

OpenVGR

はじめにお読みください

Ver. 0.9.1

2012 年 3 月 5 日

独立行政法人 産業技術総合研究所

本文書の取り扱いについて

- 本書に掲載する情報は、使用者に有用なものであるように万全を期していますが、内容の正確性、最新性、その他一切の事項について保証をするものではありません。
- 本書の著者および著作権者は、使用者がこの文書から得たまたは得られなかった情報から生じる損害に対しても、一切責任を負いません。
- 本書は使用者への事前の予告なしに変更、削除、公開の中止を行うことがあります。

【連絡先】

(独) 産業技術総合研究所 知能システム研究部門 タスクビジョン研究グループ
〒305-8568 茨城県つくば市梅園 1-1-1 中央第二

E-Mail: openvgr-contact@m.aist.go.jp

目次

1. はじめに	1
1.1. 概要	1
1.1.1. プログラムの概要、構成について	1
1.1.2. RT ミドルウェアの概要	3
1.2. 使い方の例	4
2. 動作環境	5
3. システム構成図	6
4. 用途	7
4.1. 作業対象認識モジュール	7
4.1.1. 物体の位置・姿勢を同定する	7
4.1.2. 円筒の物体認識	10
4.1.3. 直方体の物体認識	11
4.1.4. 認識結果の最大数の設定	12
4.2. 認識結果表示モジュール	13
4.2.1. 発見した対象物にモデルを重ねて画像を表示する	13
4.2.2. 対象物が見つからなかった場合の表示	15
4.3. モデル作成ツール	17
4.3.1. 対象物をもとにモデルを作成する	17
4.3.2. 用途に応じてモデルの原点と軸方向を変える	19
4.4. 3次元距離計測モジュール	20
4.5. ステレオ画像取得モジュール	22
4.6. 画像表示モジュール	23
4.7. カメラキャリブレーションツール	24

1. はじめに

1.1. 概要

この文書はロボットビジョンを主な用途とするソフトウェアについて、その概略を説明しています。このソフトウェアは複数のプログラムからなり、全体の名前を作業対象認識モジュール群といいます。

1.1.1. プログラムの概要、構成について

作業対象認識モジュール群は、ロボットシステムがカメラを使って周囲の環境から対象物を認識する機能をモジュール化しています。これは五つの RT コンポーネントと二つのツールから構成されており、ユーザはそれらを組み合わせて利用できます。

このソフトウェアは、複数台のカメラによって撮影したデジタル画像データから、対象とする物体の位置・姿勢を算出することができます。三角測量の原理を応用してシーンの距離を計測し、3次元のモデルと照合します。

なお、このソフトウェアは NEDO「次世代ロボット知能化技術開発プロジェクト」のにおいて開発されたものです。また機能の主な部分を RT ミドルウェア上で実現しています。

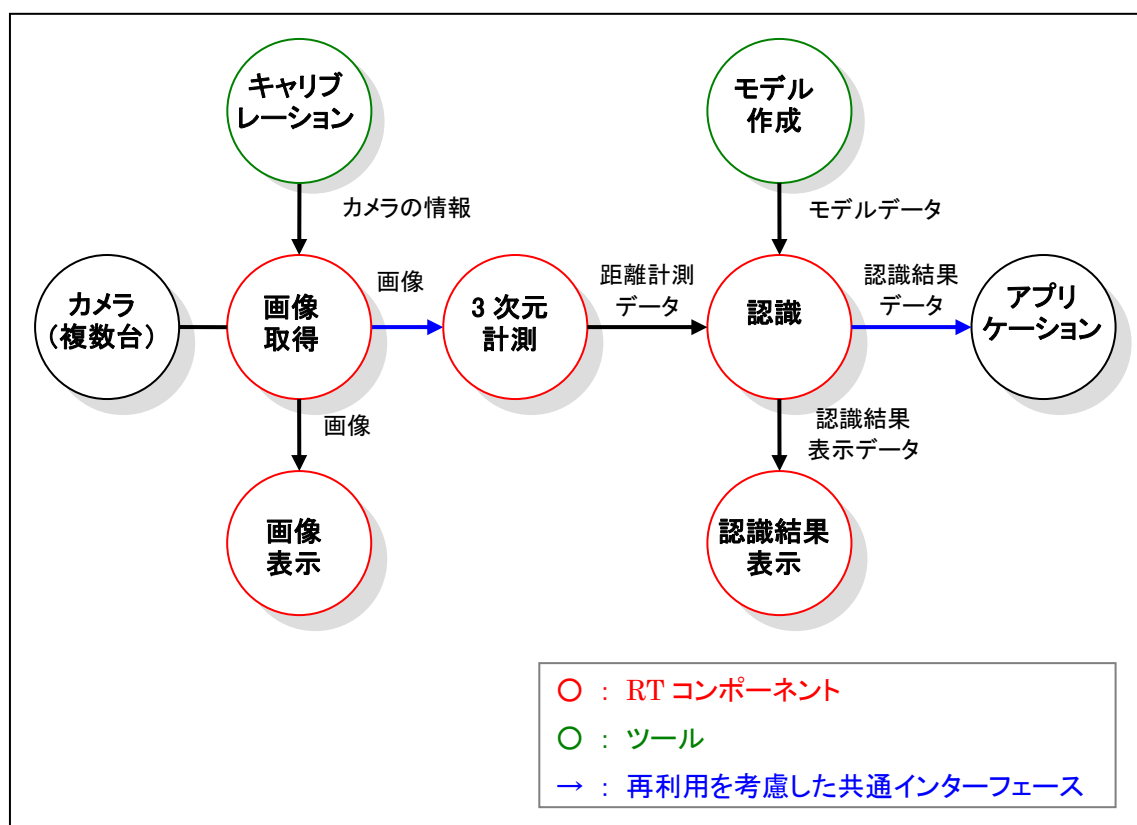


図 1 プログラムの概要

作業対象認識モジュール群は下記のモジュールから成り立ちます。

名称	機能
作業対象認識モジュール	モデルと 3 次元データを照合して位置・姿勢を計算する。
認識結果表示モジュール	認識した結果を画像に重ねたものを表示する。
モデル作成ツール	認識する対象物のモデルを作成する。
3 次元距離計測モジュール	シーンの 3 次元データを計測する。
ステレオ画像取得モジュール	ステレオ画像を取得する。
画像表示モジュール	画像を表示する。
カメラキャリブレーションツール	計測に必要なカメラのキャリブレーションを行う。

なお、この文書は作業対象認識モジュール群の目的や用途を説明するものです。実際の操作方法については、操作手順書を参照してください。

1.1.2. RTミドルウェアの概要

RT ミドルウェア (RT-Middleware: RTM) とは、ロボットシステム (RT システム) を構築する為のソフトウェアプラットフォーム一般を指す名称です。RT ミドルウェアでは、ロボットを構成するさまざまな要素のうち、まとめた機能要素 (RT 機能要素) をソフトウェアモジュール化します。複数のソフトウェアモジュールを組み合わせることで、ロボットシステムを構築することができます。

詳しくは OpenRTM-aist 公式サイトの RT ミドルウェアの説明をご参照ください。

- ・ OpenRTM-aist公式サイト <http://www.openrtm.org/openrtm/ja>
- ・ ホーム » ドキュメント » OpenRTM-aist とは? » RT ミドルウェア

1.2. 使い方の例

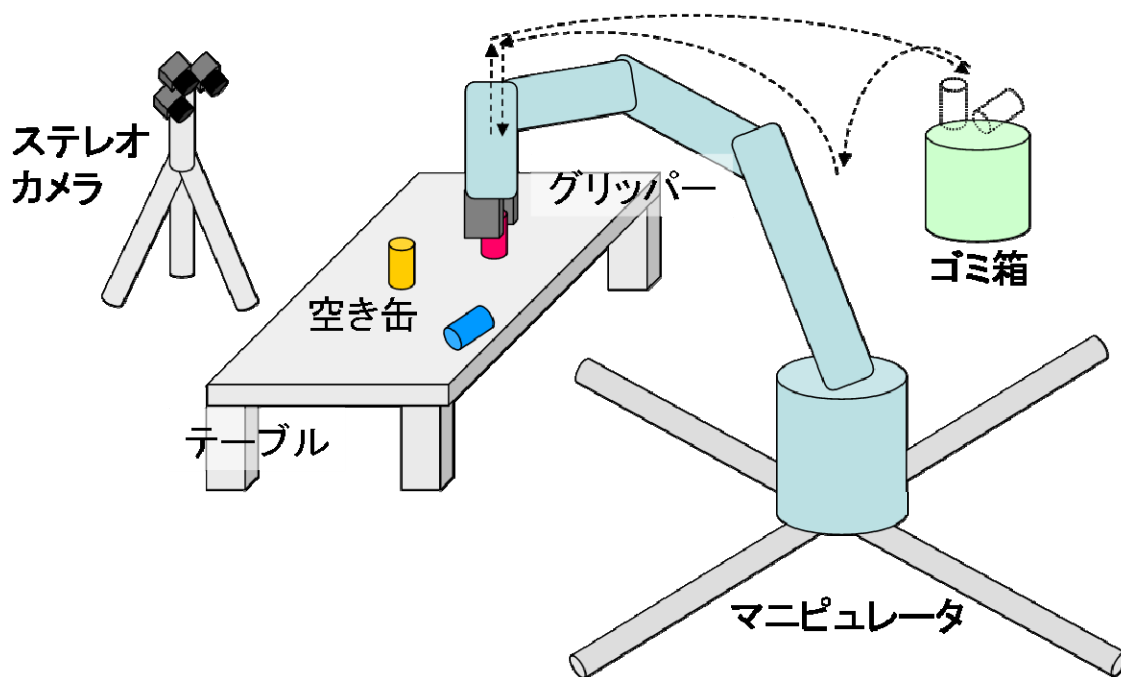


図 2 使用例

例えば、人間に代わってテーブルの上をロボットが片づけるシステムを考えます(図 2)。ここでは、ロボットアームがテーブルの上の空き缶をつかんでゴミ箱に捨てるとしましょう。ゴミ箱の位置はあらかじめロボットに教えておく事もできますが、空き缶の位置・姿勢は事前には分かりません。この場合、ロボットには自律的に缶を認識する能力が必要です。

このように、実空間でロボットに作業を行わせるためには環境についての情報を取得できるセンサが欠かせません。作業対象となる空き缶の位置や姿勢がロボットに対してどのようなものであるかが分からない限り、具体的な作業をする事が極めて難しくなります。

実空間の情報を効率的に得る手段の一つとしてカメラを使ったビジョン(人工視覚)処理があります。作業対象認識モジュール群を利用すると、カメラの画像から作業対象となる特定の物体の3次元位置・姿勢が簡単に得られるようになります。

2. 動作環境

本プログラムは以下の環境で動作します。

使用言語 : C++, C 言語

動作環境 :

PC : CPU Core2 Duo 以上, メモリ 1GB 以上, ハードディスク 10GB 以上

OS : Ubuntu 10.04 LTS Desktop 日本語 Remix (x86 32bit 環境)

カメラを使用する場合 : IEEE 1394b カメラ (IIDC 1.31 準拠)

Point Grey 社製 Flea2 カメラで動作確認

依存パッケージ :

OpenRTM-aist 1.0.0-RELEASE C++版

OpenRTM Eclipse tools 1.0-RELEASE

GNU Compiler Collection 4.4.3

OpenCV 2.0

libdc1394 2

freeglut3-2.6.0

カメラを使用する場合 : Coriander 2.0.0

3. システム構成図

作業対象認識モジュール群のシステム構成図を下記に示します。

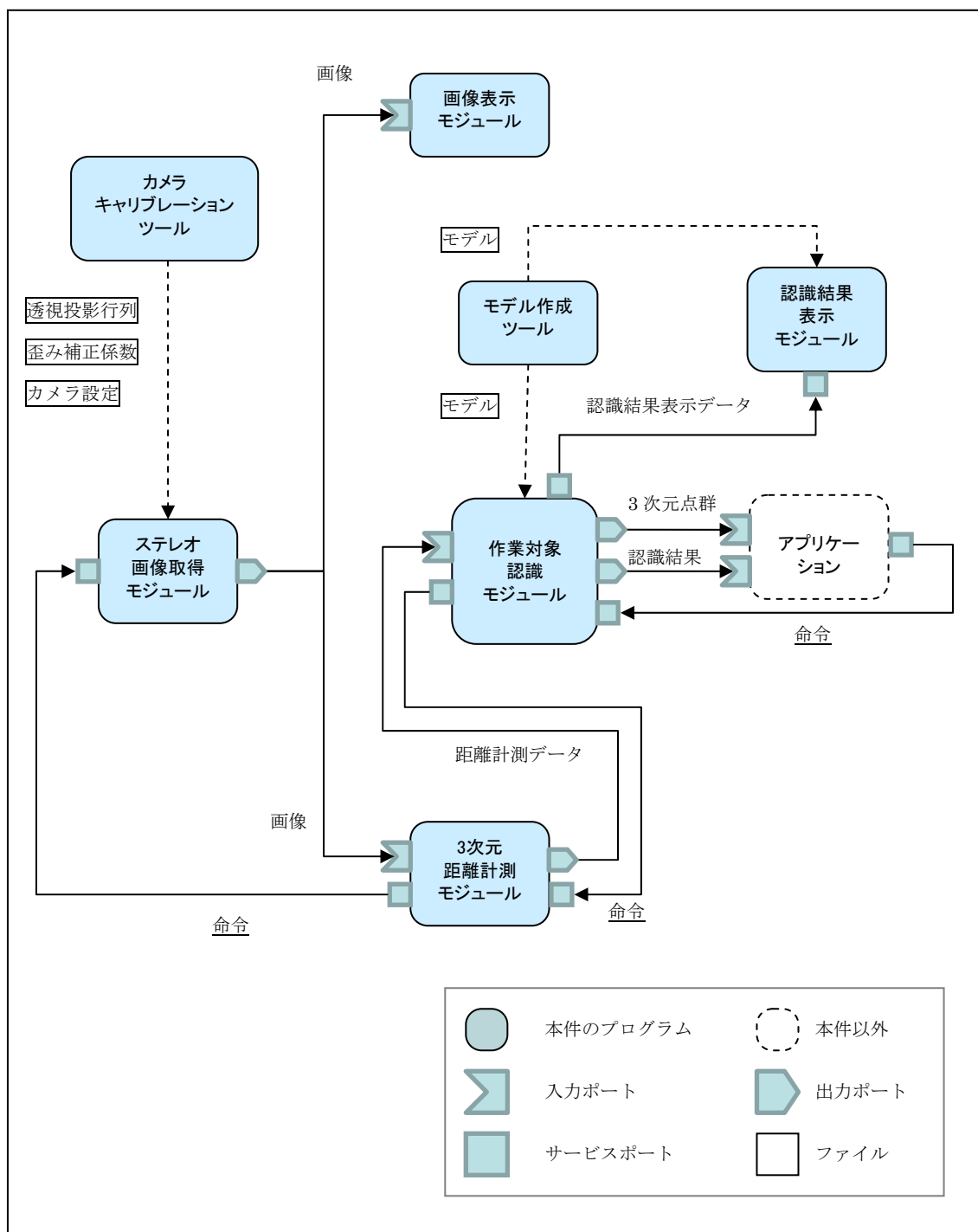


図 3 システム構成図

4. 用途

4.1. 作業対象認識モジュール

4.1.1. 物体の位置・姿勢を同定する

作業対象認識モジュールは、画像から対象物を検出してその 3 次元位置・姿勢を求めるプログラムです。以降、この処理を認識と呼びます。

ユーザはこのモジュールにモデルを与える必要があります。モデルは認識する対象物の形状をあらわす 3 次元データです。このモジュールは入力データとモデルを照合して対象物を立体的に認識し、認識結果（対象物の位置・姿勢）を出力します。

認識結果は、対象物の位置・姿勢をあらわします。これはモデルの座標系を、3 次元空間中の対象物の位置・姿勢へと移動・回転する変換行列です。認識結果によって対象物の位置・姿勢をロボットに伝えることができます。

このモジュールへの入力データは、作業対象認識モジュール群に含まれる次のモジュールの出力から得ることができます。

- ・ 距離計測データ：3 次元距離計測モジュール
- ・ モデル：モデル作成ツール

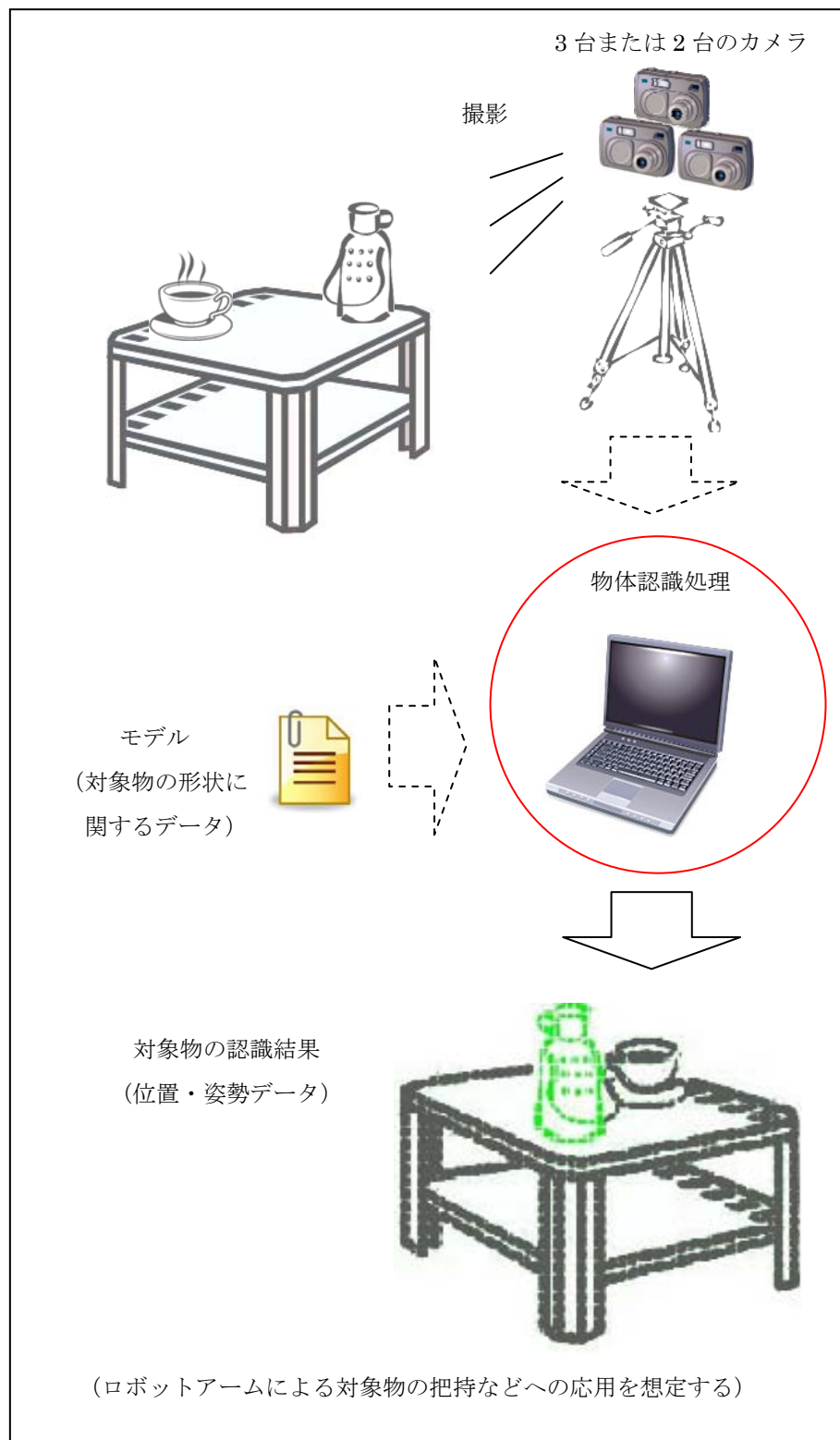


図 4 物体の位置・姿勢を同定する

作業対象認識モジュールは、作業対象認識モジュール群の中心的な役割を果たす RT コンポーネントです。現在の作業対象認識モジュールが認識できる対象物の種類は、円筒と直方体の 2 種類の形状に限られます。

またこのモジュールから後述の認識結果表示モジュールを呼び出し、認識結果を画像へ上書き表示して確認できます。



図 5 認識結果の表示

作業対象認識モジュールは、次のような処理をします。

1. 対象物を認識するよう命令を受ける。
2. 距離計測データ（画像データと 3 次元点群）を送信するよう、他のモジュールに命令を送る。
3. ID 番号によって指定されたモデルをファイルから読み込む。
4. 撮影されたシーンの距離計測データを受信する。
5. シーンについてモデルを認識し、その位置・姿勢を求める。
6. 認識結果と距離計測データをデータポートから出力する。

ユーザは作業対象認識モジュールにこれらの命令とデータが受け渡されるよう、適切な RT コンポーネントを接続する必要があります。

4.1.2. 円筒の物体認識

作業対象認識モジュールが認識できる形状の一つは円筒形です。例えば、自販機で売られているようなジュースの缶がほぼ円筒形です。ユーザは対象物と同じ寸法の円筒形のモデルを作り、このモジュールに入力します。

認識結果には、対象物の位置・姿勢をあらわす認識候補が含まれます。撮影されたシーンの中から見つけた数だけ、認識候補が出力されます。対象物が見つからなかった場合、認識候補は出力されません。複数の認識候補がある場合は、モデルによく合う候補ほど先頭に近くなるように並べられます。なお円筒形は逆さにしても同じ形状です。そのため、認識候補の数と実際に存在する対象物の数が必ずしも一致しない点に注意してください。

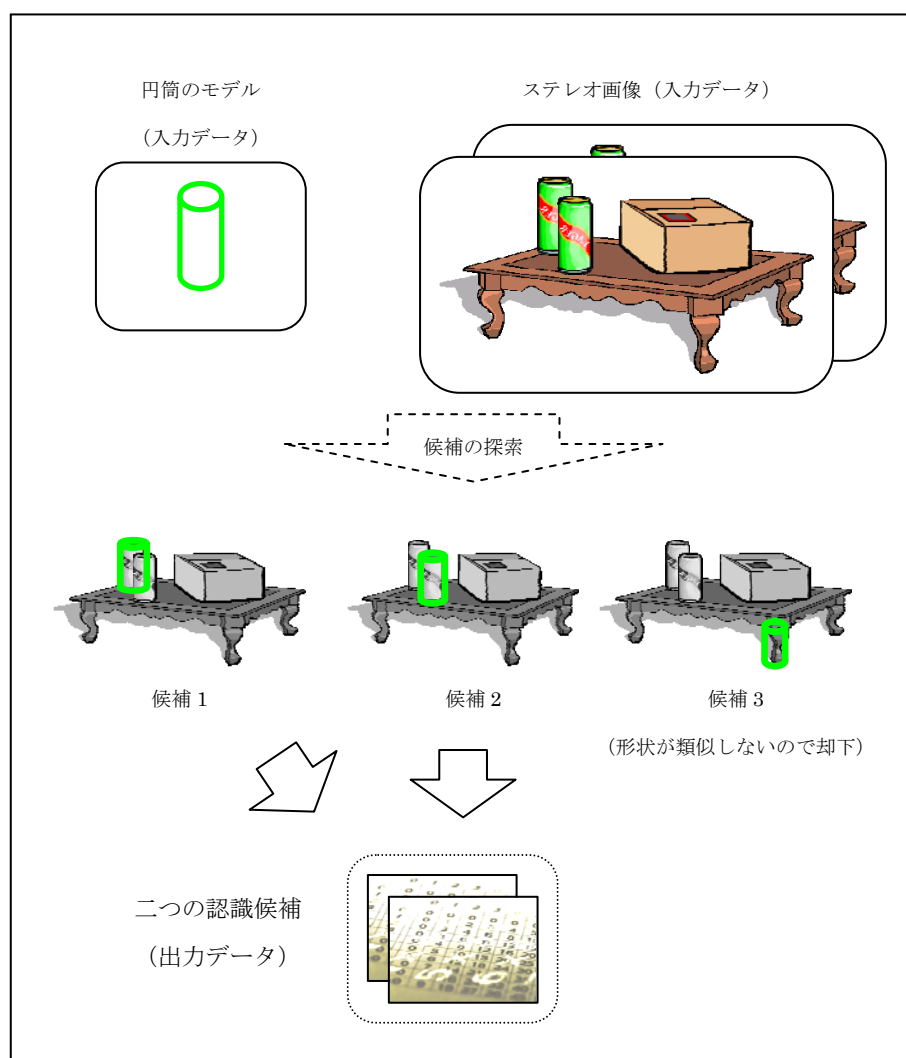


図 6 円筒の物体認識

4.1.3. 直方体の物体認識

作業対象認識モジュールが認識できる形状の一つは直方体です。例えば、段ボール箱がほぼ直方体です。ユーザは対象物と同じ寸法の直方体のモデルを作り、このモジュールに入力します。出力される認識結果の意味や注意点は円筒の場合と同じです。

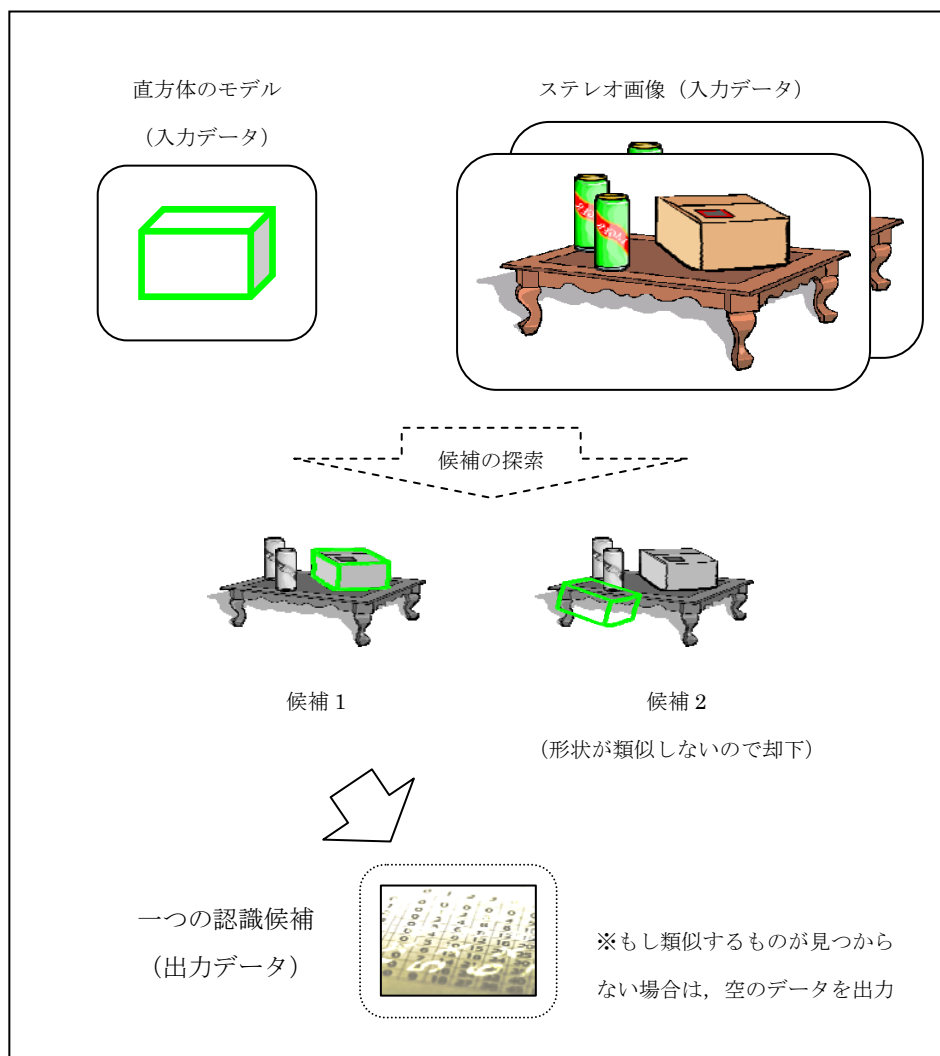


図 7 直方体の物体認識

4.1.4. 認識結果の最大数の設定

作業対象認識モジュールには、出力候補の数に上限を設けることができます。認識候補の数に最大数を設定すると、それを超える数の認識候補は出力しません。認識結果には設定した個数以下の認識候補が含まれます。ただし、これによって認識候補同士が空間中で重複することを禁止することはできません。位置・姿勢の違いがごくわずかでも、別の認識候補であれば出力されます。

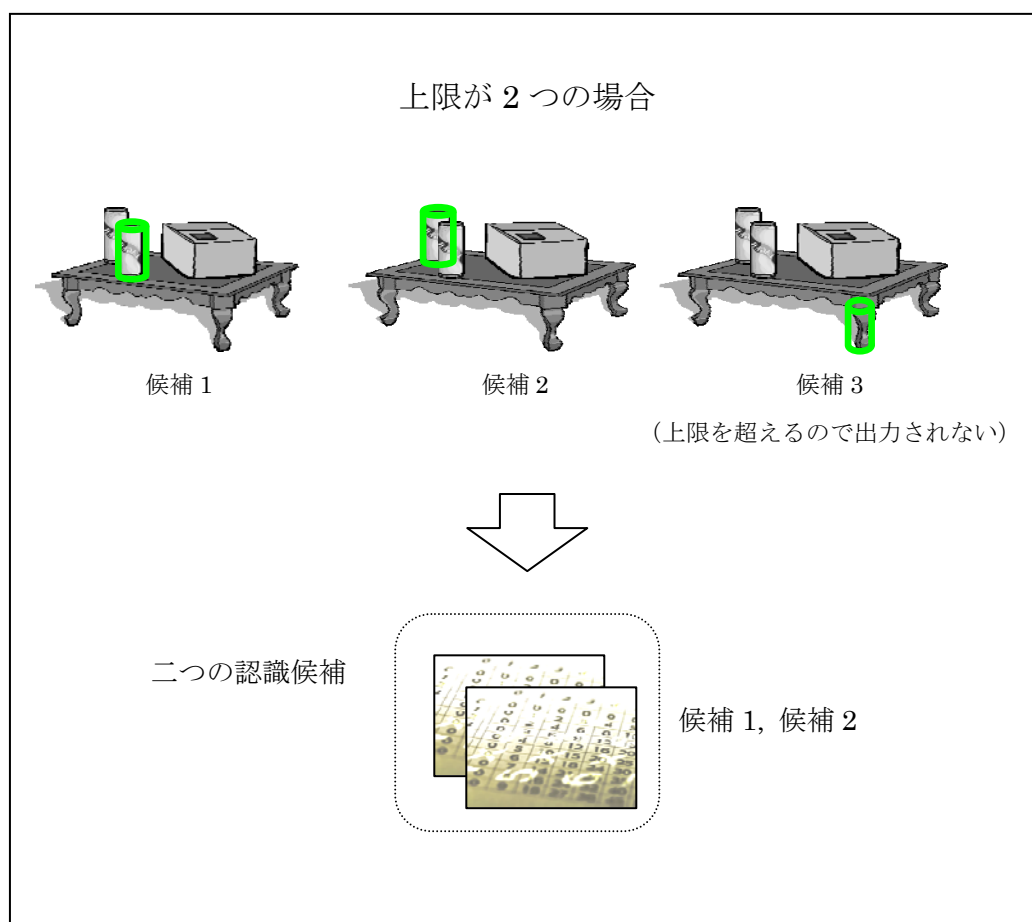


図 8 認識結果の最大数の設定

4.2. 認識結果表示モジュール

4.2.1. 発見した対象物にモデルを重ねて画像を表示する

認識結果の適切さを人が判断するには、その位置・姿勢のモデルを画像に投影することが有効な手段です。認識結果表示モジュールは、認識された中で最上位の候補を、画像に表示する機能を持っています。このモジュールは、画面表示によって作業対象認識モジュールを補佐する RT コンポーネントです。

入力された認識結果の最上位の候補によって座標変換されたモデルが、ステレオ画像に投影されます。このモデルは、緑色のワイヤーフレームで描画されます。

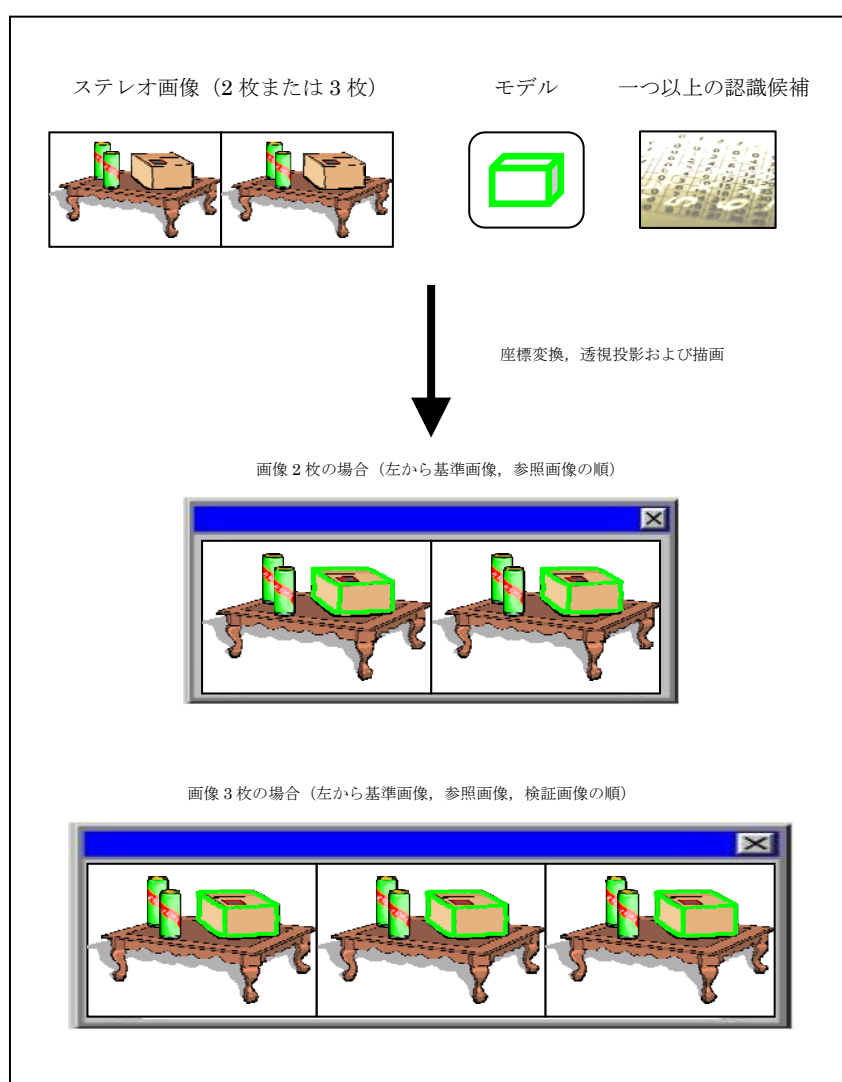


図 9 認識結果のオーバーレイ表示

認識結果表示モジュールは、ステレオ画像をウィンドウ内に左から右へ並べて表示し、さらにモデルを上書きで描画します。モデルは認識候補によって座標変換されます。画面には左から順に左カメラ、右カメラ、中央カメラの画像を表示します。物体認識に成功していれば、対象物の輪郭をなぞるようにモデルの線が緑色で描画されます。

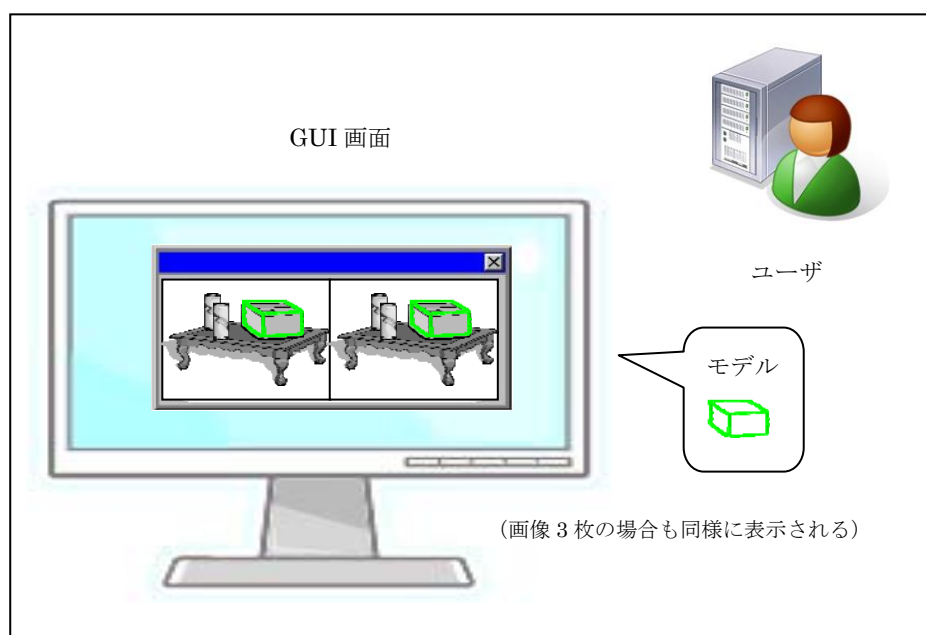


図 10 モデルを重ねて画像を表示

4.2.2. 対象物が見つからなかった場合の表示

画像上で対象物を認識することができない場合、作業対象認識モジュールにより出力される認識候補の数は 0 になります。この場合、認識結果表示コンポーネントはモデルを画像に投影しません。

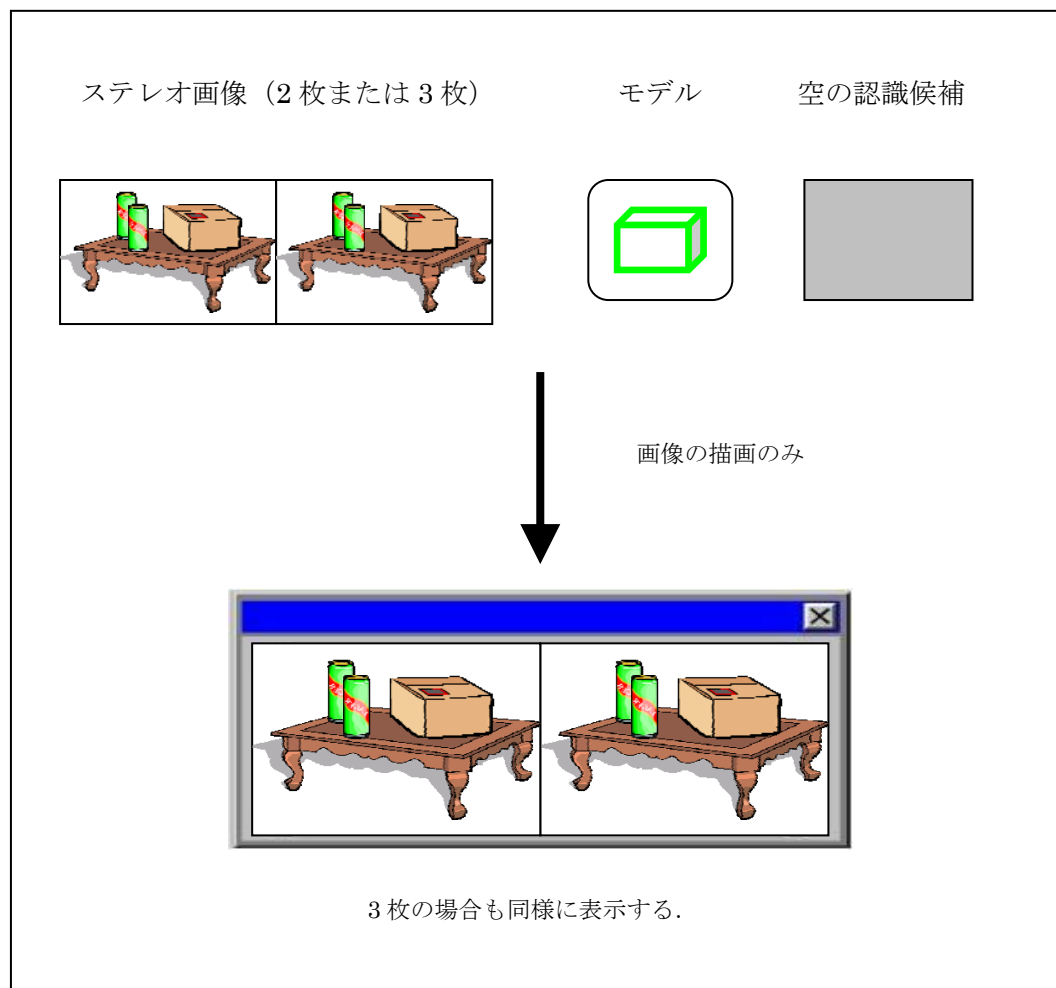


図 11 認識結果が無い場合の画像

対象物が見つからなかったとき、画像だけが表示されます。認識結果を示す緑色のワイヤースケルトンは画面にあらわれません。すなわち認識結果表示モジュールは新しい画像と認識結果が入力されると、認識結果が空でなければ画像とワイヤースケルトンの両方を表示し、空ならば画像の表示だけを行います。

ユーザのしている画面にステレオ画像が表示され、モデルを示す緑色の線画が描画されなかった場合は、認識結果に候補が 1 つも含まれなかったことを意味します。

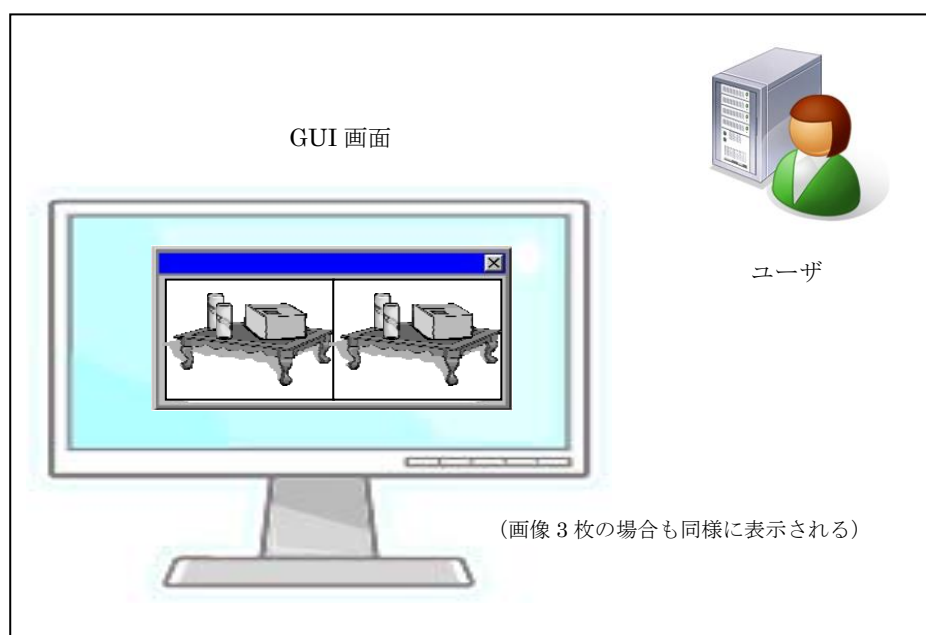


図 12 対象物が見つからなかった場合の表示

4.3. モデル作成ツール

4.3.1. 対象物をもとにモデルを作成する

モデル作成ツールは、作業対象認識モジュールで利用するモデルを作成するためのコマンドラインプログラムです。このモデルは、対象物に関する形状を記述したデータです。なお、現在のバージョンのモデル作成ツールは、円柱と直方体のみをサポートしています。

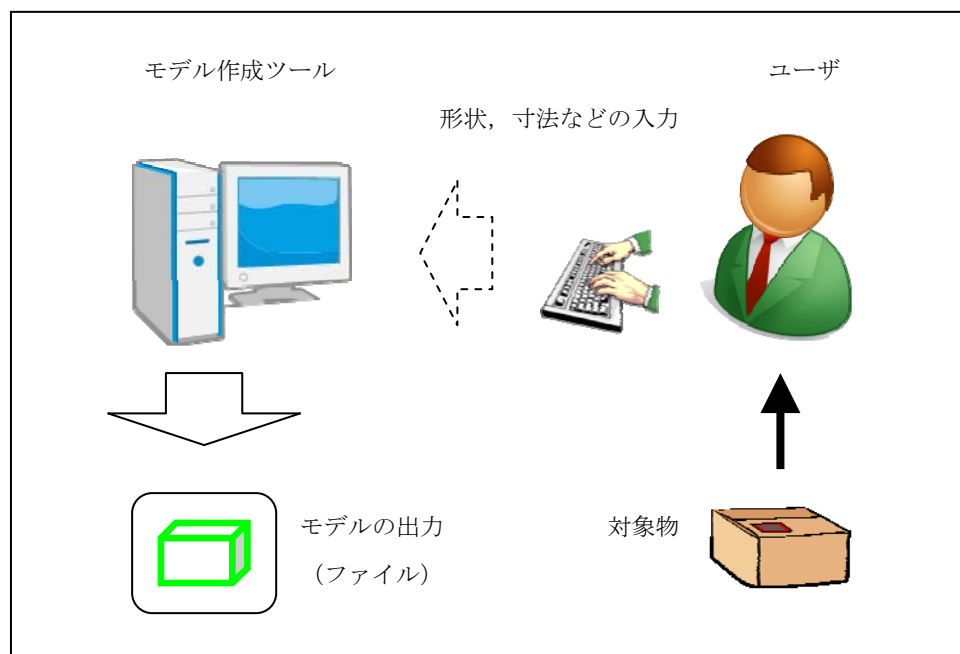


図 13 対象物をもとにモデルを作成する

モデル作成ツールを利用するには準備が必要です。ユーザは対象物を記述したテキストファイルを作成しなければなりません。このテキストファイルをモデル生成用設定ファイルと呼びます。モデル生成用設定ファイルには、形状の種類や寸法を所定の書式で記載します。モデル一つにつき、一つのモデル生成用設定ファイルが必要です。

モデル生成用設定ファイルを用意したら、それをモデル作成ツールに入力します。モデル作成ツールは、ユーザが記述したモデル生成用設定ファイルを元に、作業対象認識モジュールに入力可能なモデルを生成します。モデルにはVRML形式の表示データが付属しています。モデルのファイルはそのまま一般のVRMLビューアに入力することが可能です。これによってモデルを立体表示し、ユーザがその形状を画面で確認することができます。図 14、図 15はWhiteDuneによるモデルの表示例です。

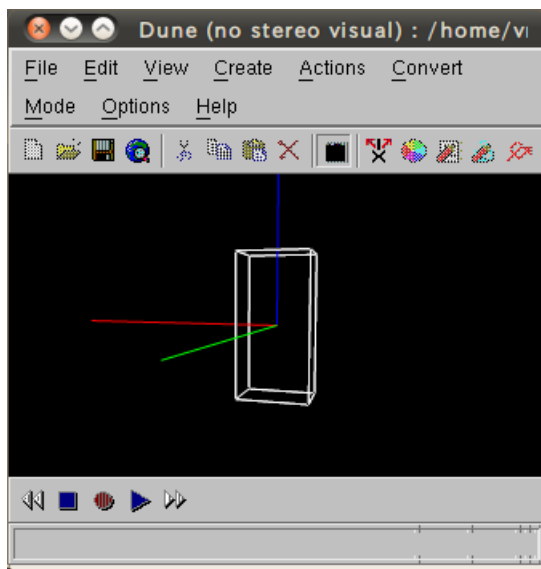


図 14 直方体の例

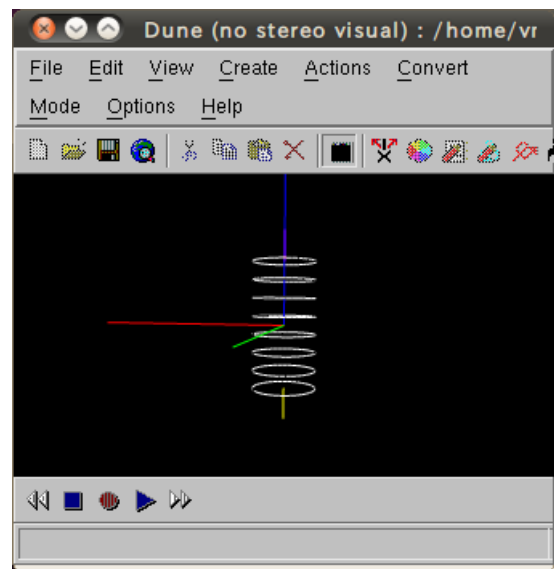


図 15 円柱の例

4.3.2. 用途に応じてモデルの原点と軸方向を変える

モデル作成ツールは、モデルの生成処理に回転や移動の操作を加えることができます。

モデルとなる形状の位置や姿勢は、モデリングするための座標系（モデル座標系）に基づいて定義されます。例えば直方体のモデルは、中心が原点に位置するように、幅・奥行き・高さ方向がそれぞれ X, Y, Z 軸方向と平行になるようにデータ化されます。このモデルを様々な用途に応用できるように、モデル作成ツールにはモデル座標系における形状の位置や姿勢を変更する機能があります。

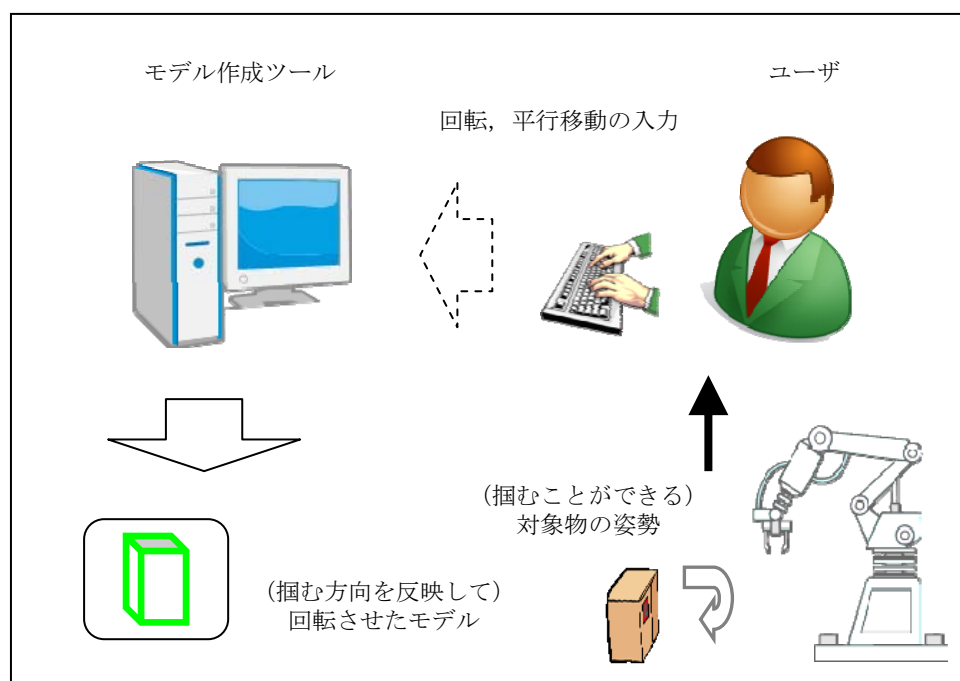


図 16 用途に応じてモデルの原点と軸方向を変える

4.4. 3次元距離計測モジュール

3次元距離計測モジュールは、物体の表面を3次元の点の集まりとして計測するRTコンポーネントです。別途、撮影されたステレオ画像から、画像処理によって各点の3次元座標を算出します。こうして求められた3次元座標のあつまりは、距離計測データに含まれる3次元点群（カメラによって撮影した物体、環境についての3次元データ）として出力されます。

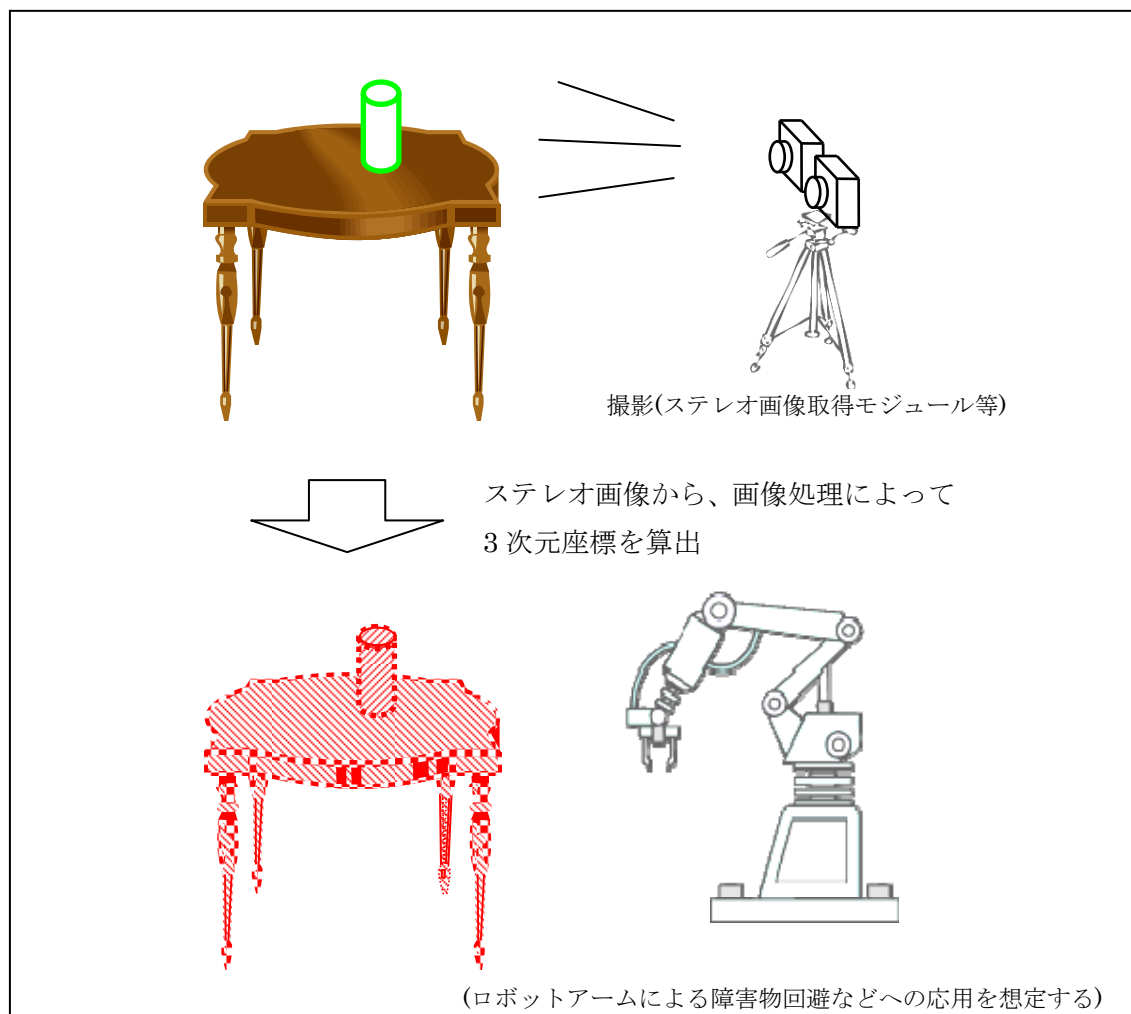


図 17 3次元距離計測モジュール

この距離計測データの利用例の一つは、ロボットの可動範囲に障害物がないか判定することです。前出の作業対象認識モジュールは、対象物しか認識しません（図 18）。対象物以外の情報は認識を命令したアプリケーションに報告されませんので、対象物をつかむ際に障害になるものが周囲にあっても検出できません。アプリケーション（ここではロボットの制御プログラム）がこのような障害物を回避しながら対象物をつかむような把持戦略をとる場合、このモジュールの3次元点群情報を活用できます（図 19）。本モジュールは

未知の物体であっても 3 次元点群として計測できますので、机のような障害物がロボットの可動範囲にあることを知る手掛かりになります。また、障害物をロボットが回避しながら対象物へとアプローチする助けになります。逆に、もし対象物の位置・姿勢以外の情報が得られなければ、ロボットが対象物にアプローチするとき、障害物に衝突する恐れがあります。

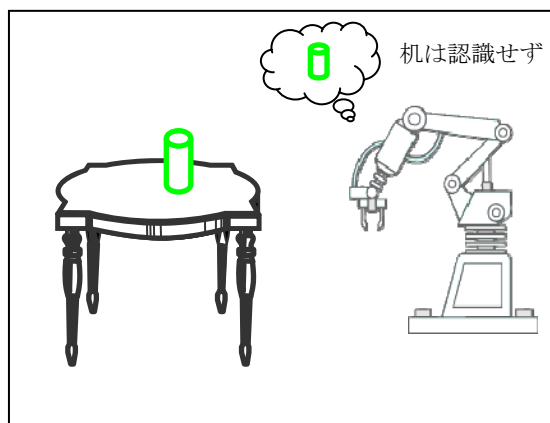


図 18 モデルによる物体認識

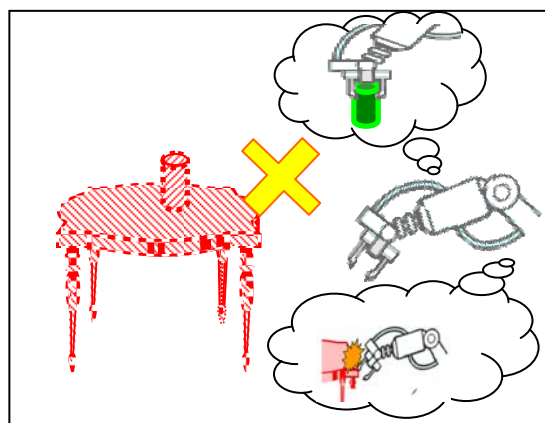


図 19 可動範囲内の障害物の判定

なお、3 次元距離計測モジュールに入力するステレオ画像は、ステレオ画像取得モジュールから得ることができます。また作業対象認識モジュールは、3 次元距離計測モジュールに処理開始の命令を与えて、距離計測データの出力を受け取ることができます。

4.5. ステレオ画像取得モジュール

ステレオ画像取得モジュールは、IHD 1.31 に準拠した複数のIEEE 1394bカメラを使ってステレオ画像を撮影するRTコンポーネントです。2 台または 3 台のカメラを制御して、2 枚または 3 枚の画像からなるステレオ画像を撮影できます。撮影の指示を受け取るとそれぞれのカメラで 1 回撮影し、左カメラ画像、右カメラ画像、中央カメラ画像（カメラ 3 台の場合）を出力します（図 20、図 21）。これらは 3 次元距離計測モジュールへの入力に用いることができます。

ステレオ画像取得モジュールは、相互運用を目的としたインターフェースに準じています。

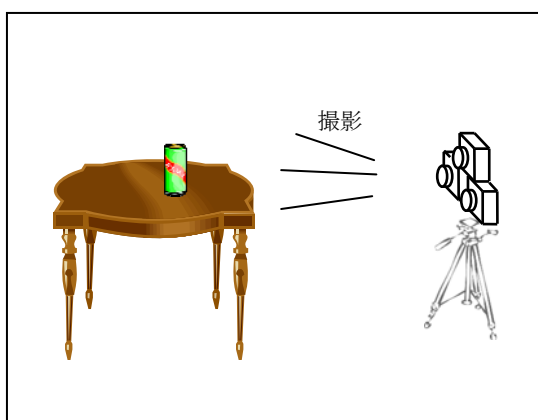


図 20 ステレオ画像の撮影（3 眼）

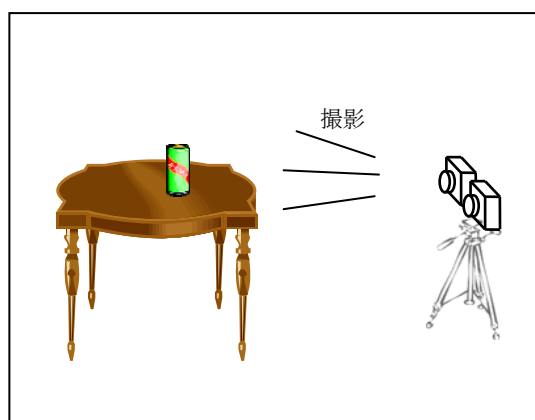


図 21 ステレオ画像の撮影（2 眼）

4.6. 画像表示モジュール

画像表示モジュールは画像を表示するプログラムです。ステレオ画像取得モジュールが出力するステレオ画像を画面に表示します。

3 眼での撮影時はモニタに向かって左手側から左カメラ、右カメラ、中央カメラの順に画像が並びます (図 22)。なお、2 眼での撮影時はモニタに向かって左手側から左カメラ、右カメラの順に画像が並びます (図 23)。



図 22 ステレオ画像の表示 (3 眼)

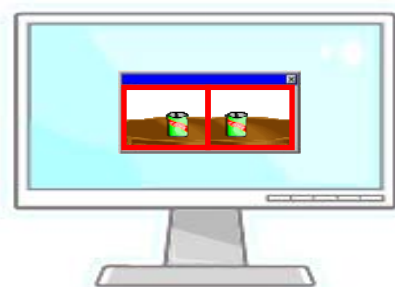


図 23 ステレオ画像の表示 (2 眼)

4.7. カメラキャリブレーションツール

カメラキャリブレーションツールは計測に必要なカメラのキャリブレーションデータを作成するプログラムです。複数のカメラ画像からチェスボードパターンの検出を行い、そこで得た格子点座標情報をもとに、各カメラの内部パラメータと外部パラメータを算出します。

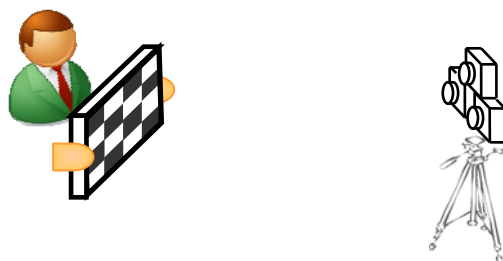


図 24 チェスパターンボードの撮影

出力される各カメラの内部・外部パラメータおよび歪み補正係数は、ステレオ画像取得モジュールに必要なキャリブレーションデータとして使われます。もしキャリブレーションデータを作成しなかった場合、ステレオ画像モジュールの動作は不完全なものになります。

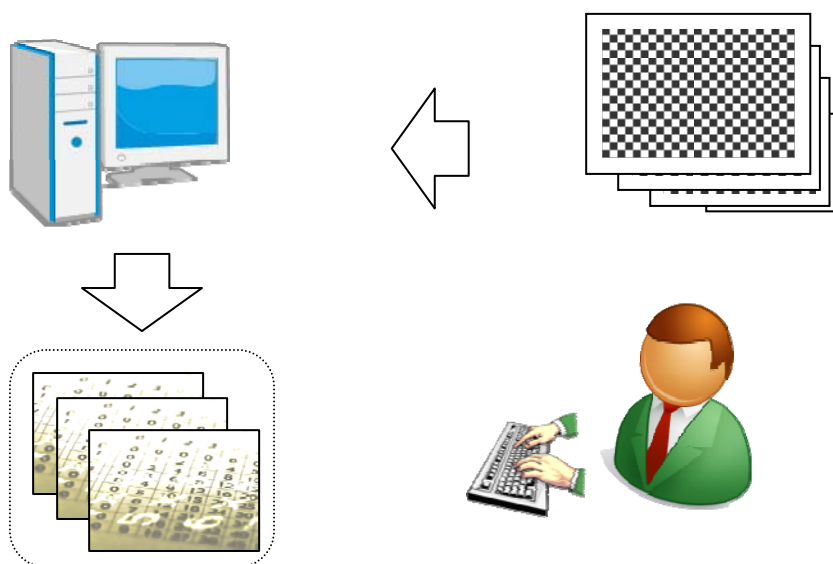


図 25 カメラのキャリブレーション

OpenVGR はじめにお読みください

Ver.0.9.1

2012 年 3 月 5 日

独立行政法人 産業技術総合研究所

知能システム研究部門 タスクビジョン研究グループ



Copyright © 2011-2012 AIST All Rights Reserved.