

機能仕様書

触覚認識モジュール

V e r . 2.1

2012 年 1 月 23 日

(株) 東芝

改版履歷

[illegible]

目次

1. はじめに	4
1. 1. 本書の適用範囲	4
1. 2. 本書を読むにあたって	4
2. 機能仕様	5
2. 1. 機能概要	5
2. 2. モジュール構成およびハードウェア構成	5
2. 3. 必要なハードウェア	6
3. RTC仕様	7
3. 1. FingerPositionComp	7
3. 1. 1. 機能概要	7
3. 1. 2. 動作環境	7
3. 1. 3. ポート情報	8
3. 1. 4. 入力データフォーマット	9
3. 1. 5. サービスポートI/F仕様	10
3. 1. 6. 設定ファイル (IDL)	10
3. 2. TactileRecognitionComp	11
3. 2. 1. 機能概要	11
3. 2. 2. 動作環境	11
3. 2. 3. ポート情報	12
3. 2. 4. 入出力データフォーマット	13
3. 2. 5. サービスポートI/F仕様	14
3. 2. 6. 設定ファイル (IDL)	14
特記事項	15

1. はじめに

1. 1. 本書の適用範囲

本書は、触覚認識モジュールの仕様について記述した文書である。触覚認識モジュールは、変位センサを装備したロボットハンドで物体（スプーン、缶など）の部分的な形状を測定し、予め登録した形状データと照合し、その物体の重心位置・姿勢を算出するものである。

1. 2. 本書を読むにあたって

本書は、RTミドルウェア、RTコンポーネント（以下、RTC）に関する基本知識を備えた利用者を対象としている。RTミドルウェア、RTCについては以下を参照のこと。

URL : <http://www.openrtm.org/>

2. 機能仕様

2. 1. 機能概要

本コンポーネントは、共通Sense I/F※を備えた触覚認識コンポーネントで、変位センサで物体（缶やスプーンなど）の部分形状を検出し、予め登録した形状と照合して、その物体の重心位置・姿勢を算出するものである。

※ 共通 Sense I/F は画像認識を行う複数の R T C インターフェースの共通化を図るという目的から作られたものである。

2. 2. モジュール構成およびハードウェア構成

本知能モジュールは、図 2-1 に示すモジュールとハードウェアの構成図において青色で示した触覚認識 R T C および指先位置 R T C である。変位センサ付きの指先をロボットハンド（後述）につけ、ロボットアームを動かすことにより、物体形状を測定し、指先位置を受け取ってから、予め登録した物体と照合し、物体の位置・姿勢を出力するものである。（図 2-1 は、ハードウェア構成の一例であり、他の構成もあり得る。）

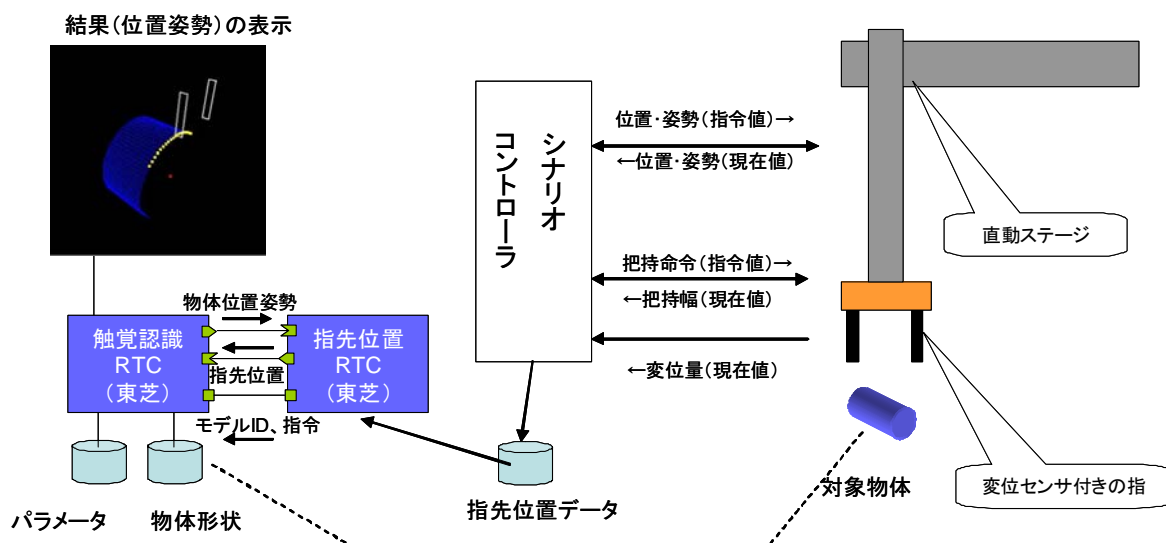
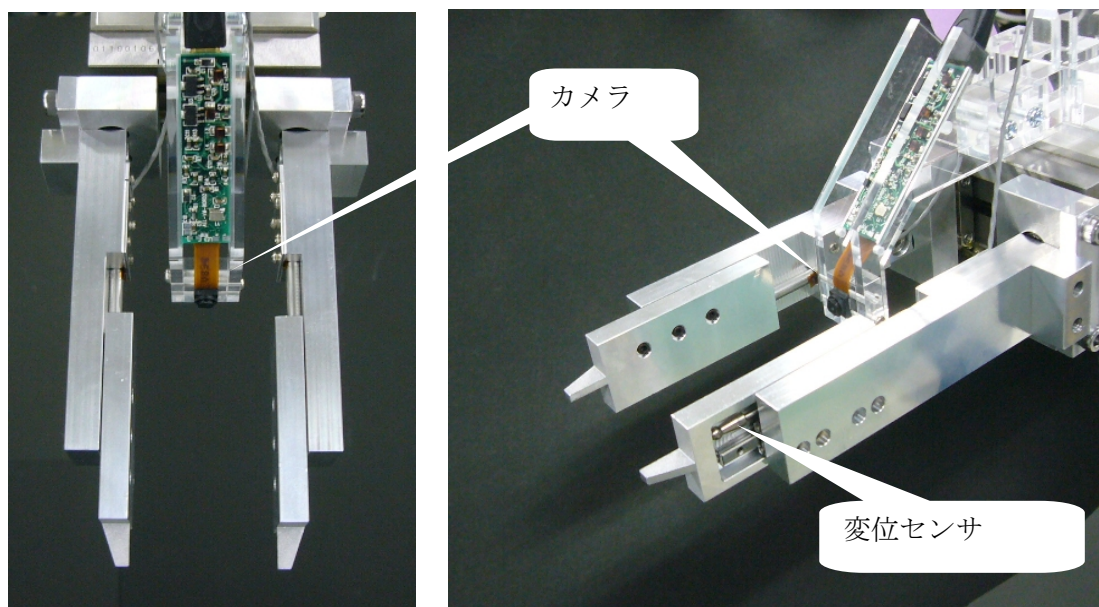


図 2-1 モジュールとハードウェアの構成図

2. 3. 必要なハードウェア

本R T Cを検証するためには、多関節ロボットと、ロボットハンド、および変位センサが必要である。多関節ロボットとしてはP A 1 0、ロボットハンドとしては、シュンクジャパン（株）のRH707 や、（株）T A I Y OのESG1 シリーズを想定している。変位センサは、L E V E X社のCLP35-008 などがある。ハンドの指先は、図 2-2 に示すように、先端が伸縮可能になっている必要があり、伸縮量を計測するために変位センサを用いる。

また、図示しないが本モジュールが搭載されたP Cには、A D変換ボードが挿入されている必要がある。これは、変位センサの値を受けけるためである。また、このP Cには、図 2-1 のシナリオコントローラも搭載されていることが好ましい。変位センサの値を読み、瞬時にアーム動作を変更させる必要があるためである。



(a)ハンドの写真（上から）

(b)ハンドの写真（斜め上から）

図 2-2 ロボットハンドの構造

3. RTC 仕様

3. 1. FingerPositionComp

3. 1. 1. 機能概要

このコンポーネントは、変位センサの値をコンポーネント内で読み取り、ハンドの指先端位置を計算して、TactileRecognitionComp に送り、TactileRecognitionComp から、物体の位置姿勢を受け取るものである。

3. 1. 2. 動作環境

本知能モジュールの動作環境（動作 OS、RT ミドルウェア、開発環境など）について記述する。

動作 OS	Windows (Xp,sp2 で動作確認済み)
開発言語	C++
コンパイラ	VisualC++2008
RT ミドルウェア／バージョン	OpenRTM-aist 1.0.0
依存パッケージ	OpenCV2.2 http://sourceforge.net/projects/opencvlibrary/ Freeglut http://sourceforge.jp/projects/sfnet_freeglut/releases/

3. 1. 3. ポート情報

A) データポート (OutPort)

名称	型	データ長	説明
finger_position	TimedDoubleSeq	可変	指先位置を複数送る

B) データポート (InPort)

名称	型	データ長	説明
object_position	TimedDoubleSeq	20	マッチングしたモデルの重心位置姿勢

C) サービスポート (Provider)

サービス名	インターフェース名	説明
モデル番号指定	TactileService	マッチングさせるモデルを指定

3. 1. 4. 入力データフォーマット

データ位置	格納値
0	カメラ I D
1	物体 I D
2	候補番号
3	座標系番号
4	識別確度
5	エラー番号
6	予約(1)
7	予約(2)
8	物体の姿勢を表す回転行列の 1 行 1 列目
9	物体の姿勢を表す回転行列の 1 行 2 列目
10	物体の姿勢を表す回転行列の 1 行 3 列目
11	物体の位置ベクトルの X 成分
12	物体の姿勢を表す回転行列の 2 行 1 列目
13	物体の姿勢を表す回転行列の 2 行 2 列目
14	物体の姿勢を表す回転行列の 2 行 3 列目
15	物体の位置ベクトルの Y 成分
16	物体の姿勢を表す回転行列の 3 行 1 列目
17	物体の姿勢を表す回転行列の 3 行 2 列目
18	物体の姿勢を表す回転行列の 3 行 3 列目
19	物体の位置ベクトルの Z 成分

センサ ID: 0～

物体 ID: 0～m-1, -1: all

認識候補 No: 0～

座標系 No: 0:センサ座標系 1:ロボット座標系 2:世界座標系

認識確度: 0～1

エラーNo: 0～

3. 1. 5. サービスポート I/F 仕様

関数名	SetModelID(in long ModelID)			
引数	名称	型	I/O	説明
	ModelID	long	out	モデル番号を指定
戻り値	値			説明
	無し			
説明	マッチングさせるモデル番号を指定。			

関数名	long GetModelID()			
引数	名称	型	I/O	説明
	なし	なし	なし	なし
戻り値	値			説明
	ModelID			モデル番号を獲得
説明	マッチングさせるモデル番号を獲得			

3. 1. 6. 設定ファイル (I D L)

```

サービスポートの interface
interface TactileService{
    long getModelID();
    void setModelID(in long ModelID);
};

```

3. 2. TactileRecognitionComp

3. 2. 1. 機能概要

このコンポーネントは、FingerPositionCopm から、ロボットの複数の指先位置を受け取り、予め登録された物体形状と比較し、物体の位置姿勢を計算して、FingerPositionCopm に送り返すものである。

3. 2. 2. 動作環境

本知能モジュールの動作環境（動作 OS、RT ミドルウェア、開発環境など）について記述する。

動作 OS	Windows (Xp,sp2 で動作確認済み)
開発言語	C++
コンパイラ	VisualC++2008
RT ミドルウェア／バージョン	OpenRTM-aist 1.0.0
依存パッケージ	OpenCV2.2 http://sourceforge.net/projects/opencvlibrary/ Freeglut http://sourceforge.jp/projects/sfnet_freeglut/releases/

3. 2. 3. ポート情報

A) データポート (InPort)

名称	型	データ長	説明
finger_position	TimedDoubleSeq	可変	指先位置を複数送る

B) データポート (OutPort)

名称	型	データ長	説明
Object_position	TimedDoubleSeq	20	マッチングしたモデルの重心位置姿勢

C) サービスポート (Provider)

サービス名	インターフェース名	説明
モデル番号指定	TactileService	マッチングさせるモデルを指定

3. 2. 4. 入出力データフォーマット

データ位置	格納値
0	カメラ I D
1	物体 I D
2	候補番号
3	座標系番号
4	識別確度
5	エラー番号
6	予約(1)
7	予約(2)
8	物体の姿勢を表す回転行列の 1 行 1 列目
9	物体の姿勢を表す回転行列の 1 行 2 列目
10	物体の姿勢を表す回転行列の 1 行 3 列目
11	物体の位置ベクトルの X 成分
12	物体の姿勢を表す回転行列の 2 行 1 列目
13	物体の姿勢を表す回転行列の 2 行 2 列目
14	物体の姿勢を表す回転行列の 2 行 3 列目
15	物体の位置ベクトルの Y 成分
16	物体の姿勢を表す回転行列の 3 行 1 列目
17	物体の姿勢を表す回転行列の 3 行 2 列目
18	物体の姿勢を表す回転行列の 3 行 3 列目
19	物体の位置ベクトルの Z 成分
20	カメラ I D

3. 2. 5. サービスポート I/F 仕様

3.1.5 節と同様であるので省略する。

3. 2. 6. 設定ファイル (I D L)

サービスポートの interface

```
interface TactileService{  
    long getModelID();  
    void setModelID(in long ModelID);  
};
```

特記事項

本モジュールをご利用される場合は、以下に示す記載事項・条件にご同意いただけたものとします。

本モジュールのライセンスは **Eclipse Public License(EPL)**に従います。利用条件の詳細については、下記サイトを参照ください、なお、本モジュールは利用条件に同意した場合にのみ利用可能となっており、本モジュールを利用した時点でライセンス条項に同意したものとみなします。

Eclipse Public License <http://www.eclipse.org/legal/epl-v10.html>