

不整地移動ロボットのための 3次元自己位置推定と環境地図の構築

石田宏 永谷圭司 五福明夫 (岡山大学)

3-D Localization and Mapping for Mobile Robot on Terrain Environment

*Hiroshi ISHIDA, Keiji NAGATANI, Akio GOFUKU (Okayama Univ.)

Abstract—Our research group aims to develop a 3-dimensional mapping's algorithm that is installed on a mobile robot to search victims in a collapsed buildings for urban search and rescue(USAR). To realize this mission, it is important to map environment information, and also the mapping requires localization simultaneously. (This is called "SLAM problem".) In this research, we use 3-D map that is constructed by laser range finder, and estimate robot's location in a global map using correlation technique. In this paper, we introduce our localization and mapping method, and report a result of preparatory experiment for localization.

Key Words: Urban search and rescue(USAR), 3-dimensional mapping, Localization, Digital elevation map

1. はじめに

1995年の阪神大震災、米オクラホマシティ連邦ビル爆破テロ以降、防災やレスキューに関するロボットの研究が多く行われるようになった。例えば、2001年にアメリカ合州国で起こった同時多発テロにおいて利用された救助ロボットなど、様々な研究が行われている¹⁾。このような背景の下、筆者らの研究室でもレスキューロボットの研究開発を行っている。

倒壊現場でレスキュー活動を行っているレスキュー犬トレーナーの話によると、建物の倒壊現場で行方不明者を捜索や環境の調査を行う場合、倒壊によって生じた高さ及び幅数十センチメートル程度の側溝を抜け、その先の空間を調査する必要があるそうである。このような環境は、余震で倒壊が進む危険もあり、レスキュー犬やレスキュー隊員にとっても危険で、大きなストレスとなる。したがって、レスキューロボットによる災害現場の環境探索の実現が期待されている。

本研究では、地震等で倒壊した建物の内部にいる要救助者を捜索するロボットの実現を目指し、このロボットに必要とされる(1) 3次元空間におけるロボットの自己位置推定、及び(2) 3次元環境地図の構築に焦点を当て研究を行う。環境地図の構築と自己位置推定では、SLAMという手法が、2次元環境において、実現されている²⁾³⁾。本研究では、3次元環境を対象とし、Fig.1に示すように、「構築した3次元環境情報と新たに獲得したセンサ情報を比較し、相関演算を行い、現在位置を推定する」という手法を用いて、(1)、(2)を互いに補完することで実現を目指す。

本稿では、3次元環境地図の構築手法および、自己位置推定手法を紹介し、この手法を評価するための予備実験の結果について報告を行う。

2. 自己位置推定と環境地図の構築手法

2.1 概要

移動ロボットが未知環境探索を行う場合、環境地図の作成と自己位置推定を行う必要がある。倒壊した建

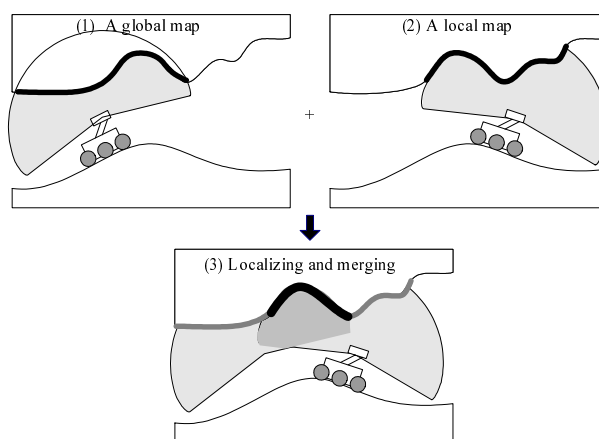


Fig.1 Introduction to our localization method

物の内部等の不整地環境においては、地面の凹凸などの情報がロボットの自己位置推定にとって有用である。そこで、以下の手順により環境地図の構築およびロボットの自己位置推定を行う。

1. ロボットを適当な距離だけ移動させる
2. ローカルマップの作成 (ローカルマップとは、ロボットの現在位置から得られるセンサ情報によって構築される物体の位置を表す3次元距離情報の集合とする。)
3. ローカルマップとそれまでに構築してきたグローバルマップとの相関を取る
4. グローバルマップにおけるロボットの自己位置を推定する
5. 自己位置推定の結果を基に、ローカルマップの情報をグローバルマップに追加する
6. 1に戻る

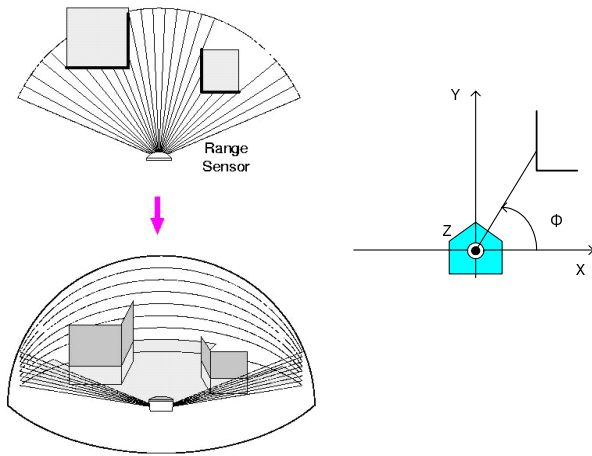


Fig.2 Acquisition of 3-dimensional environment

2.2 ロボットの姿勢情報と3次元環境情報の獲得

ロボットの姿勢情報は、3軸のジャイロセンサを用いて、ロボットのロール、ピッチ、ヨー角を検出を行う。

また、ロボットはレーザ距離計を用いて、ローカルマップを構築する。本研究で用いるレーザ距離計は、平面走査により距離情報を獲得するものである。そこで、レーザ距離計をチルト回転させることで、3次元の距離情報を獲得する (Fig.2)。ただし、後述するセンサユニットに対する、座標軸および θ, ϕ については、図のように定める。 θ は、 x 軸周りのチルト回転角を表し (水平方向を $\theta = 0 [deg]$ とする)、 ϕ は、距離計の走査角である。

2.3 Sphere digital elevation map(S-DEM)

ローカルマップを構築するために、3次元環境をBoxel(体積画素)を用いて表現すると、非常に多くのメモリと計算コストが必要となる。このコストを削減するための不整地環境の表現手法として、digital elevation map (DEM) がよく知られている⁴⁾。DEMは、環境を2次元平面の格子に区切り、それぞれの格子に登録される高さ情報のうち

最も高い情報とその格子における高さとし、3次元の環境を表現するものである。しかし、本研究で探索の対象とするのは、倒壊した建物内の空間である。このような環境は、主に閉空間であり、梁なども多く存在することが予想される。したがって、通常のDEMによって環境を表現することは困難である。

そこで、本研究では、ローカルマップの表現に sphere-DEM(S-DEM)を用いることとした。このS-DEMは、DEMが平面を格子に区切るのに対して、球面を格子に区切り作成する elevation map である。レーザ距離計を回転することによって得られるセンシングデータは、センサの位置を球の中心として $r(\theta, \phi)$ で表現できる。このため、S-DEMは、センサ情報より直接構築で

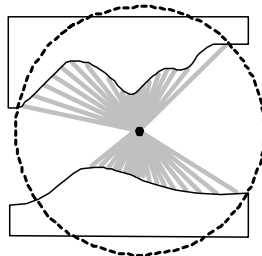


Fig.3 Construction of sphere digital elevation map(S-DEM)

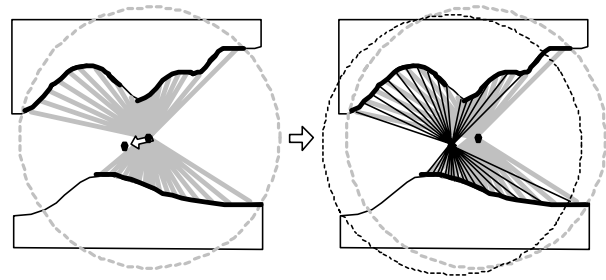


Fig.4 An example of virtual S-DEM

きるという利点をもつ。

Fig.3に2次元の場合のS-DEMの概念図を示す。この図において、円は距離計の計測範囲を表し、灰色で示した線は、物体のない領域 (自由空間と定義する) を表している。

2.4 自己位置推定手法

S-DEMを用いると、グローバルマップは複数のS-DEMと、それらの相対的な位置関係によって表現できる。また、ロボットの自己位置もグローバルマップ中において、適当な中心をとることによって構築される一つのS-DEM(仮想S-DEM)とロボットの現在の位置におけるS-DEM(ローカルS-DEM)との相関を取ることによって推定することができる。

2つのS-DEMの相対的な位置関係を求め、ロボットの自己位置を推定するために、以下に述べる手法を用いる。まず、グローバルマップにおけるS-DEM上の $r_{global}(\theta, \phi)$ をデカルト座標 (x, y, z) に変換する。次に、このS-DEMの中心を (x_g, y_g, z_g) から (x_v, y_v, z_v) に移動させる。ここで、 (x_g, y_g, z_g) は、ロボットが現在の位置に至る前にいた地点である。この (x_v, y_v, z_v) を中心とし、デカルト座標系に変換した (x, y, z) を再計算することで、仮想S-DEM $r_v(\theta, \phi)$ を構築することができる。この仮想S-DEMとローカルS-DEM $r_{local}(\theta, \phi)$ との相関値を次の式で求める。

$$d = \sum_{\theta} \sum_{\phi} (r_v(\theta, \phi) - r_{local}(\theta, \phi))^2 \quad (1)$$

この d を最小にする (x_v, y_v, z_v) は、ローカルS-DEMの中心に一致すると考えられる。したがって、 (x_g, y_g, z_g) から (x_v, y_v, z_v) の差分をロボットの移動量とし、自己位置推定を行う。

Fig.4に2次元の場合に構築される仮想S-DEMの例を示す。左の図では、S-DEMは灰色で表される。中心を図のように移動させると、右の図のように仮想S-DEMが黒の細線で表されることがわかる。

ロボットは、 (x_g, y_g, z_g) を記憶して、ローカルS-DEMとの相関値 d を最小にする (x_v, y_v, z_v) を求めることで、ローカルS-DEMの情報をグローバルマップに付加することができる。この手法を用いることで、自己位置推定を行い、環境地図の拡張を行う。

3. クローラ型移動ロボット Res-Dog

推定手法の有効性を確かめるため、不整地移動可能な機構をもつ移動ロボットを試作し、動作検証を行



Fig.5 Our target robot(Res-Dog)

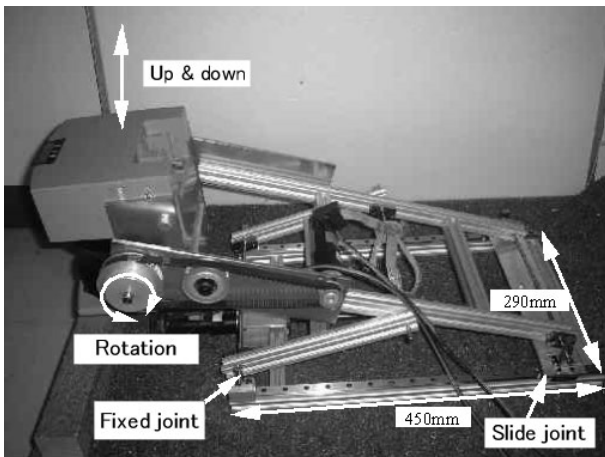


Fig.6 Developed sensor unit

うこととした．本研究で用いる，移動ロボット「Res-Dog(Fig.5)」は，クローラ型の移動機構とセンサユニットを有する．

3.1 センサユニット

対象とする環境の3次元環境情報を取得するため，本研究室でセンサユニットの設計製作を行った．このセンサユニットの外観を Fig.6 に，構成を以下に示す．

1. レーザ距離計 (SICK 社製)
レーザ距離計はセンサ中心から前方 8m の半円状の領域にある物体を測定誤差 5mm 以内で計測できる．遠くにある物体の角度誤差を考慮し，3m 以内のデータのみを用いることとする．
2. ジャイロセンサ (NEC Tokin 社製)
ジャイロセンサはジャイロだけでなく地磁気センサ方位センサを備え，ドリフトによる誤差をキャンセルする機能を有する．このジャイロセンサを



Fig.7 A target environment

用いて，ロボットの向きおよび姿勢の検出を行う

3. 上下リフト機構 (Fig.6)

ロボットが狭い側溝などを通過するためには，センサの位置を低くする必要がある．しかし，広い空間では，センサの位置が高いほうがより広範囲の環境情報を取得できる．そこで，パンタグラフ機構を用いて，レーザ距離計を上下リフトさせる機構を本センサユニットに持たせることとした．この機構は DC モータ，台形ねじ，スライド関節，固定間接によって構成される．

4. 回転機構

2.2 で述べたように，レーザ距離計を回転させる機構を実現した．回転は DC モータによって行う．

3.2 クローラ型移動機構

センサユニットを載せる移動台車として，テクノラフト社製クローラ型移動台車を用いる．この移動台車は，モータによって駆動し，バッテリー，制御用 PC 等を搭載することで「Res-Dog」の自律走行の実現を目指す．

4. 予備実験

4.1 実験環境

位置推定アルゴリズムの動作検証を行うため，研究室に簡単な環境 (Fig.7) を作り，予備実験を行った．この環境には，平らな床面にいくつかの物体が置かれている．

4.2 実験の手順

製作したセンサユニットを用いて，以下に示す手順に従い実験を行った．

- (P-1) センサユニットを水平な床の適当な位置に置き，この場所を初期位置とし，グローバル座標系の原点とした．その位置において， $-45 \leq \theta \leq 45[deg]$ でレーザ距離計をチルト回転させた．このチルト回転 $1[deg]$ 毎に，パン角 (前方 $0 \leq \phi \leq 180[deg]$) $1[deg]$ 毎の距離情報を取得する．この実験では，チルト角をジャイロセンサによって検出した．このようにして，距離情報をグローバル S-DEM $r_{global}(\theta, \phi)$ に登録する．このときの S-DEM の中心を $(x_g, y_g, z_g) = (0, 0, 0)$ とする．

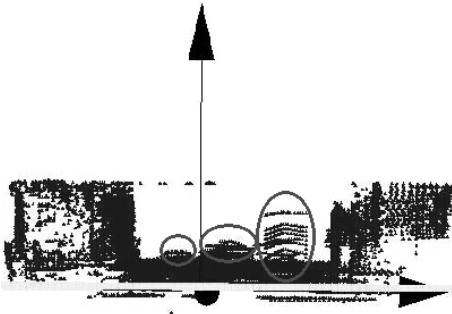
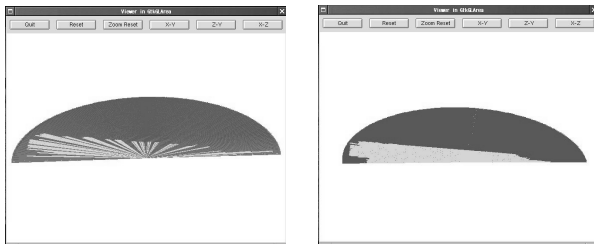


Fig.8 A result of laser range data

- (P-2) センサユニットを適当な距離だけ移動させ、さらに適当な姿勢に傾ける。ここで、(P-1)と同様に距離情報を取得し、ローカル S-DEM $r_{local}(\theta, \phi)$ として登録する。
- (P-3) センサユニットの位置の候補としてグローバル S-DEM の原点を移動させた仮想 S-DEM $r_v(\theta, \phi)$ と $r_{local}(\theta, \phi)$ との相関を (1) 式より求める。
- (P-4) 求めた相関値のうち最小となる (x_v, y_v, z_v) を現在のロボットの位置とし、実測した移動量と比較した。

4.3 実験結果

初期位置において、(P-1) に示した方法に従い、レーザ距離計を用いて得られた情報を 3次元のビューに表示したものを Fig.8 に示す。また、S-DEM を 2次元平面上にに表示することは困難であるため、 $\theta = -30[deg]$ の面で切った仮想 S-DEM を Fig.9(a) に示す。ここで灰色で示した部分が自由空間である。



(a) A global map (virtual S-DEM)

(b) A local map (S-DEM)

Fig.9 Sliced S-DEMs($\theta = -30[deg]$)

(実験 1) センサユニットの姿勢を変えずに $(x, y, z) = (22, 36, 5)[cm]$ 移動させ自己位置を推定した。結果は、 $(x, y, z) = (22, 35, 2)[cm]$ となった。これは、実際の移動量とほぼ一致したといえる。なお、Fig.9(b) は、ローカルマップを $\theta = -30[deg]$ でスライスしたものである。

(実験 2) Fig.10 のようにロボットが瓦礫によって傾いた状態にあることを想定し、 $8[deg]$ 傾けて同様の実験を行った。ロボットの移動量は、

$(x, y, z) = (22, 36, 5)[cm]$ である。このときの結果は $(x, y, z) = (24, 35, 3)[cm]$ となり、この場合も、実際の移動量とほぼ一致したといえる。

これらの実験では、グローバル S-DEM の中心の移動量は、 $-50 \leq x_v \leq 50[cm]$ 、 $-30 \leq y_v \leq 60[cm]$ 、 $-10 \leq z_v \leq 10[cm]$ とし、 θ, ϕ によって決定される S-DEM の格子の大きさはそれぞれ $1[deg]$ とした。また、グローバル S-DEM の中心を $1[cm]$ 毎に移動させ、仮想 S-DEM を構築した。したがって、今回の実験で求めた結果は妥当なものである。ただし、今回の実験では、ロボットの z 軸周りの回転角をジャイロにより正確に求めることができなかつたため、 z 軸周りの回転角は考慮に入れていない。

また、実験を天井が平らな研究室で行ったため、チルト回転角を $45[deg]$ を上限としたが、本研究が対象とする倒壊現場などの環境では $-45 \leq \theta \leq 180[deg]$ の S-DEM の構築が可能であり、より精度よく自己位置推定が行えると考えられる。

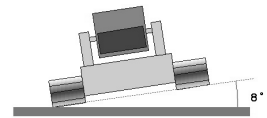


Fig.10 A tilted pose of the robot

5. まとめと今後の課題

本稿では、地震等で倒壊した建物内を探索するロボットのために、S-DEM による環境地図の構築、また構築した環境地図におけるロボットの自己位置推定手法を提案した。また、クローラ型の移動機構をもつ移動ロボット (Res-Dog) の開発を行い、予備実験において、倒壊した建物内において自己位置推定が可能となることを確認した。

今後は、ロボットの整備を進めるとともに、災害現場を模擬した環境において本手法の有効性を確認したい。

謝辞

本研究は、文部科学省大都市大震災軽減化特別プロジェクトにおける助成金により実施された。

参考文献

- 1) R.Murphy, M.Ausmus, M.Bugjska, T.Ellis, T.Johnson, N.Kelly, J.Kiefer, and L.Pollock: Marsupial-like mobile robot society, IEEE Intelligent Systems, 15,3,p.364-365 (1999).
- 2) H.Choset and K.Nagatani: Topological simultaneous localization and mapping(slam). toward exact localization without explicit localization, IEEE Trans. on Robotics and Automation, 17,2,p.125-137 (2001)
- 3) S.Thrun: A probabilistic online mapping algorithm for teams of mobile robots, International Journal of Robotics Research, 20,5,335-363 (2001)
- 4) 吉光徹雄, 久保田孝, 中谷一郎: エレベーションマップに基づく不整地における移動探査ロボットの経路計画, 日本ロボット学会誌, 18,7,1019-1025 (2000)