

平成26年度 卒業論文

指先によるひねり動作を考慮した
少数センサーデータグローブのデータ補正法

指導教員
舟橋 健司 准教授

名古屋工業大学 工学部 情報工学科
平成23年度入学 23115013番

井藤 猛

目次

第 1 章	はじめに	1
第 2 章	従来研究における補正法	4
2.1	指関節間の相関関係に基づくデータ補正	4
2.2	物体形状情報に基づくデータ補正	5
2.2.1	基本把持動作の定義	6
2.2.2	物体寸法に応じた把持動作の補正	7
2.3	代表手動作に基づくデータ補正法	9
2.3.1	センサー値の正規化	9
2.3.2	ユークリッド距離に基づく手動作推定	10
2.3.3	指関節角度の導出	11
2.4	代表手動作の決定	12
2.5	センサー値分布を考慮した補正法	13
2.5.1	センサー値のばらつきを考慮した手動作推定	14
第 3 章	ひねり動作による内転外転を含めたデータ補正法	15
3.1	ひねり動作時のセンサー出力値の計測	15
3.2	把持動作とひねり動作の境界	21
3.2.1	複数の超球による全てのサンプル点の包含	21
3.2.2	超球の決定	22
3.3	ひねり動作における関節角度推定	27
3.3.1	内転外転角度と角指関節角度の推定式	27
3.3.2	境界付近での推定方法	29

第 4 章 実験	31
4.1 実験システムの概要	31
4.2 実験結果と考察	31
第 5 章 むすび	34
謝辞	35
参考文献	36

第1章 はじめに

バーチャルリアリティ (Virtual Reality: 以下 VR) とは、現実そのものではないが、観測者が現実と同様に感じることができる仕組みや技術のことである [1]。VR 技術を利用することで、危険が伴う訓練を危険性を排除して行うことができ、また実際に体験するための環境や実物を準備するための人員およびコストを削減することができる。例えば、医療では難易度の高い手術の訓練を行うことができる [2]。他には、地震などの災害を疑似体験することで防災への意識を高めることを目的とした、ポータブル VR システムの開発が行われている [3]。また、3次元シュミレーションの結果をそのまま3次元空間で解析するために VR システムが使用される [4] など、VR 技術の応用分野は幅が広い。しかしこれらの技術や装置は、大規模または高コストであるため、一般家庭でも利用可能であるコンテンツとはいえない。一方、近年では任天堂株式会社から発売された家庭用ゲーム機「Wii」における入力デバイス「Wii リモコン」 [5] や、Oculus VR 社の製造するヘッドマウントディスプレイ「Oculus Rift」 [6] など、一般家庭でも購入できる価格の様々な入出力装置が登場してきた。今後、これらのような低コストなインターフェースの開発が進み、合わせて一般の人が日常的に利用するコンテンツが増えることで、VR システムが一般家庭に普及していくことが考えられる。

ところで、VR システムでは、ユーザが体験するその環境や操作、またその結果があたかも現実であるかのように感じられることが重要であり、そのためには直感的に操作が可能であることが重要だといえる。直感的な操作を可能とするためのデバイスは様々ある。本論文では人が行う動作は手を用いることが多いという点からデータグローブに注目する。データグローブは、実際の手の動作から各時点における各指関節の角度を取得するための手袋状の入力装置である。また一部のデータグローブでは、コンピュータからの出力により手に振動などのフィードバックを返す出力機能を有したものもあり、より現実に近い感覚を得ることが可能である。データグローブには、指の各関節部分に備えたセンサーから指関節の屈曲度

合いを取得し、手の姿勢を測定するものだけでなく、手や指に各種マーカーをつけたグローブを装着し、カメラで撮影した画像から手指の位置や姿勢を推定するものなども存在する。例えばデータグローブを用いた研究の一例として、視覚障がい者の環境把握や空間認知を支援する研究 [7] がある。このような場合には、手の姿勢を正確に測定し手に振動を返すなどを行う必要があり、センサーを多数備えたデータグローブを用いることになる。しかし、センサーを多く備えたデータグローブは高価であるため、一般の家庭で使用することは困難である。センサーの数を減らした比較的安価なデータグローブも販売されているが、センサー数が少ないため正確な測定ができない。また、比較的安価ではあるが一般家庭で使うにはまだ高価である。様々な VR コンテンツが充実し、合わせて VR システム用の入出力装置の販売数が増えれば価格も下がり、一般家庭でも利用し易くなることが期待できる。

そこで当研究室では、センサーが各指に対して一つしかなく、センサー値が合計で5つしか得られないようなデータグローブを想定し、指の全ての関節角度を推定する試みを行っている。具体的には、5DT Data Glove 5Ultra [8] を用いて実験を行っている。一連の研究では最初に、人間の手の動作を「手を開いた状態から握る」という動作に限定することで、指関節角度の相関関係に基づいて現在のユーザの関節角度を決定する手法 [9] を提案した。しかし、人間の手が行うことができる動作は多様であるため、1つの動作に限定することは望ましくない。そこで、多様な手動作に対応するために、手動作の中でもよく行われる「物体を把持する動作」を想定し、物体の形状情報を利用してユーザの手動作を推定する手法 [10] を提案している。また、人が行う把持動作の中から代表手動作を想定し、代表手動作を行ったときのセンサー出力値を基に、実際にデータグローブから得られたセンサー出力値から関節角度を推定する手法 [11] の提案も行っている。そして、代表手動作をどのような動作にすべきかの検討を行っている [12]。さらに、同一の手動作を行った場合でも、センサー値にばらつきが存在することから、データグローブのセンサー値分布を考慮したデータ補正法 [13] も提案している。

ところで、これまでは主な手動作である把持動作を対象に考えてきたが、日常の動作を再考したところ、ドアノブや容器のふたのように把持した上で「ひねる」動作も比較的多くみられる。ひねり動作は、対象を把持した上で手全体を動かす、すなわち指関節角度が変化しない場合もある。この場合には手首や腕の動きを測定する必要がある。一方、小さな対象をひねる場合には指先のみで行う場合もある。人間の手指には、握ったり開いたりする場合の

指の屈曲・伸展 (図 1.1) 動作以外に、隣り合う指の間の角度が変化する内転・外転 (図 1.2) 動作も存在する。高価なデータグローブにはこの内転外転角度を測定するためのセンサーを備えたものもあるが、安価なセンサー数の少ないグローブでは想定できない。小さな対象のひねり動作では、指の屈曲伸展に加えて内転外転も重要である。そこで本研究では、各指の屈曲伸展の組み合わせに注目し、そこから内転外転を推定することで、センサー数の少ない安価なデータグローブの測定結果からひねり動作を判定し、内転外転を含めた各指の全ての関節角度を推定する手法を提案する。

以下、第2章で従来研究の概要について述べ、第3章で提案手法について説明する。第4章で提案手法の検証を行い、第5章で本論文のまとめを述べる。

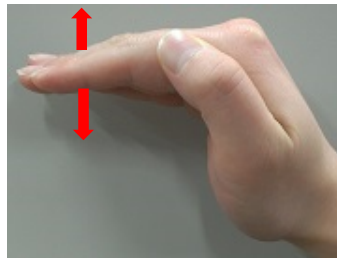


図 1.1: 屈曲・伸展

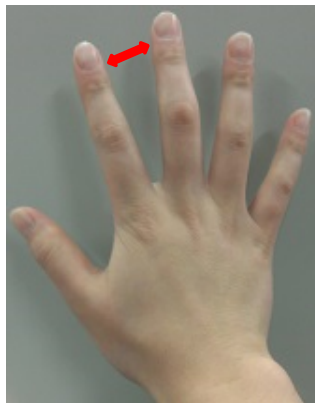


図 1.2: 内転・外転

第2章 従来研究における補正法

本章では，先行研究における指関節角度推定法について述べる．なお，本論文では以後，指の3つの関節を先端から第1関節，第2関節，第3関節と呼ぶものとする．また，本論文で利用する少数センサーデータグローブとは5DT Data Glove 5Ultra [8]である．このグローブは各指につき1本のベンドセンサーを備えている．ベンドセンサーは一本につき値は一つしか得られないが，2つの関節にかかっており，両者の影響を含んだセンサー値を得ることができる．しかしそれぞれの関節角度を求めることはできない．図2.1の円が関節であり，線の示す部分がベンドセンサーの覆っている部分である．



図 2.1: ベンドセンサーの構造

2.1 指関節間の相関関係に基づくデータ補正

本節では，少数センサーデータグローブを用いてユーザーの手動作を取得可能とするためのデータ補正法について述べる．この手法では，「手を開いた状態から握る状態」という特定の手動作に限定し，その動作を行う際の指関節角度間の相関関係を利用して，5つのセンサー値から15個の指関節角度を推定している．まず，実際にデータグローブを装着した状態で動作を行い，センサー値のサンプリングを行うことで，第3関節角度とセンサー値

の相関関係式 (2.1) を導出する. 続いて, 手を握る動作を行ったときの各指関節角度のデータをセンサーを多数備えたグローブにより計測する. この測定データから, 第2関節と第3関節角度の相関関係式 (2.2) を導出する. また, 人間の指の第1関節と第2関節の角度には式 (2.3) のような相関関係があることが知られている [14].

$$\theta_{i3} = A_i S_i + B_i \quad (2.1)$$

$$\theta_{i2} = C_i \theta_{i3} + D_i \quad (2.2)$$

$$\theta_{i1} = \frac{2}{3} \theta_{i2} \quad (2.3)$$

θ_{i1} , θ_{i2} , θ_{i3} は指 i における第1, 第2, 第3関節角度であり, S_i は指 i のセンサー値を表す. また, A_i , B_i , C_i , D_i は指 i における定数である. これらの式により, 5つのセンサー値から15個の指関節角度を決定することができる.

2.2 物体形状情報に基づくデータ補正

前節で述べた手法では, 手の動作を1つに限定しており, 人間の多様な手動作に対応できていない. 本節では, より多様な手動作を取得可能な手法について述べる. 人間が物体を把持する際, 把持に至るまでの手指の動作は把持対象となる物体の形状に大きく依存する. サイコロのような小さな物体であれば, 人差し指と親指の先端で摘むように把持し, 本のような平たく薄い物体を把持する際は, 親指と他の4指で挟みこむように把持を行うと考えられる. そこで, 仮想空間で物体を把持しようとした場合, 計算機が把持対象となる仮想物体の形状情報を保有していることに注目し, その形状情報を利用することでユーザーが物体を把持する際の手動作を推定できる. ここで, 実際には様々な形状の物体や把持の仕方があるが, 物体形状の多くは直方体や円柱などの基本的な形状で表すことが可能である. そこで, いくつかのキーとなる直方体に対する把持動作について考察し, 指関節同士の相関関係式を導出する. 続いて任意の直方体および任意の把持方向からの把持についても考慮することで, 直方体形状の物体に対する把持動作の多様な手動作の取得を可能にする.

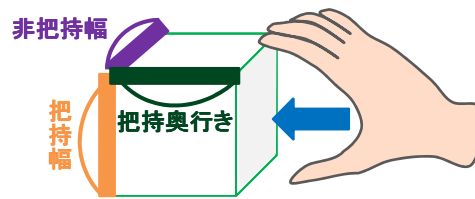


図 2.2: 手の姿勢に対する把持寸法

2.2.1 基本把持動作の定義

まず、キーとなる直方体の寸法について考える。人間が基本的な形状の物体を把持するとき、親指以外の4指の付け根を繋ぐ直線と物体の慣性主軸のいずれかが同じ方向を向くという傾向がある [15]。また、直方体を把持する場合は掌が直方体の6面のうち1面に対してほぼ正対し、掌と正対する面と隣り合う面に親指が接し、親指と接する面の向かいにある面に親指以外の4指が接するように把持すると考えられる (図 2.2)。そこで把持寸法を以下のよう

- 把持幅：親指と他の4指に挟まれる方向の長さ
- 非把持幅：親指以外の4指の第3関節を繋ぐ方向の長さ
- 把持奥行き：掌と向かい合う方向の長さ

この把持寸法に着目し、物体形状に基づく基本的な把持動作を考える。片手で扱う直方体の把持動作には様々な形態が考えられるが、ここでは「握る動作」、「摘む動作」、「挟み込む動作」の3種類の動作を日常的に行う代表的な把持動作として扱う。この3種類の動作は以下のように定義できる。

- 握る動作：親指と他の4指で、第1関節及び第2関節を屈曲させて物体を支える動作
- 摘む動作：親指と人差し指の先端で物体を支える動作
- 挟み込む動作：親指と他の4指で、第1関節及び第2関節を伸展させて物体を支える動作

それぞれの手動作の例を図 2.3 に示す。またこれらの基本的な把持動作を非把持幅と把持奥行きの2つの把持寸法からそれぞれ以下のように分類する。

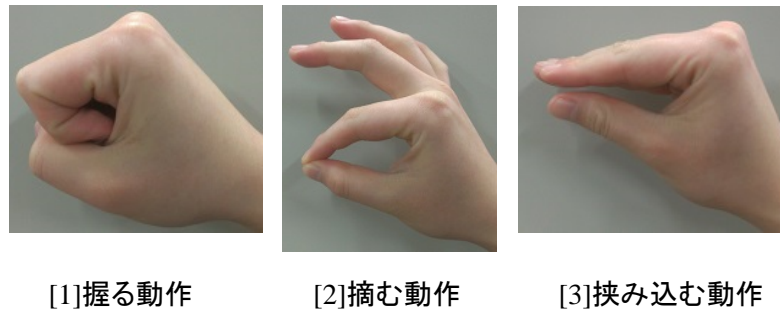


図 2.3: 3 種類の把持動作

- 握る動作：非把持幅が大きく，把持奥行きが小さい物体を対象とした把持動作
- 摘む動作：非把持幅の小さい物体を対象とした把持動作
- 挟み込む動作：非把持幅が大きく，把持奥行きが大きい物体を対象とした把持動作

続いて，把持寸法に基づき把持動作を分類する．さらに，基準となる把持寸法が大きい場合と小さい場合を考慮し，図 2.4 に示す 6 種類を基本把持動作と定義する．これらの基本把持動作について，1 節と同様に第 2 関節と第 3 関節の相関関係式を導出する (式 2.4)．

$$\theta_{i2} = C_{im}\theta_{i3} + D_{im} \quad (2.4)$$

C_{im} , D_{im} は指 i の基本把持動作 m における定数である．なお，第 1 関節角度と第 3 関節角度については 1 節の式 (2.1)，式 (2.3) により求める．これにより，基本把持寸法を持つ物体に対して把持動作を行った際のユーザーの手動作の推定が可能となる．

2.2.2 物体寸法に応じた把持動作の補正

把持対象物体が任意の把持寸法である場合，対象物体の把持寸法と基準寸法との差により基本相関関係式の各パラメータの補正をすることで，把持寸法に応じた相関関係式を導出する．ある基本把持動作について，基準寸法が小さい方の手動作を α 大きい方の手動作を β とし，現在注目している物体の寸法と基準寸法との差を m , n とすると，補正後の指 i におけるパラメータ $C_{i\alpha\beta}$ は式 (2.5) のように導出される．

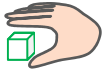


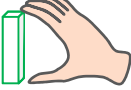

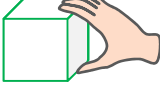
	非把持幅(小)	非把持幅(大)	
		把持奥行き(小)	把持奥行き(大)
把持幅(小)	摘み(小) 	握り(小) 	挟み込み(小) 
把持幅(大)	摘み(大) 	握り(大) 	挟み込み(大) 

図 2.4: 基本把持動作

$$C_{i\alpha\beta} = \frac{nC_{i\alpha} + mC_{i\beta}}{m + n} \quad (2.5)$$

ただし, $C_{i\alpha}$, $C_{i\beta}$ は, 手動作 α , β における指 i の相関関係式のパラメータである. なお, $D_{i\alpha\beta}$ についても同様にして求める. このように, 把持対象物体の寸法と基準寸法との差によって内分・外分を行い, 物体寸法に応じた手動作の相関関係式を導出する.

1. 物体の把持奥行きに注目し, 基本把持動作握り(小)と, 挟み込み(小)の相関関係式の内分・外分と, 基本把持動作握り(大)と, 挟み込み(大)の相関関係式の内分・外分をとる.
2. 非把持幅に注目し, 基本把持動作摘み(小)と1.で求めた(小)の把持動作同士の相関関係式の内分・外分と, 基本把持動作摘み(大)と, 1.で求めた(大)の把持動作同士の相関関係式の内分・外分をとる.
3. 把持幅に注目し, 2.で求めた2つの相関関係式の内分・外分をとる.

以上の手順により, 少数センサーデータグローブのセンサー値からでも, 物体寸法に応じたユーザーの把持動作を取得することが可能となる.

2.3 代表手動作に基づくデータ補正法

前節で述べた手法では、ユーザーの手動作を取得するために、把持対象となる物体の形状情報を利用して、本節では、把持対象となる物体の形状情報を用いず、データグローブのセンサー値のみからユーザーの手動作を推定する手法を述べる。本手法では、まず前節の手法と同様に図 2.3 に示したような「人が行う把持動作」の中からいくつかを考え、それらを代表手動作として想定する。データグローブから実際に得られるセンサー出力値が、代表手動作のいずれかを行ったときのセンサー出力値のパターンと完全に一致するような場合は、ユーザーが行っている手動作がその代表手動作であると判断でき、関節角度を求めることができる。しかし、そのような場合は非常に稀であるため、センサー出力値が一致しない場合には、各代表手動作での関節角度を基に補間を行い、実際のセンサー出力値に対する関節角度を算出する。人がある手動作を行う際の各指関節の屈曲の仕方はある程度決まっていると考えられる。そこで、データグローブから得られる5つのセンサー値を用いて、各代表手動作を行ったときのセンサー出力値のパターンと、現在のセンサー出力値との5次元ユークリッド距離を算出することで、現在のユーザーの手動作が、各代表手動作のどれに近いかを推定する。その推定結果を基に、各代表手動作の比率 $r(0 \leq r \leq 1)$ を算出し、ユーザーの手動作は各代表手動作を比率 r で含む合成動作であるとして現在の関節角度を決定する。

2.3.1 センサー値の正規化

データグローブのセンサー出力値には個体差があり、あるデータグローブで有効な手法が別のデータグローブでも有効であるとは限らない。そこで、本手法を様々なデータグローブに適用可能とするために、センサー値を正規化することを考える。正規化を行うために実際にデータグローブを装着してデータ収集を行う。まず、指の関節角度が0度のときを初期状態として、ここから第3関節角度が30度、60度、90度のときのセンサー値を収集する。このとき、第1関節・第2関節は動かさないものとする。収集した関節角度とそれぞれに対応するセンサー値に対応するデータから、各データ点の近傍を通る直線を求める。すると、ある指 i のセンサー値 s_i に対して、正規化後の値 S_i を導出する式は以下のように表せる。

$$S_i = N_{1i}s_i + N_{2i} \quad (2.6)$$

ここで、 N_{1i} , N_{2i} は定数である．この正規化後のセンサー値 S_i を用いることで、様々なデータグループで本手法を利用することが可能となる．以後、本論文で述べるセンサー値とはセンサー出力値を正規化した値を示すものとする．

2.3.2 ユークリッド距離に基づく手動作推定

事前の準備として、少数センサーデータグループを装着して各代表手動作を行い、そのときのセンサー出力値をサンプリングする．ユーザーの手動作が代表手動作のどれかである場合、センサー出力値はその代表手動作のサンプルデータに近い値になると予想されるが、実際にはサンプルデータに近い値になることは少ない．そのため、事前に得られた各代表手動作のセンサー値のサンプルデータと、現在のセンサー値 S との最短5次元ユークリッド距離 l_p を求めることで、現在のユーザーの手動作に近い動作を推定する．サンプルデータと実際のセンサー値との例を図 2.5 に示す．これは人差し指と中指の例であり、人差し指のセンサー値を横軸、中指のセンサー値を縦軸としたグラフである．なお、本来は指5本のセンサー値が存在し、5次元空間で表されるが、ここでは2次元の例を示す．現在のセンサー値 S が得られたとき、ある代表手動作 p のサンプルデータの各サンプリング点に対して、次式 (2.7) から導出されるユークリッド距離を求め、 l_{pn} の中で最も小さくなったものをその代表手動作における最短5次元ユークリッド距離 l_p とする．

$$l_{pn} = \sqrt{\sum_{i=1}^5 (S_i - s_{pmi})^2} \quad (2.7)$$

ただし、 S_i , s_{pmi} はそれぞれ現在の指 i のセンサー値、代表手動作 p における n 点目の指 i のセンサー値を表す．

各代表手動作について l_p を算出した後、 l_p が小さい動作2種類を選択する．選択された2種類の手動作 α , β の l_p をそれぞれ l_α , l_β とすると、手動作の比率 r_α は式 (2.8), r_β は式 (2.9) で表すことができる．

$$r_\alpha = \frac{l_\beta}{l_\alpha + l_\beta} \quad (2.8)$$

$$r_\beta = \frac{l_\alpha}{l_\alpha + l_\beta} \quad (2.9)$$

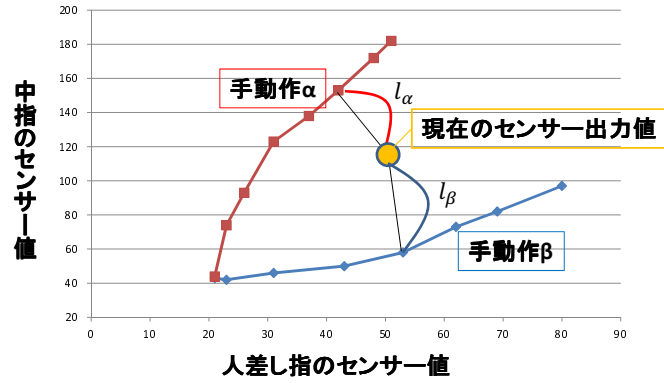


図 2.5: 代表手動作のサンプルデータと実際のセンサー値とのユークリッド距離 l_p

以上の手順により，代表手動作の中から現在のユーザーの動作に近い手動作を推定可能となる。

2.3.3 指関節角度の導出

本節では，前節で求めた比率 r_p から実際の関節角度を求める手順について述べる．まず事前に，センサーを多数備えたデータグローブにより各代表手動作を行い，そのときの各指の第2，第3関節角度を測定した上で，これを真値とする．そしてそのときの角度と少数センサーデータグローブで各代表手動作を行ったときのセンサー値とを動作の開始時刻，終了時刻が一致するように対応付けることで，最小二乗法により各指の第2，第3関節角度について以下のような関係式が導出される．ここで第1関節角度 θ_{pi1} については第2関節角度との比例関係を利用して求める [14]．

$$\theta_{pi1} = \frac{2}{3}\theta_{pi2} \quad (2.10)$$

$$\theta_{pi2} = E_{pi2}S_i^3 + F_{pi2}S_i^2 + G_{pi2}S_i + H_{pi2} \quad (2.11)$$

$$\theta_{pi3} = E_{pi3}S_i^3 + F_{pi3}S_i^2 + G_{pi3}S_i + H_{pi3} \quad (2.12)$$

ここで， θ_{pi1} ， θ_{pi2} ， θ_{pi3} ， S_i はそれぞれある手動作 p におけるある指 i の第1，第2，第3関節角度，センサー出力値を表し， E_{pij} ， F_{pij} ， G_{pij} ， H_{pij} は定数である．実際にデータグローブからセンサー値 S が得られたとき，まず前節の手順で2種類の手動作 α ， β とその比率 r_α ， r_β を導出する．次に，式(2.10)-(2.12)を使用して， α ， β におけるセンサー値 S_i の

ときの各指の第1から第3関節角度を求める．最後にそれぞれの関節角度を比率 r_α , r_β に基づいて内分することで，現在のユーザーの指関節角度を決定する(式 2.13).

$$\theta_{ij} = r_\alpha \theta_{\alpha ij} + r_\beta \theta_{\beta ij} \quad (2.13)$$

θ_{ij} は指 i における第 j 関節角度であり， $\theta_{\alpha ij}$, $\theta_{\beta ij}$ はそれぞれ手動作 α , β における指 i の第 j 関節角度である．以上の手順により，少数センサーデータグローブでも，センサー出力値から各指の関節角度を決定可能である．

2.4 代表手動作の決定

前節で述べた手法を用いるにあたって，どのような把持動作を代表手動作とするかは非常に重要である．例えば，似たような手動作ばかりを代表手動作として用いた場合，合成動作として表すことができる手動作も似たような動作ばかりになり，人間の多様な手動作を表すことができない．また，ある程度異なる手動作を代表手動作とした場合でも，人が日常行いうる動作からかけ離れた動作であると，ユーザーの手動作を表現するのに適当ではない．そこで，どのような手動作を代表手動作にするのが適しているかを文献 [12] では検討している．当該文献中では，多数ある把持動作分類法の中から，鎌倉らによる分類法 [16] [17] を基に，図 2.6 に示すような7種類の動作を選出し，それらについて以下の3つのような代表手動作の候補セットを決定している．

まず，候補1として，7種類の動作全てを代表手動作として用いる．次に，冗長な計算を削減するために，これら7種類の手動作についてウォード法による階層型クラスター分析を行い，各手動作を C_1 : Standard, C_2 : Hook-like・Lateral Contact・Index Ext, C_3 : Tripod・Tip Contact, C_4 : Parallel Ext の4つのクラスに分類する．この中で単独でクラスを形成する Standard, Parallel Ext を代表手動作の候補2として選定する．さらに，複数の動作が存在するクラスからは，各クラスの平均得点との差が最も小さい，Lateral Contact, Tripod を選出し，これらを加えた4種類の手動作を最終的な代表手動作の候補2とする．また， C_2 と C_3 に属する各手動作に対して，その平均手動作 MC_2 , MC_3 (図 2.7) を構築し，これに Standard と Parallel Ext を加えた4種類の手動作を代表手動作の候補3として選出する．な

お, 文献 [12] 中における様々な検証実験の結果により, 候補 2 が代表手動作に適しているという結論が得られている.

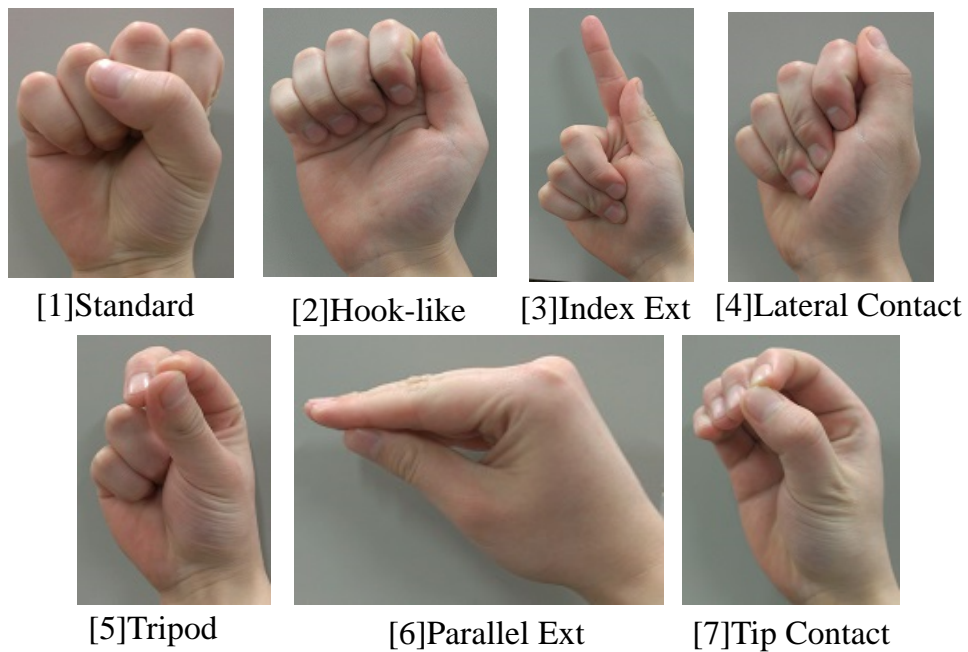


図 2.6: 代表手動作の候補

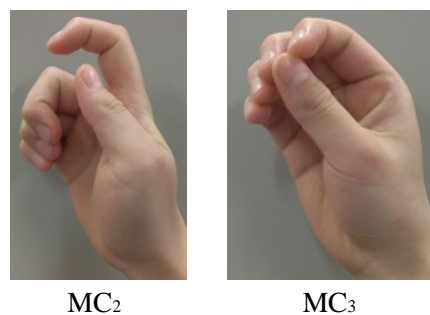


図 2.7: 平均手動作

2.5 センサー値分布を考慮した補正法

ある手動作を行った際に, データグローブの出力するセンサー値がばらつくことを考慮する必要がある [13]. そこで本節ではセンサー値分布を考慮した手動作推定の手法を述べる.

2.5.1 センサー値のばらつきを考慮した手動作推定

2.3節において述べた手法では、各代表手動作のサンプルデータが理想的なものであるとして推定を行っていた。しかし、各手動作のセンサー出力値をサンプリングし、センサー出力値に対してD'Agostino-Pearson検定(有意水準5%)を行うと有意差が認められなかった。そのため、この手法ではセンサー値が多変量正規分布に従い、ある程度のばらつきが存在すると仮定している。そして、各指のセンサー値を特徴ベクトルとした5次元特徴空間における n 点に対して、多変量正規分布の確率密度関数に基づく式(2.14)を設定する。

$$L_{pn}(\mathbf{S} : \boldsymbol{\mu}_{pn}, \boldsymbol{\Sigma}_{pn}) = \exp\left\{-\frac{1}{2}(\mathbf{S} - \boldsymbol{\mu}_{pn})^T \boldsymbol{\Sigma}_{pn}^{-1}(\mathbf{S} - \boldsymbol{\mu}_{pn})\right\} \quad (2.14)$$

ただし、 \mathbf{S} はセンサー値ベクトル、 $\boldsymbol{\mu}_{pn}$ および $\boldsymbol{\Sigma}_{pn}$ は手動作 p における n ($1 \leq n \leq 25$ を満たす整数)点目のセンサー値平均ベクトル、分散共分散行列を表す。グローブから実際にセンサー出力値が得られた場合、式(2.14)より出力される値の中で最大となるものを式(2.15)に従い選択することで、現在のセンサー出力値に対する手動作 p における尤度を得る。

$$L_p = \max_n \{L_{pn}(\mathbf{S} : \boldsymbol{\mu}_{pn}, \boldsymbol{\Sigma}_{pn})\} \quad (2.15)$$

各代表手動作に対して尤度 L_p を求め、その尤度から代表手動作の比率を導出する。最終的な各代表手動作 p の比率 r_p は式(2.16)に従い決定される。

$$r_p = \frac{L_p}{\sum_{p=1}^P L_p} \quad (2.16)$$

ここで、 P は代表手動作として設定した動作の合計数を表す。これによりセンサー値分布を考慮した手動作推定を行うことができる。

第3章 ひねり動作による内転外転を含めたデータ補正法

前章では主な手動作である把持動作に着目したデータ補正法について述べた。把持動作は人間が日常でよく用いる手動作である。また日常でよく用いる手動作の中には、ドアノブや容器のふたのように把持した上で「ひねる」動作もみられる。ひねり動作は、対象を把持した後に腕や手首を回すことでひねるために関節角度が変化しない場合もある。この場合には手首や腕の動きを測定する必要がある。一方指先のみでひねり動作を行う場合もある。高価なデータグローブでは内転・外転角度を測定するためのセンサーがついているものがあり、ひねり動作も再現することが可能である。しかし、安価なセンサー数の少ないグローブでは直接的に測定できない。一般家庭への導入という観点から考えると安価なデータグローブであっても、内転・外転を考慮した手の姿勢測定、推定ができることが必要である。そこで本章では、前章の手法に加えて、ひねり動作における内転外転角度を含めた関節角度の推定手法を述べる。

3.1 ひねり動作時のセンサー出力値の計測

まず、グローブのセンサー値から内転外転を含む各指関節角度を推定するための式を導出するために、実際に安価なデータグローブで指関節角度のセンサー出力値のサンプリングを行う。把持する物体は直径が9cmと7cmの円柱の物体であり、把持動作は5本の指全部で行う(図3.1)。指関節角度の測定では、実際に少数センサーデータグローブを装着して物体を把持し、手首や腕を動かさないように固定してひねり動作を行う。手首や腕を固定する理由は、ひねり動作を行った際に、手全体の位置や姿勢を測定することなく、指関節の変化のみに注目するためである。センサー出力値のサンプリングは、物体を把持した状態を開始状態とし、手首を動かさずにひねることができるところまで動作を行ったら終了状態とする。またひねる動作は物体を把持した後、左右に回すことができる。そのため物体を把持した状

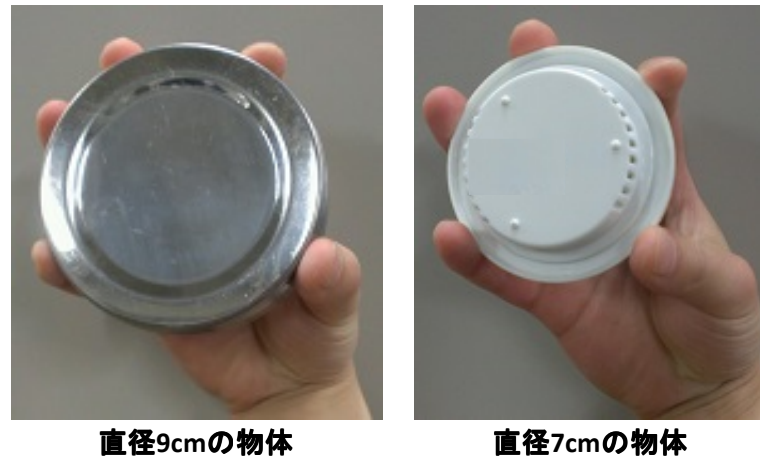


図 3.1: 把持する物体と把持の仕方

態から左に回す場合と、右に回す場合の2パターンの計測を行う。ただし、実際のひねり動作は左右どちらかの端からもう一方の端まで回すことができるので、この2パターンは一連の動きと考えられる。実際の計測結果を表 3.1、表 3.2 に示す。なお、それぞれ複数回行った試行の平均値である。またサンプル点は全部で 52 点あり、便宜的に一番左に回した状態のサンプル点を -26、一番右に回した状態のサンプル点を 25 とする。この測定結果をもとに内転外転を含む各関節角度推定のための式を導出するわけであるが、実際には想定しているグローブには内転外転を計測するためのセンサーは装着されていない。すなわち、5つのセンサーの値は、通常の把持動作を行ったときと、ひねり動作を行った時で重複することが予測される。そこで次節では、ひねり動作のための推定式を適用するための領域について検討する。その後、推定式について述べる。

ところで、大きさの近い二つの物体でひねり動作を行う際も、把持の仕方や動きなどは近い動作であることが多い。そのため、前述の直径 9cm と 7cm の対象物の操作に対して、角度推定のための式を共通化できることを期待していた。そこで後述する方法で、それぞれの対象物のための推定式を求めるために主成分分析を行った。結果の一部を表 3.3-3.6 に示す。しかし、それぞれの第一主成分は明らかに平行とはいえず、共通の推定式を見いだすことは困難と判断した。そこで、以降では直径 9cm の物体のみを対象として議論を進めた上で、大きさの異なる物体に対するひねり動作は今後の課題としたい。

表 3.1: 測定結果 (直径 9cm の物体)

サンプル点	親指のセンサー値	人差し指のセンサー値	中指のセンサー値	薬指のセンサー値	小指のセンサー値
-26	246.933	54.767	98.080	203.157	-33.460
-25	247.020	53.903	97.660	202.040	-33.573
-24	246.380	53.040	96.720	202.040	-33.573
-23	245.523	52.033	94.733	202.267	-33.687
-22	244.930	52.250	92.433	201.860	-34.140
-21	244.587	51.893	91.180	201.683	-34.310
-20	244.330	52.037	90.867	201.060	-34.367
-19	243.650	52.037	89.827	200.300	-34.310
-18	242.840	51.963	88.463	199.583	-34.367
-17	242.077	50.383	88.043	196.453	-34.707
-16	241.437	49.803	88.147	194.667	-34.480
-15	240.540	47.790	87.523	192.433	-34.820
-14	239.600	47.143	87.417	191.270	-34.820
-13	238.537	45.850	87.107	191.000	-34.933
-12	237.980	45.350	86.893	190.197	-35.160
-11	237.170	44.270	87.207	189.393	-35.330
-10	236.403	42.610	87.417	188.587	-35.387
-9	235.977	41.463	87.000	186.843	-35.783
-8	235.593	40.527	86.057	185.323	-35.897
-7	235.293	39.590	85.850	184.743	-36.010
-6	234.913	39.017	85.537	184.383	-36.123
-5	234.570	37.937	85.013	183.267	-36.180
-4	234.227	36.427	84.387	181.523	-35.953
-3	234.057	34.987	84.490	180.673	-35.783
-2	233.973	33.117	84.177	180.587	-35.613
-1	233.547	31.103	83.553	180.763	-35.387
0	233.333	18.947	50.830	168.070	-40.760
1	232.777	19.737	52.400	169.143	-40.023
2	232.097	21.030	54.177	170.393	-39.060
3	231.283	21.967	56.373	171.603	-38.440
4	230.947	21.463	58.983	172.090	-38.100
5	230.563	21.683	60.970	172.940	-37.647
6	230.010	21.823	62.330	173.433	-37.310
7	230.093	20.890	64.213	174.103	-36.463
8	230.303	19.377	65.047	173.970	-35.840
9	230.433	18.083	65.777	174.687	-35.160
10	230.223	16.210	66.617	175.357	-34.537
11	230.050	15.063	68.290	176.250	-33.857
12	229.453	14.200	69.333	176.923	-33.460
13	229.030	13.550	71.213	178.843	-32.667
14	228.473	12.830	72.153	180.097	-31.873
15	228.090	12.687	72.577	181.793	-31.703
16	227.917	11.970	73.620	182.823	-31.193
17	227.790	11.540	74.980	183.757	-30.797
18	227.450	10.747	76.547	185.590	-30.403
19	227.150	10.677	78.427	187.247	-29.610
20	226.810	11.467	79.473	188.050	-29.213
21	226.427	12.903	80.937	188.900	-28.367
22	226.003	13.480	82.507	190.687	-27.860
23	225.703	15.063	84.280	192.970	-27.010
24	225.703	17.940	84.803	193.460	-26.727
25	225.403	17.940	84.700	193.727	-27.407

表 3.2: 測定結果 (直径 7cm の物体)

サンプル点	親指のセンサー値	人差し指のセンサー値	中指のセンサー値	薬指のセンサー値	小指のセンサー値
-26	247.700	36.860	83.340	171.960	-42.290
-25	247.487	36.643	83.340	171.467	-42.233
-24	247.363	36.067	82.820	170.440	-42.290
-23	247.230	35.923	83.027	170.170	-42.460
-22	246.937	35.420	80.937	169.143	-42.630
-21	246.210	34.913	77.903	168.427	-42.687
-20	245.570	34.553	75.710	167.083	-42.800
-19	244.843	34.123	74.247	166.237	-42.913
-18	244.290	34.773	73.097	164.000	-42.800
-17	243.863	34.700	73.097	162.930	-42.630
-16	243.563	34.627	72.053	162.260	-42.347
-15	243.223	34.630	70.693	161.100	-41.950
-14	243.013	34.773	69.960	159.800	-41.610
-13	242.587	35.203	69.753	159.357	-41.270
-12	242.113	36.067	69.127	158.997	-40.873
-11	241.560	36.210	68.707	158.323	-40.590
-10	241.263	36.497	68.497	158.327	-40.533
-9	240.837	35.927	66.510	158.640	-40.477
-8	240.453	35.280	65.153	158.593	-40.363
-7	239.900	34.917	64.737	158.413	-40.250
-6	238.707	34.483	64.737	158.727	-39.797
-5	237.810	34.120	65.153	158.593	-39.173
-4	237.297	33.477	66.513	159.043	-38.437
-3	236.960	33.117	69.127	159.220	-37.760
-2	236.617	32.617	71.320	159.307	-37.313
-1	236.233	32.037	73.200	160.560	-36.860
0	238.027	40.743	76.233	165.880	-32.610
1	237.853	40.743	78.220	166.683	-31.763
2	237.637	39.380	80.103	166.593	-31.027
3	237.470	38.800	81.247	167.043	-30.177
4	237.170	38.370	82.400	168.163	-29.327
5	237.040	37.863	83.550	169.903	-27.970
6	236.787	37.650	83.653	171.420	-27.233
7	236.107	36.713	84.283	172.183	-26.047
8	235.337	34.987	84.073	173.347	-25.253
9	234.400	35.133	85.120	175.040	-23.497
10	233.587	33.833	85.953	176.430	-22.023
11	232.950	32.977	87.000	177.187	-21.233
12	232.607	32.040	88.153	178.127	-20.103
13	232.310	30.887	90.343	179.337	-18.857
14	231.500	28.873	92.227	180.677	-18.123
15	230.687	26.787	93.063	181.973	-16.990
16	230.520	26.143	94.107	184.117	-16.197
17	230.137	26.647	95.260	185.370	-15.347
18	229.627	28.730	96.510	186.980	-14.213
19	228.943	29.520	99.127	187.560	-11.327
20	228.900	29.667	100.483	188.410	-11.043
21	229.497	28.660	100.903	189.393	-11.553
22	230.093	28.583	100.693	190.330	-12.233
23	230.607	30.530	100.380	189.523	-12.177
24	230.607	29.590	99.963	189.170	-12.743
25	230.563	28.800	99.543	189.393	-13.083

表 3.3: 主成分分析 (直径 9cm, 人差し指第 3 関節内転外転)

	第 1 主成分	第 2 主成分
親指センサー値	-0.940309	0.268906
人差し指センサー値	-0.971288	0.207931
中指センサー値	-0.832950	-0.491456
薬指センサー値	-0.824731	-0.538102
小指センサー値	0.100497	-0.991351
人差し指第 3 関節内転外転	0.939402	-0.317973
寄与率	68.235750	28.841875
累積寄与率	68.235750	97.077626

表 3.4: 主成分分析 (直径 7cm, 人差し指第 3 関節内転外転)

	第 1 主成分	第 2 主成分
親指センサー値	-0.930154	-0.329519
人差し指センサー値	-0.744879	0.334521
中指センサー値	0.8947745	-0.329965
薬指センサー値	0.9145504	-0.342913
小指センサー値	0.9920840	0.068465
人差し指第 3 関節内転外転	0.7813183	0.618963
寄与率	77.529075	13.912609
累積寄与率	77.529075	91.441683

表 3.5: 主成分分析 (直径 9cm, 人差し指第 3 関節屈曲伸展)

	第 1 主成分	第 2 主成分
親指センサー値	-0.945525	0.278386
人差し指センサー値	-0.966346	0.208880
中指センサー値	-0.827241	-0.494378
薬指センサー値	-0.830196	-0.531805
小指センサー値	0.092237	-0.988572
人差し指第 3 関節屈曲伸展	-0.932281	0.315588
寄与率	67.984176	28.753782
累積寄与率	67.984176	96.737958

表 3.6: 主成分分析 (直径 7cm, 人差し指第 3 関節屈曲伸展)

	第 1 主成分	第 2 主成分
親指センサー値	-0.931625	-0.327056
人差し指センサー値	-0.744549	0.333550
中指センサー値	0.893340	-0.333855
薬指センサー値	0.913444	-0.345519
小指センサー値	0.991574	0.061485
人差し指第 3 関節屈曲伸展	-0.709854	-0.699499
寄与率	75.697131	15.702392
累積寄与率	75.697131	91.399523

3.2 把持動作とひねり動作の境界

ひねり動作は把持動作を行った後に行う動作であり、把持動作とは別の動作と考える。そのため、従来の手法で行っていた基本把持動作の複合動作として考えることができない。そこでひねり動作を、実際に得られたセンサー出力値のサンプル点により把持動作と区別することを考える。センサー出力値は各指に1つの合計5つ得られるため、サンプル点は5つの値の組で表される。ひねり動作を行った際のサンプル点を全て包含するような5次元の超球を考えることにより、その内側と外側のどちらに現在のセンサー値が位置するかでひねり動作と把持動作を分けることを試みる。ここで実際のセンサー出力値にはばらつきが存在するため、境界となる超球は最小のものではなく、全てのサンプル点を含むある程度小さな超球とする。サンプル点は、ひねり動作という一連の動作を行ったときの点であるため、指関節の変化が大きい指と小さい指が存在する。図3.2は、実際にひねり動作を行った際のセンサー出力値の各指の変化の仕方である。これより親指や小指のセンサー出力値に対して他の3本の指のセンサー出力値の変化が大きいことがわかる。1つの超球でサンプル点を包含することを考えた場合、変化の少ない軸も同じ半径で包含するため、サンプル点からの距離が遠い空間も含んでしまう。そのため超楕円球や、いくつかの超球に分けて複数の超球で全ての点を包含することを考える。だが、境界の式を多く用意することはプログラムの処理速度が低下する原因となるため、多くの超球を考えることは望ましくない。

3.2.1 複数の超球による全てのサンプル点の包含

まず、全てのサンプル点の重心を考える。そして重心から一番遠いサンプル点を求め、重心からの距離を求める。重心を中心とし、最も遠い点までの距離を半径とすることで全てのサンプル点を包含する超球を考えることができる。しかし前節で述べたとおり、この超球はサンプル点が近くにない空間を多く包含していることが考えられる。そのため以下の処理を行うことにより球を複数にわけ、小さい範囲にすることを考える。

1. 最初に求めた超球の中心から遠い点を順番に除外するように、中心付近のサンプル点のみを含む超球を求める。
2. 除外されたサンプル点を重心から最も遠いサンプル点の側にある点と、正反対の側にある点で分類する。

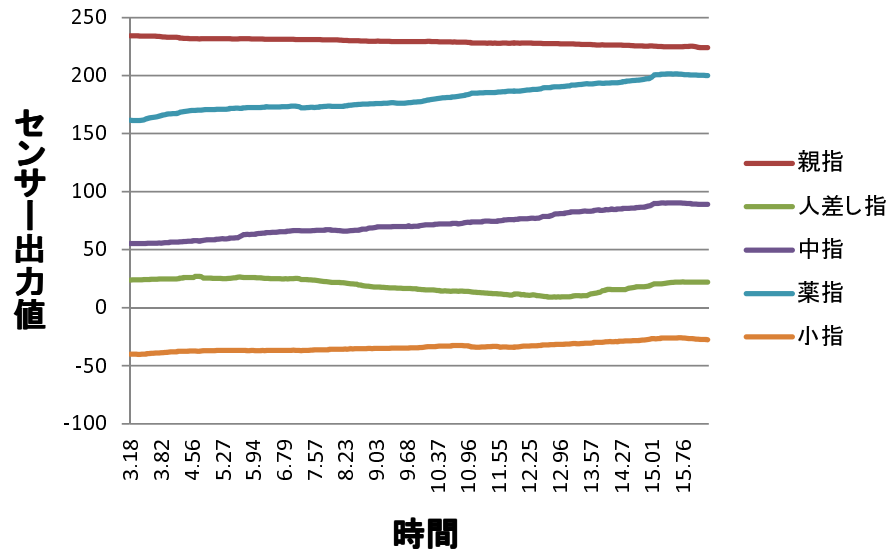


図 3.2: ひねり動作時の各指のセンサー出力値の変化

3. 分類した2つのグループのサンプル点をそれぞれ全て包含する超球を考える。

ここで、1.で作成した超球を「超球C」(中央), 3.で作成した超球を重心から最も遠い点を含むものを「超球F」(遠方), 反対側を「超球N」(近辺)とする。これにより、1つの超球で包含していた全てのサンプル点を3つの超球で包含した状態にし、サンプル点のない空間を削減できる。

3.2.2 超球の決定

ここで、1.の操作で最初に考えた超球を小さくする際に、どれくらいの点を含むかを定める必要がある。そこで、1つ目の超球がサンプル点を約55%, 60%, 65%含む3パターンを考え、それぞれパターン1, 2, 3とする。それぞれの超球は表3.7の通りである。各指のセンサー値は S_1, S_2, S_3, S_4, S_5 で表す。いずれのパターンが適しているか判断するために、各サンプル点がどのようにそれぞれの超球に含まれているか確認した。また、各パターンの超球の組で表される領域の体積の比較を行った。

確認の結果、どのパターンも超球Cと超球Nはある程度重なっていた。しかし、超球Fは超球Cと重ならなかった。ここで、ひねり動作という一連の流れを考えているため、3つの超球はどこかで他の超球と合わさっていることが望ましい。ところで分析の結果、パター

表 3.7: 超球の中心と半径

	中心					半径
	S_1	S_2	S_3	S_4	S_5	
パターン1 超球 C	236.73	33.73	81.31	179.92	-32.48	24.60
超球 F	244.83	52.43	92.80	201.04	-34.05	7.48
超球 N	229.08	16.30	68.46	179.85	-33.72	22.89
パターン2 超球 C	236.73	33.73	81.31	179.92	-32.48	25.09
超球 F	245.13	52.66	93.33	201.55	-33.98	5.78
超球 N	229.10	16.21	68.53	180.11	-33.64	23.12
パターン3 超球 C	236.73	33.73	81.31	179.92	-32.48	25.76
超球 F	245.13	52.66	93.33	201.55	-33.98	5.78
超球 N	229.81	17.28	65.25	177.81	-35.03	26.04

ン1の超球Fとパターン2とパターン3の超球Cが複数のサンプル点を重複して含んでいることが分かった。そこで領域を決定するための中央の超球はパターン2とパターン3の超球Cを候補とし、「遠方」側の超球はパターン1の超球Fを唯一の候補とする。「近辺」側の超球は、3つのパターンの中から単純に体積の一番小さい超球Nを選ぶことにする。続いて、前述の候補のそれぞれの体積の比較を行った。最初に中央の超球についての結果を述べる。パターン2とパターン3のそれぞれの超球Cの体積は以下の通りである。

$$V_{2C} = 5.23 \times 10^7 \quad (3.1)$$

$$V_{3C} = 5.97 \times 10^7 \quad (3.2)$$

V_{ij} はパターン*i*の超球*j*の体積を表す。パターン2の超球Cの方が体積が小さいので中央の超球はパターン2の超球Cに決定する。次に超球Nについて考える。超球Nの体積は以下の通りである。

$$V_{1N} = 3.31 \times 10^7 \quad (3.3)$$

$$V_{2N} = 3.47 \times 10^7 \quad (3.4)$$

$$V_{3N} = 6.30 \times 10^7 \quad (3.5)$$

以上より、一番体積が小さいパターン1の超球Nに決定する。

最後に、全てのサンプル点を含む1つの超球の体積 V_1 と、決定した3つの超球の組で表される領域の体積 V_3 を比較する。以下の通り、1つの超球で領域を決定するよりも小さな体積で効率的にひねり動作の領域を分類できる。

$$V = 36.5 \times 10^7 \quad (3.6)$$

$$V_3 = 7.56 \times 10^7 \quad (3.7)$$

求めた超球がどのようにサンプル点を含んでいるかを平面で確認するために、ある2軸を選びその2軸が与える平面からサンプル点と超球がどのように表されるかを散布図として示す(図3.3-図3.12)。

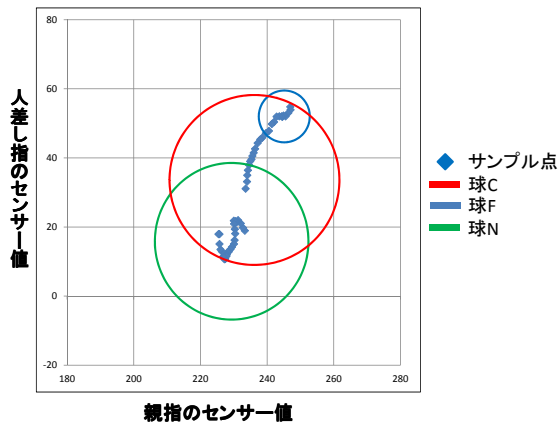


図 3.3: 親指-人差し指の軸で表される平面

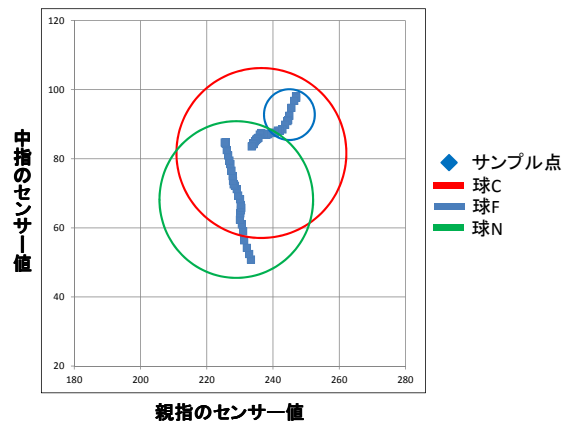


図 3.4: 親指-中指の軸で表される平面

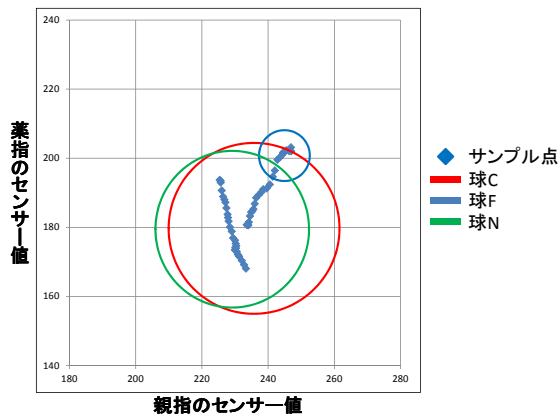


図 3.5: 親指-薬指の軸で表される平面

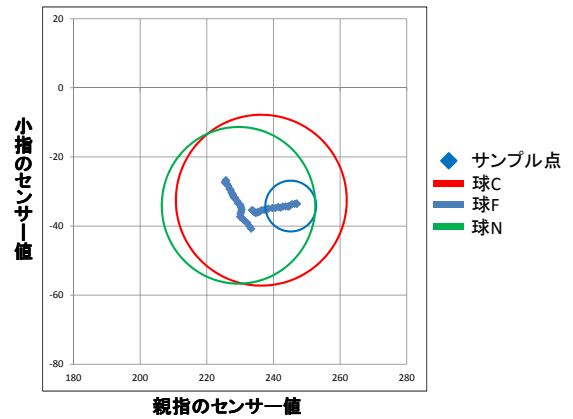


図 3.6: 親指-小指の軸で表される平面

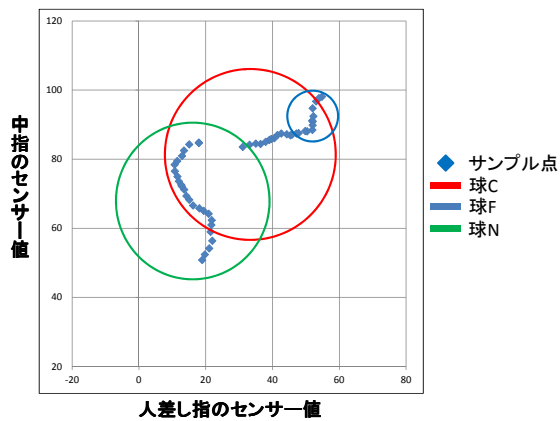


図 3.7: 人差し指-中指の軸で表される平面

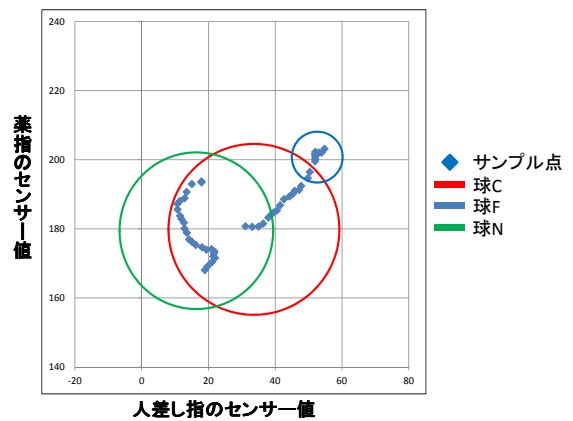


図 3.8: 人差し指-薬指の軸で表される平面

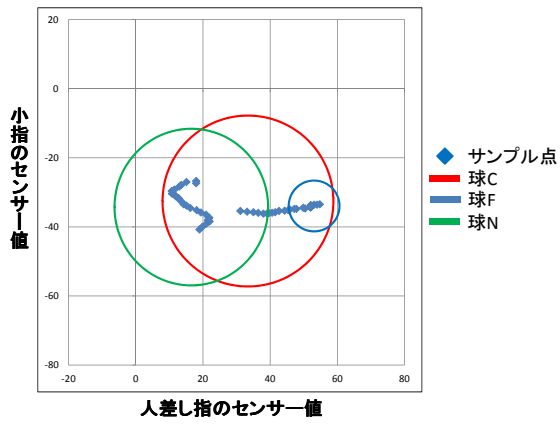


図 3.9: 人差し指-小指の軸で表される平面

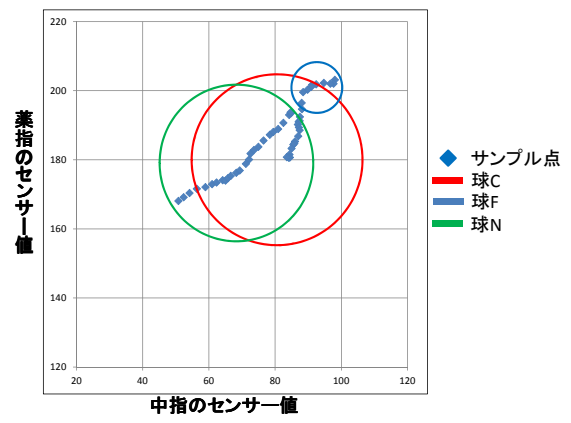


図 3.10: 中指-薬指の軸で表される平面

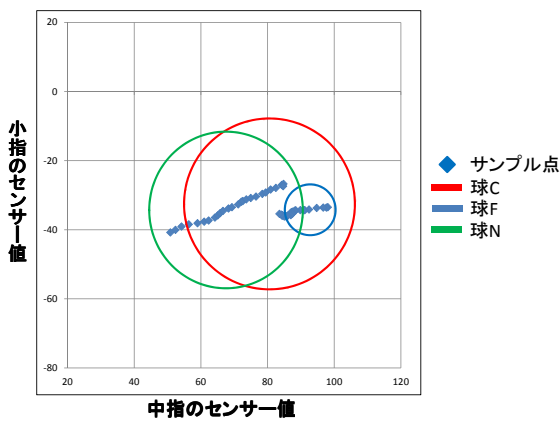


図 3.11: 中指-小指の軸で表される平面

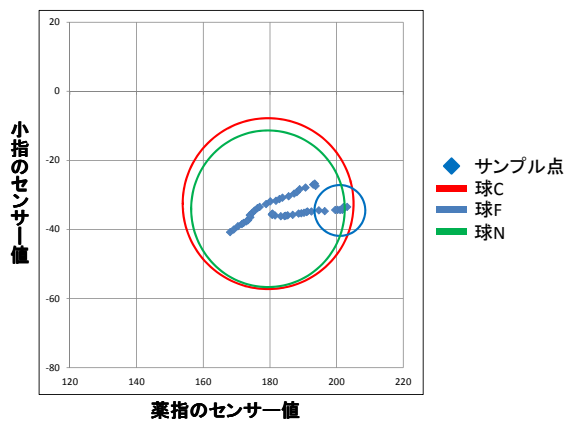


図 3.12: 薬指-小指の軸で表される平面

3.3 ひねり動作における関節角度推定

前節でひねり動作と判定するセンサー値の組み合わせの領域を導出した。本節では、実際にデータグローブを装着して操作を行った時のセンサー値がひねり動作と判定された場合の、各関節角度の推定について述べる。

3.3.1 内転外転角度と角指関節角度の推定式

少数センサーデータグローブを装着したときのひねり動作のセンサー値サンプルを3.1節で測定した。このときの各関節角度を測定したい。そこで、同様に全く同じ動作を同じサンプル間隔で、第2関節と第3関節の屈曲伸展角度に加え内転外転角度も測定できるセンサー数の多いデータグローブにより測定し、真値とする。結果を表3.8に示す(一部抜粋)。なお各サンプル点番号に対するセンサー値は表3.1を参照されたい。

表 3.8: 人差し指と中指の関節角度の真値

サンプル点	人差し指		
	第3関節 内転外転	第3関節 屈曲伸展	第2関節 屈曲伸展
-26	19.50	20.79	48.51
-25	19.50	20.79	48.51
-24	19.50	20.79	48.51
-23	19.50	20.79	47.74
-22	19.50	20.79	46.97
-21	19.50	20.02	46.97
-20	19.35	19.25	46.20
-19	19.35	19.25	44.66
-18	19.35	18.48	44.66
-17	19.35	18.48	42.35
-16	19.35	17.71	40.81
-15	19.65	17.71	40.04
-14	19.50	17.71	39.27
-13	19.80	16.94	38.50
-12	19.95	16.17	36.19
-11	20.10	15.40	36.19
-10	20.25	15.40	36.19
-9	20.40	13.86	36.19
-8	20.40	11.55	35.42
-7	20.55	10.78	35.42
-6	20.40	7.70	35.42
-5	20.40	6.93	33.88
-4	20.40	6.16	33.11
-3	20.40	6.16	32.34
-2	20.40	6.16	30.03
-1	20.40	6.16	30.03
0	21.15	4.62	40.04
1	21.15	4.62	40.04
2	21.15	4.62	40.04
3	21.15	4.62	40.81
4	21.15	4.62	40.81
5	21.30	4.62	40.04
6	21.30	4.62	40.04
7	21.60	4.62	40.04
8	21.60	4.62	40.04
9	21.75	4.62	40.81
10	21.75	4.62	40.81
11	21.90	3.85	40.81
12	21.90	3.08	41.58
13	21.90	3.08	41.58
14	22.05	2.31	42.35
15	22.05	1.54	43.12
16	21.90	1.54	44.66
17	21.90	0.77	47.74
18	21.75	0.77	51.59
19	21.90	-0.77	52.36
20	21.90	-2.31	53.13
21	22.05	-3.08	53.13
22	22.05	-3.08	53.13
23	22.20	-3.08	53.13
24	22.20	-3.85	53.13
25	22.20	-3.85	53.13

各関節角度は、5つのセンサー値から6次元空間における直線近似で求められると仮定する。各6次元空間のデータにおいて主成分分析を行う。表3.3、表3.5に、人差し指の内転外転角度と5つのセンサー値の分析結果と、人差し指の第3関節屈曲伸展角度とセンサー値の分析結果を示す。第1主成分の寄与率は比較的高いので、直線での近似は大きく間違っていないと考える。

このとき、センサー値とある関節角度の真値からなる6次元でのサンプル点の集合の重心を A 、その関節角度に対する第1主成分を V とし、近似直線 P を次式で表す。

$$P = A + \alpha V \quad (3.8)$$

$$P = \begin{pmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ X \end{pmatrix}, A = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_X \end{pmatrix}, V = \begin{pmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_X \end{pmatrix} \quad (3.9)$$

ここで α は媒介変数であり、 P は任意のセンサー値とそのときの当該関節角度である。このような推定式を全ての手指関節角度に対して求める。推定式の利用時には、いずれかのセンサー値から媒介変数 α を求めることで当該関節角度を求めることができることが期待される。しかし実際には正しく近似直線状で表現できるとは限らない。そこで、5つのセンサーから式3.10により α_1 から α_5 を求めた上で、それらの平均を α_X とする。その上で式3.11により当該関節角度を求める。

$$S_i = a_i + \alpha_i V_i \quad (3.10)$$

$$X = a_X + \alpha_X V_X \quad (3.11)$$

3.3.2 境界付近での推定方法

前述の通り、把持動作とひねり動作を領域分割した。しかし、把持動作とひねり動作を境界の内外判定のみで分けた場合、どちらかの領域からもう一方の領域へ移動した場合に指の

関節角度が大きく変化してしまう可能性がある。そこで、境界の付近に緩衝領域を設ける。具体的には、測定されたセンサー値が各超球から一定の距離に存在する場合、すなわち各超球には含まれないが、それらよりも一定の値だけ半径の大きな超球に含まれる場合には、緩衝領域にあたると判断する。この値が R であり、センサー値がひねり動作から $r(r \leq R)$ の距離にあるとする。また、把持動作と仮定した場合の推定式による推定角度を θ_h 、ひねり動作と仮定した場合の推定式による推定角度を θ_t と表した場合、求めたい関節角度の角度 θ を次式により求める。

$$\theta = \frac{r\theta_h + (1-r)\theta_t}{R} \quad (3.12)$$

このようにすることで、把持動作からひねり動作まで滑らかに推定することができる。

第4章 実験

本節では，3章で述べた提案手法の検証結果について述べる．

4.1 実験システムの概要

提案手法の検証を行うために，従来のシステムに3章で述べた境界判定と推定式を加えたシステムを構築した．実験システムは，CPU: Dual-Core CPU E5200 2.50GHz の Linux マシン上に C 言語によって構築し，推定された関節角度を基に画面上に CG 仮想手を描画する．少数センサーデータグローブは 5DT を用いる．

4.2 実験結果と考察

実際に少数センサーデータグローブを装着して腕と手首を固定した状態で，ひねり動作を行ったときの各関節角度を推定した．図 4.1 は，ひねり動作を一番左から右に回している様子である．上の画像が一番左にひねっている状態，中央の画像がひねり動作の中間で指をひねっていない状態，下の画像が一番右にひねっている状態である．把持対象の物体につけた印と，画像につけた赤色の直線から，ひねり動作を確認できる．同図に写っている CG の様子から，適切な関節角度推定が行われていることを確認できる．また，ひねり動作ではなく，通常の把持動作などを行った際は，従来のシステム通り動作していることも確認できた．実際のセンサーの出力値と推定した人差し指の指関節角度およびそのときの真値を表 4.1 に示す(一部抜粋)．各関節角度の推定結果は真値に近い数値であることが確認でき，ある程度正しい推定ができているといえる．

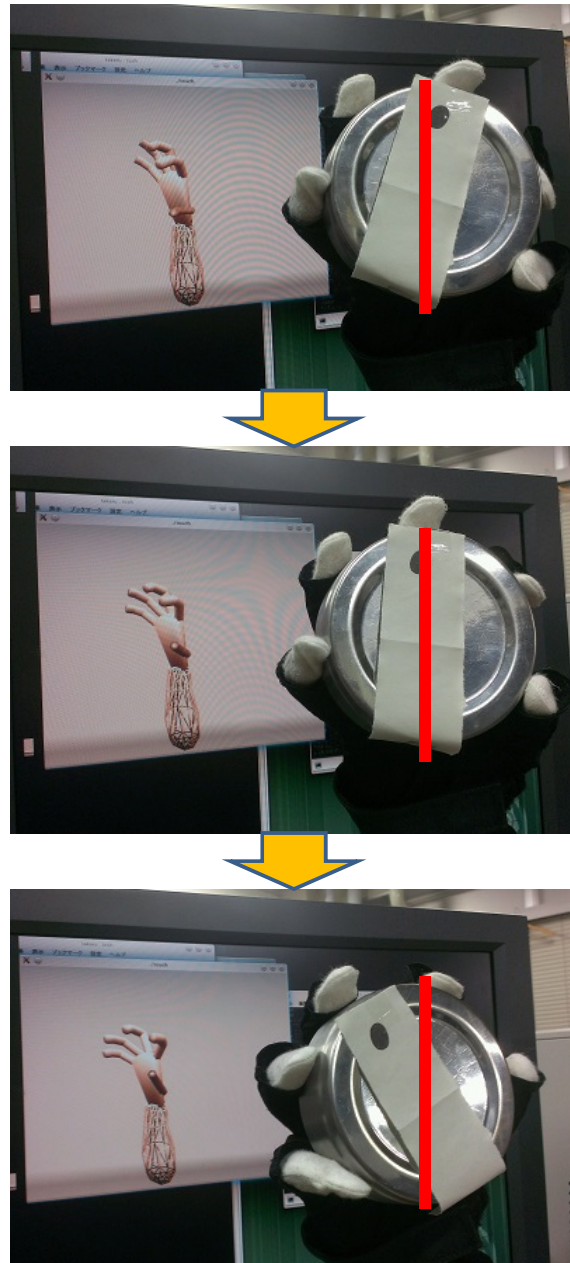


図 4.1: 実験の様子

表 4.1: センサー値と人差し指の推定関節角度と真値

親指	センサー値				人差し指の真値				人差し指の推定値				
	人差し指	中指	薬指	小指	第3関節 内転外転	第3関節 屈曲伸展	第2関節 屈曲伸展	第3関節 内転外転	第3関節 屈曲伸展	第2関節 屈曲伸展	第3関節 内転外転	第3関節 屈曲伸展	第2関節 屈曲伸展
233.973	33.117	84.177	180.587	-35.613	20.40	6.16	30.03	17.734	11.768	42.093	17.734	11.768	42.093
233.547	31.103	83.553	180.763	-35.387	20.40	6.16	30.03	18.733	10.736	42.052	18.733	10.736	42.052
233.333	18.947	50.830	168.070	-40.760	21.15	4.62	40.04	21.354	8.984	39.263	21.354	8.984	39.263
232.777	19.737	52.400	169.143	-40.023	21.15	4.62	40.04	22.091	8.132	39.495	22.091	8.132	39.495
232.097	21.030	54.177	170.393	-39.060	21.15	4.62	40.04	23.092	6.982	39.795	23.092	6.982	39.795
231.283	21.967	56.373	171.603	-38.440	21.15	4.62	40.81	23.461	6.515	40.023	23.461	6.515	40.023
230.947	21.463	58.983	172.090	-38.100	21.15	4.62	40.81	23.562	6.362	40.157	23.562	6.362	40.157
230.563	21.683	60.970	172.940	-37.647	21.30	4.62	40.04	23.802	6.051	40.326	23.802	6.051	40.326
230.010	21.823	62.330	173.433	-37.310	21.30	4.62	40.04	24.096	5.706	40.435	24.096	5.706	40.435
230.093	20.890	64.213	174.103	-36.463	21.60	4.62	40.04	25.265	4.406	40.655	25.265	4.406	40.655

第5章 むすび

本研究では，従来の把持動作に着目したデータ補正法に加え，ひねり動作における内転外転角度を含めた関節角度の推定手法を提案した．提案手法では，ひねり動作の領域を設定し，その領域内でひねり動作用の推定式を用いることで指の内転外転角度を含む関節角度を求める．またこの領域の境界を，5つのセンサー出力値に基づいて5次元超球を考えることで求めた．各手指関節角度を求めるための推定式は，主成分分析を行うことで決定した．提案の方法により，センサー数が少なく内転外転部分にセンサーのないデータグローブでも，ひねり動作における内転外転を含む各関節角度の推定が可能である．

現段階では，大きさが違う対象のひねり動作が考慮されていない．また，5本の指全てではなく，3本の指のみによるひねり動作も考慮されていない．今後は統一的な手法により様々な大きさの対象物のひねり操作を，任意の指（ただし，通常は親指を必ず含むであろうし，人差し指か中指のいずれかが含まれることがほとんどであろう）により行えるように推定手法を発展させなければならない．将来的には，人の行いうる多様な手動作を少数センサーデータグローブから推定できるようなデータ補正法を実現したい．

謝辞

本研究を進めるにあたって、日頃から多大な御尽力を頂、ご指導を賜りました名古屋工業大学、舟橋健司 准教授、伊藤宏隆 助教に心から感謝致します。また、本研究に多大な御協力頂きました舟橋研究室諸氏に心から感謝致します。

参考文献

- [1] 舘暲, 佐藤誠, 廣瀬通孝, 日本バーチャルリアリティ学会 (編), “バーチャルリアリティ学”, 工業調査会, 2010.
- [2] 近藤大祐, 飯塚博幸, 安藤英由樹, 小濱和貴, 坂井義治, 前田太郎, “腹腔鏡下手術トレーニングにおける視野共有手法による学習効果とその実証”, TVRSJ, Vol. 18, No. 4, pp. 487-496, 2013.
- [3] 坪田慎介, 大野隆造, “ポータブルVRシステムを用いた防災教育の実施”, 日本建築学会大会学術講演梗概集, 2009.
- [4] 石黒静児, 大谷寛明, “核融合プラズマを体感する ―百見は一体感に如かず―”, J. Plasma Fusion Res, Vol. 90, No. 6, pp. 325-331, 2014.
- [5] 任天堂株式会社: Wii コントローラ,

<http://www.nintendo.co.jp/wii/controllers/index.html>
- [6] 米国 OculusVR 社: Oculus Rift,

<https://www.oculus.com/rift/>
- [7] 巽久行, 村井保之, 関田巖, 宮川正弘, “視覚障がい支援のための疑似触力覚の適用”, 筑波技術大学テクノレポート, Vol. 22(1), Dec, 2014.
- [8] Fifth Dimention Technologies: 5DT Data Glove Ultra,

<http://www.5dt.com/>

- [9] 濱口真一, 舟橋健司, “VR ネットショッピングのためのセンサー数の少ないデータグローブによるデータ補正法”, 日本バーチャルリアリティ学会第 15 回大会講演論文集, 2A2-4 (DVD-ROM), 2010.
- [10] 山本三四郎, 金沢秀典, 舟橋健司, “物体形状情報を利用したセンサー数の少ないデータグローブによる補正法”, 日本バーチャルリアリティ学会第 16 回大会講演論文集, 33D-6 (DVD-ROM), 2011.
- [11] 濱口真一, 舟橋健司, “センサー数の少ないデータグローブによる手動作推定に基づくデータ補正法”, 日本バーチャルリアリティ学会第 16 回大会講演論文集, pp. 690-693, 2011.
- [12] 高橋寛昌, 舟橋健司, “医学的知見による代表手動作に基づく少数センサーデータグローブのデータ補正法”, 名古屋工業大学大学院 修士学位論文, 2014.
- [13] 高橋寛昌, 舟橋健司, “センサ値分布を考慮したセンサ数の少ないデータグローブのデータ補正法”, 2013 年電子情報通信学会総合大会 (境界・基礎) 講演論文集, pp. 216 (A-16-10), 2013.
- [14] ELKOURA G, “Handrix: Animating the Human hand”, Proceedings of the 2003 ACM SIGGRAPH /Eurographics Symposium on Computer Animation, pp. 110-119, 2003.
- [15] 中村宜孝, 宮里勉, 岸野文郎, “把持動作における preshaping の解析とターゲット予測”, 電子情報通信学会論文誌, Vol. J80-A, No. 6, pp. 1003-1013, 1997.
- [16] 鎌倉矩子, 大村道子, 石井晴美, 三星文子, 三浦頼子, “常手の把握の様式-分類の試み”, リハビリテーション医学 14, pp. 57-72, 1978.
- [17] Kamakura N, Matsuo, Ishii H, Mitsuboshi F, Miura Y, “Patterns of static prehension in normal hands”, Am J Occup Ther 34, pp. 437-445, 1980.