

回転カメラによるモンテカルロ自己位置同定を用いたタスク支援

堀口 研一[†] 神尾 正太郎[‡] 伊庭 斉志[‡]

[†]東京大学 工学部電子情報工学科 [‡]東京大学大学院 新領域創成科学研究科基盤情報学専攻

1. はじめに

近年、研究が盛んに行われている 2 足歩行のロボット (いわゆるヒト型ロボット) の研究は、よりヒトに近い動きをさせることで、より複雑なタスクを遂行できることを目的としている。中でも、予定された計画に沿ってロボットが自律して行動するというタスクは、自分の位置や向いている方向などの情報をロボット自身がセンサ情報から把握しなければならず大変困難を極める。それに対して視覚やセンサを用いるなど、さまざまな方法が提案されている。ヒューマノイドロボットは移動によるずれが大きく、与えられた経路どおりに行動させることが難しい。

本論文では、ロボット (図 1:HOAP-1:富士通オートメーション開発) の位置を認識するため、自己位置同定手法を単眼の回転カメラで実現する。自己位置同定手法には Monte Carlo localization (MCL) を用いる。

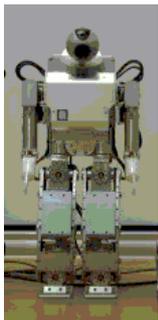


図 1 使用ロボット

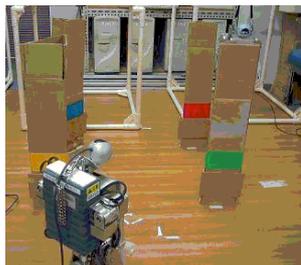


図 2 実験風景

2. Monte Carlo localization

MCL は Markov Localization の一種であり、自己位置を多数のパーティクルと呼ばれる点を用いて離散的な確率分布で表現する自己位置同定手法である [1][2]。パーティクル $s_t^{(i)} (i=1,2,3,\dots,N)$ は、おのおのがパラメータとして姿勢 $l_t^{(i)} = (x_t^{(i)}, y_t^{(i)}, \theta_t^{(i)})$ と重み $w_t^{(i)}$ を持ち、 $s_t^{(i)} = (l_t^{(i)}, w_t^{(i)})$ と表現される。一般に、パーティクルの数 N が大きくなるほど、近似精度は高くなる。

ロボットの移動 a_{t-1} をパーティクルの分布に反映するためには、次の手続きを $i=1,2,\dots,N$ について行い、 $s_t^{(i)} (i=1,2,\dots,N)$ を生成する。

(1) $s_{t-1}^{(j)} (j=1,2,\dots,N)$ から、重み $w_{t-1}^{(j)}$ の示す確率に従って、 $s_{t-1}^{(i)}$ を選択

(2) $p(l | l_{t-1}^{(k)}, a_{t-1})$ に従って l を選択

(3) パーティクル $s_t^{(i)} = (l, 1/N)$ を生成

センサ情報 (d_t とする) を反映させ、重みを正規化するために、

$$\alpha = \sum_{i=1}^N P(d_t | l_t^{(i)}) w_t^{(i)} \dots \dots \dots (1)$$

という近似で求め、 $i=1,2,\dots,N$ に対し、

$$w_t^{(i)} \leftarrow P(d_t | l_t^{(i)}) w_t^{(i)} / \alpha \dots \dots \dots (2)$$

という手続きによって重みを変更する。

3. アルゴリズムの実装

このアルゴリズムは、環境中に色分けされたマーカーを置き、ランドマークとして用いる。ヒューマノイドロボットは一行動ごとに、頭部に装着した回転カメラでランドマークの画像を捉え、推定経路をリアルタイムで計算する。

3.1. アルゴリズム

自己位置同定のアルゴリズムは以下のようにになっている。

1. ランドマークの位置情報を読み込む。
2. ロボットの初期位置、及び向きを入力する。
3. 移動命令を入力し、ロボットがそれに従って移動する。
4. 現在位置 (l_{t-1}) と移動命令から仮の移動先を計算する。

$$(l_t \leftarrow l_{t-1} + a_{t-1}) \quad (a_t \text{ は移動命令をモデル化したもの})$$

5. その位置 (l_t) から見えるランドマークの中から、見やすい

(カメラの回転量の少ないもの) を選んでカメラの回転に必要な角度を計算する。

6. カメラを回転させ、画像を得る。(画像を、誤差の少ない画面の中心付近で捉えられるように回転角度を補正する)。(3.2. 節で説明)

7. 複数の画像が得られるように 5, 6 を数回繰り返す。

8. 移動命令に基づいて解候補の位置・姿勢情報を移動後の位置・姿勢に更新する。

$$(l_t^{(i)} \leftarrow l_{t-1}^{(i)} + a_{t-1}) \quad (3.3. 節で説明)$$

9. 解候補は誤差を付加されて、散らばる。

$$(l_t^{(i)} \leftarrow l_t^{(i)} + \text{noise}) \quad (3.3. 節で説明)$$

10. 散らばった解候補ひとつひとつについて、推定画像情報を計算する (3.4. 節で説明)。

11. 推定画像情報と画像データのマッチングを取り、よりマッチした解候補の重み $w_t^{(i)}$ を更新する (3.5. 節で説明)。

12. 最も重みの値 $w_t^{(i)}$ が大きいもの (重みの等しかった場合はカメラ画像と推定画像の重心の距離の小さいもの) の位置・

Task Support for a Humanoid Robot with MCL using an Orbit Camera

[†]Kenichi Horiguchi, TheUniversity of Tokyo

[‡]Shotaro Kamio, Hitoshi Iba, Graduate School of Frontier Sciences, The University of Tokyo

姿勢情報 $l_t^{(i)}$ が現在のロボットの状態と推定される。

13. 命令を待つ。(後は3~13を繰り返す)

以下にそれぞれ具体的な実装を示す。

3.2. カメラの回転角度の補正

2歩目以降は推定位置をもとにカメラの向く方向を決定するため必ずしもカメラの中心付近でランドマークの色画像を捉えられるとは限らない。そこで、ランドマークが画面の端に見えた場合に、ランドマークが中心に来るようにカメラの方向を自動で修正するようにした。

3.3. ロボット動作のモデル化

ロボットは前進、左回転、右回転の三通りを利用して移動する。これらの移動量及び移動量のばらつき(ガウス分布の定数)は実測値をもとに設定した。

3.4. 推定画像情報の計算

自己位置同定は、事前に計測してあるランドマークの各頂点の座標 (x, y, z) をデータとして保持し、カメラの位置 (x, y, z) 及びロボットの向き θ 、カメラの回転方向 ϕ 、 ϕ から各々の頂点のカメラへの見え方を計算する。

3.5. 重み計算機構の実装

本研究では、パーティクル (particle: 解候補) の持つロボットの位置情報及び向きから、視界に入るべき情報を計算により求め、その情報と回転カメラによって得られた画像データとのマッチングをはかり (画像が複数得られた場合はその積を用いる)、その度合い (マッチングの数) によって重みに変化を加えるという手法をとった。

4. 実測経路と推定経路の比較実験

図3に実際にロボットが歩いた経路と、アルゴリズムで算出した推定経路を重ねた図を示す。ロボットを用いた実験では、ロボットの歩行にかなりの誤差が含まれるため、同じ命令でも実際には少しずつ異なる経路をたどり、再現性が低い。そのため、図には、あるひとつの経路とそのときの推定経路を一例として示した。パーティクル数は100、距離のガウス分布定数を10[cm]、角度のガウス分布定数を5[度]とした。

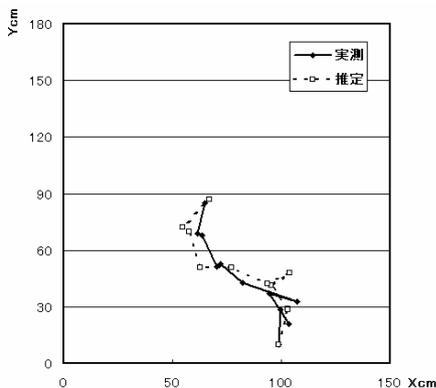


図3 実測経路と推定経路

5. 考察

図4は図3の実験において実機の経路と推定経路の誤差の推移をグラフにしたものである。この歩行における誤差の平均は7.2[cm]となった。

本実験の誤差の要因としては

- ・カメラの歪みによる画像データ処理の難しさ
- ・見る角度によってカメラに写るランドマークの色が微妙に変化し(カメラのホワイトバランスの変化やランドマーク自身の反射光が原因)、正しい画像が得られないことがある。
- ・ロボット自身の歩行時の誤差

などがある。この経路では途中、ホワイトバランスの影響で画像がうまく取れない事態が発生し、大きく誤差が膨らんだが、その後の歩行で、再び誤差が減少した。このことから、一度誤差が大きくなっても、その後鮮明な画像を捉えることが出来れば、正しい経路を推定可能であると言える。

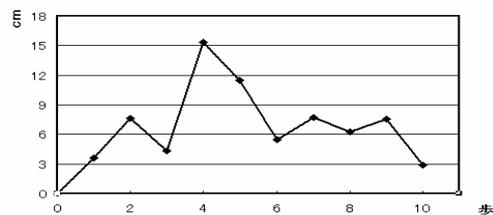


図4 実測経路と推定経路の誤差の推移

6. おわりに

本研究では、ヒューマノイドロボットに回転カメラを搭載し、自己位置同定機構を適用した。ランドマークの位置を最初に覚えさせておき、カメラを向ける方向をロボット自身に決定、修正させることで、より誤差の少ない推定を実現した。

応用例として3次元迷路内でのタスクをロボットにさせる実験も試みている(図5参照)。

また、誤差の割合を減少させるためには、マッチング条件の等しい解候補の更なる絞込み方法の検討が課題であり、今後改善を行う予定である。

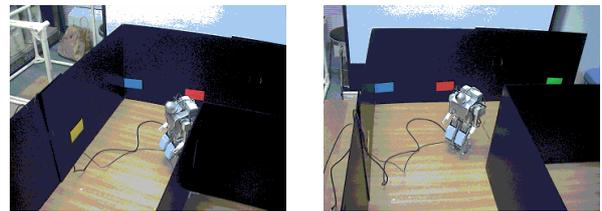


図5 3次元迷路

参考文献

- [1] D.Fox, et al., "Monte Carlo Localization: Efficient Position Estimation for Mobile Robots", In Proc. AAAI, pp.343-349, 1999.
- [2] F.Dellaert, et al., "Monte Carlo Localization for Mobile Robots", In Proc. of IEEE ICRA, pp.1332-1328, 1999.