

視覚誤差修正能力を有する観察からの組み立て動作理解

高松淳¹ 小川原光一² 木村浩³ 池内克史³

¹ 東京大学大学院情報理工学系研究科

² 東京大学大学院工学系研究科

³ 東京大学生産技術研究所

概要

ロボットプログラムの簡略化のために様々な方法が提案されてきた。その際の問題の一つとして、いかにして行わせたい動作を教示するかということがあげられる。ロボットビジョンによる教示は、ビジョンがロボットに標準装備されているものであり特殊なデバイスを用いる必要がないこと、また教示側も教示動作をビジョンの前で行うだけでよく、大変使いやすいものであるという長所がある。その反面、ビジョンの誤差が教示動作の理解、特に組み立て作業のような細かな動作が多い作業の理解を困難にしている。そこで本手法では、視覚による組み立て動作理解のために、ビジョンの誤差を、接触状態、及び、それらの間の可能な直接遷移を用いて修正する方法について述べる。

Assembly Task Recognition with Vision Error Correction Ability from Observation

Jun Takamatsu†, Koichi Ogawara†, Hiroshi Kimura†, and Katsushi Ikeuchi†

¹ Graduate School of Information Science and Technology, University of Tokyo

² Graduate School of Engineering, University of Tokyo

³ Institute of Industrial Science, University of Tokyo

Abstract

Although various methods for simplification of robot programming have been proposed, the key is how to teach the operation to the robot. Using a robot vision system has two merits: first, no special equipment is necessary because a robot vision system is normal equipment; second, the method is very easy because we only have to perform the operation in front of the vision system. Unfortunately, the method has the disadvantage that vision errors prevent operation recognition. This is particularly the case in assembly tasks, in which much fine operation appears. In this paper, we propose a method for correcting vision errors by using contact relations and those transitions for assembly task recognition from observation.

1 はじめに

ロボットプログラムの簡略化のために様々な方法が提案されてきた。その際の問題の一つとしてとして、いかにしてロボットに行わせたい動作を教示するか、があげられる。例えば、ティーチペンダント、テレオペ

レーションのように、ロボットに動きそのものを何らかの手段で教える方法が提案されたが、動的な環境変化に弱いなどの欠点があった。特に、組み立て作業に適応する場合、それが重大な問題となるため、動きを細かく解析し柔軟さを身につけることにより、問題を

解決する方法が主流となっている。

本論文では、ロボットビジョンを用いた教示方法を考察する。ビジョンを用いた教示方法では、1. ビジョンはロボットには必須のデバイスであり標準的に使用可能であること、2. 教示者は行なわせたい動作を見せるだけで良いので手軽である、という良い特徴がある反面、ビジョンのデータの誤差が動作解析を難しくするという欠点があり、組み立て動作のような細かな作業が含まれるものについては、致命的な問題となってしまう。そこで本論文では誤差を含んだビジョンデータから組み立て動作を理解する方法を提案する。

2 組み立て動作理解

組み立て動作は目的とした接触状態の変化を実現することに等しい。無限の接触状態、及び、その遷移が存在し得るため、組み立て動作理解のためには、それらを分析するための手段が必要となる。ここでは、接触状態と物体の可動範囲には密接な関係があることを手がかりとして、可動範囲を分析することで、接触状態を解析する方法について述べる。

2.1 動作次元の分類

以前の APO システム [1] では、並進運動に関して把持物体の動作自由度を以下の3つにわけ、それらの遷移によりロボットの動作を決定していた。(図1参照)

並進維持自由度 接触を維持しながら並進することのできる自由度

並進離脱自由度 接触を維持しながら並進することのできない自由度

並進拘束自由度 全く並進することのできない自由度

本論文では並進運動だけでなく回転運動も扱いたいので、回転に関しても上の3つの自由度を定義する。

回転維持自由度 接触を維持しながら回転することのできる回転軸の向きの自由度¹

回転離脱自由度 接触を維持しながら回転することができない回転軸の向きの自由度

¹3次元空間において位置姿勢6つの自由度が存在する。並進運動の自由度が3であるので、残り3つの自由度が回転運動の自由度であるが見ながちであるが、並進運動は、軸が無限遠に離れた回転運動と見なすことができることから、その両者を厳密に分離することは難しい。そのため、ここでは、回転軸の向き(これは3自由度である)を用いる。

回転拘束自由度 全く回転することができない回転軸の向きの自由度

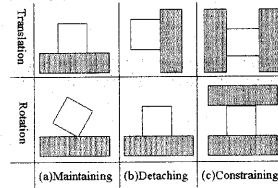


図1: 維持、離脱、拘束次元

2.2 物体の可動範囲の表現法

動作自由度を接触状態から抽出するために、物体の位置姿勢の6つの自由度に対応する座標を持つ configuration-space (C-space) [4] で物体の可動範囲について考える。各物体はめり込むことができないので、取り得ない位置姿勢が存在し、それらは c-obstacle と呼ばれる多面体として表現される。例えば、図2の左に見られるような長方形と穴に対して、長方形の可動範囲を c-space 上で表現すると、図2の右ようになる。

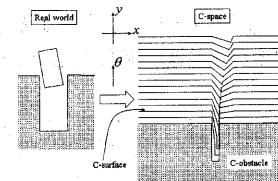


図2: 実空間と c-space との対応関係

C-space 上での組み立て動作計画は、把持物体が点として表現できるので把持物体の形に依存しない解析ができるという長所がある反面、c-obstacle を求めるためには時間がかかり、また、c-obstacle を表わす式が非線形であるので取り扱いが難しい。しかし、把持物体の動作計画を、微小動作の繰り返しによって行なうことにすれば、c-space 上で物体の微小可動範囲だけを知れば十分である。screw 理論 [5] を用いると物体の微小可動範囲を線形形式で近似表現することができる。

2.3 screw 理論

screw 理論 [5] では、物体の運動をある軸に沿った並進運動とそれと同じ軸を中心とする回転運動の組み合わせで表現する。図3に示すように、軸の向きを S_0 、軸の位置を P 、回転運動に対する並進運動の割合を p とすると、その変移は、 $[S_0, S_1]$ の6次元のベクトルとして表現される。ただし $S_1 = P \times S_0 + pS_0$ である。 $p=0$ の時は純粋な回転運動を表わし、 $p=\infty$ の時、つまり、 $[0, S_1]$ の時は、純粋な並進運動を表わす。

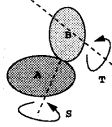
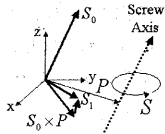


図 3: screw を用いた変移の表現法 図 4: 2 物体の接触

次に図 4 に見られる 2 物体の接触では、screw $S = (s_1, \dots, s_6)$ が接触点を中心とし、接触面の法線を軸とする回転運動を表現するとき、物体 B の可動範囲を表わす screw $T = (t_1, \dots, t_6)$ は式 (1) を満たす。

$$s_1 t_4 + s_2 t_5 + s_3 t_6 + s_4 t_1 + s_5 t_2 + s_6 t_3 \geq 0 \quad (1)$$

式 (1) の等号が成り立つときは 2 物体が接触を維持したままの物体 B の動きを、不等号が成り立つときは物体 A との接触を離す物体 B の動きを表わしている。

多面体の物体では接触を、点と面、面と点、辺と辺の接触の組み合わせとして表現することができ、screw 理論を用いて把持物体の可動範囲は式 (2) のような線形連立不等式で表わすことができる。

$$\begin{pmatrix} a_{11} & \dots & a_{16} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{n6} \end{pmatrix} \begin{pmatrix} t_1 \\ \vdots \\ t_6 \end{pmatrix} \geq \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \quad (2)$$

2.4 接触状態の分析

可動範囲を表わす連立線形不等式を解析する手段として、凸多面錘理論 [6] を紹介する。

$$\begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{p1} & \dots & a_{pn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \geq \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \quad (3)$$

凸多面錘理論によると、式 (3) で表現される連立線形不等式が与えられた時、その解領域 $A^* = \{X | A_1 X \geq 0, \dots, A_p X \geq 0\}$ の face とは、添字 1 から p の適当な部分集合 H が与えられた時、 $h \notin H$ の時は $A_h X = 0$ 、 $h \in H$ の時は、 $A_h X > 0$ に対応する解領域 A^* の部分集合 F_H のことを言い、定義から 2^p 通りの face が存在する。(ただし face の中には空集合の場合も存在する。) 空でない face F_H が与えられたとき $d_H = n - r_H$ をその face の次元と定義する。ただし、 r_H は線形独立な等式の最大数を表わす。この空でないすべての face が取り得る次元の範囲を知ることにより、3 つの自由度を知ることができる。

取り得る face の次元の範囲の簡単な導出法を示す。はじめに、式 (3) を式 (4) のように、任意の不定値を取り得る変数を用いて等式として表現する。

$$\begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{p1} & \dots & a_{pn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} u_1 \\ \vdots \\ u_p \end{pmatrix} \quad (\forall i, u_i \geq 0) \quad (4)$$

次に、以下に示すアルゴリズムを適用する:

1. 線形独立でない行の組を探す。
2. それらから x_1, \dots, x_n を消去することにより、式 (5) のような u_i のみの関係式を得る。
$$c_1 u_{i_1} + \dots + c_l u_{i_l} = 0 \quad (5)$$
3. もし $\forall k, c_k > 0$ か $u_{i_k} = 0$ が成り立つか、 $\forall k, c_k < 0$ か $u_{i_k} = 0$ が成り立つとき、 $u_{i_1} = \dots = u_{i_l} = 0$ と解くことができる。
4. 新たに $u_i = 0$ と解かれる u_i が現れなくなるまで 1 から 3 を繰り返す。

このアルゴリズムを適用させることにより、連立線形不等式 (2) は、それと同等な解を持つ連立線形方程式と不等式の最適な組み合わせの式 (6) に変形することができる。

$$BX = 0 \quad CX \geq 0 \quad (6)$$

すべての空でない face が取り得る次元の範囲は、 $m - r_{all}$ から $m - r_{eq}$ までである。ただし、 r_{all} は行列 A のランクを、 r_{eq} は行列 B のランクを表わす。

2.4.1 並進運動における 3 つの自由度の抽出

局所的可動領域が式 (2) で表現されているとき、並進可能範囲を表わす解領域は、並進運動を表わす screw 表現は $[0, S_1]$ という形式で表現されることから、式 (7) の解領域として表わされる。

$$\begin{pmatrix} a_{14} & a_{15} & a_{16} \\ \vdots & \vdots & \vdots \\ a_{n4} & a_{n5} & a_{n6} \end{pmatrix} \begin{pmatrix} t_4 \\ \vdots \\ t_6 \end{pmatrix} \geq \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \quad (7)$$

この解領域のうち維持自由度は $\{X | AX = 0\}$ で表わされ、拘束自由度は等式行列 B の行ベクトルの線形和で表わされ、離脱自由度はその残りということになる。この連立線形不等式の face の次元の範囲、及び、行列 A のランクより、表 1 から自由度数を求められる。

表 1: 並進運動自由度

rank	face の次元	維持	離脱	拘束
0	{3}	3	0	0
1	{2}	2	0	1
	{2,3}	2	1	0
2	{1}	1	0	2
	{1,2}	1	1	1
	{1,2,3}	1	2	0
3	{0}	0	0	3
	{0,1}	0	1	2
	{0,1,2}	0	2	1
	{0,1,2,3}	0	3	0

2.4.2 回転運動における 3 つの自由度の抽出

回転運動における運動自由度を抽出するために、回転軸に関する式、つまり、 t_1, t_2, t_3 の関係式を導出するため、まず、式 (4) を式 (8) の形に変形する。

$$A_r \begin{pmatrix} t_4 \\ t_5 \\ t_6 \end{pmatrix} = \begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix} + A_l \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix} \quad (\forall i, u_i \geq 0) \quad (8)$$

$$A_r = \begin{pmatrix} a_{14} & a_{15} & a_{16} \\ \vdots & \vdots & \vdots \\ a_{n4} & a_{n5} & a_{n6} \end{pmatrix}, A_l = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & a_{n3} \end{pmatrix}$$

次に、以下に示すアルゴリズムを適用する:

1. 行列 A_r を構成する行要素の中で線形独立でない組み合わせを見つける。
2. それらから t_4, t_5, t_6 を消去することにより、式 (9) のような関係式を得る。

$$d_1 t_1 + d_2 t_2 + d_3 t_3 = c_1 u_{i_1} + \dots + c_n u_{i_n} \quad (9)$$

3. もし $\forall k, (c_k > 0 \text{ or } u_{i_k} = 0)$ ならば、関係式 $d_1 t_1 + d_2 t_2 + d_3 t_3 \geq 0$ を得る。
4. もし $\forall k, (c_k < 0 \text{ or } u_{i_k} = 0)$ ならば、関係式 $d_1 t_1 + d_2 t_2 + d_3 t_3 \leq 0$ を得る。
5. 新たな関係式が得られなくなるまで 1 から 4 を繰り返す。

そうして得られた、 t_1, t_2, t_3 の関係式に対して、並進運動における自由度を求めるアルゴリズムを適用する。

この場合、表 1 における維持自由度は両方向回転可能な回転軸に、離脱自由度は片方向回転可能な回転軸に、拘束自由度は回転不可能な回転軸に対応する。

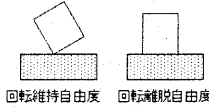


図 5: 両回転可能な自由度と回転維持、離脱自由度

片方向回転可能、及び、回転不可能な回転軸はそのまま回転における離脱、及び、拘束自由度を表わすが、両回転可能な回転軸はそのまま維持自由度を表わしてはいない。(図 5 参照) 回転に関する維持自由度は、全運動における維持自由度から、並進における維持自由度を取り除いたものであることから、まず全体に対する維持自由度を求め (式 (2) の行列 A のランクに等しい)、それから、並進維持自由度を取りのぞく。それぞれの自由度数は以下の式により求められる。

$$\text{回転維持} = \text{全維持} - \text{並進維持}$$

$$\text{回転離脱} = 3 - \text{回転維持} - \text{回転拘束}$$

$$\text{回転拘束} = \text{回転不可能な軸の自由度数}$$

2.5 特異な接触

把持物体の位置姿勢が C-space で C-obstacle の凸辺もしくは凸頂点上にある場合を考える。この場合、物体の動作可能範囲は凸ではないので、それを式 (2) の形で表現することはできない。この場合、式 (10) の形で表現することができる。但し、 A_i, B_{ij} は、 1×6 行列、 $T = {}^t(t_1 \dots t_6)$ である。

$$\left(\bigcap_i A_i T \geq 0 \right) \cap \left(\bigcup_{i,j} B_{ij} T \geq 0 \right) \quad (10)$$

このように、凸辺もしくは凸頂点上に存在するためには、特異な接触と呼んでいる 2 つの接触を含むことが必要十分である。このような場合、これらの特異な接触がないものとして解析することにする。つまり、式 (11) に対して、今までの解析を適用する。但しこの場合、それぞれの次元のことを、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ことにする。

$$\bigcap_i A_i T \geq 0 \quad (11)$$

2.6 6 つの自由度間の遷移解析

接触状態の変化は可動範囲の自由度の変化をもたらす。全部で 6 つの運動自由度があるので、それらの間

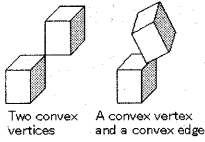


図 6: 特異な接触

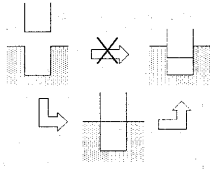


図 7: 直接遷移不可能

の遷移は ${}^6C_2 = 15$ 種類存在する。

今、特異な接触を維持したままの組み立て動作が存在しないものと仮定すると²、特異な3つの自由度間の3つの遷移が取り除かれる。

次に、図7に見られるように、維持、離脱自由度から、(特異)拘束自由度への遷移を考えた時、特異維持、特異離脱自由度という、入り口が必要となり、直接遷移は不可能である。つまり、4つの遷移が取り除かれ、図8で示される8つの遷移しか存在し得ない。

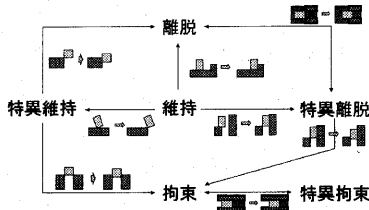


図 8: 可能な遷移

これらの8つの遷移は、大きく分けて2つの型に分けることができる。1つは、維持-離脱、維持-特異維持、維持-特異離脱の3つであり、これらは物体の運動方向と変化する自由度の方向が等しく、これをクラス1サブスキルと呼んでいる。クラス1サブスキルは、実際の組み立て動作に対応するものであり、これらの遷移を調べることで、並進、回転つき当て動作、及び、並進、回転すべらし動作の4つのクラス1サブスキルが定義される。(図9参照。)

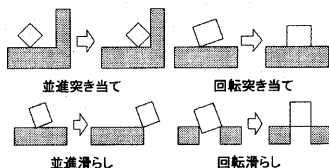


図 9: クラス1サブスキル

もう1つは、残りすべての遷移で、これらは物体の運動方向と変化する自由度の方向が直交しており、これをクラス2サブスキルと呼んでいる。クラス2サブ

²本論文の目的は、人間の組み立て動作を認識することであり、我々も組み立て動作の際そのような動作をしないことから、この仮定は妥当なものである。

スキルは、次の接触状態へ行くために満たすべき条件を表現している。例えば、特異維持-拘束では、変化する自由度方向の位置姿勢制御が必須であることが分かり、特異維持-離脱では、変化する自由度方向のある程度の位置姿勢誤差が許されることが分かる。(図10参照。)

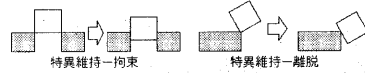


図 10: クラス2サブスキル

2.7 サブスキルの割り付け

サブスキルの定義からクラス1サブスキルの割り付けは以下になると考えられる。

- 並進 (回転) 維持自由度が並進 (回転) 離脱自由度に変化したとき、並進 (回転) つき当て。
- 並進 (回転) 特異維持自由度が並進 (回転) 離脱自由度に変化したとき、並進 (回転) すべらし。

1つ目のルールは正しいが、2つ目は正しくない。

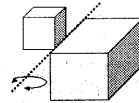


図 11: 真な特異維持自由度?

例えば、図11に見られる、平行な辺同士の接触を考えたとき、図で示された回転軸を中心とした回転運動では、接触を維持したまま回転を行うことができ、維持自由度の性質を有している。つまり、特異維持自由度から、維持自由度を分離した、真に特異維持自由度を考える。その自由度においては、接触を維持しながら動くことができないので、まず、接触しながら動くことのできない自由度を考え(当然、離脱、拘束自由度はこの中に含まれる)束縛自由度と定義する。

物体の可動範囲が、式(10)の形で表現されているとき、定義より束縛自由度は、行列 $R = [{}^t A_1 \dots {}^t B_{11} \dots]$ の行要素の線形和によってつくられる空間に等しく、束縛自由度数は行列 R のランク r に等しい。真の特異維持自由度数 m_s は式(12)を用いて計算される。但し、 c, d はそれぞれ、特異拘束、特異離脱自由度数である。

$$m_s = r - c - d \quad (12)$$

また、クラス1サブスキル割り付けの正しいルールは以下のようになる:

- 並進 (回転) 維持自由度が並進 (回転) 離脱自由度に変化したとき、並進 (回転) つき当て。
- 真の並進 (回転) 特異維持自由度が真の並進 (回転) 特異離脱自由度に変化したとき、並進 (回転) すべらし。

このことは、クラス2サブスキルの際にも成り立つ。

3 組み立て動作観察

本論文で用いるロボットには、3次元画像とそれに対応する2次元画像とを同時に得ることのできる、マルチベースラインリアルタイムステレオカメラシステムを搭載している。この章では、それによって得られた画像を処理することにより、前章で定義したサブスキルを用いて組み立て動作を理解するための提案手法について述べる。

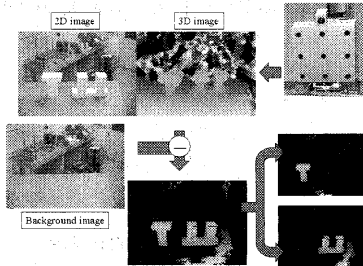


図12: 視覚処理システム

3.1 組み立て物体の3次元位置姿勢抽出

まず、ロボットから見た画像から組み立て作業物体のみを取り除く。組み立て作業を見ている間に、光源環境が大きく変化することは少ないので、あらかじめ2次元の背景画像を持たせておくことにより、背景差分法を用いることにより2次元画像上で組み立て物体のみを取り除く。そしてそれに対応する3次元画像上の点のみを取り出すことにより、3次元画像上での組み立て作業物体のみを取り除く。(図12参照。)

次に、組み立て作業物体のみの画像の時系列変化を見ることにより、背景画像は必ず同じところにいるという性質を用いて、ヒストグラムを用いて把持物体と環境物体を分離することができる。

最後に、上で得られた把持、環境物体分離画像に、3次元テンプレートマッチングシステム(3DTM)[3]を用いて、3次元上の把持物体の軌道、及び、環境物体の位置を求める。

3.2 接触状態を用いた視覚誤差修正

ビジョンにより得られたデータには誤差が乗っているため、そのままでは解析に用いることができない。そこで、比較的誤差に強い接触状態を用いて修正する方法が考えられる。しかし、ある接触状態を満たす物体の位置姿勢を求めるためには、非線形冗長連立方程式を解かねばならず難問である。本手法においては、視覚によりおおよそその解を得られているためかを示す方法を用いて解くことができる[2]:

1. 与えられた接触状態を実現するために満たすべき物体の位置姿勢の条件式を求める。
2. 得られた式をテイラー展開し線形近似する。
3. 最小二乗法を用いて近似解を求める。
4. 十分満足な解が得られるまで繰り返す。

3.2.1 接触状態の定式化

上で述べた通り、ロボットビジョンにより得られた物体の位置姿勢には誤差が含まれている。そのため、物体を微小に動かすことでその誤差を修正することを考える。今修正のために、物体を z, y, x 軸を中心に γ, β, α 回転し、 x, y, z 軸にそって x, y, z 並進したとすると、物体上の点の位置 v 、及び、法線の向き n の動いたあとの位置 v' 、向き n' は、式(13)によって計算される。但し $R = R_x(\alpha)R_y(\beta)R_z(\gamma), T = \begin{pmatrix} x & y & z \end{pmatrix}$ であり、 $R_x(\theta), R_y(\theta), R_z(\theta)$ はそれぞれ、 x, y, z 軸中心に θ 回転したときの回転行列を表わしている。

$$\begin{aligned} n' &= Rn \\ v' &= Rv + T \end{aligned} \quad (13)$$

単接触

点-面接触

図13に見られる点-面接触の場合、誤差修正の結果、修正後の点 v' は面 f 上に乗っている必要があるので、式(14)が得られる。

$$\Delta e_{vf} = n \cdot (Rv + T - f) = 0 \quad (14)$$

面-点接触

図13に見られる面-点接触の場合、誤差修正の結果、修正後の面 n' の上に点 v が乗っている必要があるので、式(15)が得られる。

$$\begin{aligned} \Delta e_{fv} &= Rn \cdot (v - Rf - T) \\ &= Rn \cdot v - n \cdot f - Rn \cdot T = 0 \end{aligned} \quad (15)$$

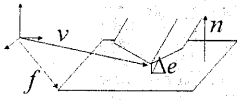


図 13: 点-面接触

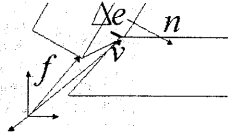


図 14: 面-点接触

辺-辺接触

図 15 に見られる辺-辺接触の場合、誤差修正の結果、修正後の辺 e_1' の上に辺 e_2 が接触している必要があるため、式 (16) が得られる。

$$\Delta e_{ee} = \frac{(Re_1 \times e_2) \cdot (p_2 - Rp_1 - T)}{|Re_1 \times e_2|} = 0 \quad (16)$$

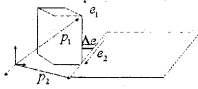


図 15: 辺-辺接触

多点接触

多点接触の場合、それぞれの接触を定式化したものを連立させた方程式が求めるべき方程式となる。

テイラー-展開

式 (14) (15) (16) にテイラー-展開を適用することにより、式 (17) (18) (19) が得られる。但し、 $q = {}^t(x, y, z, \alpha, \beta, \gamma)$ 、 $P = \frac{p_2 - p_1 + (n \cdot (p_2 - p_1))n}{|e_1 \times e_2|}$ である。

$$\Delta e_{vf} = n \cdot (v - f) - [{}^t n^t (n \times v)]q \quad (17)$$

$$\Delta e_{fv} = n \cdot (v - f) - [{}^t n^t (v \times n)]q \quad (18)$$

$$\Delta e_{ee} = n \cdot (p_2 - p_1) - [{}^t n^t \{p_1 \times n + (e_2 \times P) \times e_1\}]q \quad (19)$$

すべての式は式 (20) の形に表現することができる。

$$\Delta e = c - [{}^t a^t b]q \quad (20)$$

最小二乗法

視覚誤差修正のための式は導出できたので、理論的には $\Delta e = 0$ となる線形連立方程式を解けば良いことになるのであるが、テイラー-展開による近似によりその方程式が解けるという保証はない。連立方程式を解くかわりに、式 (21) を最小にする解を求める。

$$\sum (\Delta e)^2 \quad (21)$$

この式が最小となるためには、次の条件を満たす必要がある。

$$\frac{\partial}{\partial q} \sum (\Delta e)^2 = 0$$

$$\left[\sum_i^n \begin{pmatrix} a_i \\ b_i \end{pmatrix} \right] [{}^t a_i \quad {}^t b_i] q = \sum_i^n c_i \begin{bmatrix} a_i \\ b_i \end{bmatrix} \quad (22)$$

これを特異値分解を用いて解く。

3.3 接触状態間の直接遷移可能性

得られた接触状態から視覚の誤差を修正する方法について述べたが、誤差のある状態で得られた接触状態に誤差が含まれる可能性もありうる。本論文では視覚により得られた接触状態間が直接遷移可能かどうかを判定することにより、よりロバストな視覚誤差修正能力を獲得させる方法を提案する。

3.3.1 直接遷移可能性と c-obstacle 同士の隣接

図 16 の左に示された接触状態間の遷移について考える。矢印はそれらの遷移間が直接遷移可能であることを示す。図 16 の右はそれぞれの接触状態を満たす c-obstacle を示している。

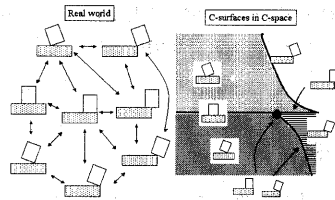


図 16: c-obstacle と直接遷移可能性との関係

ある 2 つの接触状態間が直接遷移可能であるための必要十分条件は、以下のようになる：

$\{a_i\} \in c\text{-obstacle } A, b \in c\text{-obstacle } B, \lim_{i \rightarrow \infty} \{a_i\} = b$ なる数列 $\{a_i\}$ が存在するか

$\{b_i\} \in c\text{-obstacle } B, a \in c\text{-obstacle } A, \lim_{i \rightarrow \infty} \{b_i\} = a$ なる数列 $\{b_i\}$ が存在する、

つまり、2 つの c-obstacle が隣接していることである。

3.3.2 接触状態間の直接遷移可能性判定アルゴリズム

対応する c-obstacle の隣接から 2 接触状態間の直接遷移可能性を判定することができるが、c-obstacle を求めることは難しい。そこで、実用上問題のない高速な直接遷移可能性判定アルゴリズムを提案する。そのために、まず c-obstacle の性質について述べる。

1. 接触状態が異なる c-obstacle 同士は重ならない。

2. ある接触要素を維持しながら動ける範囲は (ほかの要素は離脱してもよい)、閉集合である。

定理 1 2つの接触状態 A と B が与えられたとき、もし A を構成する接触要素がすべて B に含まれてなく、かつ、 B を構成する接触要素がすべて A に含まれていないならば、対応する c -obstacle C_a, C_b は隣接していない、直接遷移は不可能である。

証明 今 C_a と C_b が隣接していると仮定すると、 $\{a_i \in C_a, b \in C_b, \lim_{i \rightarrow \infty} \{a_i\} = b\}$ なる数列 $\{a_i\}$ が存在する。(逆に関しては、 C_a と C_b を入れ替えて考えれば良い。) A に含まれ B に含まれない接触要素を p とすると、接触要素 p は接触状態 A から B に移る際に離脱しなくてはならない。しかし、接触要素 p を維持している物体の位置姿勢を表わす c -obstacle はその性質上閉集合であり、それは、 $\{a_i \in C_a, b \in C_b, \lim_{i \rightarrow \infty} \{a_i\} = b\}$ が成り立たないことを意味し、仮定に矛盾する。

定理 2 2つ接触状態 A, B があり、 A の接触要素はすべて B に含まれているとする。もし、接触状態 A における維持、離脱自由度が接触状態 B において拘束自由度に変化しているならば、直接遷移は不可能である。

証明 図 8 を参照のこと

4 実験

本論文で提案したシステムにペグインサージョン動作を適用した。最初にリアルタイムステレオカメラシステムの前で動作をみせた。次に、接触状態による視覚誤差修正を行い全体として図 17 で示される接触状態遷移を得た。これに出現するすべての接触状態遷移は正しいものであると判定され、その後、組み立て動作認識を適用した。

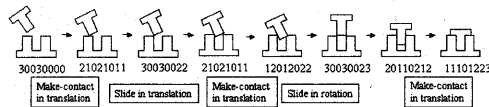


図 17: サブスキル適用

その結果を用いて、ロボットが同じ組み立て動作をする様子を、図 18 に示す。

5 考察

本論分では、ビジョンによるロボスタな組み立て動作認識手法を提案した。まず、組み立て動作理解に必

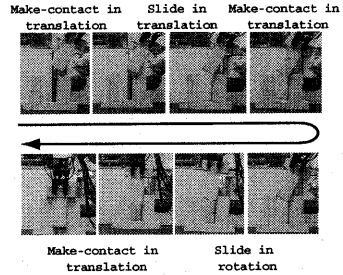


図 18: 実行

要な接触状態解析について述べ、組み立て動作に必要な不可欠な組み立て動作要素であるサブスキルを導出した。次に、ビジョンの誤差を接触状態および、その遷移を用いてロボスタに修正する方法について述べた。

参考文献

- [1] K. Ikeuchi and Takashi Suehiro, "Toward an Assembly Plan from Observation Part I: Task Recognition with Polyhedral Objects", IEEE Transactions on Robotics and Automation, Vol 10, No. 3, June, 1994.
- [2] Takashi Suehiro and Katsushi Ikeuchi, "Towards an Assembly Plan from Observation: Part II: Correction of Motion Parameters based on Fact Contact Constraints", IEEE International Conference on Intelligent Robots and Systems, pp. 2095 - 2102, July 7 - 10, 1992.
- [3] Mark. D. Wheeler, "Sensor Modeling, Probabilistic Hypothesis Generation, and Robust Localization for Object Recognition", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 28, pp 293 - 331, 1986.
- [4] Q. J. Ge and J. M. McCarthy, "An Algebraic Formulation of Configuration-Space Obstacles for Spatial Robots", IEEE Conference on Robotics and Automation, pp. 1542 - 1547, 1990.
- [5] M. S. Ohwovoriole and B. Roth, "An Extension of Screw Theory", Journal of Mechanical Design, Vol. 103, pp 725 - 735, October, 1981.
- [6] H. W. Kuhn and A. W. Tucker, "Linear Inequalities and Related Systems", Annals. of Mathematics Studies, Vol. 38, 1956