. Delamarre and O. Faugeras : “3D articulated models and multiview tracking with physical forces,” *International Journal of Computer Vision and Image Understanding, Special Issue on Modelling People*, Vol. 81, pp. 328–357, Mar. 2001.
- [26] 高橋和彦, 坂口竜己, 大谷淳 : “三眼視による実時間非接触非装着型三次元人物姿勢推定法,” 電子情報通信学会パターン認識・メディア理解研究会, PRMU, pp. 47–54, Oct. 1999.
- [27] 大野一, 池内克史 : “多視点ステレオカメラによる手のトラッキング,” 電子情報通信学会パターン認識・メディア理解研究会, PRMU, pp. 45–52, May. 1999.
- [28] E. Ueda, Y. Matsumoto, M. Imai, and T. Ogasawara : “Hand pose estimation using multi-viewpoint silhouette images,” In *Proc. of The 2001 IEEE/RSJ International Conf. on Intelligent Robots and Systems (IROS2001)*, pp. 1989–1996, Maui, Hawaii, USA, Oct. 2001.
- [29] 鎌倉矩子 : “手のかたち 手のうごき,” 医歯薬出版株式会社, 1989.
- [30] Y. Yasumuro, Q. Chen, and K. Chihara : “Three-dimensional modeling of the human hand with motion constraints,” *Image and Vision Computing*, Vol. 17, No. 2, pp. 149–153, 1999.
- [31] 安室喜弘 : “運動の制約を考慮した手の3次元CGモデリングに関する研究,” Ph. D. thesis, 奈良先端科学技術大学院大学 情報科学研究科情報処理専攻, Feb. 2000.

参考文献

- [32] P. J. Neugebauer : “Reconstruction of real-world objects via simultaneous registration and robust combination of multiple range images,” *International Journal of Shape Modeling*, Vol. 3, No. 1&2, pp. 71–90, Jun. 1997.
- [33] R. Sagawa, K. Nishino, M. D. Wheeler, and K. Ikeuchi : “Parallel processing of range data merging,” In *Proc. IEEE/RSJ International Conf. on Intelligent Robots and Systems (IROS)*, Vol. 1, pp. 577–583, Maui, Hawaii, USA, Oct./Nov. 2001.
- [34] W. E. Lorensen and H. E. Cline : “Marching cubes: A high resolution 3D surface construction algorithm,” In *Proc. of Computer Graphics (SIGGRAPH '87)*, pp. 163–169, Anaheim, USA, 1987, ACM Press.
- [35] 剣持雪子, 小谷一孔, 井宮淳 : “点の連結性を考慮したマーチング・キューブ法,” 電子情報通信学会パターン認識・メディア理解研究会, PRMU, pp. 197–204, Jan. 1999.
- [36] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle : “Mesh optimization,” In *Proc. of Computer Graphics (SIGGRAPH '93)*, pp. 19–26, NY, USA, 1993, ACM Press.
- [37] P. J. Besl and N. D. McKay : “A method for registration of 3-D shapes,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 14, No. 2, pp. 239–256, Feb. 1992.
- [38] Z. Zhang : “Iterative point matching for registration of free-form curves and surfaces,” *International Journal of Computer Vision*, Vol. 13, pp. 119–152, 1994.
- [39] J. H. Friedman, J. L. Bentley, and R. A. Finkel : “An algorithm for finding best matches in logarithmic expected time,” *ACM Trans. on Mathematical*

- Software*, Vol. 3, No. 3, pp. 209–226, Feb. 1977.
- [40] R. M. Haralick, H. Joo, C. N. Lee, X. Zhuang, V. G. Vaidya, and M. B. Kim : “Pose estimation from corresponding point data,” *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 19, No. 6, pp. 1426–1446, Nov./Dec. 1989.
- [41] D. G. Lowe : “Robust model-based motion tracking through the integration of search and estimation,” *International Journal of Computer Vision*, Vol. 2, No. 8, pp. 113–122, Aug. 1992.
- [42] M. D. Wheeler and K. Ikeuchi : “Sensor modeling probabilistic hypothesis generation,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 17, No. 3, pp. 252–265, Mar. 1995.
- [43] 河村憲太郎, M. D. Wheeler, 山下修, 佐藤洋一, 池内克史 : “メッシュモデルと M-推定法に基づく配電器材の位置推定手法,” *日本ロボット学会誌*, Vol. 18, No. 4, pp. 600–611, 2000.
- [44] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery : “Modeling of data,” In *NUMERICAL RECIPES in C*, chapter 15, pp. 656–706, Cambridge University Press, 1988.
- [45] R. O. Duda, P. E. Hart, and D. G. Stork : “Pattern Classification, Second Edition,” John Wiley & Sons, Inc., New York, USA, 2001.
- [46] B. K. P. Horn : “Robot Vision,” McGraw-Hill Book Company, New York, USA, 1986.
- [47] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery : “Minimization or maximization of functions,” In *NUMERICAL RECIPES in C*, chapter 10, pp. 394–455, Cambridge University Press, 1988.

参考文献

- [48] 徐剛, 辻三郎 : “3次元ビジョン,” 共立出版, 1998.