

ロボット動作の自動生成のための観察による手作業の抽象化

高松 淳[†] 池内 克史^{††}

[†] 東京大学生産技術研究所

^{††} 東京大学情報学環

〒 153-8505 東京都目黒区駒場 4-6-1 駒場 II キャンパス E 棟
東京大学生産技術研究所 第 3 部 池内研究室
03-5452-6242
j-taka@cvl.iis.u-tokyo.ac.jp

あらまし 様々な作業を行うロボット動作を自動生成する方法として、「観察による行動獲得」手法が近年注目されている。この手法では、まず教示者の作業を観察、認識することにより、作業を抽象化して表現する。それを用いて、動作プリミティブの繰り返しとして、目的の作業を再現することができる。動作プリミティブとは抽象化して表現された作業中の状況を再現するために必要不可欠となる動作群である。つまり、この手法の適用の際には、作業表現の方法と作業を再現するために十分な動作プリミティブの定義が非常に重要となる。

本論文では日常の手作業を対象とし、組み立て作業、回転ジョイントにより接続されたリンク物体の操作を扱う。

組み立て作業では 2 物体間の接触状態により作業を抽象化して表現することができる。しかし、無数の接触状態を扱う困難さを解消するため、各接触状態における物体の可能な微小変位を指標化した運動自由度を定義し、作業の再現に十分な動作プリミティブを定義する。

次に、回転ジョイントにより接続されたリンク物体の操作作業を扱う方法を提案する。ドアノブを回す作業を考えたとき、その回転軸の向きや中心の位置 (ジョイントパラメータ) を知っていれば、ノブを回すことができる。そこで、実際に作業の様子を観察することにより、視覚誤差に対してロバストに、ジョイントパラメータを推定する方法を提案する。

キーワード 観察による行動獲得, 組み立て作業, リンク物体の操作

Abstraction of Manipulation Tasks to Automatically Generate Robot Motion from Observation

Jun TAKAMATSU[†] and Katsushi IKEUCHI^{††}

[†] Institute of Industrial Science, The University of Tokyo

^{††} Graduate School of Interdisciplinary Information Studies, The University of Tokyo

Institute of Industrial Science, The University of Tokyo, 3rd Dept. Ikeuchi Laboratory
4-6-1 Komaba, Meguro-ku, Tokyo, 153-8505, JAPAN
+81-3-5452-6242
j-taka@cvl.iis.u-tokyo.ac.jp

Abstract For a framework to automatically generate robot motion to execute various tasks, the *Learning from Observation* (LFO) paradigm has recently been proposed. In this paradigm, a robot system observes and recognizes a demonstration of a task, builds an abstract representation of the task, and finally generates robot motion which consists of repetitions of movement primitives. Movement primitives are essential operations to realize the situation corresponding to the abstract representation. Here, the key to the successful implementation depends on the representation of the target task and the definition of sufficient movement primitives.

The target tasks of this paper are everyday manipulation tasks including an assembly task and manipulation of linkages which are connected by a joint.

In assembly tasks, a contact relation between the two objects is employed as the abstract representation. Because of overcoming the difficulty to deal with numerous kinds of contact relations, we define a *Motion Degree of Freedom* (DOF), which is an index that represents the quality of possible local displacement of an object in each contact relation and define sufficient movement primitives in advance using Motion DOFs.

Next, we propose a method to deal with manipulation of linkages which are connected by a revolute joint. Considering the task of rotating a doorknob, which is connected to a door by a revolute joint, it is possible to rotate the knob, if one knows the axis direction and the center of rotation (parameters of the joint). Therefore, we propose a method to estimate the parameters from noise-contaminated observation.

Keywords Learning From Observation, Assembly Tasks, Manipulation of Linkages

1 はじめに

1.1 背景

人型ロボットなど、多目的ロボットの社会進出が目まぐるしい今日において、一般家電製品のような感覚で、簡単にロボットが使える枠組みを構築することが強く求められてきている。そのため、作業目的の情報からロボット動作を自動的に生成する研究が、現在まで数多く行われてきた [1]。しかし動作生成のためには、NP 完全な問題を数多く解く必要があり、結果として動作生成は非常に困難であることが認知されている。

その困難さを解消する方法として、「観察による行動獲得」手法が注目されている [2, 3, 4, 5]。この手法では、動作生成の際、作業目的の情報のみでなく、作業の観察により得られる情報を用いることで困難さを解消している。さらにこの手法の特徴として、教示者が無意識のうちに行っている作業のコツを獲得できることが挙げられており、それゆえ人工知能や神経科学の分野でも注目されている [6]。

具体的には、次のような段階を経て、ロボット動作が生成される [4]: まず教示者の行った作業を観察し、認識する。次に作業の抽象表現を得る。最後に作業に対応する動作プリミティブの列に変換する。動作プリミティブとは、抽象表現に合致する作業中の状況を再現するために必要不可欠な動作のことを指す。「観察による行動獲得」手法を実装する際には、1. どのように作業を抽象的に表現するか、2. 任意の作業を再現するのに十分な動作プリミティブをどのように定義するか、を熟考する必要がある。

1.2 対象とする作業領域

本論文では、ステレオビジョンシステムによる観察を通じて、剛体を操る作業を理解する方法を提案する。ここでいう理解とは、作業の抽象化および動作プリミティブ列への変換を指す。

まず剛体を操る作業について概観する。単一物体の状態は一定であるため、複数物体間の位置関係が作業の抽象化において非常に重要であると考えられる。また剛体を操る作業では、たいていの場合、接触による動作の拘束がおこる。この拘束に注目し、拘束された動作の情報を用いて作業を認識する方法を提案する。提案手法は、基本的にはほとんど全ての剛体を操る作業を扱うことができると考えている。

拘束の情報を抽出する方法に関して、次に示す2つの方法があると考えられる: 1つは接触状態から直接抽出する方法である。この場合、接触状態が作業の抽象化に対応する。この方法では、物体の正確な形状はあらかじめわかっているものとする。この方法の具体的な応用として、我々は組み立て作業を扱う。それにより、机の上にものを置くといった簡単な動作から、コンセント、ケーブル、鍵などの挿入動作が再現可能となる。

もう1つは拘束された動作の種類を表すプリミティブ

を定義し、そのプリミティブに関する情報を抽出することである。もちろんプリミティブの種類が作業の抽象表現となる。本論文では、いわゆるジョイントと呼ばれるものをプリミティブとして用い、ジョイントによって接続されたリンクを操る作業をその応用として取り上げる。それにより、扉を開ける、蛇口をひねる、コーヒー豆をひくなどの作業が再現可能となる。本論文では回転ジョイントのみを扱うが、他の種類のジョイントへの応用を考えた際、提案手法は非常に重要な役割を果たす。

拘束されてない動作の情報も作業認識において重要な役割を担うが、本論文では明示的にそれを扱わない。結果として、例えばクランク回しの作業を考えたとき、我々の提案手法はクランクを回すことはできるが、効率よく回すことはできない。なぜならば効率よく回すコツは、拘束されてない動作の情報に含まれているからである。しかし効率よく回せるためには、まず回せることが必要不可欠であり、決して提案手法が役に立たないと言うわけではない。

1.3 関連研究

この節では、まず「観察による行動獲得」手法に関する関連研究を次の2つの観点から概観する:

- どのように対象作業を観察するか?
- どのように動作プリミティブを設計、実装するか?

その手法の実装の仕方は、これら2つの観点と密接な関係がある。

まず「どのように対象作業を観察するか?」に関して、「観察による行動獲得」手法を概観する。いくつかの研究において、haptic インターフェース [7] やロボットアーム [8] によって動作を直接指定する方法や、VR 空間内で教示者が作業を教示する方法 [9, 10] などが提案されている。これらの方法では、特殊なデバイスを必要としたり、教示者が特殊な教示方法に熟れなければならないという問題点がある。

それに対して、ビジョンシステムを用いて作業を観察する方法 [2, 3, 11] では、以下に示す優位性がある:

- 通常多目的ロボットは、ビジュアルフィードバックのためにビジョンシステムを標準装備しているため、手法実装のために特別なデバイスを必要としない。なおビジョンシステムの中には、本論文で用いているステレオビジョンシステムも含まれる。
- 教示の際、教示者はいつでもどおり作業を行うことができる。

次に「どのように動作プリミティブを設計、実装するか?」に関して「観察による行動獲得」手法を概観する。この問題の解決策として、次に示す2つの解決法が提案されている:

- あらかじめ、目標作業に対して十分な数の動作プリミティブを予測し実装する

- 観察を通じて動作プリミティブを自動的に抽出し、システム内に取り込む

前者の長所は、あらかじめ必要な動作プリミティブがわかっているため、実装が簡単であることである。短所は、未実装な動作プリミティブを含む作業に対応できないことである。これは予測された動作プリミティブの数が不十分であったことに起因する。つまり動作プリミティブの十分性には深く注意を払うべきである。

それに対して、後者の長所は、未実装な動作プリミティブに対し、それを観察により抽出し動的に獲得することにより、任意の作業を再現できることである。もちろん短所は、そのような仕組みを構築することが非常に困難であることである。

我々が前者を選択した理由として、後者の方法では本論文で対象とする作業は扱えないと考えるからである。我々の対象とする作業では、位置制御に加え力制御も必要となる。これは位置、力に関する多くのパラメータを調整しなくてはならないことを意味しており、このパラメータの調整は、専門的な知識を持ち、かつ作業全体の構造を十分知っていたとしても非常に困難である。

実際に動作プリミティブの実装例を示す。ほとんどの研究では、動作そのものを動作プリミティブとして定義している。例えば、教示者の手足や関節の動き [12, 13, 14, 15] や、対象物体の動き [16] などが動作プリミティブとして用いられている。いくつかの研究においては、ユーザーによって与えられた評価関数を最適にするように、動作を改編するものもある。これらの方法(本論文では Motion-based 法と呼ぶ)では、観察により得られた動作をそのままざることが動作プリミティブの実現につながり、その実装は比較的容易である。これらの方法では、外界との作用を伴わない動作(例えば、手の到達運動やジェスチャーなど)は扱えるが、そうでない動作は、それが机の上にもものを置くような簡単な動作であっても、その扱いは非常に困難である。なぜならば動作プリミティブが位置制御に基づいている実装されているからである。

いくつかの研究では、抽象化された作業表現に対応する状態を実現する動作を動作プリミティブとして定義している [2, 3, 17, 18, 10]。これらの方法(本論文では State-based 法と呼ぶ)では、動作プリミティブは対象とする作業の際に現れる任意の状態を実現できなくてはならない。もちろんその条件を満たす動作プリミティブを実装することは非常に困難であるが、本論文の対象とする作業を含む、外界との相互作用を伴う作業を実現するロボット動作を自動的に生成するためには、State-based 法しかないといわれは確信している。

2 観察による 2 多面体物体を用いた組み立て作業の認識

本節では、2 多面体物体を用いた組み立て作業を、観察により認識する方法について述べる。ここでは、観察により以下のようなデータが得られているものとする：

- 接触状態の遷移
- 各接触状態における物体の位置姿勢

接触状態は接触要素の集合からなっており、接触要素は接触する 2 つの物体要素(頂点、辺、面)からなっている。実際にステレオビジョンシステムを用いてそれらの情報を得る方法については [19] に示している。

前述の通り、組み立て作業は 2 物体間の接触状態により作業を抽象化して表現することができる。しかし、無数の接触状態を扱う困難さにより、この表現のみからロボット動作を生成することは非常に困難である。そこで、各接触状態における、対象物体の可能な微小変位に関する指標群を用いて認識する方法を提案する。また、実際に物体の 3 次元 CAD モデルと接触状態から指標群を計算する方法を述べる。

組み立て作業中に現れるあらゆる接触状態遷移において、可能な指標群の変化を調べることにより、組み立て作業の実現に十分な動作プリミティブを定義し、実際に指標群の変化から適切な動作プリミティブを選択する方法を述べる。具体的な動作プリミティブの実装に関しては [20] に示す。

本論文では、すべての多面体物体の形状は既知であるとし、変形しないものとする。また、把持された物体(以後、把持物体と記述する)のみ動くものとし、それ以外の物体(以後、環境物体と記述する)は固定されているものとする。

我々の提案手法は、Ikeuchi と Suehiro が提案した手法 [2] の改良である。本手法の新規性は、1. 面接触を含む任意の接触状態と 2. 回転運動を扱えることにある。ちなみに、彼らの手法は並進運動と面接触しか扱うことができなかった。その改良の困難さは、次に示す 2 つであり、その解決法が本手法の重要な話題となる：

- 回転変位に対する指標群の定義
- 扱う接触状態およびその遷移の急激な増加

2.1 可能な微小変位の定式化

比留川らは、多面体物体間の接触における物体の可能な微小変位を、幾何モデルから導出する方法を提案している [21]。それによると、可能な微小変位は、式 (1) の形で表現することができる。ただし、 \mathbf{F}_{ij} は分離面の法線(例えば、把持物体の点と環境物体の面が接触している場合、 \mathbf{F}_{ij} は面の外向き法線に等しい)、 $\Delta \mathbf{X}, \Omega$ はそれぞれ位置および姿勢の微小変位、 N は接点の数、 $M(i)$ は i 番目の接点における分離面の数を表す。また J_i は、 i 番目の接点の微小変位と物体の位置姿勢の微小変位との関係を表すヤコビ行列である。

$$\bigcap_i \bigcup_j^{M(i)} \mathbf{F}_{ij}^T J_i \begin{pmatrix} \Delta \mathbf{X} \\ \Omega \end{pmatrix} \geq 0 \quad (1)$$

本論文では、微小変位を screw 表現 [22] を用いて表す。screw 表現では、微小変位をある軸に沿った並進運

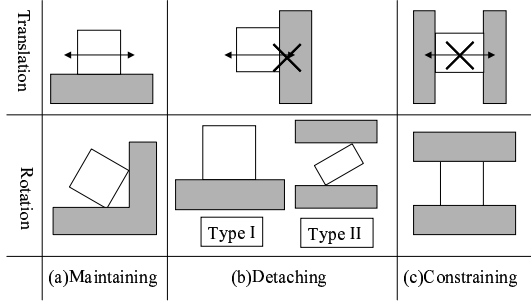


図 1: 維持, 離脱, 拘束変位

動と, その軸を中心とする回転運動の組み合わせとして表現する. 具体的には, 軸の向きを \mathbf{S}_0 , 軸の位置を \mathbf{P} , 回転運動に対する並進運動の割合を p とすると, その微小変位は, 6次元ベクトル $[\mathbf{S}_0, \mathbf{S}_1]$ として表現される. ただし, $\mathbf{S}_1 = \mathbf{P} \times \mathbf{S}_0 + p\mathbf{S}_0$ である. $p = 0$ のときは純粋な回転運動を表し, $p = \infty$ のとき, つまり $[\mathbf{0}, \mathbf{S}_1]$ のときは, 純粋な並進運動を表す.

実際に screw 表現を用いると, 式 (1) は式 (2) のように書き換えられる [22]. ただし, \mathbf{P}_i は i 番目の接触点の位置を表す.

$$\bigcap_i \bigcup_j \mathbf{F}_{ij} \cdot \mathbf{S}_1 + (\mathbf{P}_i \times \mathbf{F}_{ij}) \cdot \mathbf{S}_0 \geq 0 \quad (2)$$

2.2 微小変位の指標群

まず $\forall i, M(i) = 1$, つまり式 (2) が連立線形不等式である場合 (以後, 非特異接触状態と記述する) と, それ以外の場合 (以後, 特異接触状態と記述する) に分けて物体の微小変位に関する指標群について考察する.

2.2.1 非特異接触状態における指標群

非特異接触状態の場合, 把持物体の微小変位は, その変位にともなって接触状態がどのように遷移するかに基づき, 以下に示す 3 種類に分類できる (図 1 参照)[2]:

維持変位 接触状態を維持する変位

離脱変位 接触要素が減少する方向に接触状態を変化させる変位

拘束変位 接触により消失した変位

並進変位における維持, 離脱, 拘束変位の自由度 m_t, d_t, c_t は, 式 (3) のように定式化される. ただし V_t は式 (2) に $\mathbf{S}_0 = \mathbf{0}$ を代入したときの \mathbf{S}_1 の解領域, $d(V_t)$ は解領域 V_t の face の最大次数 [23], 行列 $R_t = (\mathbf{F}_{11} \cdots \mathbf{F}_{N1})$, $\text{Rank}(R_t)$ は行列 R_t のランクであるとする.

$$\begin{aligned} m_t &= 3 - \text{Rank}(R_t) \\ d_t &= 3 - (m_t + c_t) \\ c_t &= 3 - d(V_t) \end{aligned} \quad (3)$$

実際に $d(V_t)$ を計算する方法は, Kuhn と Tucker[23] や, Hirukawa ら [24] により提案されている. それぞれ並進維持, 並進離脱, 並進拘束自由度と呼ぶ. 次に回転変位に関する指標つまり自由度の定義を行う.

まず任意運動における維持, 離脱, 拘束変位の自由度 m_a, d_a, c_a を式 (4) のように定義する. ただし V_a は式 (2) の $[\mathbf{S}_0, \mathbf{S}_1]$ の解領域, 行列

$$R_a = \begin{pmatrix} \mathbf{F}_{11} & \cdots & \mathbf{F}_{N1} \\ \mathbf{P}_1 \times \mathbf{F}_{11} & \cdots & \mathbf{P}_N \times \mathbf{F}_{N1} \end{pmatrix}$$

であるとする.

$$\begin{aligned} m_a &= 6 - \text{Rank}(R_a) \\ d_a &= 6 - (m_a + c_a) \\ c_a &= 6 - d(V_a) \end{aligned} \quad (4)$$

これらを全維持, 全離脱, 全拘束自由度と呼ぶ.

さらに作業理解の質を改善するために, 我々は回転軸の向きに関して, 次に示す 3 種類の指標群を定義する (図 1 参照):

回転維持自由度 維持変位における回転軸の向きの自由度

回転離脱自由度 離脱変位における回転軸の向きの自由度

回転拘束自由度 拘束変位における回転軸の向きの自由度

実際にこれらの自由度を, 式 (2) から計算する方法について述べる.

命題 1 部分空間 $V_r = \{\mathbf{S}_0 | [\mathbf{S}_0, \mathbf{S}_1] \in V_a\}$ のとき,

$$c_r = 3 - d(V_r),$$

ただし c_r は回転拘束自由度である.

証明 定義より自明. □

命題 2

$$d_r = 3 - (m_r + c_r),$$

ただし m_r, d_r はそれぞれ回転維持, 回転離脱自由度である.

証明 回転軸の自由度は 3 であり, すべての自由度は維持, 離脱, 拘束のいずれかであるため. □

命題 3

$$m_r = m_a - m_t,$$

入力: 基底 $B = \{[s_{10}, s_{11}], \dots, [s_{m_a 0}, s_{m_a 1}]\}$

1. 集合 $\{s_{10}, \dots, s_{r_0}\}$ の中から, 線形独立な最小の組み合わせ C を探す. ただし, すべての $s_{i0} \in C$ に対して $s_{i0} \neq \mathbf{0}$ である. もしこのような組み合わせが見つからなければ, 集合 B を出力して終了する.
2. その組み合わせを用いて, 次の式を得る:

$$\sum_{s_{i0} \in C} a_i s_{i0} = \mathbf{0} \quad (a_i \neq 0)$$

3. 組み合わせ C に含まれる要素のうちの一つを集合 B から取り除く.
4. 要素 $[s'_0, s'_1]$ を集合 B に加える. ただし

$$[s'_0, s'_1] = \sum_{s_{i0} \in C} a_i [s_{i0}, s_{i1}].$$

5. 1. に戻る.

図 2: 回転維持自由度を計算するためのアルゴリズム

証明 式 (5) のある適当な解の基底を $\{[s_{10}, s_{11}], \dots, [s_{m_a 0}, s_{m_a 1}]\}$ とすると, 任意運動における維持変位はその線形和で表すことができる.

$$\bigcap_i^N \mathbf{F}_i \cdot \mathbf{S}_1 + (\mathbf{P}_i \times \mathbf{F}_i) \cdot \mathbf{S}_0 = 0 \quad (5)$$

次に, その解の基底に図 2 に示すアルゴリズムを適用することにより, $\{[\mathbf{0}, s_{11}], \dots, [\mathbf{0}, s_{t1}], [s_{t+1,0}, s_{t+1,1}], \dots, [s_{m_a 0}, s_{m_a 1}]\}$ で表される解の基底が得られる. ただし, $\{s_{t+1,0}, \dots, s_{m_a 0}\}$ は線形独立である. このとき $\{[\mathbf{0}, s_{11}], \dots, [\mathbf{0}, s_{t1}]\}$ の線形和のみで表される並進変位は維持変位を表すことから, t は m_t に等しい. また $\{s_{t+1,0}, \dots, s_{r_0}\}$ は線形独立であることから, m_r は $m_a - t$ に等しい, つまり $m_r = m_a - m_t$.

□

これに対し, $d_r = d_a - d_t$ や $c_r = c_a - c_t$ は常に成り立つとは限らない. これより次の事実が成り立つ.

命題 4 回転離脱自由度にあたる回転軸の向き s_0 について考えたとき, 次に示す 2 つの場合が存在する (図 1 参照):

- $s_0, -s_0$ の両方が部分空間 V_r に含まれる場合
- s_0 のみが含まれる場合

前者の場合, 把持物体は両方向に回転でき, 後者の場合, 一方方向にしか回転できない. ちなみに, 回転維持自由度はその定義から常に両方向回転可能であり, 拘束自由度は常に両方向回転不可能であるので, そのような区別を必要としない.

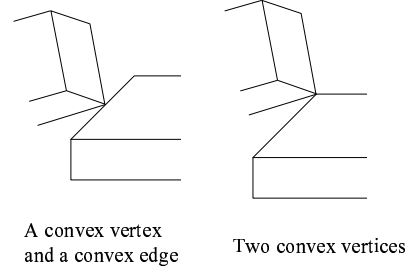


図 3: 2 種類の特異接触要素

そこでさらに, タイプ I, タイプ II 回転離脱自由度を次のように定義する:

タイプ I 回転離脱自由度 離脱変位において, 前者に対応する回転軸の向きの自由度

タイプ II 回転離脱自由度 離脱変位において, 後者に対応する回転軸の向きの自由度

命題 5 タイプ I 回転離脱自由度 d_{r1} は, 次に示す手順を経て計算することができる:

1. 部分空間 V_r を解に持つ不等式 (6) を生成する. その生成の仕方はすでに提案されている [23, 24].

$$\bigcap_i G_i \cdot S_0 \geq 0 \quad (6)$$

2. 行列 $G = (G_1 \dots G_n)$ のランクを計算する. d_{r1} は $3 - \text{Rank}(G) - m_r$ に等しい.

証明 式 (6) の解は可能な微小変位 (維持および離脱変位) における回転軸の向きを表しており, $3 - \text{Rank}(G)$ は式 (7) の解の次元を表している. 式 (7) の解は両方向回転可能な回転軸の向きを表しており, これより $3 - \text{Rank}(G) = m_r + d_{r1}$ が常に成り立つ.

$$\bigcap_i G_i \cdot S_0 = 0 \quad (7)$$

□

命題 6

$$d_{r2} = d_r - d_{r1},$$

ただし d_{r2} はタイプ II 回転離脱自由度である.

証明 自明. □

2.2.2 特異接触状態における指標群

[21] によると, 図 3 に示す 2 つの接触要素 (以後, 特異な接触要素と記述する) を含むとき, 複数分離面が存在し $\exists i, M(i) \neq 1$ となるため, 微小変位を単なる連立線形不等式の形で表すことはできない.

このような場合、これらの特異な接触要素が存在しないものとして解析することにする。つまり、式(8)に対して今までの解析を適用する。ただしこの場合、それぞれの自由度のことを、通常と区別して特異維持、特異離脱、特異拘束自由度と呼ぶ。

$$\bigcap_{\{i|M(i)=1\}} \mathbf{F}_i \cdot \mathbf{S}_1 + (\mathbf{P}_i \times \mathbf{F}_i) \cdot \mathbf{S}_0 \geq 0 \quad (8)$$

ちなみに、特異接触状態では維持、離脱、拘束自由度は0であるとする。また、非特異接触状態では特異維持、特異離脱、特異拘束自由度は0であるとする。IkeuchiとSuehiro[2]による手法では、特異接触状態は明示的に扱われていない。

2.3 接触状態遷移解析

この節では、任意の組み立て作業を再現するのに十分な動作プリミティブ(サブスキルと呼ぶ)を定義し、指示された組み立て作業に対応するサブスキルの列を獲得する方法について述べる。

IkeuchiとSuehiro[2]は、並進維持、並進離脱、並進拘束自由度の違いにより、すべての面接触状態を10種類に分類できることを示した。そして、その10種類の接触状態の間に、13種類しか可能な遷移がないことを示し、13種類の遷移を具体的に調べることににより、組み立て作業の解析を行っていた。

まず、我々は並進および回転変位における維持、離脱、拘束自由度の違いにより、すべての非特異接触状態を分類しようと試みた。しかし138種類もあったため、その間の可能な遷移をすべて調べることは現実的ではないと結論づけた。そこで本論文では、接触状態の遷移に伴い、それらの自由度(以降運動自由度と記述する)のうちいずれかが増加し、いずれかが減少することに注目し、その情報を用いて組み立て作業の解析を行う方法を提案する。

2.3.1 運動自由度の変化の種類

定義より、並進および回転における6種類の運動自由度の和は常に一定、つまり3である。つまり、ある運動自由度が増加すれば、必ず別の運動自由度が減少することになる。この増加、減少した運動自由度の種類に注目すると、全部で6種類の運動自由度が存在するので、 ${}^6P_2 = 30$ 種類の組み合わせが存在する。この増加、減少のことを運動自由度遷移と呼ぶ。A自由度からB自由度への運動自由度遷移とは、A自由度が1減少し、B自由度が1増加する接触状態遷移のことを意味する。

理論的には、すべての30種類の運動自由度遷移は起こりえるが、本論文の目的は、人間の組み立て作業の認識であり、30種類の遷移のうち実際に発生するもののみを対象とする。通常の組み立て作業では発生しない、取り除くべき遷移の候補は以下の2種類である：

- 特異な接触を維持したまま把持物体を動かすことにより発生する6種類の運動自由度遷移

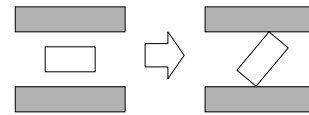


図4: 維持自由度から拘束自由度へ

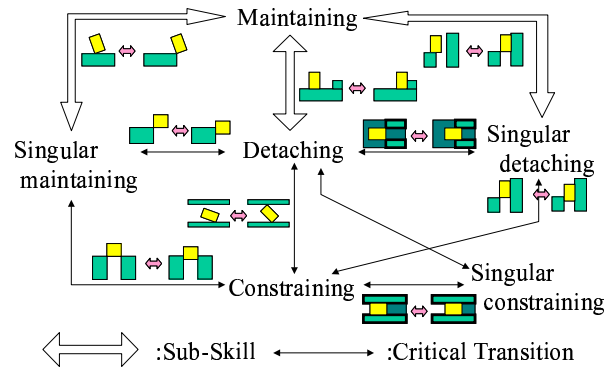


図5: 通常の組み立て作業で現れる運動自由度遷移

- 維持、(特異)拘束自由度間の4種類の運動自由度遷移

前者を取り除く理由は、我々人間にとっても特異な接触を維持したまま把持物体を動かすことが困難であるからである。後者を取り除く理由は、例えば維持自由度から拘束自由度への遷移は、図4に示す2つの接触状態間を回転運動により直接遷移した場合しか発生せず、このような遷移も実現が困難であるからである。

前者より、特異維持、特異離脱、特異拘束間の6つの遷移が、後者より、維持↔拘束、維持↔特異拘束間の4つの遷移が取り除かれる。結果として、通常の組み立て作業では図5に示す20種類の遷移しか現れないことになる。

2.3.2 運動自由度遷移の意味

接触状態が遷移する瞬間の把持物体の変位の方向と運動自由度遷移に対応する基底の方向との関係により、これらの20種類の運動自由度遷移は、次に示す2種類に分類することができる: 1つは、維持自由度↔離脱自由度、維持自由度↔特異維持自由度、維持自由度↔特異離脱自由度の6つである。この場合、2つの方向は一致している。我々は、この運動自由度遷移を引き起こす把持物体の動作を、組み立て動作における必要不可欠な動作要素とみなし、サブスキルと定義する。

もう1つは、残りすべての14種類の運動自由度遷移である。もちろん、この場合2つの方向は一致していない。これらは、実行時の誤差が組み立て作業の進行にどのように影響するかを判断するのに有用である。特に、実行時の誤差が組み立て作業の進行に深刻な影響を与える遷移を、critical transitionと定義する。はじめに、サブスキルとしてどのようなものがあるかについて記

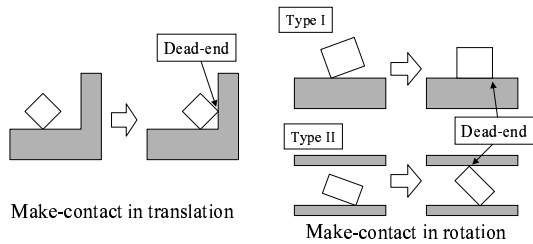


図 6: 維持自由度から離脱自由度へ

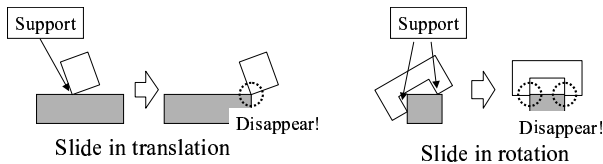


図 7: 維持自由度から特異維持自由度へ

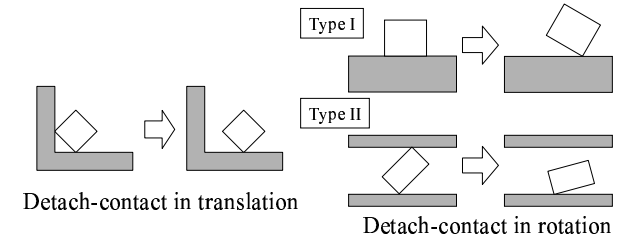


図 9: 離脱自由度から維持自由度へ

述し、次にどのような遷移が critical transition であるかについて記述する。

2.4 サブスキル

節 2.4 から 6 の図では、白色および灰色の物体はそれぞれ把持、環境物体を表すものとする。

2.4.1 維持自由度から離脱自由度へ

図 6 の左側に、並進維持自由度から並進離脱自由度への運動自由度遷移を引き起こす動作の例を示す。この例では、水平方向の変位（これは動作方向に一致する）が、維持変位から離脱変位に変化する。結果として、この動作は前述の運動自由度遷移を引き起こす。我々はこの動作を並進つき当てサブスキルと定義する。

図 6 の右上および右下に、回転維持自由度からタイプ I およびタイプ II 回転離脱自由度への運動自由度遷移を引き起こす動作の例を示す。この例では、紙面垂直軸回りの変位（これは動作方向に一致する）が、維持変位から離脱変位に変化する。結果として、この動作は前述の運動自由度遷移を引き起こす。我々はこれらの動作をタイプ I およびタイプ II 回転つき当てサブスキルと定義する。ちなみに、図の右下の例では、垂直方向において、並進離脱自由度から並進拘束自由度への運動自由度遷移が同時に発生している。

これらのサブスキルは、遷移前の接触状態を維持しながら「つき当たり」接触要素に接触するまで把持物体を動かすことで実現できる。この性質は実装に役に立つ。

2.4.2 維持自由度から特異維持自由度へ

図 7 の左側に、並進維持自由度から並進特異維持自由度への運動自由度遷移を引き起こす動作の例を示す。

この例では、水平方向において、その運動自由度遷移が発生している。我々はこの動作を並進すべらしサブスキルと定義する。ちなみに、垂直方向において、並進離脱自由度から並進特異維持自由度への運動自由度遷移が同時に発生している。

図 7 の右側に、回転維持自由度から回転特異維持自由度への運動自由度遷移を引き起こす動作の例を示す。この例では、紙面垂直軸回りにおいてその運動自由度遷移が発生している。我々はこの動作を回転すべらしサブスキルと定義する。ちなみに、水平、垂直方向において、並進離脱自由度から並進特異維持自由度への運動自由度遷移が同時に発生している。

つき当てサブスキルと比較して、遷移後の接触状態には「つき当たり」接触要素がない。そのかわり、この運動自由度遷移は、常に並進離脱自由度から並進特異維持自由度への遷移を伴う。結果として、これらのサブスキルは、接触状態を維持しながら「支え」接触要素がなくなるまで動かすことで実現できる。

2.4.3 維持自由度から特異離脱自由度へ

図 8 に、並進維持自由度から並進特異離脱自由度への運動自由度遷移を引き起こす動作の例を示す。この例では、水平方向において、その運動自由度遷移が発生している。この動作は、並進つき当て動作にも並進すべらし動作にも似ている。実際に、遷移前の接触状態には「支え」接触要素が含まれているし、遷移後の接触状態には「つき当たり」接触要素が含まれている。我々はこのような運動自由度遷移を引き起こす動作を、並進つき当てサブスキルで代用する、つまり新たなサブスキルは定義しない。

回転の場合も同様に、回転維持自由度から回転特異離脱自由度への運動自由度遷移を引き起こす動作を、回転つき当てサブスキルで代用する。

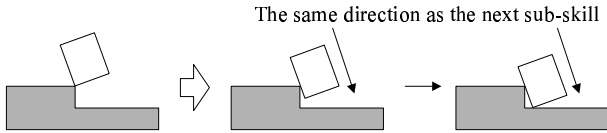


図 10: 特異維持自由度から離脱自由度へ

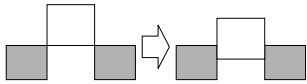


図 11: 特異維持自由度から拘束自由度へ

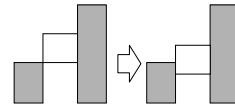


図 12: 特異離脱自由度から拘束自由度へ

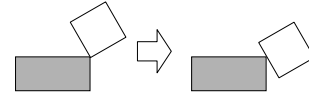


図 13: 特異維持自由度から離脱自由度へ

2.4.4 その他の運動自由度遷移

図 9 の左側 (右側) に、並進 (回転) 離脱自由度から並進 (回転) 維持自由度への運動自由度遷移を引き起こす動作の例を示す。それぞれの動作は並進 (回転) つき当て動作の逆の動作である。我々はこれらの動作を並進 (回転) つき放し動作と定義する。しかし、組み立て作業は維持自由度が少なくなる方向に進行する傾向があるため、これらの動作はめったに現れることはない。また、並進 (回転) 特異離脱自由度から並進 (回転) 維持自由度への運動自由度遷移を引き起こす動作は、並進 (回転) つき放し動作で代用することにする。もちろんこれらの動作もめったに現れることはない。

図 10 に、特異維持自由度から維持自由度への運動自由度遷移を引き起こす動作の例を示す。この動作はすべらし動作の逆の動作である。この動作は、その次の状態に移るための動作の一部と見なすことができるため、新たなサブスキルは定義しない。

2.5 Critical Transitions

2.5.1 特異維持自由度から拘束自由度へ

図 11 に、特異維持自由度から拘束自由度への運動自由度遷移を伴う接触状態遷移の例を示す。この例では、水平方向と紙面垂直軸回りにおいて、前述の運動自由度遷移が発生している。

このような運動自由度遷移を伴う接触状態遷移では、把持物体は狭い「入り口」を通り抜けて次の接触状態へ遷移しなければならないため、運動自由度遷移に関する変位方向において正確な位置姿勢制御を必要とする。よって、この遷移は critical transition である。これは、[25] において critical dimension と定義されているものに対応し、そこではビジュアルフィードバックにより位置姿勢制御を行っている。

2.5.2 特異離脱自由度から拘束自由度へ

図 12 に、特異離脱自由度から拘束自由度への運動自由度遷移を伴う接触状態遷移の例を示す。この例では、水平方向と紙面垂直軸回りにおいて、前述の運動自由度

遷移が発生している。

この場合も、次の接触状態に遷移するためには、運動自由度遷移に関する変位方向において正確な位置姿勢制御を必要とするため、この遷移は critical transition である。しかしこの場合、遷移前の把持物体は、その変位方向において一方向にしか動けないため、位置姿勢制御は比較的簡単である。

2.5.3 特異維持自由度から離脱自由度へ

図 13 に、特異維持自由度から離脱自由度への運動自由度遷移を伴う接触状態遷移の例を示す。たしかに、特異接触状態を実現することは難しいが、この場合、実行時に多少の誤差が発生したとしても次の状態へ遷移することができる。よって、この遷移は critical transition ではない。

2.5.4 特異離脱自由度 ↔ 離脱自由度および特異拘束自由度 ↔ 拘束自由度

図 14 に、特異離脱自由度と離脱自由度の間の運動自由度遷移を伴う接触状態遷移の例を示す。その運動自由度遷移に関する方向において、把持物体は遷移前後で一方向にしか動かせないため、誤差はめったに発生しない。よって、この遷移は critical transition ではない。同様に特異拘束自由度と拘束自由度の間の運動自由度遷移を伴う接触状態遷移も critical transition ではない。

2.5.5 その他の運動自由度遷移

図 15 に、離脱自由度から拘束自由度への運動自由度遷移を伴う接触状態遷移の例を示す。この運動自由度遷移は、タイプ II 回転つきあてサブスキルを実行する際に発生する。この運動自由度遷移に関する方向において、遷移前、把持物体は一方向にしか動かせないため、誤差はめったに発生しない。よって、この遷移は critical transition ではない。

図 16 に、次に示す運動自由度遷移を伴う接触状態遷移の例を示す：

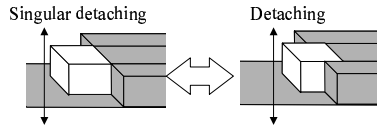


図 14: 特異離脱自由度と離脱自由度の間の運動自由度遷移

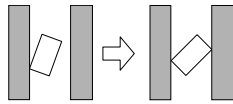


図 15: 離脱自由度から拘束自由度へ

- (a) 拘束自由度から特異維持自由度へ
- (b) 拘束自由度から特異離脱自由度へ
- (c) 離脱自由度から特異維持自由度へ

この運動自由度遷移は、すべらしサブスキルを実行する際に発生する。1 番目と 2 番目の運動自由度遷移に関する方向において、遷移前、把持物体は全く変位することができない、つまり誤差は全く発生しない。よって、この遷移は critical transition ではない。3 番目の運動自由度遷移に関する方向において、遷移前、把持物体は一方向にしか変位することができないので、誤差はめったに発生しない。よって、この遷移は critical transition ではない。

2.6 サブスキルの割り付け、および critical transition の抽出

サブスキルの定義から、その割り付けの方法は以下のようにになると考えられる:

- 並進 (回転) 維持自由度から並進 (回転) 離脱自由度か並進 (回転) 特異離脱自由度への運動自由度遷移が発生したとき、並進 (回転) つき当てサブスキルを割り付ける。
- 並進 (回転) 離脱自由度か並進 (回転) 特異離脱自由度から、並進 (回転) 維持自由度への運動自由度遷移が発生したとき、並進 (回転) つき放しサブスキルを割り付ける。
- 並進 (回転) 維持自由度から並進 (回転) 特異維持自由度への運動自由度遷移が発生したとき、並進 (回転) すべらしサブスキルを割り付ける。

1, 2 番目のルールは正しいが、3 番目のルールは正しくない。特異接触状態の解析における簡略化がこの結果を引き起こしたと考えられる。

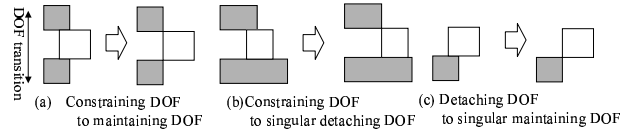


図 16: すべらしサブスキルに伴って発生する運動自由度遷移

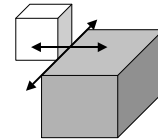


図 17: True singular maintaining DOF or not

例えば、図 17 に示す平行な辺-辺接触 (特異接触状態である) について考える。並進特異維持自由度は 3 であるので、任意の並進変位は同じ性質を持つことが期待される。辺に沿った並進変位は接触状態を維持し続けられるのに対し、辺に垂直な並進変位は接触状態を遷移させてしまう。すべらしサブスキルの割り付けのためには、これら 2 つの特異維持変位を区別する必要がある。

そこで、特異維持自由度から維持自由度の性質を有するものを分離することを考える。維持自由度の性質を持たない特異維持自由度に関する方向の変位は、接触状態を維持し続けられないことに注目し、新たに束縛自由度を定義する。

束縛自由度は、束縛変位の自由度である。束縛自由度は、非特異、特異接触状態の区別なく、接触状態を維持できない変位であると定義する¹。

物体の可能な微小変位が、式 (2) の形で表現されるとする。定義より並進、回転、全束縛自由度は、以下のように定式化される:

並進束縛自由度 行列 R_t のランク

回転束縛自由度 全束縛自由度 - 並進束縛自由度

全束縛自由度 行列 R_a のランク

ただし、

$$R_t = (\mathbf{F}_{11} \cdots \mathbf{F}_{NM(N)})$$

$$R_a = \begin{pmatrix} \mathbf{F}_{11} & \cdots & \mathbf{F}_{NM(N)} \\ \mathbf{P}_1 \times \mathbf{F}_{11} & \cdots & \mathbf{P}_N \times \mathbf{F}_{NM(N)} \end{pmatrix}.$$

束縛自由度を用いると、サブスキル割り付けの正しいルールは以下ようになる:

- 並進 (回転) 維持自由度から並進 (回転) 離脱自由度か並進 (回転) 特異離脱自由度への運動自由度遷移が発生したとき、並進 (回転) つき当てサブスキルを割り付ける。

¹定義より、非特異接触状態の並進、回転、任意運動における束縛自由度は、離脱自由度と拘束自由度の和に等しい。しかし、特異接触状態において、束縛自由度はそれらの和と必ずしも等しくない。

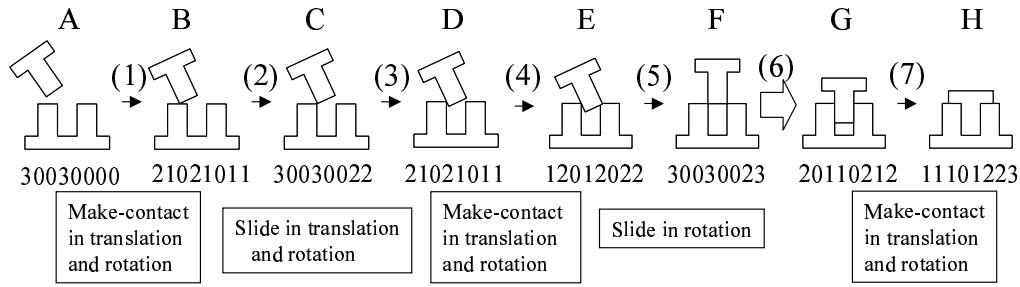


図 18: 組み立て作業理解の例

- 並進 (回転) 離脱自由度から並進 (回転) 特異離脱自由度から、並進 (回転) 維持自由度への運動自由度遷移が発生したとき、並進 (回転) つき放しサブスキルを割り付ける。
- 並進 (回転) 維持自由度から並進 (回転) 特異維持自由度への運動自由度遷移が発生し、かつ並進 (回転) 束縛自由度が増加したとき、並進 (回転) すべらしサブスキルを割り付ける。

また、critical transition は以下に示す運動自由度遷移が発生した接触状態遷移である：

- 並進 (回転) 特異維持自由度から並進 (回転) 拘束自由度への運動自由度遷移を伴う接触状態遷移
- 並進 (回転) 特異離脱自由度から並進 (回転) 拘束自由度への運動自由度遷移を伴う接触状態遷移

2.7 組み立て作業の認識例

この節では、図 18 に示す組み立て作業を提案手法を用いて解析した例を示す。見た目は 2 次元であるが、解析は 3 次元空間中で行われている。図 18 の各接触状態の下の数字は、左から並進維持、並進離脱、並進拘束、回転維持、回転離脱、回転拘束、並進束縛、回転束縛自由度数を表す。ただし状態 C、F は特異接触状態であり、数字は並進特異維持、並進特異離脱、並進特異拘束、回転特異維持、回転特異離脱、回転特異拘束、並進束縛、回転束縛自由度を表す。また白抜き矢印は critical transition を表す。

(1) の遷移では、並進、回転維持自由度がそれぞれ 1 減少し、並進、回転離脱自由度がそれぞれ 1 増加していることより、並進、回転つき当てサブスキル²が割り付けられる。この回転離脱自由度はタイプ I であることより、さらに回転つき当てタイプ I であることがわかる。(4)、(7) の遷移でも、同様の自由度の変化が見られるため、並進、回転 (タイプ I) つき当てサブスキルが割り付けられる。

(2) の遷移では、並進、回転維持自由度がそれぞれ 2 減少し、並進、回転特異維持自由度がそれぞれ 2 増加しており、かつ並進、回転束縛自由度がそれぞれ 1 増加し

²ペグの辺とホール面の面を合わせる動作に対応。

ていることより、並進、回転すべらしサブスキル³が割り付けられる。ここでは、さらに並進、回転離脱自由度がそれぞれ 1 減少し、並進、回転特異維持自由度がそれぞれ 1 増加する変化がみられるが、これはすべらしサブスキルに付随して発生した遷移である。

同様に (5) の遷移でも、並進、回転維持自由度がそれぞれ 1 減少し、並進、回転特異維持自由度がそれぞれ 1 増加する変化が見られるが、この場合回転束縛自由度のみが 1 増加しているため、回転すべらしサブスキルのみが割り付けられる⁴。

(3) の遷移では、並進、回転維持自由度がそれぞれ 2 減少し、並進、回転維持自由度がそれぞれ 2 増加する変化もみられ、次のサブスキルの一部であると見なせる。また、並進、回転特異維持自由度がそれぞれ 1 減少し、並進、回転離脱自由度がそれぞれ 1 増加しており、それ以外の変化がみられないことから、この遷移は critical transition ではないことがわかる。

しかし (6) の遷移では、並進特異維持自由度が 1 減少し、並進拘束自由度が 1 増加していること、および回転特異維持自由度が 2 減少し、回転拘束自由度が 2 増加していることから、この遷移は critical transition であることがわかる。

3 観察による回転ジョイントパラメタの推定

本節では、回転ジョイントでつながれた物体の操作に必要なとなる、ジョイントパラメタを観察により推定する方法を提案する。

ジョイントでつながれた物体の操作に関する研究として、例えば Mason は図 19 に示す 6 つのジョイントの静的な特徴を調べ、position/force hybrid 制御によりそれらの物体を操作する方法を提案している [26]。またジョイントの静的、および動的な特徴に関しては機構学において活発に研究されている。

しかし、そのような物体の操作の際に必要な不可欠であるジョイントパラメタを、人間が手作業で与えなければならなかった。その結果、そのような操作をおこなうロ

³ペグの辺とホール面の辺を合わせる動作に対応。

⁴(2) の遷移と異なり、姿勢が決定されれば、位置は一意に決定されることから、並進すべらしサブスキルが割り付けられていない

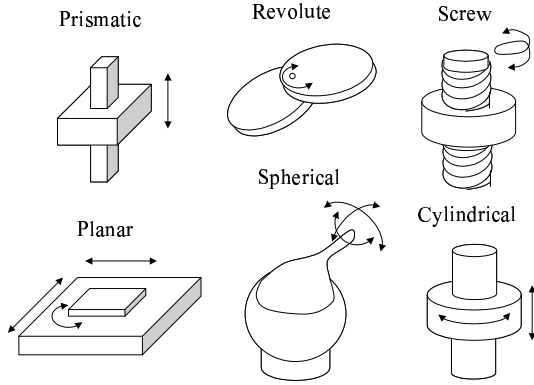


図 19: 低次対偶

ボットプログラムの生成を困難にしていた。そこで本論文では、観察により回転ジョイントのパラメタ (回転軸の向きや回転中心) を獲得する方法を提案する。

実際に、3次元トラッキングを用いてそれらのパラメタを推定する方法はあまり提案されておらず、提案されている手法もパラメタの一部である回転中心を求めるのみであり [27, 28], 回転軸の向きを推定する方法は我々の知る限り提案されていない。

そこで我々は、誤差を含んだ3次元トラッキングの結果から、回転ジョイントのパラメタの推定を行う手法を提案する。具体的に3次元トラッキングの結果からジョイントでつながれた2つの物体の相対位置姿勢が得られているものとする。本論文では誤差はガウシアンノイズであると仮定する。

3.1 前準備

2物体間の姿勢変位が 3×3 直行行列 Θ で表されているとする。その姿勢変位は、ある回転軸 \mathbf{n} 回りに θ ラジアン回転したものと表現し直すことができる。本論文では、回転量の絶対値 $|\theta|$ を行列 Θ の大きさであるとし、それを $\|\Theta\|$ と表す。

2つの物体 A, B が、回転ジョイントにより接続されているとする。本論文では、物体 B からみた物体 A の相対位置姿勢 ${}^B\mathbf{q}_A = ({}^B\mathbf{t}_A, {}^B\hat{\Theta}_A^i)$ が3次元トラッキングシステム等により得られたとき、回転ジョイントのパラメタを推定する方法を提案する。ただし、 ${}^B\mathbf{t}_A^i$ は相対位置を表す3次元ベクトル、 ${}^B\hat{\Theta}_A^i$ は相対姿勢を表す 3×3 直行行列であるとする。系列の順番をあらわす変数 $i (\geq 1)$ を、場合に応じて各項の上付きの文字として表現することもある。

3次元トラッキング等により得られた相対位置姿勢は通常誤差を含んでいる。そこで $\hat{\mathbf{q}}_i = ({}^B\hat{\mathbf{t}}_A^i, {}^B\hat{\Theta}_A^i)$ は正確な (推定された) 相対位置姿勢、 $\Delta\mathbf{q}_i = (\Delta\mathbf{t}^i, \Delta\Theta^i)$ は正確なものとトラッキングにより得られたものとの差を表すものとする。

3.2 回転ジョイントのパラメタ推定

3.2.1 定式化

回転ジョイントのパラメタは、接続された物体 A, B から見た回転軸の向き ${}^A\mathbf{1}$ と ${}^B\mathbf{1}$, 物体 A, B から見た回転中心の位置 ${}^A\mathbf{c}$ と ${}^B\mathbf{c}$ からなる。ただし $|\mathbf{1}| = |\mathbf{c}| = 1$ が成り立っているものとする。

回転ジョイントのパラメタと接続された物体 A, B の相対位置姿勢の間には、すべての i に対して式 (9) と (10) に示す関係が成り立っている。

$${}^B\mathbf{1} = {}^B\hat{\Theta}_A^i {}^A\mathbf{1} \quad (9)$$

$${}^B\mathbf{c} = {}^B\hat{\Theta}_A^i {}^A\mathbf{c} + {}^B\hat{\mathbf{t}}_A^i \quad (10)$$

3.2.2 パラメタ推定

まず回転軸の向きに関するパラメタ ${}^A\mathbf{1}$, ${}^B\mathbf{1}$ を式 (9) を用いて推定する方法を提案する。式 (9) は姿勢誤差を表す項 $\Delta\Theta^i$ を用いて式 (11) のように書き直すことができる。

$${}^B\mathbf{1} = \Delta\Theta^i {}^B\hat{\Theta}_A^i {}^A\mathbf{1} \quad (11)$$

まず姿勢誤差の項 $\Delta\Theta^i$ を、式 (12) に示す2つの回転行列の積 (1. 回転軸 $\mathbf{1}_i$ 回りに θ_{2i} ラジアン回転, 2. 回転軸 ${}^B\mathbf{1}$ 回りに θ_{1i} ラジアン回転) で表すものとする。

$$\Delta\Theta^i = R({}^B\mathbf{1}, \theta_{1i})R(\mathbf{1}_i, \theta_{2i}) \quad (12)$$

回転行列 $R(\mathbf{l}, \theta)$ は回転軸 \mathbf{l} 回りに θ ラジアン回転したものを表しており、すべての i において、 ${}^B\mathbf{1} \cdot \mathbf{1}_i = 0$ が成り立っているものとする。

式 (12) を式 (11) に代入することにより、式 (13) が得られる。

$$R({}^B\mathbf{1}, \theta_{1i})^T {}^B\mathbf{1} = R(\mathbf{1}_i, \theta_{2i}) {}^B\hat{\Theta}_A^i {}^A\mathbf{1} \quad (13)$$

式の左辺にあたる項 $R({}^B\mathbf{1}, \theta_{1i})^T {}^B\mathbf{1}$ の値は、 θ_{1i} の値に関わらず一定である。これは、回転軸 ${}^B\mathbf{1}$ 回りの回転が誤差によるものかどうかを区別できないことを意味している。ここでは、姿勢誤差の項 $\Delta\Theta^i$ を最小化するため、 $\theta_{1i} = 0$ であるとする。

次に、 ${}^B\mathbf{1}^T$ を式 (13) の両辺に右側からかけることにより、式 (14) を得る。なお、記述を簡単にするため、 θ_{2i} を改めて θ_i と記述する。

$$R(-\mathbf{1}_i, \theta_i) {}^B\mathbf{1} {}^B\mathbf{1}^T = {}^B\hat{\Theta}_A^i {}^A\mathbf{1} {}^B\mathbf{1}^T \quad (14)$$

式 (14) の左辺は、式 (15) のように書くことができる。ただし、

$$[(x, y, z)]_{\times} = \begin{pmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{pmatrix}$$

は、いわゆる歪み対称行列と呼ばれるものである。

$$(I - \sin \theta_i [\mathbf{1}_i]_{\times} + (1 - \cos \theta_i) [\mathbf{1}_i]_{\times}^2) {}^B\mathbf{1} {}^B\mathbf{1}^T \quad (15)$$

実際に計算することにより、以下の式が成り立つことが確認できる:

$$\begin{aligned}\text{Tr}({}^B\mathbf{1}\mathbf{1}^T) &= 1 \\ \text{Tr}([\mathbf{l}_i]_{\times} {}^B\mathbf{1}\mathbf{1}^T) &= 0 \\ \text{Tr}([\mathbf{l}_i]_{\times}^2 {}^B\mathbf{1}\mathbf{1}^T) &= ({}^B\mathbf{1} \cdot \mathbf{l}_i)^2 - 1 = -1\end{aligned}$$

これらを用いることにより、式 (16) が得られる。

$$\text{Tr}(R(-\mathbf{l}_i, \theta_i) {}^B\mathbf{1}\mathbf{1}^T) = \cos \theta_i \quad (16)$$

この結果により、我々は最適化に適した評価関数を手に入れることができる。 $1 - \cos \theta_i$ が最小となるとき、姿勢誤差の項の大きさ $\|\Delta\theta^i\|$ 、つまり $|\theta_i|$ の値は最小となることから、非線形最適化手法を用いて式 (17) の値を最小化することにより、パラメタ A_1, B_1 を推定することができる。

$$\sum_i (1 - \cos \theta_i) = \sum_i (1 - \text{Tr}({}^B\Theta_A^i A_1 B_1^T)) \quad (17)$$

次に、推定された相対姿勢 ${}^B\hat{\Theta}_A^i$ を計算する方法を述べる。これは3次元ベクトル ${}^B\Theta_A^i A_1$ と B_1 の外積を用いることにより、簡単に計算することができる。

最後に、回転中心に関するパラメタ A_c, B_c を式 (10) を用いて推定する方法を提案する。式 (10) は、位置誤差を表す項 Δt^i を用いて、式 (18) のように書き直すことができる。

$$\Delta t^i = B_c - {}^B\hat{\Theta}_A^i A_c - B t_A^i \quad (18)$$

これらのパラメタは最小自乗法を用いて推定することができる。具体的には、式 (19) を解けばよい。ただし $A_i = \begin{pmatrix} -{}^B\hat{\Theta}_A^i & I \end{pmatrix}$ である。

$$\left(\sum_i A_i^T A_i \right) \begin{pmatrix} A_c \\ B_c \end{pmatrix} = \sum_i A_i^T B t_A^i \quad (19)$$

なお回転中心の選び方には曖昧性が生じるため、行列 $\sum_i A_i^T A_i$ は必ずしも正則であるとは限らない。そこで我々は特異値分解を用いてこの方程式を解いている。

3.3 実装と実験

3.3.1 実装

この節では、実際の実装の方法について述べる。実装にあたって、次に示す2つの問題を解決する必要がある:

- 回転軸および姿勢をどのようにパラメタ化するか?
- どのような非線形最適化を用いるか?

1番目の問題に関して、我々は回転軸の向きを極座業表現を用いてパラメタ化した。具体的に、 A_1, B_1 はパ

ラメタ p, q, r, s を用いて式 (20) のように表すことができる。

$$\begin{aligned}A_1 &= (\sin p \cos q, \sin p \sin q, \cos p)^T \\ B_1 &= (\sin r \cos s, \sin r \sin s, \cos r)^T\end{aligned} \quad (20)$$

また姿勢をロールピッチヨー表現を用いてパラメタ化した。具体的に、 ${}^B\Theta_A$ はパラメタ α, β, γ を用いて式 (21) のように表すことができる。ただし $c_\alpha = \cos \alpha, s_\alpha = \sin \alpha, c_\beta = \cos \beta, s_\beta = \sin \beta, c_\gamma = \cos \gamma, s_\gamma = \sin \gamma$ であるとする。

$${}^B\Theta_A = \begin{pmatrix} c_\alpha c_\beta & c_\alpha s_\beta s_\gamma - s_\alpha c_\gamma & c_\alpha s_\beta c_\gamma + s_\alpha s_\gamma \\ s_\alpha c_\beta & s_\alpha s_\beta s_\gamma + c_\alpha c_\gamma & s_\alpha s_\beta c_\gamma - c_\alpha s_\gamma \\ -s_\alpha & c_\beta s_\gamma & c_\alpha c_\gamma \end{pmatrix} \quad (21)$$

これらの表現方法はいくつかの特異点を有しており、それが非線形最適化における収束を困難にする場合がある。しかし実際に実験を行った結果、そのような問題は見られなかった。

2番目の問題に関して、我々は非線形最適化手法として共役勾配法を用いた。なお、共役勾配法における初期解 A_{1in}, B_{1in} を以下のような手続きを経て設定した:

1. $\|{}^B\Theta_A^{iT} {}^B\Theta_A^j\|$ の値が十分 $\frac{\pi}{2}$ に近くなるような i, j を選ぶ。
2. A_{1in} の値を $R(A_{1in}, \theta) = {}^B\Theta_A^{iT} {}^B\Theta_A^j$ を解くことにより決定する。
3. B_{1in} の値を $B_{1in} = {}^B\Theta_A^i A_{1in}$ を用いて決定する。

なお、式 (22) は誤差のない状況では常に成立している。

$${}^B\hat{\Theta}_A^{iT} {}^B\hat{\Theta}_A^j A_1 = A_1 \quad (22)$$

この式より回転行列 ${}^B\hat{\Theta}_A^{iT} {}^B\hat{\Theta}_A^j$ は回転軸 A_1 まわりの回転を表しているといえる。

3.3.2 シミュレーション実験

はじめに、シミュレーション実験において、真値と推定値を実際に比較することにより、我々の提案手法の性能評価を行った。本実験の目的は以下のとおりである:

- 我々の提案手法と他の手法との性能を比較する
- 設定条件と推定精度の関係を調べる

実際に、我々は以下に示す方法で、シミュレーション実験を行った:

1. 適当に回転ジョイントのパラメタを設定する
2. 設定されたパラメタに従って、正確な相対位置姿勢の列を生成する
3. 生成された相対位置姿勢の列にノイズを加える
4. ノイズを加えた相対位置姿勢の列に対して、パラメタ推定手法を適用する

表 1: 提案手法と他の手法との性能評価

A_1 に関して推定された値と真の値との差

回転量 - サンプル数	手法 A 平均 (標準偏差) [deg]	提案手法 平均 (標準偏差) [deg]
45[deg] - 50	3.857 (2.104)	1.330 (0.6914)
45[deg] - 100	3.800 (2.157)	0.9352 (0.5167)
45[deg] - 200	3.884 (2.143)	0.6989 (0.3877)
90[deg] - 50	2.128 (1.120)	0.6916 (0.3455)
90[deg] - 100	2.053 (1.147)	0.4881 (0.2552)
90[deg] - 200	2.077 (1.132)	0.3485 (0.1891)
180[deg] - 50	1.999 (1.117)	0.3951 (0.2000)
180[deg] - 100	2.051 (1.177)	0.2690 (0.1424)
180[deg] - 200	2.044 (1.130)	0.1904 (0.09464)

B_1 に関して推定された値と真の値との差

回転量 - サンプル数	手法 A 平均 (標準偏差) [deg]	提案手法 平均 (標準偏差) [deg]
45[deg] - 50	3.909 (2.122)	1.329 (0.6838)
45[deg] - 100	3.741 (2.157)	0.9411 (0.5109)
45[deg] - 200	3.862 (2.126)	0.6904 (0.3826)
90[deg] - 50	2.096 (1.199)	0.6890 (0.3474)
90[deg] - 100	2.051 (1.158)	0.4929 (0.2558)
90[deg] - 200	2.060 (1.134)	0.3397 (0.1907)
180[deg] - 50	2.058 (1.151)	0.3818 (0.1970)
180[deg] - 100	2.048 (1.170)	0.2701 (0.1468)
180[deg] - 200	2.042 (1.151)	0.1934 (0.1010)

5. 推定されたパラメタと真のパラメタを比較する

実際に、上に示すステップを 1000 回繰り返し行うことにより性能評価を行った。

まずノイズの加え方について具体的に述べる。実際に式 (23) に従ってノイズを加えた。ただし Δt , $\Delta \theta$ はそれぞれ位置、姿勢に関するノイズである。

$$\begin{aligned} {}^B \mathbf{t}_A^i &= \Delta \mathbf{t} + {}^B \hat{\mathbf{t}}_A^i \\ {}^B \Theta_A^i &= \Delta \Theta + {}^B \hat{\Theta}_A^i \end{aligned} \quad (23)$$

$\Delta \mathbf{t}$ の大きさは、0 から 10[mm] の間の一様分布に従ってランダムに決定した。もちろん $\Delta \mathbf{t}$ の向きは、ランダムに設定してある。姿勢に関するノイズを表す項 $\Delta \theta$ は、式 (24) に従って生成した。ただし $\Delta \theta$ は 0 から 5[deg] の間の一様分布に従ってランダムに設定しており、また \mathbf{l} もランダムに設定した。

$$\Delta \Theta = R(\mathbf{l}, \Delta \theta) \quad (24)$$

まず、我々の提案手法と他の手法との比較を行った。ここでは、本手法の中心的話題である回転軸の向きの推定の精度に注目して性能比較を行った。比較対象として次のような方法 (以降、手法 A と記述する) を選んだ: それは $\|{}^B \Theta_A^i - {}^B \Theta_A^j\|$ ができるだけ $\frac{\pi}{2}$ 離れた 2 つの相対姿勢 ${}^B \Theta_A^i, {}^B \Theta_A^j$ を用いて回転軸を推定する方法である。

確かに、手法 A によって得られた回転軸の向きの平均をとることにより、さらに良い推定を行うことは可能

表 2: 設定条件と推定精度の関係

A_1 に関して推定された値と真の値との差

	50 平均 (標準偏差) [deg]	100 平均 (標準偏差) [deg]	200 平均 (標準偏差) [deg]
45[deg]	1.330 (0.6914)	0.9352 (0.5167)	0.6989 (0.3877)
90[deg]	0.6916 (0.3455)	0.4881 (0.2551)	0.3485 (0.1891)
180[deg]	0.3951 (0.2000)	0.2690 (0.1424)	0.1904 (0.09464)

B_1 に関して推定された値と真の値との差

	50 平均 (標準偏差) [deg]	100 平均 (標準偏差) [deg]	200 平均 (標準偏差) [deg]
45[deg]	1.329 (0.6838)	0.9411 (0.5109)	0.6904 (0.3826)
90[deg]	0.6890 (0.3474)	0.4929 (0.2558)	0.3397 (0.1907)
180[deg]	0.3818 (0.1970)	0.2701 (0.1468)	0.1934 (0.1010)

であるかも知れない。しかしその際 S^2 つまり球面空間における和を定義する必要がある。一般的に球面空間で和を定義することは困難であり、このような計算により得られた結果の数学的正当性は、非常に希薄なものである。

実際に以下に示す 9 つの条件下において性能比較を行った:

- 相対位置姿勢のサンプル数: 50, 100, 200
- 回転量: 45[deg], 90[deg], 180[deg]

表 1 に比較した結果を示す。

この表から、我々は次のような考察を得た: まず我々の提案手法は手法 A にくらべて性能が良い。当然のことではあるが、手法 A の推定精度は回転量が $\frac{\pi}{2}$ 以下になると急激に悪くなる。またサンプル数の増加に対して推定精度の向上がまったくみられない。それに対して我々の手法では、サンプル数、回転量の増加により推定精度の向上が見られる。

次に、設定条件と回転軸の向きの推定精度の関係について述べる。本実験でも、先と同じ 9 つの条件下で、推定精度との関係を調べた。表 2 にその結果を示す。この表から、我々は次のような考察を得た: 回転量に比例してパラメタの推定精度が向上している、つまり回転量が 2 倍になると推定誤差の平均および標準偏差は半分になる。また推定精度の 2 乗がサンプル数に比例する、つまりサンプル数が 4 (= 2²) 倍になると推定誤差の平均および標準偏差は半分になる。この結果から、回転量の小さい回転ジョイントのパラメタ推定は困難であると結論づけられる。



図 20: 実験対象物: おもちゃのペンチ

3.3.3 実物体を用いた実験

– レーザレンジファインダを用いた場合 –

実際に、実物体を用いて回転ジョイントのパラメタの推定を行った。本実験では、図 20 に示すおもちゃのペンチを実験に用い、以下に示す手順で実験を行った:

1. ペンチの各パーツ間の角度を変えながら、その3次元形状をレーザレンジファインダ(本実験では、コニカミノルタホールディングス株式会社製のVivid 910を用いた)を用いて複数回計測する
2. ペンチを構成する2つのパーツのうちの1つを、適当な2枚のレンジデータ間で位置あわせを行う
3. 位置あわせされた2枚のレンジデータ間の差分(共通している部分とそうでない部分)をとることによりそれぞれのパーツをモデル化する(図 21 参照)
4. 生成されたモデルを用いて、2つのパーツの相対位置姿勢の列を得る
5. 得られた相対位置姿勢から、回転ジョイントのパラメタを推定する

位置合わせには3次元テンプレートマッチング法 [29] を用いた。これはいわゆる Iterative Closest Point 法 [30] の一種であるが、重みつき最小自乗法を用いることにより、誤差にロバストであるという特徴がある。

図 21 にペンチのそれぞれのパーツをモデリングした結果を示す。部分的に正しくパーツ分けされていないところがあるが(図中の赤丸で囲まれた部分)、3次元テンプレートマッチング法を用いて相対位置関係の列を得るには十分なモデル化である。

実際に、このペンチの回転量は約 30[deg] であるので、推定は非常に難しいと考えられる。図 22 に推定結果を示す。図中の青い線が推定された回転軸を示す。

3.3.4 実物体を用いた実験

– ステレオビジョンを用いた場合 –

この節では、ステレオビジョンを用いて、回転ジョイントをもつ実物体のパラメタの推定を行った。前節で用いたおもちゃのペンチを、ステレオビジョンを用いてトラッキングするのは非常に困難であったため、今回は図 23 に示す Lego のパーツを用いた。

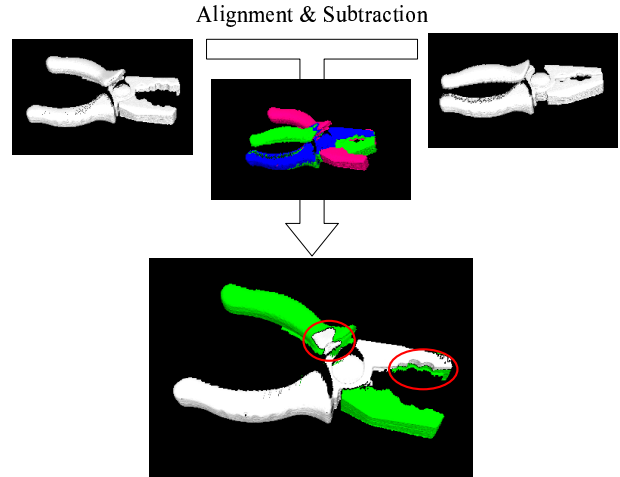


図 21: パーツのモデル化

ステレオビジョンの精度の関係上、トラッキングのために正確なパーツモデルは必要ないと判断したため、CAD システムを用いて大まかなパーツモデルを生成した。なおトラッキングには、前述の3次元テンプレートマッチング法 [29] を用いた。

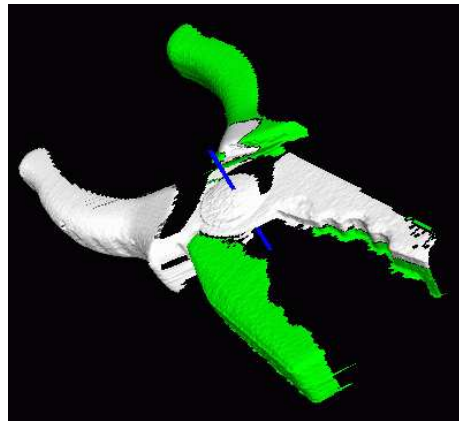
図 24 に推定結果を示す。図中の赤い線が推定された回転軸を示す。図中白色のパーツ(図 23 において青色の物体)に関しては、比較的正確に回転軸の向き、位置ともに推定されていると言える。しかし図中緑色のパーツ(図 23 において赤色の物体)に関しては、あまり正確な推定とは言いがたい。我々は視覚の誤差をガウシアンノイズであると仮定したが、実際その仮定は正しくないため、推定精度が悪くなったと考えられる。ただし、一般的に視覚の誤差分布を先見的に予測することは非常に困難であることを考慮すれば、良く推定できていると考えられる。

4 まとめ

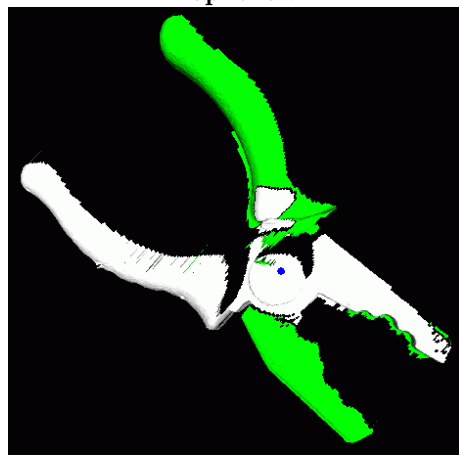
本論文では、ステレオビジョンシステムによる観察を通じて、剛体を操る作業、具体的には以下に示す2つの作業を理解する方法を提案した。ここでいう理解とは、作業の抽象化および動作プリミティブ列への変換を指す。

- 2 多面体物体を用いた組み立て作業
- 回転ジョイントによって接続されたリンクを操る作業

本論文では、回転動作を含めた視覚による組み立て作業理解の方法を提案した。まず、[2] で定義されている運動自由度を、回転運動を含めた上で拡張し、実際に導出するための方法について記述した。さらに作業理解の質を改善するために、回転軸に関する運動自由度を定義した。



Top View



Side View

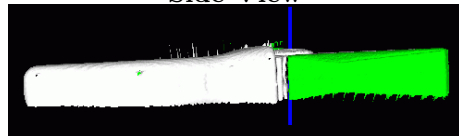


図 22: 回転ジョイントのパラメタ推定の結果

次に、通常表れる運動自由度の遷移に注目し、組み立て作業を記述する上で必要不可欠となるサブスキル、および組み立て作業の進行上、重要なポイントとなる critical transition を定義し、実際の接触状態遷移が与えられたときにそれらを導出する方法を示した。

実際に、ペグインサージョン動作に適用した例を示した。ペグインサージョン動作のような、様々な接触状態が表れる組み立て作業を扱えることから、本手法は多くの組み立て作業に対して有効であると我々は考えている。

さらに本論文では、回転ジョイントによって接続された2つの物体の相対位置姿勢から回転ジョイントのパラメタを推定する方法について提案した。その際、相対位置姿勢は3次元トラッキングシステム等によって得られるものとし、多少の誤差(ガウシアン分布であると仮定した)が含まれているものとした。

まず、回転ジョイントにおいて、パラメタと相対位置姿勢の間の拘束を定式化した。そして我々が定義した評価関数を最小化することによりパラメタ推定を行う

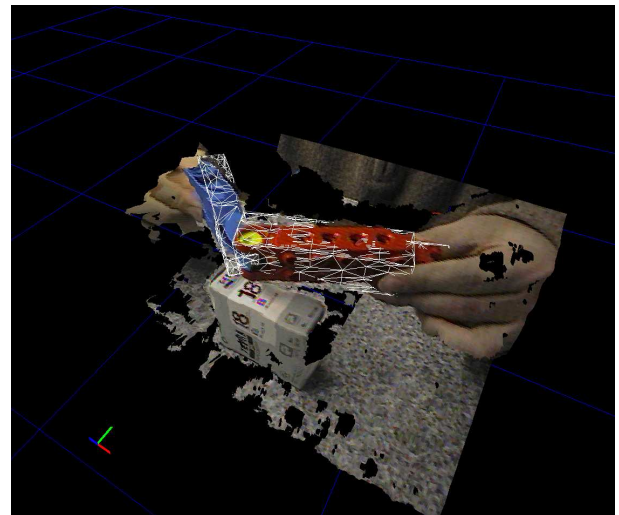


図 23: トラッキングの様子

方法を提案した。

次に我々はシミュレーション実験と実物体を用いた実験を通じて提案手法の性能評価を行った。シミュレーション実験では、本手法の統計的な性質を示した。それにより、本手法は推定精度が回転量に比例していること、および推定精度の2乗がサンプル数に比例していることを示した。その結果は統計的視点からみて妥当なものであった。

謝辞

なお本研究は、文科省科研費補助金特定領域研究(C)課題番号 15017222 及び、独立行政法人科学技術振興機構・池内 CREST プロジェクトの補助を受けている。

参考文献

- [1] J. C. Latombe: "Robot motion planning," Kluwer Academic Publishers.
- [2] K. Ikeuchi and T. Suehiro: "Toward an assembly plan from observation part i: Task recognition with polyhedral objects," IEEE Trans. on Robotics and Automation, **10**, 3 (1994).
- [3] Y. Kuniyoshi, M. Inaba and H. Inoue: "Learning by watching: Extracting reusable task knowledge from visual observation of human performance," IEEE Trans. on Robotics and Automation, **10**, 6 (1994).
- [4] S. Schaal: "Is imitation learning the route to humanoid robots?," Trends in Cognitive Sciences, **3**, pp. 233 - 242 (1999).
- [5] R. Dillmann and M. Bordegoni: "Learning robot behavior and skills based on human demonstration and advice: The machine learning paradigm," Int. Symp. on Robotics Research (1999).
- [6] V. Gallese and A. Goldman: "Mirror neurons and the simulation theory of mind-reading," Trends in Cognitive Sciences, **2**, 12, pp. 493 - 501 (1998).
- [7] M. Skubic and R. Volz: "Acquiring robust, force-based assembly skills from human demonstra-

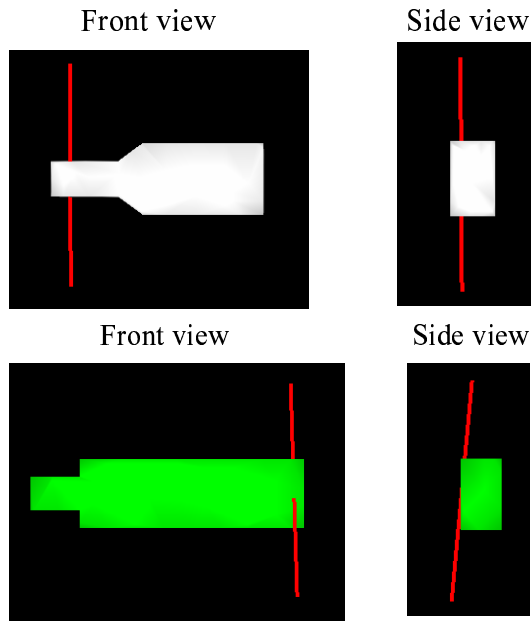


図 24: 回転ジョイントのパラメタ推定の結果

tion,” *IEEE Trans. on Robotics and Automation*, **16**, 6 (2000).

- [8] T. Fukuda, M. Nakaoka, T. Ueyama and Y. Hasegawa: “Direct teaching and error recovery method for assembly task based on a transition process of a constraint condition,” *IEEE Int. Conf. on Robotics and Automation*, pp. 1518 – 1523 (2001).
- [9] T. Takahashi, H. Ogata and S. Muto: “Robotic assembly operation based on task-level teaching in virtual reality,” *IEEE Int. Conf. on Robotics and Automation* (1992).
- [10] H. Onda, T. Ogasawara, H. Hirukawa, K. Kitagaki, A. Nakamura and H. Tsukune: “A telerobotics system using planning functions based on manipulation skills and teaching-by-demonstration technique in vr,” *Journal of the Robotics Society of Japan*, **18**, 7, pp. 979 – 994 (2000).
- [11] Y. Maeda, N. Ishido, H. Kikuchi and T. Arai: “Teaching of grasp/graspless manipulation for industrial robots by human demonstration,” *IEEE Int. Conf. on Intelligent Robots and Systems*, pp. 1523 – 1528 (2002).
- [12] H. Miyamoto and M. Kawato: “A tennis serve and upswing learning robot based on bi-directional theory,” *Neural Networks*, pp. 1331 – 1344 (1998).
- [13] S. Schaal, C. G. Atkeson and S. Vijayakumar: “Real-time robot learning with locally weighted statistical learning,” *IEEE Int. Conf. on Robotics and Automation* (2000).
- [14] A. J. Ijspeert, J. Nakanishi and S. Schaal: “Movement imitation with nonlinear dynamical systems in humanoid robots,” *IEEE Int. Conf. on Robotics and Automation*, pp. 1398 – 1403 (2002).
- [15] T. Inamura, Y. Nakamura, H. Ezaki and I. Toshima: “Imitation and primitive symbol acquisition of humanoids by the integrated mimesis loop,” *IEEE Int. Conf. on Robotics and Automation*, pp. 4208 – 4213 (2001).
- [16] C. G. Atkeson and S. Schaal: “Robot learning from demonstration,” *Int. Conf. on Machine Learning*, **2**, 12, pp. 12 – 20 (1997).
- [17] S. B. Kang and K. Ikeuchi: “Toward automatic robot instruction from perception - recognizing a grasp from observation,” *IEEE Trans. on Robotics and Automation*, **9**, 4, pp. 432 – 443 (1993).
- [18] M. Tsuda, T. Takahashi and H. Ogata: “Generation of an assembly-task model analyzing human demonstration,” *Journal of the Robotics Society of Japan*, **18**, 4, pp. 535 – 544 (2000).
- [19] J. Takamatsu, K. Ogawara, H. Kimura and K. Ikeuchi: “Correcting observation errors for assembly task recognition,” *IEEE Int. Conf. on Intelligent Robots and Systems*, pp. 1208 – 1213 (2002).
- [20] J. Takamatsu, H. Kimura and K. Ikeuchi: “Calculating optimal trajectories from contact transitions,” *IEEE Int. Conf. on Intelligent Robots and Systems*, pp. 232 – 237 (2002).
- [21] 比留川, 松井, 高瀬: “多面体間の接触による拘束条件を幾何モデルから導出する一般的なアルゴリズム,” *日本ロボット学会誌*, **9**, 4, pp. 415 – 426 (1991).
- [22] M. S. Ohwovoriole and B. Roth: “An extension of screw theory,” *Journal of Mechanical Design*, **103**, pp. 725 – 735 (1981).
- [23] H. W. Kuhn and A. W. Tucker: “Linear inequalities and related systems,” *Annals of Mathematics Studies*, **38**, (1956).
- [24] H. Hirukawa, T. Matsui and K. Takase: “Automatic determination of possible velocity and applicable force of frictionless objects in contact from a geometric model,” *IEEE Trans. on Robotics and Automation*, **10**, 3, pp. 309 – 322 (1994).
- [25] J. Miura and K. Ikeuchi: “Task-oriented generation of visual sensing strategies in assembly tasks,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **20**, 2 (1998).
- [26] M. T. Mason: “Compliance and force control for computer controlled manipulators,” *IEEE Trans. on Systems, Man, and Cybernetics*, **11**, 6 (1981).
- [27] J. H. Challis: “Estimation of the finite center of rotation in planar movements,” *Medical Engineering and Physics*, **23**, pp. 227 – 233 (2001).
- [28] G. K. M. Cheung, S. Baker and T. Kanade: “Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture,” *Proc. of CVPR* (2003).
- [29] M. D. Wheeler and K. Ikeuchi: “Sensor modeling, probabilistic hypothesis generation, and robust localization for object recognition,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **17**, pp. 252 – 265 (1995).
- [30] P. Besl and N. McKay: “A method for registration of 3-d shapes,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **14**, 2 (1992).