

# 全方位カメラによる自己位置推定のための特徴点追跡

猿渡 章太郎\* 胡振程† Thomas Feraud‡

\*熊本大学工学部 〒860-8555 熊本市中央区黒髪2丁目39番1号

†熊本大学工学部 〒860-8555 熊本市中央区黒髪2丁目39番1号

‡Institut Pascal, Blaise Pascal University, France

E-mail: shotaro@its.cs.kumamoto-u.ac.jp hu@cs.kumamoto-u.ac.jp

あらまし SLAMとは自己位置を計測しながら障害物マップを作ることであり、ロボットや自律走行分野に広く応用されてきた。従来ではジャイロセンサーや車速センサーを等の手法が主流であるが、タイヤの空回りやスリップに対応できないため、近年画像処理に基づいた Visual SLAM の手法が注目を集めている。

本研究では、複数台のカメラを接続した全方位カメラから取得した画像を用いた Visual SLAM を提案する。この手法は単眼カメラより範囲が広く、曲面ミラーを用いた全方位カメラより使える部分が多い画像を用いることでより正確な自己位置の計測を可能にしている。

キーワード 全方位カメラ, Visual SLAM, カルマンフィルタ

## Tracking of Feature Points for Visual SLAM with Multiple Cameras

Shotaro Saruwatari\* Zhencheng Hu† and Thomas Feraud‡ (IP)

\*Faculty of Engineering, Kumamoto University 2-39-1 Kurokami, Chuo-ku, Kumamoto, 860-8555 Japan

†Faculty of Engineering, Kumamoto University 2-39-1 Kurokami, Chuo-ku, Kumamoto, 860-8555 Japan

‡Institut Pascal, Blaise Pascal University, France

**Abstract** SLAM is method of measuring vehicle's position and making obstacle map. It used for robotic and Autonomous travel. Formerly, SLAM was based on vehicle speed sensor or gyro sensor but they can't cope idle or slip. So Visual SLAM based on image process has lately attention.

This paper suggest Visual SLAM using connect cameras. It has wider visibility than single camera and higher resolution than omni camera.

**Keyword** Omni camera, Visual SLAM, Kalman filter

### 1. はじめに

SLAMとは自分の位置を計測しながら障害物マップを作ることであり、近年画像処理に基づいたSLAMの手法が注目を集めている。地理情報を用いない自己位置推定の方法、タイヤの回転数やハンドルがきられた角度を用いて車両がどのように移動したかを求める所謂自律走行は、カーナビやロボット計測等の分野でよく使われている。しかしこの方法だとタイヤがスリップ、空回りした時に実際の軌道と求めた移動距離がずれていってしまう。そこで、車両から画像を取りその画像から現在の車両の位置を推定する方法が提案されている。

従来は単眼カメラや曲面ミラーを用いた全方位カメラを用

いて行っていたが[1][2], 1方向カメラでは得られる画像の範囲が狭い、曲面ミラーを用いた全方位カメラには歪みが大きく、画像に用いれない部分があるなどの欠点があった。

それらの欠点を解消するため、各方向を向いた複数のカメラを接続し、全方位カメラを用いる方法がある[3]。この研究は、車両の全方向の画像を取ることで撮影できる範囲の狭い単カメラや歪みが大きく画像として使えない部分がある曲面ミラーを用いた全方位カメラより正確に画像から車両の位置を推定することを目的としている。

### 2. 従事研究

Visual SLAMは車から撮影した画像から特徴点を検出し、

他の時刻に撮った画像と対応付けすることで車や特徴点の三次元位置を計測する方法である (図1)。

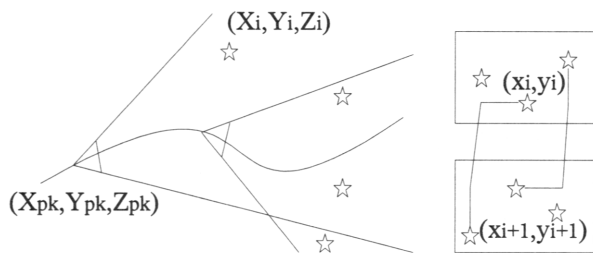
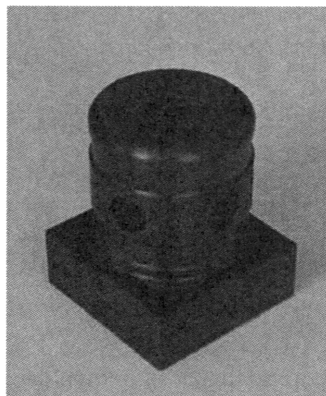


図1 Visual SLAM

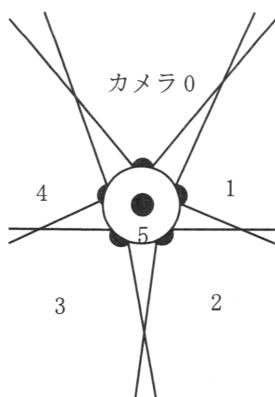
この実験では全方向の画像を取るために各方向に向けて設置された6つのカメラを接続して作られた全方位カメラを使用する。

図2(a)は今回使用したカメラである。図のように円筒の側面に等間隔に5つ、頂点に1つのカメラが付いている。側面のカメラには右回りにカメラ0, 1, 2, 3, 4, 頂点のカメラにはカメラ5の名称が付いており、ここでは、カメラ0を正面として実験を行った。カメラ0~5の撮影する範囲を図2(b)に、それぞれのカメラで撮影される画像の一例を図2(c), 図2(d), 図2(e), 図2(f), 図2(g), 図2(h)に示す。

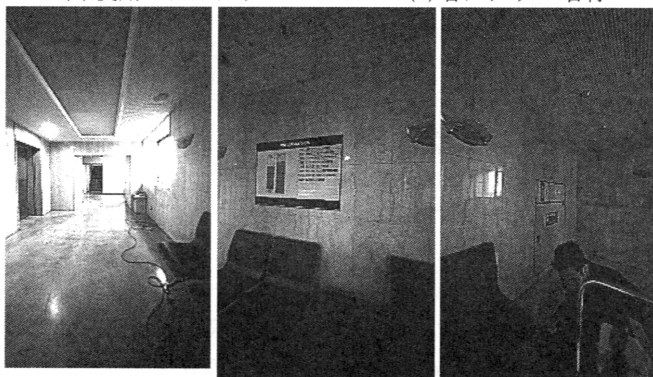
またこのカメラにはそれぞれのカメラで撮影した画像を合成してパノラマビューを作る機能も付いているが、これはピンホールカメラモデルではないため使用しない。



(a) 使用したカメラ



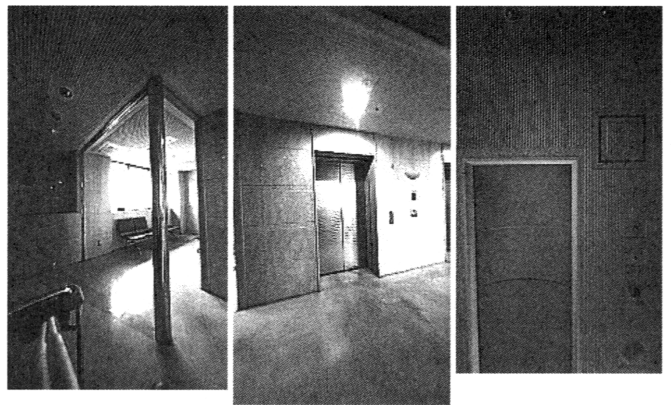
(b) 各カメラの名称



(c) カメラ0

(d) カメラ1

(e) カメラ2



(f) カメラ3

(g) カメラ4

(h) カメラ5



(i) パノラマビュー

図2 全方位カメラ Lady Bug

### 3. 提案手法

#### 3.1 特徴点の検出とマッチング

まず現在のカメラの位置と1単位時間後のカメラの位置から画像を取る。次に2つの画像から特徴点を検出し、対応付けをする。

特徴点の検出にはハリス検出器を使用した。また特徴点の対応付けは二つの特徴点周りの画像をテンプレートとして正規化相互相関 (ZNCC) をとることで類似度を評価、類似度が一定の値を超えた場合、対応付けするという方法を取った。

ここで対応付けされた特徴点を同じ物体とし、物体番号をつけることでどの物体が何枚目の画像の座標  $(x, y)$  に映っている、というデータを取った。このデータはカルマンフィルタで現在地を予測するとき使用する。

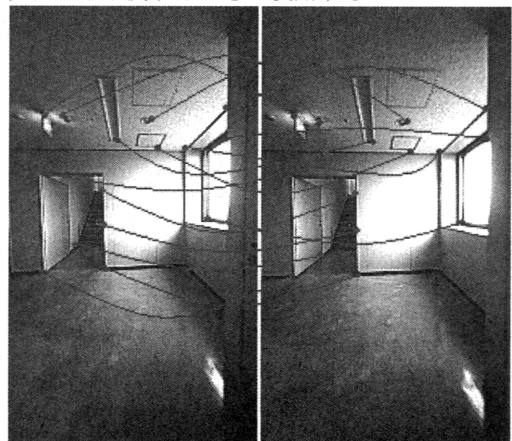


図3 特徴点の対応付けの例

図3は特徴点の対応付けの例である。線がつながっている点同士が対応付けされている。

### 3.2 カルマンフィルタと車両位置の予測

カルマンフィルタとは誤差のある観測から時間と共に変化する量を推定するために使われるフィルタである。ここでは式(1)を用いて画像の特徴点、カメラパラメータからカメラの位置を予測するために用いる。

$X_{vi}$ を時刻*i*で予測した自己位置、 $X_{pi}$ を時刻*i*で予測した障害物の位置、 $P_i$ を時刻*i*での誤差の共分散行列、 $\Delta$ を予測した値と観測値の誤差、 $h=(u,v)$ を画像上の特徴点の座標、 $R$ を観測誤差として、式(2)で自己位置の予測と予測の更新ができる。

$$\begin{aligned} X_{i+1} &= X_i + K \Delta \\ P_{i+1} &= P_i - KHP_i \end{aligned} \quad (2)$$

$K$ は出力の誤差を状態に変換する役割を持つカルマンゲインであり、式(3)で求められる。

$$K = PH^T(HPH^T + R)^{-1} \quad (3)$$

また $H$ は状態を観測に変換する観測モデルであり、式(4)のように観測量 $h$ をカメラと障害物の状態 $X_{vi}$ 、 $X_{pi}$ で偏微分することで求められる。

$$\begin{aligned} X &= \begin{pmatrix} X_{vi} \\ X_{pi} \end{pmatrix} \\ H &= \frac{\partial h}{\partial X} \end{aligned} \quad (4)$$

### 3.3 使用カメラの決定

この実験ではスタート地点から直進し、90度カーブ、再び直進した後Uターンし、最初に曲がったカーブの地点まで直進し再び90度カーブ、スタート地点に戻ってくる、という軌道で画像を取り、カルマンフィルタにかけた。

図4はUターン時と最後のカーブ時に撮影された画像の内の1つであるが、(a)を見るとカメラ0はUターンする際壁の方を向き、そこで特徴点が見つからない画像を取ってしまったことが分かる。

また図5(a)を見ると窓からの光が逆光になってしまい、特徴点が写りにくくなってしまっている。

これにより特徴点がない部分が出てきてしまい、軌道が歪んでしまっていると考えられる。そこで、カメラ0では特徴点がない時、他のカメラから画像と特徴点を取り、それをフィルタにかける方法を用いる。

どのカメラで撮った画像をカルマンフィルタにかけるかの判断のために、最初のカーブ、Uターン、最後のカーブ周辺における画像の一枚当たりの特徴点の数を表1に示す。

表1 画像一枚当たりの特徴点の数

	カメラ0	カメラ1	カメラ2	カメラ3	カメラ4	カメラ5
カーブ(行)	15.89	10.75	12.67	12.31	10.92	13.53
Uターン	10.86	9.33	9.3	11.67	15	7.71
カーブ(帰)	9.53	14.32	10.56	17.78	12.44	11.67

この表からカーブ付近では正面のカメラ0、Uターン付近ではカメラ0の左隣のカメラ4、帰りのカーブではカメラ0の左後ろのカメラ3が多く特徴点を取れていることが分かる。

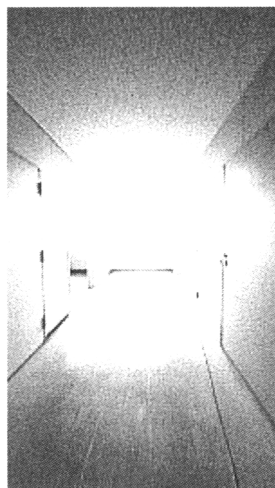
ここでは特徴点の多さのほかに画像を高さと幅の半分で四分割し、一番少ない部位でも特徴点が平均1つ以上あることをカメラの使用条件とした。



(a)カメラ0

(b)カメラ4

図4 Uターン時画像



(a)カメラ0

(b)カメラ3

図5 最後のカーブ時画像

図3(b)や図4(b)を見ると正面のカメラ0ではあまり特徴点

が取れていない場所でも他のカメラでは特徴点が取れそうな場所を写していることが分かる。

#### 4. 実験

今回はタイヤが空回り，スリップしないような綺麗な床面の廊下で実験を行った。台車に全方位カメラを載せ，図6のようなルートを通った。図7は車速センサーとジャイロセンサーにより取れた軌道であり，これを基準とする。

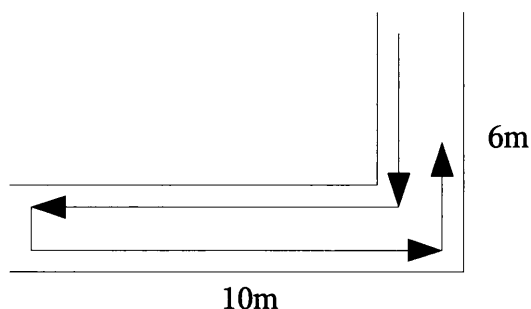


図6 実験ルート

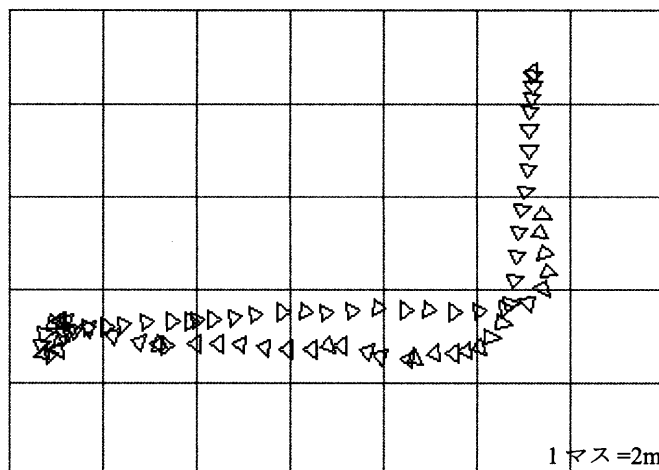


図7 車速センサーとジャイロセンサーで取れた軌道

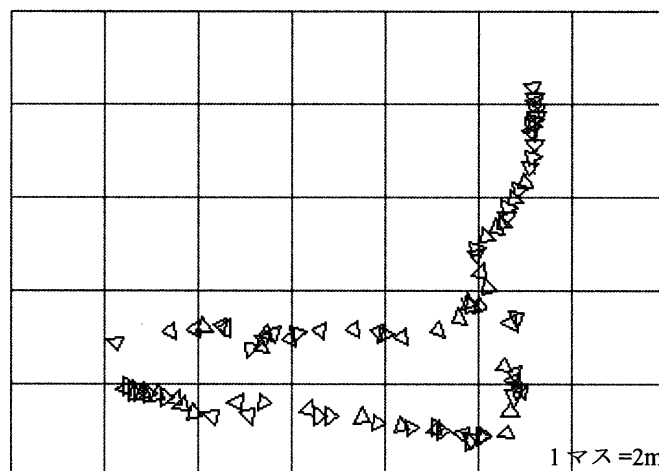


図8 カメラ0のみの画像から予測された軌道

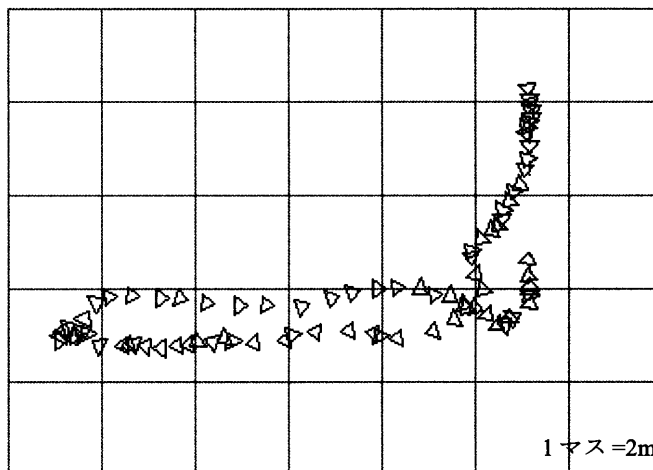


図9 今回の提案手法で予測された軌道

図8は正面のカメラのみで画像取得を行い，そこから軌道予測をした結果である。最初直進し90°曲がったということは予測できているものの，Uターン時に図4(a)のような特徴点のない壁を取ってしまったり帰りは窓からの逆光で特徴点が取れなかったりとなってしまうため途中から軌道が大きく歪んでしまっていることが分かる。

一方図9は全方位カメラを用いた方法で予測したルートである。単カメラでは大きく歪んでしまっていたUターン，帰りのカーブでも特徴点が映っている画像を取得できているため，Uターンもカーブも予測できスタート地点に戻ってこれていることが分かる。

#### 5. まとめ

全てのカメラの中から特徴点を多く写しているカメラを利用することで程度綺麗な軌道を取れるようになったが，より全方位カメラの利点を活かすには全方位への特徴点追跡，つまりカメラ間の特徴点の受け渡しができるようになることが望ましい。そのためには撮影された角度が違うカメラ間での特徴点の対応付けの方法を考える必要がある。

また，カメラが5台ある分画像も通常の5倍処理しなければならず，時間がかかる。より処理を高速化するためにはプロセスの更なる改良が必要である。

#### 参考文献

- [1]Oscar Martinez Mozos, Arturo Gil, Monica Ballesta, Oscar Reinoso, Interest Point Detectors for Visual SLAM, 2007
- [2]Teruko Yata, Chuck Thorpe, Static Environment Recognition Using Omni-camera from a Moving Vehicle, 2002
- [3]Sei IKEDA, Tomokazu SATO and Naokazu YOKOYA, High-resolution Panoramic Movie Generation from Video Streams Acquired by an Omnidirectional Multi-camera System