

人混み中での移動ロボットの自己位置推定に有効な 測域センサデータ処理法の提案

- 蓄積処理と近傍点削除による壁面の抽出 -

○藤井 祐介^{†1}, 大矢 晃久^{†1}, 坪内 孝司^{†1}

Effective Localization Method for Mobile Robots in Crowded Environments Using Range Data

-Wall Extraction by the Process of Data Accumulation and Near Point Deletion-

*Yuusuke FUJII^{‡1}, Akihisa OHYA^{‡1}, Takashi TSUBOUCHI^{‡1}

Abstract—The purpose of this study is to enable a mobile robot to localize itself in a crowded environment. Scan matching is an example of localization method. However, it is not effective in environments with many dynamic obstacles, i.e., crowded environments. In this study, we propose a method to obtain data usable for scan matching in crowded indoor environments by the process of scan data accumulation and near point deletion. With this method, we can obtain wall data that is similar to when there is no crowd. Experiments were carried out to investigate the effectiveness of the method. As a result, the method was found to be effective in localization in environments with many dynamic obstacles.

Keywords: Mobile robot, Localization, Scan-Matching, Crowded environments

1. はじめに

現在、人が生活する空間で自律的に移動し、活躍する移動ロボットの開発が進んでいる。ロボットが自律的に移動するには、センサで周囲の環境を観測して、そのデータから自分の位置を推定する必要がある。

移動ロボットの自己位置推定の方法の一つとしてスキャンマッチングがある。これは既知の物体の位置や壁面の形状が記された地図データと、周囲の環境をレーザスキャナ等で観測して得られるスキャンデータとを照らし合わせて自己位置を求める方法である [1]。しかし、駅構内などの人が密集している場所だと、人に遮られてセンサから周囲の環境が見えず、周囲の見え方の変化も激しいために、スキャンマッチングでは人混み中だと自己位置推定に大きな誤差が生じる問題がある。実世界でロボットの活動場所を広げるためには、人混みの中でも自己位置推定ができることが必要となる。

そこで本研究では、人混み中でのスキャンマッチングに有効な測域センサデータ処理方法を提案し、その方法が人混み中での自己位置推定にどの程度有効なのか検証する。

2. 関連研究

人混み環境での自己位置推定に関する先行研究は盛んであり、すでにいくつかの方法が提案されている。データの取得方法に関する研究として山田らの手法 [2] では、センサを高い位置に搭載し、かつ斜め上方向に向けることで、形状変化が少ない人より高い所にある物体をランドマークとして観測し、自己位置推定を行っている。また、マッチングの手法を改良した研究もある。Masuda らの手法 [3] では、ICP アルゴリズム [4] において地図データとスキャンデータの対応点間距離の平均値と標準偏差を用いた外れ値除去の要素を入れた方法を提案している。このように、従来の自己位置推定の研究

では、データの取得やマッチングに改良を加えたものが多い。これに対して本研究では、センサの水平方向スキャンのみを使用し、データ取得とマッチングの間のデータ処理に改良を加えることで、人混み中でのスキャンマッチングによる自己位置推定を可能にすることを目的とする。

データ処理を工夫する関連研究としては、Fox らの手法 [5] が挙げられる。これは人の密度が高い動的な環境において、地図と異なっているセンサの値を検出し、ロボットが正しい位置にいることの信頼度を上げるスキャンデータのみを取り出す方法である。本研究は、データ処理の際に地図データを必要としない点が大きく異なる。

3. 蓄積と近傍点削除によるスキャンデータ処理

3.1 アイデア

人混み中で自己位置推定に失敗する原因は、オクルージョンなどにより地図データに記されているものが観測できないことと、地図データに記されていない人などのスキャンデータが得られてしまうことの 2 つが考えられる。そこで、それぞれへの対策を考える。

まず、人混み中では人に遮られて周囲の環境の一部しかセンサから観測できないことが多い。しかしロボットが動いたときには、人の隙間から見える部分が変わり、見えていなかった場所が見えるようになる。そして、それぞれの場所での周囲が一部分ずつ見えているデータを集めれば、Fig. 1 のように周囲の人のデータなどと同時に遮られて見えない部分が無い環境すべてのデータが得られると考えられる。そこで、オクルージョンにより地図にあるものが観測できないことへの対策として、ロボットが走行中にデータを蓄積することを考える。もしロボットが動くことに加えて人も動いた場合、見える部分が変わりやすくなるため、蓄積することで見えなかった部分のデータがさらに得やすくなる。

次に、Fig. 2 のように屋内などの壁に囲まれた状況でのスキャンマッチングを考えてみる。地図データとなる壁面の形状と、人などの障害物の位置関係を見てみると、ロボットから

^{†1} 筑波大学 大学院 システム情報工学研究科

^{‡1} Graduate School of Systems and Information Engineering,
University of Tsukuba

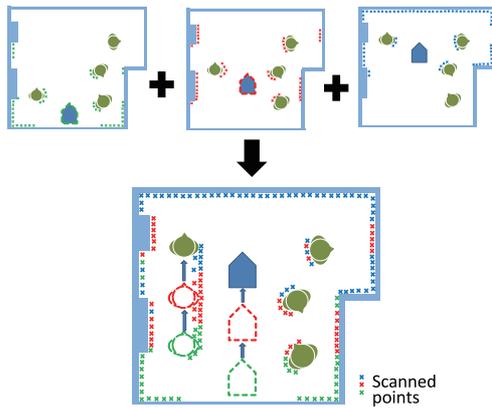


Fig. 1 各地点で得られるデータを蓄積したときのスキャンデータ

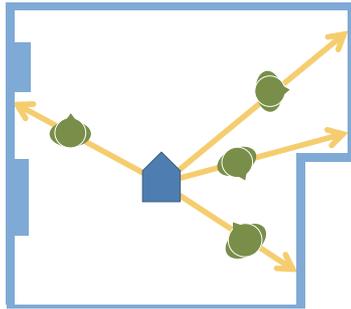


Fig. 2 周囲を壁に囲まれている状況でのロボットと人の位置関係
見て障害物は手前にあり、地図となる壁面は遠くにある。そこで、地図に記載されていないものが観測されてしまうことへの対策として、前に述べたデータの蓄積後、ロボットから見て一番遠くにあるものだけを取り出せば壁のデータだけが得られるのではないかと考えた。これは周囲が壁に囲まれたような状況で有効となるが、通常の屋内環境では多くの場合これに当てはまると考えられる。また、人がいない状況においても遠くにあるものが壁であることに変わりはないため、同様に壁面のデータのみが得られる。

3.2 データ処理のアルゴリズム

上記のアイデアを実現するアルゴリズムを下記のように構築した。

- 1) ロボットが動きながら一定距離を走行する間、測域センサのスキャンデータをロボットの自己位置により世界座標系に変換したデータを蓄積する。
- 2) Fig. 3 のように、蓄積したデータに対してロボットの現在位置を中心に、微小角度ごとに区切る。
- 3) Fig. 4 のように、微小角度の範囲内でロボットの位置から見てもっとも遠い距離にあるデータだけを取り出す。

4. 評価実験

4.1 実験の方法

人混みにおいて、提案するデータ処理により、どの程度自己位置推定ができるか調べるため実験を行った。実験に使用したロボットは、縦横の幅と高さが約 0.3 [m] の車輪型移動ロボットである。ロボットには、外界センサとして北陽電機製の測域センサ UTM-30LX を、地面から高さ 0.2 [m] の位置に搭載した。このセンサは、角度分解能 0.25 [deg] で走査角度 270 [deg] の範囲の距離データを得ることができる。今回の実験では、ロボット後方はセンサの死角となり、壁のデータが蓄

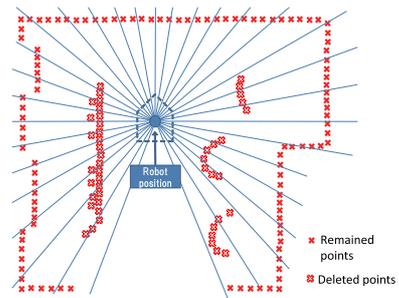


Fig. 3 ロボットの位置を中心に蓄積したデータを微小角度ごとに区切る処理

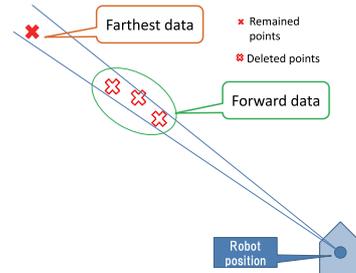


Fig. 4 区切られた微小角度内で一番遠い距離にあるデータを残す処理



Fig. 5 人に周囲を囲まれた時の実験の様子



Fig. 6 行列を再現した時の実験の様子

積されないため、3.2 で述べたアルゴリズムの 3) の処理はセンサの視野角度内のみに対して行うことにした。また、区切るときの角度の値は測域センサの角度分解能と同じ 0.25 [deg] に設定した。さらに、データを蓄積する走行距離は、1 [m] とした。

本実験では、人混みとしてロボットの周囲に人が集まって一緒に移動する状況 (Fig. 5) と、人の行列がある状況 (Fig. 6) の 2 種類で実験を行った。

実験は、人がいないときの事前走行により作成した地図データを、人混み中でデータ処理により得られたスキャンデータを、初期位置をずらした状態から ICP アルゴリズムによりマッチングさせた。その結果推定された位置が、データを取得した時のロボットの自己位置にどれだけ近いかを調べた。なお、提案手法との比較のために、その位置で得られる一回のスキャンデータを用いたときについても同様の実験を行った。

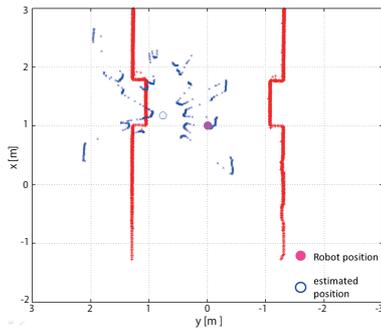


Fig. 7 人に周囲を囲まれたときのスキャンマッチング結果 (1 スキャンデータの時)

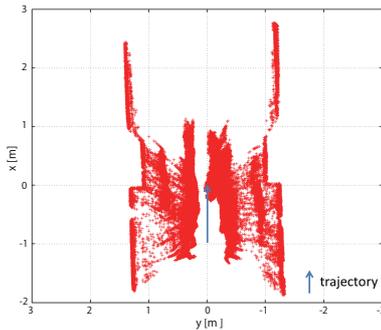


Fig. 8 人に周囲を囲まれたときの蓄積スキャンデータ

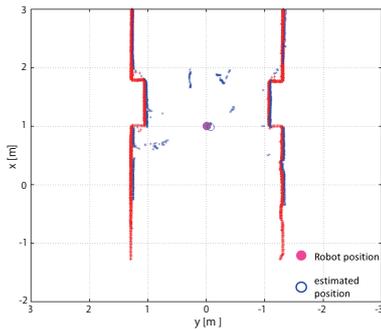


Fig. 9 人に周囲を囲まれたときのスキャンマッチング結果 (データ処理をしたとき)

4.2 データ処理と自己位置推定の結果

4.2.1 人に周囲を囲まれたとき

Fig. 7 に 1 スキャンデータを用いたときのマッチング結果を示す。地図データ (赤色) に対して、大きく形状が異なったスキャンデータ (青色) が得られているため、正しくマッチングできていない。

次に、Fig. 8 に人に囲まれながらロボットが 1 [m] 走行したときに蓄積したデータを示す。人が一緒に動いて隙間から見える壁の部分も移り変わったため、蓄積することで壁面の形状を欠かすことなく得ることができた。そして、Fig. 9 に蓄積データに処理を加えて得られたスキャンデータを地図データに対しマッチングをさせた結果を示す。障害物データが消えて壁面の形状のみがうまく取れたため、正しくマッチングできている。

Table 1 に、1 スキャンデータを使用したときとデータ処理をしたときの、自己位置推定結果の誤差の値を示す。1 スキャンデータを使用したときに比べて誤差を大きく軽減することができた。

Table 1 人に周囲を囲まれたときの自己位置推定結果の誤差

	Δx [cm]	Δy [cm]	$\Delta \theta$ [deg]
1 スキャンデータの時	+15.2	+75.1	-17.8
データ処理をしたとき	-2.0	-7.0	-1.5

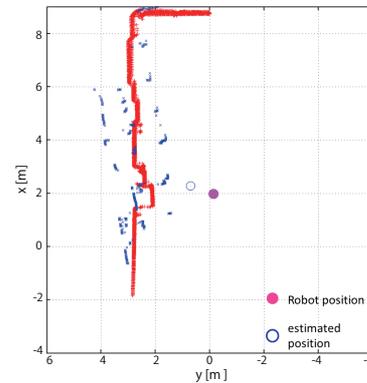


Fig. 10 人が行列を形成しているときのスキャンマッチング結果 (1 スキャンデータの時)

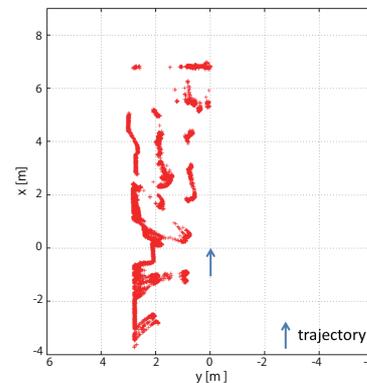


Fig. 11 人が行列を形成しているときの蓄積スキャンデータ

4.2.2 人が行列を形成しているとき

行列がある方向の壁のみで自己位置推定できるかを調べるため、実験中ではロボット正面から壁がある左側のみのスキャンデータを使用した。Fig. 10 に 1 スキャンデータを用いたときの自己位置推定結果を示す。こちらも囲まれたときと同様に人のスキャン点が多く存在し、地図データと形状が大きく異なっているためマッチングに失敗している。

次に、Fig. 11 に行列の横を走行しながら蓄積したデータを示す。壁の大部分は蓄積できていないが、Fig. 10 中の赤い地図データと比較した時、左上の壁の角があるところが人に遮られて蓄積できていない。Fig. 12 に、蓄積したデータに処理をしてからマッチングした結果を示す。蓄積した時に奥の壁が見えなかったため、その方向にある前方の人のデータを削除しきることができず、完全にマッチングさせることができなかった。

Table 2 に、1 スキャンデータを使用したときとデータ処理をしたときの、自己位置推定結果の誤差を示す。データ処理をしないときに比べれば誤差の値は軽減されたが、データ処理をしたときでも主に角度の誤差が大きく残ってしまった。

5. 考察

行列のような人が動かない場合、データ処理のみでは障害物データを削除しきることができず、自己位置推定に失敗す

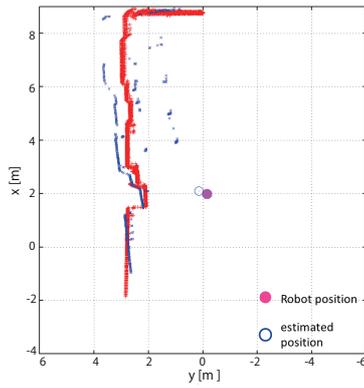


Fig. 12 人が行列を形成しているときのスキャンマッチング結果（データ処理をしたとき）

Table 2 人が行列を形成しているときの自己位置推定結果の誤差

	Δx [cm]	Δy [cm]	$\Delta \theta$ [deg]
1 スキャンデータの時	+27.7	+71.4	+9.9
データ処理をしたとき	+9.1	+15.5	+6.0

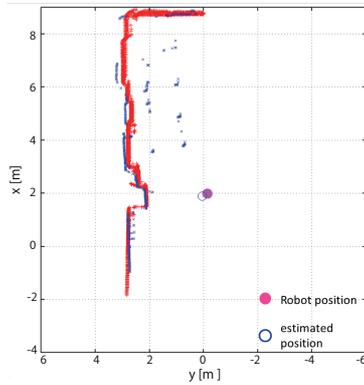


Fig. 13 人が行列を形成している時のスキャンマッチング結果（データ処理と外れ値除去を組み合わせた時）

ることがあった。データ処理により大部分の壁面みのデータを得ることができても、多少でも障害物データが残ってしまうと自己位置推定に誤差が出てしまう。そこで残ってしまう障害物データに対して、データ処理に加えて従来の外れ値除去を加えた場合、自己位置推定結果が改善されるかを検証した。

行列があるときの実験において、データ処理をした後、ICP アルゴリズムによるマッチング時に Masuda らの手法 [3] を参考にした対応点間距離の平均と分散を用いた外れ値除去の要素を加えた。Fig. 13 にその時のマッチング結果を示す。外れ値除去によって壁面から離れて残ったデータがはじかれて、データ処理のみのときよりマッチング結果が良くなっている。Table 3 に、外れ値除去も組み合わせた場合の自己位置推定結果の誤差の値を示す。自己位置推定の結果も、x 方向の誤差はあまり変わらなかったが、y 方向と角度の誤差については外れ値除去を入れない場合と比べて大きく誤差の値を減らすことができた。

結果としては、データ処理に加えてマッチング手法の改良も組み合わせることで、人混み中での自己位置推定の誤差をさらに大きく減らすことができることもわかった。

Table 3 外れ値除去の併用による自己位置推定結果

	Δx [cm]	Δy [cm]	$\Delta \theta$ [deg]
データ処理と外れ値除去	-11.2	+4.5	+2.7

6. まとめと今後の課題

本研究では、データを蓄積し遠くにあるスキャン点のみを取り出すことで人混みの中でも周囲の壁のみのスキャンデータを得る手法を提案し、この手法が人混みの中でのスキャンマッチングによる自己位置推定にどの程度効果があるのか検証するため、人混みを再現して自己位置推定の実験を行った。

結果は、人が動き回る環境では、データ処理が人混み中での自己位置推定に効果があることが分かった。一方、人が立ち並んでいるような環境では、データ処理をしても自己位置推定に誤差が残ってしまう場合があることが分かった。

今後の課題として、静止障害物への対応や蓄積距離などのパラメータ調整などがあり、様々な人混みでのスキャンマッチングの実験を通して検証していく必要がある。

参考文献

- [1] Y. Hara, H. Kawata, A. Ohya, S. Yuta: "Mobile Robot Localization and Mapping by Scan Matching using Laser Reflection Intensity of the SOKUIKI Sensor", *Proc. of IECON'06*, 2006.
- [2] 山田 大地, 石田 卓也, 関口 誠, 岡村 公望, 福永 和海, 大矢 晃久: "高所特徴による自己位置推定を用いた明示的な動作計画に基づく屋外自律ナビゲーション", *日本ロボット学会誌*, vol. 30, no. 3, pp. 253-261, 2012.
- [3] T. Masuda, K. Sakaue, N. Yokoya: "Registration and Integration of Multiple Range Images for 3-D Model Construction", *Proc. ICPR*, vol. 1, pp. 879-883, 1996.
- [4] 増田 健: "ICP アルゴリズム", *情報処理学会研究報告 CVIM-168*, no. 23, 2009.
- [5] Dieter Fox, Wolfram Burgard, Sebastian Thrun, Armin B. Cremers: "Position Estimation for Mobile Robots in Dynamic Environments", *J. of Artificial Intelligence Research*, vol. 11, pp. 391-427, 1999.