

手書き地図における直線情報観測確率を活用した 屋内向け SLAM

○鈴木龍紀（中央大学） 池勇勲（JAIST）Sarthak Pathak（中央大学）
梅田和昇（中央大学）

1. 序論

現在、自律移動ロボットによる人間の業務代替が注目されている。自律移動ロボットの運用において SLAM (simultaneous localization and mapping) 技術は必要不可欠である。地図構築の精度を向上させるべく、事前情報を活用した手法が考案されてきた。森らは建物の内装地図などの概ね形状が一致している簡易地図を事前地図として活用しロボットの自己位置推定を行った[1]。町中らはGoogleマップから得た情報を元にロボットの軌跡地図を作成し自己位置推定に活用した[2]。しかしこれらはいずれも、事前情報の準備やパラメータの調整に時間がかかる。またこれらの手法はいずれも、自己位置推定にとどまっており、地図構築はできない。

一方、事前情報の中でも人の事前知識による手書き地図の準備は容易に可能である。BahramらはMCL (Monte Carlo localization) [3]による自己位置推定手法を手書き地図上でも可能にした[4]。これにより、手書き地図を用いた自律移動ロボットの運用の手法が数多く提案された[5,6,7]。しかし、これらの手法は地図構築における手書き地図の活用方法ではなく、移動ロボットの経路生成や手書き地図上での自己位置推定にとどまっている。そこで著者らは、手書き地図上の直線情報と観測した直線情報間の幾何学的対応関係を求め、SLAMに応用する手法を提案した[8]。この手法では、観測した直線間の配置関係と手書き地図上の直線間の配置関係が単一で一致する場合のみ幾何学的に対応したと判断するため、類似形状が多数観測される環境では動作せず、汎用性に乏しい。そこで本研究では、類似形状が多数存在する環境においても動作する手書き地図を事前情報として活用した屋内向けSLAMによる地図生成システムを提案する。

2. 手法概要

本手法は測域センサ搭載車輪型移動ロボットを前提としている。提案手法の全体的なプロセスを図1に示す。本手法は、手書き地図上における処理と実環境上における処理の2つの大枠に分けられる。

まず手書き地図上における処理について説明する。初めに手書き地図から確率的ハフ変換を利用し、直線情報を抽出する。次に、オドメトリを用いて、手書き地図上におけるMCL[4]を行い、手書き地図上のロボット位置を推定する。MCL[4]は一般的なMCL[3]と同様にパーティクルを活用し、ロボットの位置推定を行う他、状態変数に地図のスケールも加

え手書き地図のスケールも同時推定する。続いて、レイトレーシングを用いて疑似測域センサデータを点群として取得する。この時、疑似測域センサデータの各点が、手書き地図から抽出した直線のうち、どの直線に属しているかも同時に算出する。

次に、実際環境上における処理について説明する。初めに、オドメトリによるロボットの位置更新を行う。次に、ロボットに搭載した測域センサで観測した点群からRANSAC (random sample consensus) を用いて直線を抽出する。続いて、手書き地図上における処理で取得した疑似測域センサ点群と実測域センサ点群の対応を求めることで、手書き地図における直線と実際に観測した直線の対応を算出し、確率密度関数による直線の観測確率を求める。その後、直線の観測確率が閾値を超えた直線を用いて、回転成分の自己位置推定を行う。また、スキャンマッチングに基づく並進方向の自己位置推定を行う。

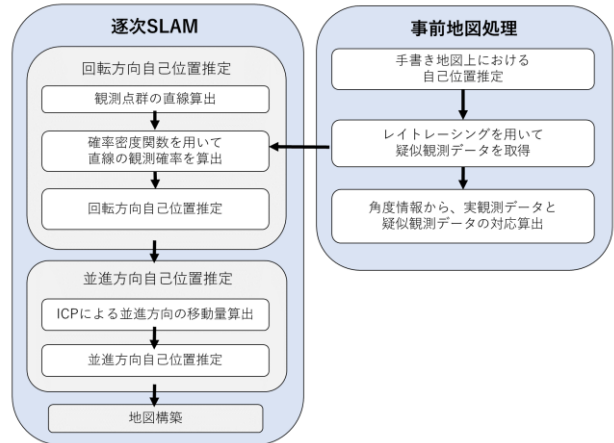


図1 手法概要

3. 直線観測確率算出

本手法では、手書き地図上の各直線が観測されたかの判別に次式の直線観測確率を活用する。

$$f(r) = \frac{1}{\sqrt{2\pi}\sigma^2} \exp \left\{ -\frac{(r_i^t - s_t \hat{r}_i^t)^2}{2\sigma^2} \right\} \quad (1)$$

ただし、 r_i^t と $\hat{r}_i^t$ はそれぞれ t フレームで取得した点群の i 番目の点と t フレームにおける事前地図上のロボット位置から取得した仮想点群の i 番目の点である。また、 σ は測域センサの不確かさである。 $f(r)$ は各直線の存在確率である。

直線観測確率是对应関係にある手書き地図上にて疑似測域センサから得られた点群の各点と、実際

に得られた点群の各点の点間ユークリッド距離に基づいて算出される．上記の各点の対応関係は，点群のインデックスで求める．すなわち，式(1)の直線観測確率が閾値を超えた時に直線は観測されたとみなす．

4. 自己位置推定

4.1 回転方向自己位置推定

3 章にて得られた手書き地図上における観測された直線と実環境で観測された直線を用いて，次式のように回転方向の自己位置推定を行う．

$$\theta = \frac{1}{N} \sum_{i=1}^M (\theta_{k_i} - \theta_i) \quad (2)$$

ただし， θ は回転方向のロボットの修正量であり k_i は i 番目の観測直線に対応する事前地図上の直線インデックスである．また， M は対応が算出された直線の数である．

4.2 並進方向自己位置推定

回転方向の自己位置推定終了後，筆者らの手法[9]と同様に point-to-plane ICP (iterative closest point) を用いて並進方向の自己位置推定を行う．この時の並進方向の移動量は次式のような距離の二乗和 E を評価値として算出する．

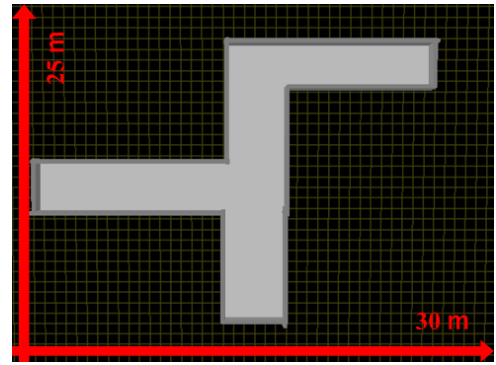
$$E = \min \sum_{i=1}^R \left\| \mathbf{n}^T (\mathbf{y}_{u_i} - (\mathbf{x}_i + \mathbf{T})) \right\|^2 \quad (3)$$

ただし， $\mathbf{x}$ と $\mathbf{y}$ はそれぞれ前のフレームでの既存スキャン点群（以下，ターゲット点群）の点と現在フレームでスキャンした点群（以下，ソース点群）の点であり， $\mathbf{n}$ は法線ベクトルである．また， R と u_i はそれぞれソース点群の点の数（繰り返し計算の試行回数）とターゲット点群中の i 番目の点に対応するソース点群のインデックスである． $\mathbf{T}$ は変換行列の並進成分ベクトルである．

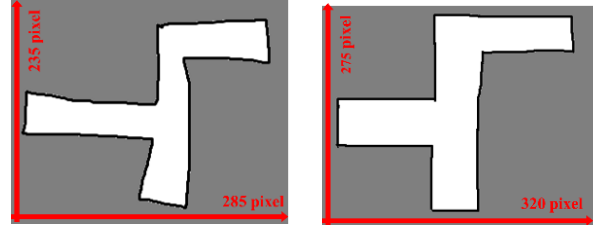
5. 実験

5.1 概要

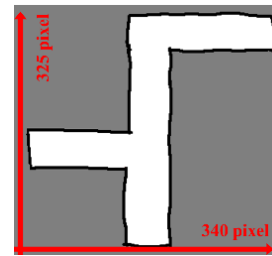
本手法の検証のため図 2(a)に示すシミュレーション環境にて実験を行った．手書き地図は，Windows 10 内蔵ソフト「ペイント」を使用して簡単に作成した．作成した手書き地図での 1 pixel は 10 cm として処理される．本実験にて使用した手書き地図を図 2(b)～(d)に示す．図 2(b)は乱雑に書いたため曲線が目立つ．図 2(c)は実験環境に比べて左側の形状の横比率にズレがある．図 2(d)は左の形状の横比率及び上に続く経路の縦比率にそれぞれズレがある．また，仮想測域センサの最大測域距離は 30 m とし，仮想エンコーダの誤差は正規分布の乱数を用いて定義した．



(a) シミュレーション環境



(b) 手書き地図① (c) 手書き地図②



(d) 手書き地図③

図 2 実験環境と手書き地図

5.2 実験結果

実験では，図 2(b)～(d)を用いた提案手法による SLAM の精度を，オドメトリと ICP による SLAM と比較して評価した．図 3 と表 1 に示す結果より，手書き地図を事前地図として活用することで地図の点群の散らばりが減少していることが分かる．しかしながら図 2(d)の手書き地図を用いた場合は，手書き地図上における MCL[4]がうまくいかず地図構築も成されなかった．使用した手書き地図上の通路形状の縦横比が場所により大きく異なっているためである．実際，本手法で使用した MCL について詳細に書かれている文献[4]では，実環境と手書き地図上の縦横比が大きく異なる場合，自己位置推定が正しくなされないと明記されている．提案手法では手書き地図上における大まかな自己位置が分かれば地図構築が成されるため，文献[4]に比べ縦横比の変化には対応できるものの，急激な縦横比の変化により大まかな自己位置推定が成されず，地図構築が不可能であった．一方図 2(c) の手書き地図を用いた場合は図 3 (c)に示すように，縦横比に大きなズレがあるものの，表 1 に示すように高い精度で地図構築が成された．これは地図全体の大きさは違うものの，地図の縦横比に大きな違いがないためである．

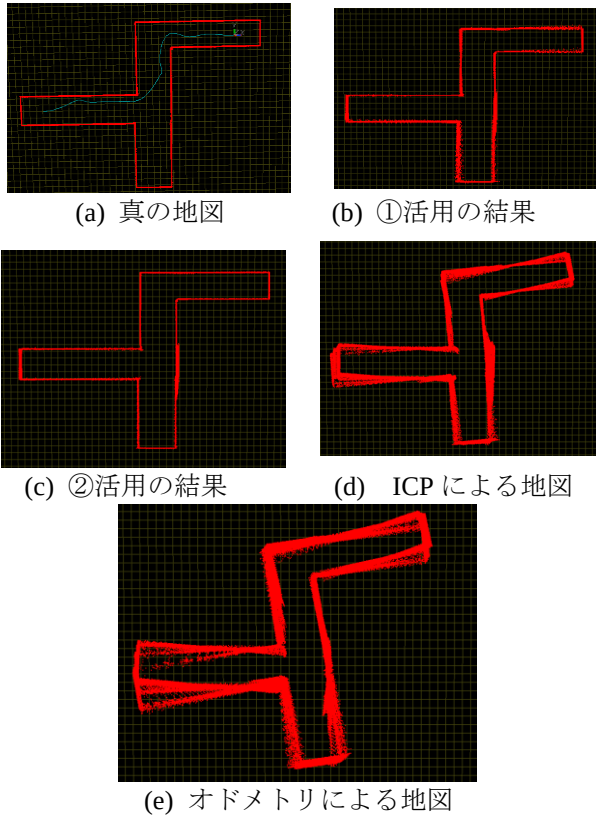


図3 生成された地図

表1 地図の誤差 [m]

オドメトリ	6.157
ICP	2.72
手書き地図①	0.270
手書き地図②	0.297
手書き地図③	-

6. 結論

本研究では、手書き地図上におけるロボットの自己位置推定を行った後、直線情報の対応関係による屋内向け SLAM を提案した。実験は、シミュレーション環境にて行った。走行環境に近い手書き地図や、スケールがほぼ一定の手書き地図を用いた場合、高精度な地図構築が成されたものの、縦横比の変化が大きな手書き地図では地図構築が成されなかった。

よって今後の展望としては、縦横比の変化が大きな手書き地図上において本手法がうまく動作しない問題に対処する。

参考文献

- [1] 森武俊, 栗原誠, 黒田藍子, 野口博史, 田中雅行, 福井類, 下坂正倫, 佐藤知正: “パーソナルモビリティのための簡易マップを手がかりとする自己位置同定と詳細マップの生成 (移動ロボットの自己位置推定と地図構築),” ロボティクス・メカトロニクス講演会講演概要集 2011, 2011.
- [2] 町中希彰, 深橋遼太, 細田佑樹, 金馬誠郎, 黒田洋司: “ノード・エッジグラフに基づいた内界センサのみに

よる自己位置推定,” 第23回ロボティクスシンポジウム講演論文集, pp. 139-144, 2018.

- [3] Frank Dellaert, Dieter Fox, Wolfram Burgard, and Sebastian Thrun, “Monte Carlo localization for mobile robots,” In Proc. of the 1999 IEEE International Conference on Robotics and Automation (ICRA1999), pp. 1322-1328, 1999.
- [4] Bahram Behzadian, Pratik Agarwal, Wolfram Burgard, and Gian Diego Tipaldi, “Monte Carlo localization in hand-drawn maps,” In Proc. of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2015), pp. 4291-4296, 2015.
- [5] Federico Boniardi, Abhinav Valada, Wolfram Burgard, and Gian Diego Tipaldi, “Autonomous indoor robot navigation using a sketch interface for drawing maps and routes,” In Proc. of the 2016 IEEE International Conference on Robotics and Automation (ICRA2016), pp. 2896-2901, 2016.
- [6] Federico Boniardi, Bahram Behzadian, Wolfram Burgard, and Gian Diego Tipaldi, “Robot navigation in hand-drawn sketched maps,” In Proc. of the 2015 IEEE European Conference on Mobile Robotics (ECMR2015), pp. 1-6, 2015.
- [7] Kevin Chen, Marynel Vázquez, and Silvio Savarese, “Localizing Against Drawn Maps via Spline-Based Registration,” In Proc. of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2020), pp. 8521-8526, 2020.
- [8] 鈴木龍紀, 池勇勲, 梅田和昇: “手書き地図の直線情報の対応関係を活用した屋内向け SLAM,” 第38回日本ロボット学会学術講演会予稿集, 2020.