

## 二足歩行ロボット用小型ステレオビジョンと そのアプリケーションの開発

芦ヶ原 隆之† 佐部 浩太郎‡

†ソニー（株）ブロードバンドアプリケーション研究所

‡ソニー（株）デジタルクリーチャーズラボラトリー

二足歩行型エンターテインメントロボット SDR-4X は、家庭環境において自律行動を行えることを目標としている。ロボットの環境認識や HMI(Human Machine Interaction)のために物体・シーンの三次元情報を得ることは極めて有用であるが、加えて本体に搭載するため小型であることやリアルタイム性が重要になってくる。今回、歪曲収差を補正したカラー画像やシーンの奥行き情報である距離画像をリアルタイムに出力する小型ステレオビジョンシステムを開発した。さらに、このビジョンシステムからの情報を用いて簡単な環境認識を行い、それに基づいて歩行経路を導出して実行する一連の技術を開発したので報告する。

### A Stereo Vision System for Humanoid Robot and Its Application

Takayuki Yoshigahara† Kohtaro Sabe‡

† Broadband Applications Laboratories, Sony Corporation

‡ Digital Creatures Laboratory, Sony Corporation

We developed a prototype small humanoid entertainment robot "SDR-4X" that can adapt its performance to its environment and situations found in the home. It is important for a robot to get 3D information of the scene in real-time. The small stereo vision system has been developed, which can create depth image and distortion-less color image in real-time. These images allow the robot to perceive the existence of floor and range between itself and the object to produce a path for collision avoidance.

## 1 はじめに

二足歩行型エンターテインメントロボット SDR-4X (図 1) は家庭環境において自律行動を行えることを目標としている。

従来の車輪型移動ロボットでは、超音波センサなどを使い壁等の遮蔽物からの距離を求めて建物内での位置同定や地図生成を行う技術が提案されてきている。しかしながら歩行ロボットにとっては床面の状態や床面上の物体の有無がより重要な情報となる。そのため SDR-4 では新規に搭載されたステレオビジョンシステムを使って簡単な環境認識を行い、それに基づいて歩行経路を導出して実行する一連の技術を開発した。ここで利用されるステレオビジョンは、外界とのインタラクションやロボットの即時性のためリアルタイム性が重要であり、またロボット頭部に搭載するために小型化を図る必要がある。

本稿では、歪曲収差を補正したカラー画像やシーンの奥行き情報である距離画像をリアルタイムに出力する小型ステレオビジョンシステムと、そ



図 1 : SDR-4X

の情報を用いた障害物回避アプリケーションについて述べる。

まず、ヒューマノイド型エンターテインメントロボット SDR-4X の概要、ハードウェア・ソフトウェアのシステムの構成、そして今回開発したロボット用小型ステレオビジョンシステムについて述べ、その後、画像処理による平面検出とロボットの姿勢から観測点を床面と障害物に切り分ける手法、観測点を 2 次元グリッド地図に投射し、観測モデルと移動モデルから統計的に自己中心障害物地図を更新していく手法、地図をもとに経路を導出し歩行パターンを生成する手法、の 3 つの特徴的な技術に関して解説し、最後に実施例を示す。

## 2 SDR-4X

2002 年 3 月に発表され、ROBODEX2002 において一般公開されたヒューマノイド型エンターテインメントロボット SDR-4X の機能を簡単に紹介する[4]。

SDR-4X では、各関節を駆動する小型アクチュエータの出力性能を向上させるとともに、内蔵された各種センサからの情報に基づいて全身 38 箇所の関節部をリアルタイムで制御する『実時間統合適応制御システム』を新たに開発している。不整地面や傾斜面における二足歩行だけでなく、外部から力が加えられた場合の姿勢保持制御なども可能となり、より高度な運動性能を実現している。

さらに、歩幅や旋回角度など、状況に応じて必要な歩行動作パターンをリアルタイムで生成することで、安定かつ柔軟な歩行が可能となっている。

また、2 つの CCD カラーカメラを搭載しており、カメラの視差を利用して被写体までの距離を検出することが可能になっている。これにより、床面の存在や障害物とロボットとの間隔などを認識し、障害物を避けた経路を自動的に生成して歩行することもできる。画像・音声認識技術や音声生成技術に加えて、記憶に基づく対話や行動の制御技術を採用し、人とのより豊かなコミュニケーションを実現している。CCD カメラから入力された画像から、正面顔を検出し、誰であるかを識別することができる。また、頭部に配置された 7 つのマイクロフォンを使用して、音源方向を検出するとともに、話者識別を行うことが可能である。さらに内蔵ワイヤレス LAN 機能を利用することにより SDR-4X の CPU に加え、外部接続されたコンピュータとの連動も可能となり、大語彙の連続発話の音声認識も実現している。

画像認識で得られた人や物の場所など情報は、短期・長期の記憶情報として利用され、記憶に基づいてより複雑な対話や行動を実現することがで

きる。

さらに、楽譜データや歌詞データを入力することで、音声合成によりビブラートを含む歌声を生成、感情や動作に合わせた歌唱など、エンターテインメント性の向上も図られている。

## 3 システム構成

本章では、小型ステレオカメラシステムのハードウェアの構成と、障害物回避タスクに用いられるソフトウェアの構成について述べる。

### 3.1 ハードウェア構成

ビジョンシステムの構成は、2 台のカメラ、FPGA、SRAM、flash ROM、制御用マイコンからなる。これらの回路はモジュール化され、頭部に搭載されている。距離画像の生成処理はこのモジュール内で行うため、メインの CPU への負荷はない。メイン CPU へは処理結果画像をデジタル YUV ビデオ信号として出力する。また OPEN-R Bus[9]と呼ばれるシリアルバスを介して、メイン CPU との間で演算処理用のパラメータやカメラ制御信号の受け渡しが可能となっている。

カメラには CIF サイズ (352 × 288 画素) の CCD イメージセンサを用いている。ロボットビジョンセンサという用途には小型、低消費電力は言うまでもないが、ゲイン、ホワイトバランスのコントロールを外部から行えることが必須である。また、ステレオビジョンセンサに関してはこれらに加えて 2 つのカメラの外部同期機能が重要になる。このクラスのカメラには CMOS イメージセンサが多くあるが、現状のものはフレーム内の画素間でシャッターの同時性が保証されず、モーションぶれを起こすために動的な環境の認識には向かない。

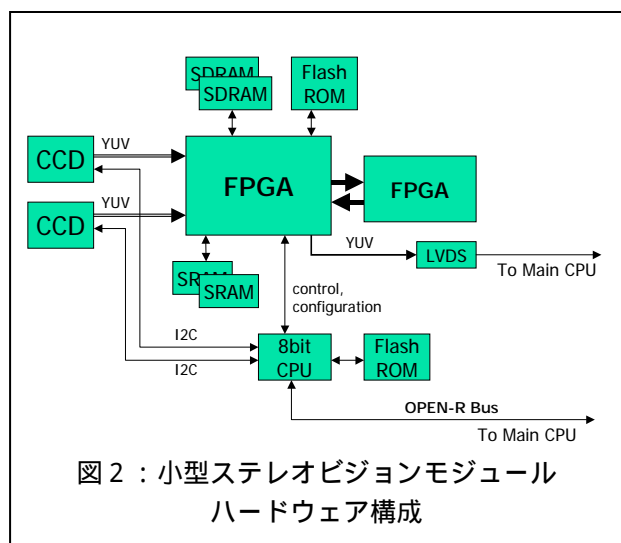


図 2：小型ステレオビジョンモジュール  
ハードウェア構成

### 3.2 ソフトウェア構成

図3が障害物を認識するソフトウェアの構成である。

四角で囲まれたブロックがこのタスクで利用されるソフトウェアオブジェクトで並列実行される。矢印がデータの流れを示す。メッセージ通信によりオブジェクトが invoke されて実行されるようなイベントドリブンフレームワークで構成されている[9]。

以下に主なソフトウェアモジュールの役割を説明する。

#### Kinematics & Odometry

ここでは画像（カラー画像とステレオカメラによる距離画像）とセンサ情報（ロボットの関節角）を入力データとして受け取り、これら2つの情報の時間的対応を取る。つまり画像が撮像された時刻の関節角を探す。次にこの関節角を使ってロボット中心に固定された座標系から頭に付いたカメラの座標系への座標変換を求める。具体的には同次変換行列を導出する。この座標変換行列と対応する画像をそれぞれ画像認識を行うオブジェクトへ送信する。また LocomoEngine で実行された歩行制御による移動量を一步毎に受け取り、ロボット中心座標系での移動量として求める。これをオドメトリと言う。このオドメトリも出力する。

#### PlaneExtractor

このオブジェクトでは、ステレオカメラから得られる視差画像から、キャリブレーションで求めたパラメータを利用して三次元の位置データ（レンジデータ）を算出する。次にこのレンジデータからハフ変換などを使い平面を検出する。また Kinematics&Odometry から来る座標変換行列からロボットの足底が接地している平面との対応を取り、床面を選択する。床面上に無い点を障害物として床面からの距離 OccupancyGridObject へ出力する。

#### OccupancyGridObject

ここでは床面上に障害物があるかないかの確率を保持する OccupancyGrid を PlaneExtractor から得られる障害物の観測結果と Kinematics&Odometry から来るロボットの移動量（オドメトリ）から確率的手法で更新する。この OccupancyGrid はロボットを中心とした周囲4mの障害物情報を保持しており、この認識結果を出力する。

#### SituatedBehaviorLayer

ここでは、与えられた障害物地図情報に基づいて目的地に向かっての安全な経路を計画し、それを実現するための歩行コマンドを出力する。また地図の上の観測状況に応じて視線方向の制御（首の制御）コマンドを独立に生成して出力する。

#### LocomoEngine

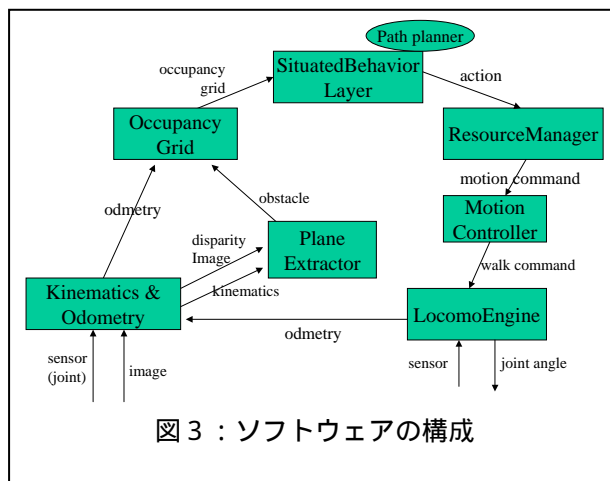


図3：ソフトウェアの構成

このオブジェクトでは歩行コマンドに基づいて、下肢関節の軌道計画と安定化制御を実時間で行う。また脚の運びから求めた実際の移動量を出力する。

## 4 ステレオビジョンシステム

本章では、SDR-4X に搭載されたステレオビジョンシステムについて述べる。

### 4.1 仕様

SDR-4X では、床に置いた状態から立っている人間を見上げながら対話したり、歩行しながら先の方にある障害物を事前に認識して回避するなどの要求が考えられる。SDR-4X の身長は約 60cm なので 1.5m 先の床面や身長 175cm の人間の顔の位置までの距離を考えると少なくとも 2m までは距離の計測ができなくてはならない。逆に近いところでは顔の目の前に物が出現した時にも反応できるように少なくとも 15cm 程度からの計測ができなくてはならない。また、デザイン上の制約から極端にベースラインを長くすることはできない。さらに、本システムはロボット頭部に搭載されるため、回路規模にも制限がある。これらの要求を考慮して、本ステレオビジョンシステムは以下のような仕様になっている。

- カメラ間距離（ベースライン）：5cm
- 距離画像サイズ：QCIF（176×144画素）
- フレームレート：25 fps
- 距離画像画素値ビット幅：8
- 距離画像生成時間：40ms 以内
- 画素値補間：線形
- 距離探索サンプリング数：32
- 距離値補間：パラボラフィッティング
- マッチング関数：正規化相関
- テンプレートサイズ：5×5, 7×7, 9×9, 11×11

## 4.2 カメラキャリブレーション

2台のカメラを、その位置関係を保つのに十分な強度のフレームに固定した後、パターンを描いた平面を両カメラで撮像し、その画像を用いてカメラキャリブレーションを行って、歪曲収差補正に必要なパラメータと距離画像生成に必要なパラメータを推定している。

今回用いた小型 CCD カメラは、携帯電話や PDA に載せて画像を撮ることを主な目的として設計されているので、レンズの歪曲収差が大きく、画質も画像計測にとっては比較的悪い。しかもベースラインが 5cm と短いので、このカメラキャリブレーションでいかに精度よくパラメータを推定できるかがポイントとなる。

### 歪曲収差パラメータ

歪みのない画像平面上の点を  $\mathbf{n}_u = [x_u, y_u]^T$ 、歪みを考慮した画像平面上の点を  $\mathbf{n}_d = [x_d, y_d]^T$  とし、

$$\begin{aligned} x_u &= x_d + (x_d - c_x)\kappa r_d^2 \\ y_u &= y_d + (y_d - c_y)\kappa r_d^2 \end{aligned} \quad (1)$$

というモデルを使って点  $\mathbf{n}_u$  をあらわす[3]。ただし、 $\kappa$  は歪み係数、 $(c_x, c_y)$  は歪み中心、 $r_d$  は歪み中心からの距離を表し、次式で定義される。

$$r_d = \sqrt{\left(\frac{x_d - c_x}{s_x}\right)^2 + (y_d - c_y)^2} \quad (2)$$

ここで、 $s_x$  は画像のアスペクト比である。これらの式より、歪曲収差の影響を取り除いた画像を生成することができる。キャリブレーションでは、白と黒のチェッカーパターンを描いた平面を各カメラで撮像し、その画像の直線であるべきところが直線になるように、歪曲収差パラメータ  $\kappa$ 、 $c_x$ 、 $c_y$ 、 $s_x$  を推定する。

### ステレオパラメータ

視差の算出は、基準画像（左画像）上の画素に対応する画素を検出画像（右画像）のエピポーラライン上の画素から探索することで行っている。このエピポーララインを算出するために、距離が既知である3枚の平面をステレオカメラで撮像し、各々の基準画像から検出画像への射影変換行列を求める。この射影変換行列は、2台のカメラで撮影した平面上のパターンの画像の輝度誤差が最小になるように合わせこむこと（Image Registration）で、高精度のパラメータ推定が可能となっている[1]。また、2枚の平面の射影変換行列を用いる手法があるが[2]、求めた視差と実際の距離との関係をより高精度に求めるためにあえて3枚の平面を利用している。

ここで求めた射影変換行列を用いて、対応点を

探索するためのエピポーラライン上の 32 のサンプル点を算出する。サンプル点はエピポーラライン上で等間隔にとっているため、カメラからの距離が遠いほど、計測距離分解能は粗くなる。表 1 に、カメラからの距離とそこでの距離分解能の対応を示す。実際に本ステレオシステムで求められたパラメータから算出したもので、画像の中心の画素での数値となっている。これによれば、2m 付近の遠いところではたとえば人物を切り出す程度のが可能で、カメラから 30cm 付近の近いところでは細かな形状が得られることがわかる。これはロボットの環境認識や HMI に際に適した分解能であると言えるだろう。

距離 Z[mm]	Z[mm]	距離 Z[mm]	Z[mm]
300	1.4	1500	34.8
500	3.8	1700	45.2
700	7.4	1900	56.8
900	12.2	2100	68.3
1100	18.6	2300	83.6
1300	26.1	2500	98.1

表 1：カメラからの距離と距離分解能

## 4.3 対応点探索

エピポーラライン上の対応点探索は area-based マッチングにより行われ、類似度を示す関数には正規化相関を用いている。上記射影変換行列から求められるエピポーラライン上の 32 (5 ビット) のサンプル点についてスコアを求め、最大のスコアを持つ点の前後のスコアに 2 次曲線を当てはめ、ピーク位置の下位 3 ビットの補間を行っている。

実装上は、画像の各画素について 32 のサンプル点でのマッチングスコアを順次求めるのではなく、各サンプル点について画像全体の画素におけるスコアを求めていくことで、計算コストの軽減を図っている[1]。

## 4.4 信頼度画像

area-based マッチングによる対応点探索では、テクスチャの無い領域や細かい繰り返しテクスチャ領域ではマッチングの誤対応を起こしやすい。このために上記のような状況を取り除くために以下のような指標を使って計測の信頼性を表す。

ブロックマッチングはエピポーラ線上のサンプル点に対して行われ、そのスコア（正規化相関）のピーク点近傍で補間演算される。その補間によって求められたピーク点と近傍サンプル点からピークの急峻度を求めることで信頼度とする。つまりいくらマッチングのスコアが高くても近傍点と似通ったスコアである場合には前述した状況である可能性が高く、求めた視差の信頼性が低いと考えることができる。

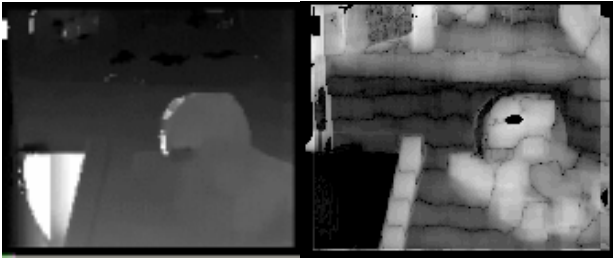


図4：視差画像と信頼度画像

この信頼度は視差計測の各点に対して算出するために視差画像に対応する信頼度画像として出力される。図4の上がカラー画像，左下が視差画像，右下が信頼度画像の例である。

信頼度の画像の暗い部分が信頼度の低い画素で明るい部分が信頼度の高い部分である。左下の白い紙が置いてある部分は信頼度が低いのにに対して、茶色い模様が入っている床面は画面手前のテクスチャが良く見える部分ほど信頼度が高くなっていることがわかる。

#### 4.5 出力画像

本ステレオシステムからの出力の画像形式は選択可能となっており、カラー画像（歪曲収差補正前と後）、距離画像、距離信頼度画像、フレーム間差分画像などの処理結果の中から選べる。選択はフレーム内で3つの画像スロット（通常のビデオ信号のY,U,Vに相当）に対して行うことができる。また2種類のフレームを構成して、出力頻度の比を設定することもできるので最大で6種類の画像を得られる。実際に使用されている例としては、同じ時刻に撮像された画像に対するYUVのカラー画像と差分、視差、信頼度画像を交互のフレームで出力するように設定している。

このように同時刻のカラー画像と距離画像が利用可能であり、ロボットのタスクは距離画像を使った環境認識だけではないので、両方の画像を得られることで距離画像とカラー画像のセンサフュージョンなどへの応用が実現しやすくなる。

### 5 平面検出

#### 5.1 視差 - レンジデータ変換

距離画像から床面を検出するためにはまず距離

情報から三次元の位置情報（レンジデータ）へ変換する必要がある。これは、カメラキャリブレーションの際にこの変換に必要なパラメタを求めておき、このパラメタを使ってレンジデータを算出する。

#### 5.2 Hough 変換

多数の計測点からのパラメタの抽出などの問題には Hough 変換がよく用いられる。Hough 変換の解法には順写像解法、逆写像解法、特徴点对による解法など多くの手法が提案されているが、ここではデータの量とパラメタの次元からメモリ、演算的に有利な Randomized Hough 変換を利用する。Randomized Hough 変換を使った平面検出は岡田ら[10]が提案している手法があるが、本稿で用いた手法では以下の点で特徴を持っている。

- 平面上の3点をすべてランダムに選択することでより早くデータ全体の統計量を反映させることができる。
- 投票するパラメタ空間を単純な $(\theta, d)$ のグリッドではなく $(\theta, \cos \theta, d)$ で構成することで公平な投票が行われる。
- 信頼度による重み付け投票で誤差に強い推定
- 重み付き平均によるピーク推定の精度向上
- 多重解像度視差画像の利用と投票打ち切りによる高速化

図5に平面検出の全体の処理手順を示す。

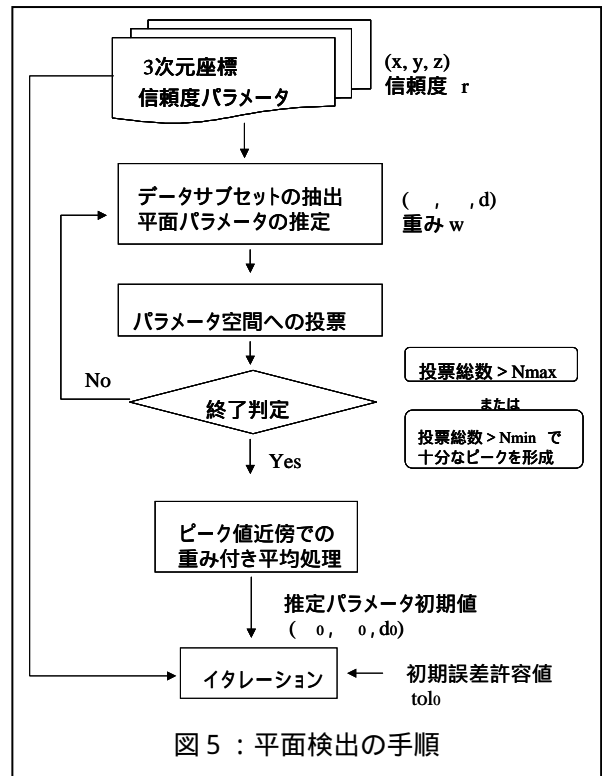
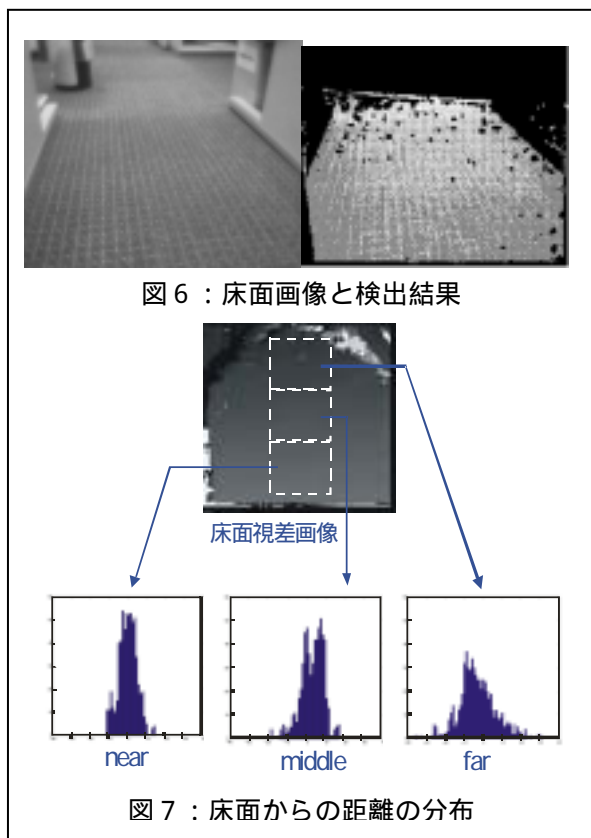


図5：平面検出の手順

### 5.3 結果

前述のアルゴリズムを適用して処理した結果例を示す．図 6 の左側は SDR-4 が立位状態で首のピッチを 30° 下に向けて取った左眼画像である．右側が検出平面からの距離が ±10mm 以内に収まる画素を左眼画像から抽出した結果である．奥行き 200cm 以遠の部分の結果は切り捨てている．遠方で切り出しに失敗している画素があるがほぼ良好に検出に成功している．

図 7 は検出された床面のパラメタから実際に計測された画素の 3 次元位置から検出された床面までの距離のヒストグラムを取ったものである．左から足元 30cm ~ 50cm, 50cm ~ 100cm, 100cm 以遠の 3 種類の分布を示している．グラフ中の一目盛りが 10mm を表しているので足元近くではほぼ ±10mm の精度で推定が可能であることがわかる．100cm 以遠の計測結果に対しても ±30mm の範囲で収まっている．



## 6 環境地図の保持と更新

前章で解説した平面検出によりカメラの撮像範囲内にある障害物もしくは歩行可能領域を把握することができる．しかしながらカメラの画角は広くはないのでこの情報だけを使って行動計画するには情報量は十分ではない．また人間は視野角範囲外でも以前観測した結果を短期的に記憶して周

囲の状況として把握することができる．そこで測定結果を自己中心の座標系にて表現し、これらの測定結果の累積をエゴモーションによって更新していく環境地図を導入する．

環境地図はロボットを中心とする床面上の 2 次元グリッドとして表現する (OccupancyGrid)．各グリッドは、床面である確率、障害物がある確率を保持している．

### 6.1 観測モデル

床面と障害物の認識結果 (観測) から環境地図上の確率を更新する方法については、Wijk[5]によって様々に手法の比較が為されている．Bayes の更新則、Dempster Shafer の更新則、Borenstein[8]の手法を実際に実装し比較を行ったところ、Bayes と Dempster Shafer の方法ではほぼ同じ結果が出た．Borenstein の手法は動く障害物があるような動的な環境に向いている手法である．ここでは最も一般的な Bayes の手法を用いて観測による確率の更新を行う．

### 6.2 モーションモデル

ロボットの姿勢の変化、移動に伴ってこの環境地図を更新するが、ロボットの移動量がグリッドの大きさよりも小さいときはグリッドの更新はせずにグリッド内のロボットの移動位置だけを保持 (図 8 (a)) し、移動量がグリッドよりも大きい場合には移動したグリッドの数だけ環境地図をシフト移動させる (図 8 (b))．つまり、障害物が表現される最小単位であるグリッドの大きさを CS (Cell Size)、グリッド内のロボットの位置を (Bx, By)、ロボットの 2 次元方向の移動量を (dx, dy) とすると、グリッドのシフトの大きさ (Sx, Sy) は式 3 のようになる．

$$(Sx, Sy) = \left( \left\lfloor \frac{Bx + dx}{CS} \right\rfloor, \left\lfloor \frac{By + dy}{CS} \right\rfloor \right) \quad (3)$$

ここで、括弧 "⌊" は、内部の値の小数点以下を切り捨てた値をあらわすものとする．また、グリッド内のロボットの位置 (Rx, Ry) は、式 4 のようになる．

$$(Rx, Ry) = (dx - CS \times Sx, dy - CS \times Sy) \quad (4)$$

ロボットの姿勢情報に関しては、環境地図を環境に固定された座標系に対する角度として表現する．つまり、ロボットが姿勢方向 d だけを変化させた場合は、ロボットの向きのみが更新され、環境地図は更新されない (図 9)．

これらの環境地図更新は、ロボットが 1 歩歩くごとに行うようにすることで、移動していないときに更新することによる計算コストを押しさえることができる．

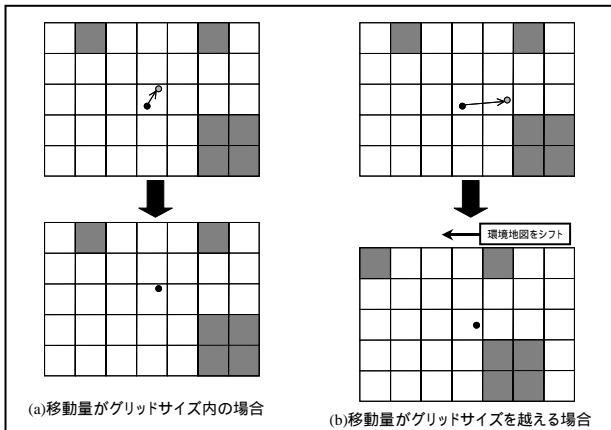


図 8 : 環境地図の更新方法 (並進移動成分)

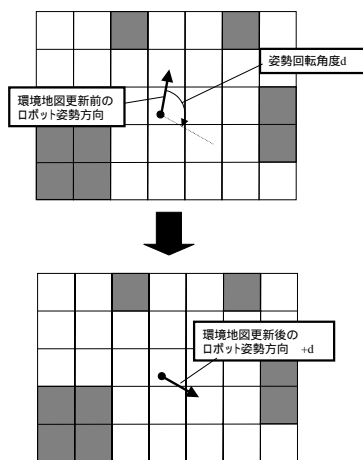


図 9 : 環境地図の更新方法 (回転成分)

## 7 歩行経路計画

自己の周囲の状況を反映した環境地図を利用することで安全に移動する経路を算出することができる。

経路計画問題を、環境地図の各グリッドをノードとする探索の問題に帰着し、 $A^*$ 探索により、最適経路を求めることができる。探索における評価関数として、環境地図における障害物の存在確率によるポテンシャル場を用いる。ノードにおけるポテンシャル場は、障害物の存在確率  $P(n)$   $[0,1]$ を用いて、障害物からの斥力ポテンシャルとして次式で求める。

$$U(n) = \begin{cases} \max\left(\frac{d_0}{d(n)} \cdot P(n')\right) & d_0 < d(n) \\ P(n') & d(n) < d_0 \end{cases} \quad (5)$$

$d(n)$  :  $n$  から  $n'$  までの距離。  $d_0$  : 障害物にロボットが接触しないための安全距離。

$A^*$ 探索における、評価関数として次式を用

いた。

$$f(n) = g(n) + \alpha \cdot h(n) \quad (6)$$

ここで、 $g(n)$  : スタートノード  $n_s$  から現在のノード  $n$  までの経路コスト、 $h(n)$  : 現在のノード  $n$  からゴールノード  $n_g$  までの経路の見積もりコスト、 $\alpha$  : 重み係数である。従って、 $f(n)$  は  $n$  経由の最適解の見積もりコストとなり、最小の  $f$  を持つノード  $n$  を展開していくことによって、最適解を求めることができる。

$$g(n) = \sum_{n_s}^n U(n) + L(n_s, n) \quad (7)$$

$$h(n) = \hat{L}(n_g, n) = \|n_g - n\| \quad (8)$$

$L(n_s, n)$  :  $n_s$  から  $n$  までの経路長。  
 $\hat{L}(n_g, n)$  :  $n$  から  $n_g$  までの最短経路長。

求められた経路の接線方向に一定の歩幅となるように歩行コマンドを継ぎ足していくことで止まらない滑らかな歩行を実現している。

## 8 実施例

図 11 が ROBODEX2002 においてステージ上に置かれた障害物を避けながら歩行をしている様子である。デモでは左端から右端を目指して歩行している。図 10 は実際に観測された OccupancyGrid で、下の図 11 の写真にだいたい対応する位置での観測となっている。但し進行方向が上側となっている。環境地図の大きさは  $4m \times 4m$  で各グリッドの大きさを  $40mm \times 40mm$  としているのでグリッドの数は  $100 \times 100$  個ある。白い部分が高い確率で床面であるグリッドで、黒い部分が高い確率で床面でないグリッドである。灰色は確率 0.5 に近い部分で、青色部分は十分に観測されていない部分である。OccupancyGrid の中心にある黒い塊が SDR-4 の頭の形状を表しており、SDR-4 の向きに合わせて回転している。

ステージの大きさは  $1m \times 2m$  であるが OccupancyGrid からステージの形状と大きさとその上の障害物の位置関係が伺え、良好に周囲の環境が認識されていることがわかる。

## 9 おわりに

小型ステレオビジョンシステム、平面検出を用いた路面と障害物の認識、観測と移動に基づいた環境地図の生成、環境地図上の経路計画と歩行コマンドの生成という一連の技術開発と二足歩行ロボット上への実装を行い、家庭環境下において自

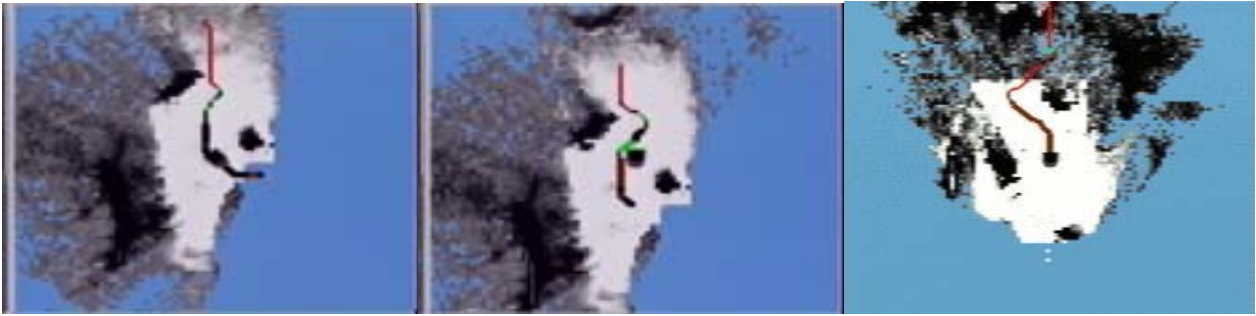


図 10 : 環境地図の更新例



図 11 : 障害物を回避しながらの歩行

律的に移動できるような機能を達成した。この機能はロボットが環境を自己中心座標系にて捉えることで環境に対する前提知識をまったく持つ必要がないという利点を持っている。

今後は、ステレオビジョンに限らず、認識に必要な他の機能を取り込んで、アプリケーション側からより使いやすいビジョンシステムとしていく必要がある。また、ロボットにグローバルな環境地図を持たせ、その上での位置を認識することで目的をもった行動（移動）を実現していく。

## 参考文献

- [1] 横山, 三輪, 芦ヶ原, 林, 小柳津, 後, “3次元映像の入力技術”, 映像情報メディア学会誌, Vol.54, No.3, pp.328-331, 2000.
- [2] 蚊野, 金出, “任意カメラ配置におけるステレオ視とステレオカメラ校正”, 信学論, Vol.J79-D-II, No.11, pp1810-1818, 1996.
- [3] R.Tsai, “An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision”, Proc. of CVPR, pp.364-374, 1986.
- [4] ソニーWeb サイト プレスリリース  
<http://www.sony.co.jp/SonyInfo/News/Press/200203/02-0319>
- [5] Olle Wijk, “Triangulation Based Fusion of Sonar Data with Application in Mobile Robot Mapping and Localization”, PhD thesis, Royal Institute of Technology, Stockholm, Sweden, 2001.
- [6] A. Elfes, “A Probabilistic Framework for Robot Perception and Navigation”, PhD thesis, Carnegie-Mellon University, 1989.

[7] D. Pagac, E. Nebot, and H. Durrant-Whyte, “An evidential approach to map-building for autonomous vehicles”, IEEE Transactions on Robotics and Automation, 14(4), 623-629, 1998.

[8] J. Borenstein and Y. Koren, “Error eliminating rapid ultrasonic firing for mobile robot obstacle avoidance”, IEEE Transactions on Robotics and Automation, II(1), 132-138, 1991.

[9] 佐部, 藤田, “エンターテインメントロボットシステムソフトウェア”, 第19回日本ロボット学会学術講演会予稿集, 2001.

[10] 岡田, 冬野, 加賀美, 稲葉, 井上, “プレーンセグメントファインダ: ハフ変換を用いた実時間平面検出器”, 第18回日本ロボット学会学術講演会予稿集, 2001

[11] 佐部, 藤田, “エンターテインメントロボットの商品化”, 日本ロボット学会誌 Vol.18 No.2, 2000.

[12] 佐部, 大橋, 河本, グットマン, 福地, 芦ヶ原, “二足歩行ロボットによるステレオ画像を用いた障害物回避と歩行計画”, 第8回画像センシングシンポジウム講演論文集, pp.237-242, 2002.