

人つながり支援のための人物位置の実時間マッピングに基づく ロボットナビゲーション

上間圭祐 平井裕麻 松尾直志 島田伸敬 (立命館大学)

1. はじめに

画像認識や音声対話機能など多様なロボティクス技術の向上によって、家族間や社会において人どうしの協働や対話を円滑にするためのロボットが期待されている。顔認識、発話理解、移動機能などが個別に研究開発されてきたが、ROS(Robot Operating System)[1]やOpenRTM[2]に代表されるロボットミドルウェアの登場によって統一されたプラットフォーム上で個々の機能モジュールとして有機的に統合されるようになってきた。

本研究では、誰と誰がいつどこで会い対話をしたかといった人のどうしの関わり合いをロボットの視点で観察し、時に応じてスムーズな人間関係を支援することを目的としたパートナーロボットの開発を目指している。コミュニティ内での正しい情報の共有を促進するために、コミュニティ内で会話をしつつ情報を収集するロボットを開発する研究がこれまでも行われているが[3]、著者らは誤った情報を持った人物を発見した場合に、その人が十分信頼してくれるだろう根拠を添えて説得を試みる対話戦略[4]について研究してきた。そのような対話支援機能や人の行動観察機能は、ネットワーク・インフラ上に構築したロボットミドルウェア上のモジュールとして構成することで複数のロボットプラットフォームから再利用できるようになる。本稿では、ROS上のネットワークサービスとして顔検出・識別機能、人物の存在場所・行動記録機能を実装し、自律移動可能なロボットがオフィス内で自由に移動しながら時々刻々と変わる人物の配置をマッピングして、音声等による名前の指示でその人のいるところまでお使いに出かけるロボット制御プラットフォームを構築したのでその概要について報告する。

2. システムの概要

本システムは大きく分けて、以下の三つのサブシステムで構成される。

- 自律移動パートナーロボットシステム
- 人物配置マッピングシステム
- 音声による目的地指定システム

自律移動パートナーロボットシステムは、車輪走行ロボットにRGB-D センサ、レンジセンサ ならびにPCを搭載し、オフィス内において顔検出・顔識別をしながら目的地まで自律移動する。人物配置マッピングシステムは、顔を識別した人物の最新の観測位置をオフィス内地図の上に記録し、他のモジュールからの問い合わせに対して指定人物が最後に観測された位置の座標を返答する。音声による目的地指定システムは、ロボットへの移動指示のための音声認識インターフェースである。

以上三つのサブシステムは、以下のソフトウェアとハードウェアを使用して実装した。

• ソフトウェア

ROS(Robot Operating System) Willow Garage 社が開発したロボットアプリケーション作成を支援するライブラリとツールを提供するミドルウェア。複数のモジュールが分散協調動作する。

顔認識用サーバモジュール 本研究においてオリジナルに実装した画像をネットワーク経由で入力すると顔認識結果と個人属性データを応答するサーバ。

MMDAgent 名古屋工業大学国際音声技術研究所で開発された音声インタラクションシステム構築ツールキット [5]。

Julius 音声認識システム開発・研究のための高性能な汎用大語彙連続音声認識エンジン [6]。

• ハードウェア

Pioneer 3-DX MobileRobots 社製車輪走行型移動ロボット。

URG-04LX-UG01 北陽電機 (株) 社製スキャナ式レンジセンサ。

Xtion PRO LIVE ASUS 社製 RGB-Depth センサ。

Logicool Wireless Headset H600 Logicool 社製ノイズキャンセリングヘッドセット。

ノート PC ROS を起動させる PC (MacBook Pro を使用)。

タブレット PC MMDAgent を使用し音声認識を行う端末 (Microsoft Surface Pro2 を使用)。

自律移動パートナーロボットシステムは、複数の ROS のモジュールを動作させる PC に URG-04LX-UG01 と Xtion PRO LIVE および Pioneer3-DX を接続して構成される。自律移動を行うために、自律移動モジュールが Pioneer3-DX からオドメトリ情報を、URG-04LX-UG01 から距離データを ROS の機能を利用して Subscribe し移動を制御する。また画像キャプチャモジュールや顔認識サーバモジュールを含む複数の ROS モジュールがネットワーク通信により協調することで地図上の人物配置情報を時々刻々更新する。タブレット PC 上の MMDAgent はヘッドセットを通して発話された音声を認識し、その認識結果を ROS 上で動作するコマンド解釈モジュールにその結果を出力する。コマンド解釈モジュールは自律移動モジュールに対して解釈した目的地の座標を Publish することで目的地への移動を実現する。

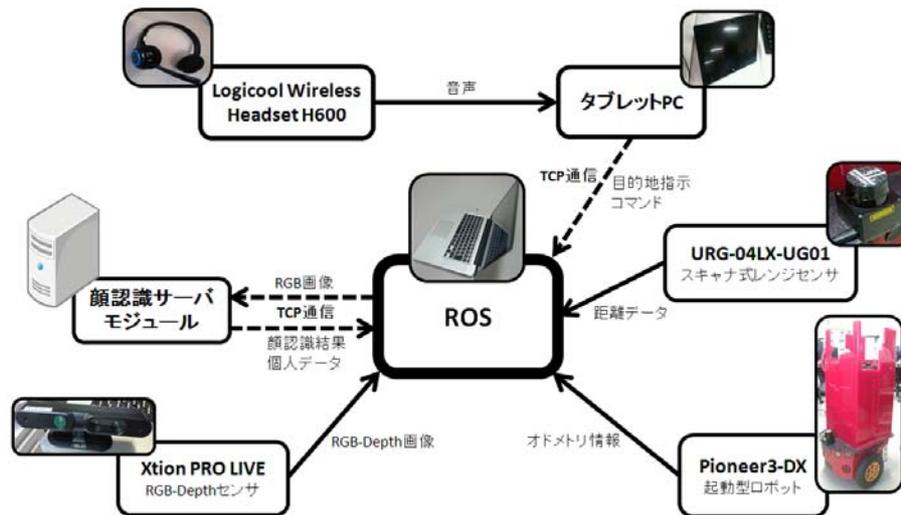


図 1 システム全体図

3. 自律移動パートナーロボットシステム

自律移動パートナーロボットシステムでは、自律移動と顔認識を行う。

自律移動モジュールは、ROS の hydro ディストリビューションの標準パッケージに含まれている ROS モジュール `amcl` と `move_base`[7] を用いて構成される。`amcl` と `move_base` を利用するためには移動ロボットとレンジセンサの他に、2D 地図が必要である。ダルムシュタット工科大学にて開発された `hector_mapping` モジュール [8] を使い、レンジセンサの距離データから 2D 地図を作成する。`amcl` は移動ロボットのオドメトリとレンジセンサの距離データから、所与の地図上における自己位置を推定する。`move_base` は自己位置推定結果を利用して環境内における自律移動の制御を行い、目的地までの経路生成と障害物を検知した場合の回避行動とを行う。目的地は 2D 地図上の (x, y) 座標で表現される。

顔認識機能については、RGB 画像を受信して顔認識結果と個人データを返信する顔認識サーバモジュールをネットワークサービスとして構築した。移動ロボットに搭載された RGB-D センサから毎時刻の画像を取得し、顔認識サーバに送信することで人物の検知と認識を行う。図 2 は顔認識の結果から、顔の部分を四角図形で囲み、認識した人物の識別 ID (以下、人物 ID) と名前を表示した図である。

OpenNI2 と NiTE2 を組み合わせて作成した ROS モジュール `OpenNiTE2` は、RGB-D センサからカラー画像と Depth 画像を取得する。顔認識サーバモジュールと TCP 通信を行う ROS モジュール `okao_client` は、得られた RGB 画像を顔認識サーバに送信して処理結果を受け取る。

顔認識サーバモジュールは、以下三つのサーバから構成される。

顔認識サーバ オムロン社が開発した OKAO Vision を用いた顔認識ライブラリを用いたサーバである。ネットワーク経由で画像を受け取ると、顔検出お



図 2 顔認識結果画像

よび人物識別を行い、顔の位置、視線の角度、人物 ID をネットワーク経由で返信する。

個人属性データベースサーバ 個人属性データを MySQL に保持したサーバである。人物 ID を受け取ると、人物 ID をキーにしてその人物の個人属性データ (人物の名前、所属、学年) を検索しネットワーク経由で返信する。

仲介サーバ クライアントと顔認識サーバ・個人情報データベースサーバの仲介をするサーバである。まず、`okao_client` から画像を受け取ったあと顔認識サーバにその画像を送り、顔検出結果を受け取る。次に、顔検出結果に含まれる人物 ID を個人情報データベースサーバに送り、個人属性データを取得する。最後に、取得した顔情報と個人属性データを組んで `okao_client` に返す。

4. 人物配置マッピングシステム

人物配置マッピングシステムとは、認識した人物の名前と位置座標をオフィス内の地図上に記録するシステムである (図 3)。独自に実装した以下の三つの ROS モジュールを相互接続することで実現する。



図3 人物マッピング

people_info モジュール オフィス内における人物の位置を、3章で取得した Depth 画像とロボットの自己位置から計算する ROS モジュールである。人物位置は顔が検出された三次元位置 (x, y, z) で表現される。Depth 画像を受け取ると、深度センサから見た人物の位置を計算し、それを ROS が提供する座標変換ライブラリ tf を用いてオフィス内 2D 地図における位置座標に座標変換する。変換した人物位置座標を、3章で取得した人物 ID と結び付け、人物配置地図を更新する要求メッセージ “update” を添えて `people_position_server` モジュールに送信する。

people_position_server モジュール ROS 内のネットワークでやりとりされるメッセージの履歴は自動的に記録されないため、認識した人物の位置情報を登録し記憶するモジュールが必要となる。`people_position_server` モジュールは人物の位置情報を受けて地図上に登録する。このモジュールは以下の3つの要求メッセージを受け付ける。“update” 要求を受けると、人物 ID と人物位置を対応付けて人物配置地図に登録する。“request” 要求と人物 ID を受けると現在の人物配置地図に登録されているその人物 ID の最新の人物位置を返す。また、“allid” メッセージを受けた場合は、人物配置地図に現在登録されている全ての人物の ID を返す。

face_mapping_tool モジュール ROS が提供する画面表示用モジュール rviz に人物の位置を表示する ROS モジュールである。`people_position_server` に “allid” を送信して人物配置地図に登録されているすべての人物の ID を取得したあと、個々の人物 ID に対する “request” 要求を送り、対応する人物の最新の登録位置を受け取る。人物 ID をモジュール内で人物の名前に変換したのちその人物の位置を表す ROS トピックに対して人物位置座標を publish すると、rviz に当該人物の位置にその人物の名前を表示することができる。

`people_info` は人物 ID と Depth 画像からオフィス内における人物の位置を出力し、`people_position_server` は人物 ID と人物の位置を記録する。`face_mapping_tool` は、`people_position_server` に人物の位置を問い合わせ、地図上に人物の位置をマッピングする。

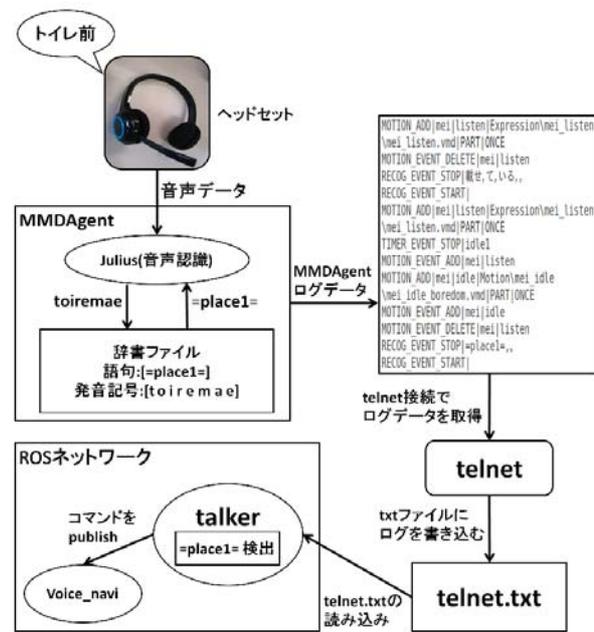


図4 音声情報からコマンド抽出

5. 音声による目的地指定システム

音声による目的地指定サブシステムの処理の流れは大きく分けて、

1. 音声情報からコマンドを抽出する処理
2. コマンドを解釈して移動ロボットに目的地を指示する処理

からなる。コマンドとは、目的地と対応付けた文字列である。

5.1 音声情報からのコマンド抽出

音声情報からコマンドを抽出する処理の流れを図4で示す。ヘッドセットから取得した音声情報は、音声インタラクションシステムである MMDAgent に入力される。MMDAgent の内部では音声認識エンジン Julius が、音声情報と単語辞書を用いて音声を文字列に変換出力する。MMDAgent はソケット通信プラグイン (「MMDAgent & Project-NAIP wiki」[10]) を使用することで、変換結果のログデータを送信する TCP/IP の telnet サービスを起動する。現状の実装では変換結果を含むログデータを telnet クライアントがファイル出力する。ファイルから登録語句を検出してコマンドに変換する ROS モジュール talker が抽出したコマンドを ROS メッセージとして publish する。talker があらかじめ定義したコマンドを抽出できるように、Julius の標準の辞書ファイルを編集して、コマンドとして意味を持つ単語の先頭と終わりに “=” 記号を付加しておく。talker モジュールは “=” 記号ではさまれた文字列をログデータから検出し、コマンドとして解釈する。辞書ファイル書き方の例を図5に示す。

音声認識コマンドを解釈して動作する ROS モジュールはこの ROS メッセージを適宜 subscribe して使用する。

語句のジャンル	登録語句	登録語句	発音記号
動作:ドウサ:動作	@0.0=place1=:トイレマエ=:place1=	[=place1=]	toiremae
動作:ドウサ:動作	@0.0=place2=:イリグチマエ=:place2=	[=place2=]	iriguchimae
名前:ナマエ:名前	@0.0=Uemas=:ウエマ=:Uemas=	[=Uemas=]	uema

図 5 Julius 辞書ファイル書き方例



図 6 目的地の指定

5.2 移動ロボットの目的地の指定

目的地の名称で移動ロボットの目的地を設定するモジュールの構成を図 6 で示す。独自に作成した ROS モジュール Voice_navi には、目的地の名称とその場所に対応した 2D 地図上の座標の組をあらかじめ登録しておく。音声コマンドの検出結果メッセージを Voice_navi が subscribe し、コマンドに含まれる目的地名と登録してある目的地名を照らし合わせ位置座標を取得する。取得した目的地の位置座標を自律移動モジュール move_base に ROS メッセージを publish することで目的地に設定する。目的地の座標が変更されると、move_base がロボットの現在の自己位置推定結果をもとに移動経路を自動的に生成し、移動を開始する。

6. 動作実験

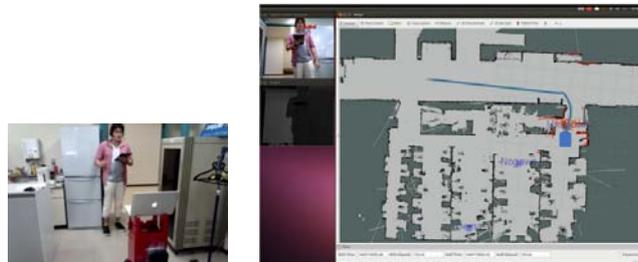
移動ロボットを、自律移動パートナーロボットシステムによって起動させ、人物マッピングシステムで人物の位置を地図上に記録しながら、音声による目的地指定システムを用いて動作させる実験を行った。ここでは「トイレ前」という音声を入力し、移動ロボットをトイレ前まで自律移動させる例について結果を示す。

図 7 は動作実験開始から 3 秒後、26 秒後、80 秒後の研究室の出入り口からトイレ前までの移動風景と内部状態を表示した画面である。内部状態画面には、rviz モジュールに表示された人物配置地図とロボットの現在の自己位置ならびに移動経路プラン、さらに人物観測画像と顔認識結果を並べている。

ロボットが移動しながら人物配置を地図に動的に記述していき、後からその人物が観測された場所まで自律的に移動することができた。

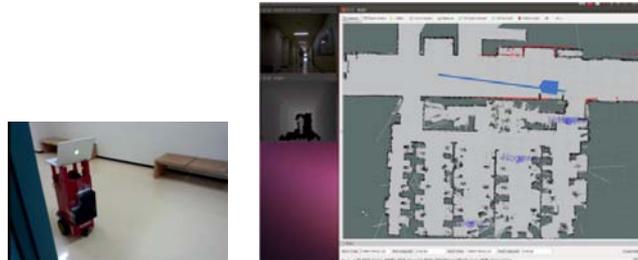
7. おわりに

ROS 上のネットワークサービスとして顔検出・識別機能、人物の存在場所・行動記録機能を実装し、自律移動可能なロボットがオフィス内で自由に移動しながら時々刻々と変わる人物の配置をマッピングして、音声等による名前の指示でその人のいるところまで移動するロボット制御プラットフォームを構築した。本ロボットは同じ空間に実装した環境埋め込み型の深度画像センサーやネットワークディスプレイと連動するこ



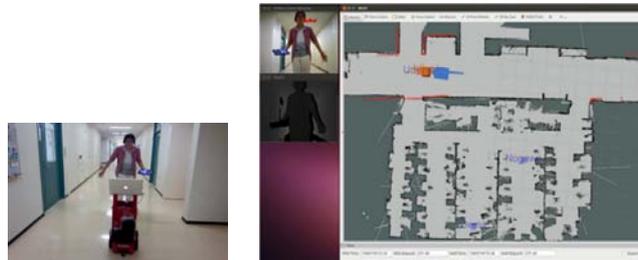
移動風景 (3 秒後)

内部状態画面 (3 秒後)



移動風景 (26 秒後)

内部状態画面 (26 秒後)



移動風景 (80 秒後)

内部状態画面 (80 秒後)

図 7 動作実験

とが可能であり、今後はコミュニティ内における人々の対話的関わりの深さを画像認識や音声認識を応用した観察をもとに、人同士の関わりあいの支援を行う方法について研究を進める予定である。

参考文献

- [1] Robot Operating System, <http://www.ros.org/>
- [2] RT ミドルウェア OpenRTM-aist, <http://www.openrtm.org/openrtm/ja>
- [3] H. Asoh, Y. Motomura, F. Asano, I. Hara, S. Hayamizu, K. Itou, T. Kurita, T. Matsui, N. Vlassis, R. Bunschoten, B. Kroese: Jijo-2: An Office Robot that Communicates and Learns, IEEE Intelligent Systems, Vol.16, No.5, pp.46-55, 2001.
- [4] 小林俊介, 島田伸敬, 「ロボットによる説得対話のための翻意を扱う知識状態モデル」, 2013 年度人工知能学会全国大会論文集, 1K4-OS-17b-3, 富山国際会議場, 2013.
- [5] 李 晃伸, 大浦敬一郎, 徳田 恵一, 「魅力ある音声インタラクションシステムを構築するためのオープンソースツールキット MMDAgent」, 電子情報通信学会技術研究報告. NLC, 言語理解とコミュニケーション, 111(364), pp.159-164, 2011.
- [6] 河原達也, 李晃伸, 「連続音声認識ソフトウェア Julius」, 人工知能学会誌, Vol.20, No.1, pp.41-49, 2005.
- [7] navigation, <http://wiki.ros.org/navigation>
- [8] hector_slam, http://wiki.ros.org/hector_slam
- [9] MMDAgent, <http://www.mmdagent.jp>
- [10] MMDAgent, & Project-NAIP wiki <http://cube370.wiki.fc2.com/>