説 解

ヒューマノイドロボットの環境モデルと歩行計画 —3D グリッドマップの利用—

Map Building and Whole Body Locomotion Planning of Humanoid Robots
 —3D Occupancy Grid Map—

金 広 文 男* *(独) 産業技術総合研究所

Fumio Kanehiro* *National Institute of Advanced Industrial Science and Technology (AIST)

1. は じ め に

ヒューマノイドは移動する空間の幅や高さに制約がある場合でも多数の自由度を利用してぐぐったり、すり抜けたることが可能である。ただしこの利点を生かすには環境を三次元で観察してモデル化し、その結果に応じた移動動作を生成できなければならない。

環境モデルの生成手法に関してこれまで車輪型移動ロボットを中心として非常に数多くの研究が行われてきたが、その多くは二次元マップの生成を対象としていた。ヒューマノイドにおいては障害物を観察しながら2足歩行の着地位置を計画する、フットステッププランニングに関する研究が行われているが、多くの場合観察の範囲は足元に限定されている。

本稿では、知覚情報から生成した環境モデルを用いてヒューマノイドの歩行計画を行う研究に着目し、従来研究を概観するとともに筆者らが取り組んでいる3Dグリッドマップを用いた歩行計画手法を紹介する。

2. 従 来 研 究

これまでに、いくつかのヒューマノイドは視覚情報に基づく歩行計画を実現している。それらの研究は、マップの表現法に着目すると以下のように分類できる。

一つめは知覚情報から移動可能な平地部分を抽出し、その領域内に着地位置を配置するものである。この場合環境モデルは二次元で表現される。SONYのQRIOは、頭部に搭載した二つのCCDカメラから得た画像に相関ベースのステレオ処理を適用してレンジデータを生成し、ハフ変換によって平地部分を検出、occupancy gridを更新して歩行軌道の計画に用いている[1]。岡田らは同様にハフ変換に基づく平面検出をPlane Segment Finderと呼び、これを用いて歩行可能な領域を検出して歩行経路の計画に用いている[2]。

図1 環境モデルの表現方法による違い

藤本らは床のかわりに壁面を観察対象として平面検出を行うことで壁面の開口部を検出し、その大きさに応じて横歩きや四つんばい等を選択、実行する行動を実現している[3]。

二つめは二次元のマップに高さ情報を追加することで階段移動や段差の踏み越え動作、またぎ動作までを可能としたものである。この高さ情報を持った二次元マップを加賀美らは2.5D modelと呼んでいる[4]。Loachらはエッジベースのステレオ処理によって路面上の障害物を検出し、障害物の高さによってまたぐか避けるかを選択する行動を実現している[5]。岡田らは前述のPlane Segment Finderを用いて段差を検出し、段に登る動作を実現している[6]。Gutmannらは2.5D grid mapと呼ぶ形式でマップを表現し、段差の踏み越えや階段の昇降を実現している[7]。

これらの手法は環境モデルを2足歩行の着地位置を計画するために用いることを目的としているため、その表現方法は二次元や2.5次元で十分であった。しかし門状の空間をくぐり抜けたりする場合等には、三次元で環境モデルを表現しなければ対応できない。図1にマッピングの例を示す。左端の世界を二次元、2.5次元、三次元でそれぞれマッピングした結果が右の三つである。2.5次元のマッピングでは奥の門の中や脇の空間を通り抜けられるかどうか判断できず、さらに二次元のマッピングでは前方の箱が踏み越えたり、またいだりできる形状なのかも判断できないことが分かる。

一方で環境の三次元モデルは既知として複雑な動作を行う研究は行われているものの、三次元の環境モデル生成と歩行計画とを同時に行う研究は少ないのが現状である。

3. 3D グリッドマップによる環境モデル生成

3Dグリッドマップとは三次元空間を等間隔の格子(グ

原稿受付 2008年1月29日

キーワード: Humanoid Robot, Stereo Vision, 3D Grid Map, Whole Body Locomotion

*〒305-8568 つくば市梅園 1-1-1

*Tsukuba-shi, Ibaraki

Table 1 グリッド属性更新のアルゴリズム

Algorithm 1 UPDATE_GRID(g_R)

```

1:  $g_C \leftarrow \text{Map2Camera}(g_R)$ 
2:  $g_S \leftarrow \text{Camera2Screen}(g_C)$ 
3: if  $\text{InsideOfView}(g_S)$  then
4:   if  $\text{RangeData}(g_S)$  then
5:      $z \leftarrow Z(g_C)$ 
6:      $z' \leftarrow Z(\text{RangeData}(g_S))$ 
7:     if  $|z - z'| \leq s$  then
8:        $\text{label}(g_R) \leftarrow \text{Observed}$ 
9:        $\text{cnt}_o(g_R) \leftarrow \text{cnt}_o(g_R) + 1$ 
10:    else if  $z - z' > s$  then
11:       $\text{label}(g_R) \leftarrow \text{Observed}$ 
12:       $\text{cnt}_v(g_R) \leftarrow \text{cnt}_v(g_R) + 1$ 
13:    end if
14:  end if
15: end if

```

リッド)で区切り,それぞれのグリッドが持つ属性によってその近傍の空間の状態を表す表現形式である.環境モデルの生成は,(1)レンジデータの取得,(2)レンジデータを用いたグリッド属性の更新という二つのステップからなる.

3.1 レンジデータの取得

筆者らは実機としてヒューマノイドロボット HRP-2 [8]を用いており,頭部に搭載されたカメラからの画像に対して再帰相関演算法に基づくステレオ画像処理を行うことでレンジデータを得ている.ステレオビジョンシステムの他,レーザレンジファインダや赤外線 TOF 方式三次元距離測定カメラ等の出力も利用可能である.

3.2 グリッド属性の更新

レンジデータから得られるのは物体が存在する点の集合であるが,これらを空間に並べた Point Cloud のような表現方法では点がないところは何もないのか,物陰になって見えないのか判別できない.そこでグリッドに label , cnt_o , cnt_v と呼ぶ三つの属性を与える. label は *Observed*, *Unknown* の2種類の値を持ち,それぞれ観察済み,状態が不明に対応する. cnt_o はそのグリッドが観察された際に物体が存在すると判定された回数, cnt_v は何もないと判定された回数である.

label , cnt_o , cnt_v の初期値をそれぞれ *Unknown*, 0, 0 とし,レンジデータが生成されるごとに Table 1 に示す UPDATE_GRID() をそれぞれのグリッドに対して適用して各属性を更新する.1行目の $\text{Map2Camera}()$ はマップ座標系におけるグリッドの座標をカメラ座標系における座標に変換する関数であり,2行目の $\text{Camera2Screen}()$ はカメラ座標系におけるグリッドの座標をスクリーン座標系における座標に変換する関数である.また,3行目の $\text{InsideOfView}()$ はスクリーン座標が視野内のものであるかどうかを判別する関数,4行目の $\text{RangeData}()$ はスクリーン座標に対応するレンジデータ(3D座標)を取り出す関数,5行目,6行目の $Z()$ は z 成分を取り出す関数であり, z , z' はカメラ座標系における値である.7行目の s はグリッドの間隔

図2 3Dグリッドマップの生成例(上:入力画像,左下:視差画像,右下:3Dグリッドマップ)

である.

以上の手順によってグリッドの属性を更新した後, *Observed* ラベルの付いたあるグリッド g_R に障害物が存在する可能性 $p(g_R)$ を式(1)によって計算し,これを適当な閾値と比較することで障害物があるかどうかを判別する.このような投票形式を取ることによって,相関演算の誤対応や計測誤差による影響を抑えている.

$$p(g_R) = \frac{\text{cnt}_o(g_R)}{\text{cnt}_o(g_R) + \text{cnt}_v(g_R)} \times 100 \quad (1)$$

3.3 3Dグリッドマップの生成例

3Dグリッドマップの生成例を図2に示す.上段が OpenHRP [9] を用いて生成した入力画像,下段左が視差画像,下段右が3Dグリッドマップである.入力画像のサイズは 616×472 [pixel] である.視差画像は明るい部分が距離が近く,暗い部分が遠いことを意味する.グリッドの間隔は 2 [cm],マッピングを行った空間のサイズは $5 \times 2 \times 1.5$ [m] とした.以上のような条件設定で, Pentium4 3.06 [GHz] を搭載した PC で処理時間は 1.3 [s] であった.またマップが使用するメモリ量はおおよそ 10 [MB] であった.

環境マップを3Dで記述する際の問題点は,メモリとCPUを大量に消費するという点にあり,それらはグリッド数に比例する.グリッドの数はマッピングを行う空間のサイズやグリッドのサイズによって大きく変化するため,アプリケーションによって必要とされる空間のサイズや情報の粒度を考慮しながら設定しなければならない.

4. 3Dグリッドマップを用いた歩行計画**4.1 ステップ情報の決定**

構築した3Dグリッドマップを用いて歩行計画を行う.歩行計画は1歩を定義するステップ情報のシーケンスとして

表現され、ステップ情報は腰の高さ、着地位置・姿勢、単脚支持期・両脚支持期の長さ、上体姿勢からなる。

ここではステップ情報の内、ロボットの高さ z_n 、歩幅 s_n と上体姿勢をローカルな環境情報に基づいて変化させる方法を紹介する (n は n 歩目のパラメータであることを示す。図 3 参照)。

4.2 歩幅と腰高さの決定

歩幅と腰高さの決定は以下の方針に基づいて行う。

- 腰を落としての移動は膝にかかる負担が大きくなり、可動範囲の制約によって歩幅を大きく取ることができないため、極力腰を高くする。
- 目の移動可能な空間のサイズに合わせて 1 歩ごとに腰を上げ下げするのではなく、滑らかな動作となるようにロボット前方のある一定区間の情報を用いてステップ情報を決定する。この空間を歩行計画領域と呼び、その奥行きを d と表す。

高さ優先とするため、まず最初に歩行可能な最小幅 y_{min} で移動可能な領域の高さ $h(x)$ を歩行計画領域の全区間にわたって求める。 $h(x)$ は幅 y_{min} で 3D グリッドマップに内接する長方形の高さである。 $h(x)$ を用いて次式で計算される Δz_{min} を高さの変化率として選択することで高さをスムーズに変化させる。

$$\Delta z_{min} = \min_{x_n < x \leq x_n + d} \frac{h(x) - z_n}{x - x_n} \quad (2)$$

歩幅は腰の高さによって制限を受けるため、次式によって計算する。ここで s_s , z_s はそれぞれ標準歩行姿勢での歩

図 3 ステップ情報の計画

幅、ロボットの高さであり、 k は適当な正の数である。

$$s_{n+1} = s_s + k(z_n - z_s) \quad (3)$$

Δz_{min} と s_{n+1} から次の 1 歩でのロボットの高さは次のように求められる。

$$z_{n+1} = z_n + s_{n+1} \Delta z_{min} \quad (4)$$

4.3 上体姿勢

次に歩行計画領域の全区間にわたって $h(x)$ で通過可能な最大幅 $w(x)$ を求め、 Δz_{min} の場合と同様の方法で Δy_{min} を求める。これを用いて次の 1 歩でのロボットの幅 y_{n+1} を次のように決定し、その値を用いて腰関節および腕の関節の角度を決定する。具体的には腰を捻り、脇を閉めて腕を胴体に密着させることで左右幅を減少させる。

$$y_{n+1} = y_n + s_{n+1} \Delta y_{min} \quad (5)$$

図 4 実験環境の設定 (左) と視差画像 (右)

図 5 生成された 3D グリッドマップ

図 6 実現されたくぐりぬけ動作のスナップショット (単位:[s])

また高さを上下させた場合でも前方の領域観察を極力広い範囲で行うために、腰高さの変化に合わせて視線の制御を行う。

4.4 計画を行う歩数

環境モデルの更新処理は計算量が多いために時間がかかるのに対して、1歩の時間は0.8[s]と短いため、環境モデルが更新されるごとに1歩分の情報しか生成しないと、1歩進むごとに立ち止まることになってしまう。そこで歩行計画開始時に実行されずに残っているステップ情報の数をチェックし、それがあある歩数分以上となるようにステップ情報を追加する。この歩数を処理時間に応じて設定することで、常に歩き続けることが可能となる。

5. 適用事例

HRP-2を用いて、高さに制約のある門状の空間を通り抜ける実験を行った。環境は図4左のような幅1.1[m]、高さ1.35[m]の開口部を持つ環境である。ロボットは壁面から3[m]離れた位置から3Dグリッドマップの更新、歩行計画、歩行動作の実行をオンラインで行いながら前方に5[m]移動する。

この実験ではマッピングの領域を前方2[m]から奥行き方向に5[m]、幅を2[m]、高さを1.7[m]の領域とし、グリッドのサイズは2[cm]とした。またマップの重ね合わせはロボットの前方2.5[m]から4.5[m]の区間に限って行い、歩行計画領域は1[m]の領域とした。図4左に初期位置での視野画像、右に視差画像を示す。また図5に生成された3Dグリッドマップを示す。カメラの画像出力は640×480[pixel]であるが、処理時間短縮のためにこれを320×240[pixel]に縮小し、以降の処理を行った。この場合、体内PC(Pentium M 2[GHz])上でおよそ1[s]周期で処理を行うことができている。

図6に実験時の1.6[s]ごとのスナップショットを示す。このように立ち止まることなく環境を観察しながらそれに応じた移動動作をオンラインで生成・実行する行動が実現できている。

6. まとめ

本稿では知覚情報から環境モデルを構築して歩行計画を行う従来研究を概観し、3Dグリッドマップによって環境をモデル化し、それを用いて歩行計画を行う手法を紹介した。本稿で紹介した手法は非常にプリミティブなものであり、今後自己位置推定機能との統合、レンジセンサの性能を加味したグリッドの属性の更新手法の導入などによって

マップの信頼性を高めることが不可欠である[10][11]。また人間や他のロボットとの共存を実現するには環境の変化にも対応できなければならない。

参考文献

- [1] Y. Kuroki, M. Fujita, T. Ishida, K. Nagasaka and J. Yamaguchi: "A Small Biped Entertainment Robot Exploring Attractive Applications," Proc. of the 2003 IEEE International Conference on Robotics & Automation, pp.471-476, 2003.
- [2] K. Okada, M. Inaba and H. Inoue: "Walking Navigation System of Humanoid Robot using Stereo Vision based Floor Recognition and Path Planning with Multi-Layered Body Image," Proc. of the IEEE/RSJ International Conference on Intelligent Robotics and Systems (IROS'03), pp.2155-2160, 2003.
- [3] 藤本隆司, 西脇光一, 加賀美聡, 稲葉雅幸, 井上博允: "ヒューマノイドにおける狭隘空間通り抜け動作生成法", 日本機械学会ロボティクスメカトロニクス講演会 2003, pp.2P2-1F-F3, 2003.
- [4] S. Kagami, K. Nishiwaki, J.J. Kuffner, K. Okada, M. Inaba and H. Inoue: "Vision-based 2.5D terrain modeling for humanoid locomotion," Proc. of the 2003 IEEE International Conference on Robotics & Automation, pp.2141-2146, 2003.
- [5] O. Lorch, A. Albert, J. Denk, M. Gerecke, R. Cupec, J.F. Seara, W. Gerth and G. Schmidt: "Experiments in Vision-Guided Biped Walking," Proc. of the IEEE/RSJ International Conference on Intelligent Robotics and Systems (IROS'02), pp.2484-2490, 2002.
- [6] K. Okada, T. Ogura, A. Haneda and M. Inaba: "Autonomous 3D walking system for a humanoid robot based on visual step recognition and 3D foot step planner," Proc. of the 2005 IEEE International Conference on Robotics & Automation, pp.625-630, 2005.
- [7] J. Gutmann, M. Fukuchi and M. Fujita: "A Floor and Obstacle Height Map for 3D Navigation of a Humanoid Robot," Proc. of the 2005 IEEE International Conference on Robotics & Automation, pp.1078-1083, 2005.
- [8] 五十棲隆勝, 赤地一彦, 平田勝, 金子健二, 梶田秀司, 比留川博久: "ヒューマノイドロボット HRP-2 の開発", 日本ロボット学会誌, vol.22, no.8, pp.1004-1012, 2004.
- [9] 金広文男, 藤原清司, 梶田秀司, 横井一仁, 金子健二, 比留川博久, 中村仁彦, 山根克: "ヒューマノイドソフトウェアプラットフォーム OpenHRP", 日本ロボット学会誌, vol.21, no.6, pp.609-614, 2003.
- [10] 三浦純, 白井良明: "不確かさを考慮した移動ロボットのための視覚とそのプランニング", 情報処理学会論文誌, vol.44, no.SIG 17 (CVIM 8), pp.37-49, 2003.
- [11] F. Saidi, O. Stasse, K. Yokoi and F. Kanehiro: "Online Object Search with a Humanoid Robot," Proc. of the IEEE/RSJ International Conference on Intelligent Robotics and Systems (IROS'07), pp.1677-1682, 2007.

金広文男 (Fumio Kanehiro)

1999年東京大学大学院工学系研究科情報工学専攻博士課程修了。博士(工学)。1998年より日本学術振興会特別研究員。2000年電子技術総合研究所入所。2007年仏国 LAAS-CNRS 客員研究員。現在産業技術総合研究所知能システム研究部門主任研究員。ヒューマノイドロボットのシステム構成法、全身行動制御に興味がある。IEEEの会員。(日本ロボット学会正会員)