

# 建築物構造を利用した地図作成による自己位置認識の精度向上

○木村海渡（東京電機大学） 日高浩一（東京電機大学）

## 1. はじめに

人口減少による労働力不足や宅配サービスの増加などにより、人との協働が可能なロボットの実用化が要求されている。この要求に対していち早く対応ができている分野に組み立て作業の自動化に利用されている産業用ロボットがある。一方で労働力不足から自動化の要求が強い運搬分野や施工分野の自動化用のロボット技術は、実環境に対応が難しい技術が多数あり、実用化がまだできていない分野である [1]。これらの分野ではロボットの移動は重要な一要素技術である。移動では自己位置および環境状況認識は重要技術であり、Simultaneous Localization and Mapping (SLAM) 技術が開発され、自動運転などで利用され成果を上げている [2, 3]。使用される SLAM は Lidar や RGB-D カメラによる観測データに基づく Visual-SLAM がある。ORB-SLAM[4] や MONO-SLAM[5] などの Visual-SLAM では、環境情報の特徴点を利用される。そのため廊下や壁など特徴変化が少ない環境下では、特徴点の取得が少なくなり精度の低下が起こっていた。そこで特徴点だけでなく環境から取得可能な線も SLAM のランドマークとして使用する研究が提案されている [6, 7]。しかし [6] の Line-SLAM のような線特徴のみをランドマークに使用したとき精度低下が起こる [7]。そこで [7] の Struct-SLAM では一般的な人工建築物は長方形の組み合わせで表現可能であり、柱や床と壁の境目などの建築物の構造から取得される規則的に並んだ線のみを線ランドマークとして使用する事で誤差の少ない SLAM が提案されている。しかし内部構造を環境情報の制限にのみ使用し loop-Closer など、地図の精度変化には使用されていない。そこで我々は建築物の内部構造と消失点を利用して環境位置情報を修正し SLAM の精度向上を行う方法を提案している。本発表では、提案する方法のベースとなる Struct-SLAM のアルゴリズムについて解説し、改良研究の提案を行う。

## 2. Struct SLAM の概要

SLAM では、センサの初期位置と姿勢を決定し、センサの移動を行いながら自己位置を推定後自己位置を利用して環境位置推定を行う処理を繰り返すことで環境地図と移動位置を推定するアルゴリズムである。従来の SLAM で利用される環境情報は点群データであるため、室内ではカメラをセンサに使用する Visual SLAM では特徴点取得が難しい環境が存在する。Struct SLAM では環境中の構造線データを取得することで精度向上を行う。この章では [5, 4] に基づき Struct-SLAM の概要を示すとともに、問題点を指摘し提案する改良法について説明する。

### 2.1 Struct SLAM の環境情報

Struct SLAM は環境の特徴点と、構造線と呼ばれる柱や床と壁の境目などの建築物の構造から取得される規則的に並んだ線を使用して自己位置推定を行う。構造線の規則性から方向を 1 つに限定する事ができ、限定された方向によりセンサの姿勢が固定できることで姿勢による誤差を減少しより正確な自己位置推定が可能となる。また特徴点では認識が難しい壁や通路などの環境で構造線を使用することで有効的に環境情報を取得する事が可能となる。構造線の取得は、図 1 に示す消失点が建築物の構造線延長上の交点である性質より、消失点を求める事で構造線の方向を推定する。

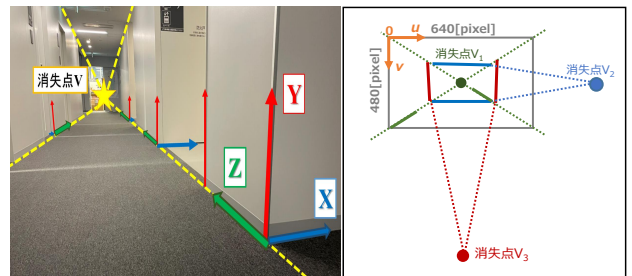


図 1 左：廊下での構造線と消失点，右：構造線と消失点の関係

### 2.2 自己位置推定および環境位置推定

カメラが搭載されているロボットを対象としているため、ロボットとカメラは移動および姿勢が同じとなる。このシステムで環境計測をするとき、カメラの動的モデルおよび各特徴点と構造線から構成される観測モデルを拡張カルマンフィルタを用いて更新し自己位置を推定する。カメラの動的モデルは Mono-SLAM で使用されている 6DoF を持つ動的モデルと同様なモデルを使用する [5]。動的モデルの状態ベクトル  $x_c$  を、カメラに関する世界座標での位置  $p_w$ 、姿勢  $q^{wc}$ 、世界座標でのカメラ速度  $v^w$  およびカメラ座標系でのロボットの角速度  $\omega^c$  で与える。観測値は特徴点の状態ベクトル  $X_p$  および構造線の状態ベクトル  $X_l$  とする。 $X_p$  はカメラ座標系での三次元位置  $a_i = [x_c, y_c, z_c]^T$  であり、 $X_l$  は図 2 に示すように、世界座標でのカメラの位置、カメラまでの距離および角度である  $l_{t1} = [x_w, z_w, h, \theta]^T$  で設定した。以上より状態、観測式が (1) 式で与えられる。ここで  $l$  は観測個数を表し、観測時間ごとに異なる値となり、(1) 式の数  $v_t^w$  と角速度  $\omega_t^c$  には外乱が含まれる。

$$f_c(x_c) : \begin{bmatrix} \hat{p}_{t+1}^w \\ \hat{q}_{t+1}^{wc} \\ \hat{v}_{t+1}^w \\ \hat{\omega}_{t+1}^c \end{bmatrix} = \begin{bmatrix} p_t^w + v_t^w \Delta t \\ q_t^{wc} \cdot q(\omega_t^c) \Delta t \\ v_t^w \\ \omega_t^c \end{bmatrix} \quad (1)$$

$$X_{tp} = \begin{bmatrix} a_{t1} \\ a_{t2} \\ \vdots \\ a_{tl} \end{bmatrix}, X_{tl} = \begin{bmatrix} l_{t1} \\ l_{t2} \\ \vdots \\ l_{tl} \end{bmatrix} \quad (2)$$

測定関数はカメラから観測された特徴点の三次元位置  $p(x)$  と線分の測定関数  $m(x)$  を使用する。このとき線分の測定関数は図2のような直前の動作で観測された構造線との差を使用すると以下で与えることができる。

$$m_{ij}(x) = \begin{bmatrix} \frac{s_j^a \cdot \bar{l}_i}{\sqrt{(\bar{l}_i^1)^2 + (\bar{l}_i^2)^2}} \\ \frac{s_j^b \cdot \bar{l}_i}{\sqrt{(\bar{l}_i^1)^2 + (\bar{l}_i^2)^2}} \end{bmatrix} \quad (3)$$

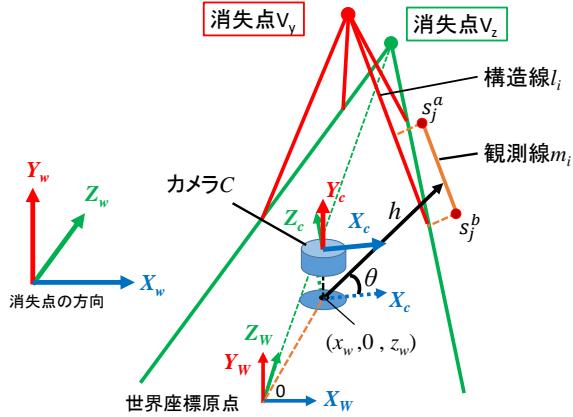


図2 消失点と観測線分の関係

各観測点を自己位置推定と同時に3次元の環境情報として記録する。このとき各構造線の中点近傍の画像をテンプレートとして保存する。このテンプレートは特徴の記述子として使用し、直前に観測された構造線と新たな観測線が観測される度にテンプレートマッチングを行う事で現在までに観測されていた構造線と新規に観測された観測線とのデータ関連付けを行う。

### 3. カルマンフィルタを利用するテンプレートマッチングの誤認識に対する改良法

構造線データと観測により得られた構造線上の観測値と、データの対応に利用するテンプレートマッチングでは、(i) 構造線の中点付近の画像のテンプレートでは情報が少ないため、他の線との誤認識が発生する、および (ii) 設定したテンプレート画像と類似の特徴を持つ環境での誤認識が発生する、問題がある。そこでこれらの問題に対する対策として以下の方法を提案している。

- 1) 線の端点の座標を中心とした画像を使用して線分以外の情報を取得することで特徴量の増加を行う。
- 2) 現在の画像内の線分と直前の観測画像中の線分の動きをカルマンフィルタを利用して推定し、次の線分の移動推定から画像がマッチングする範囲を予測する。そして予測範囲内でテンプレートマッチングを実行することで類似する特徴を有する環境での誤認識を防ぐ。

ここでは2)の提案法について説明する。一つ前の画像でのマッチング結果の中心座標  $n_{t-1} = [u_{t-1}, v_{t-1}]^T$  に対して、カルマンフィルタを利用して移動した中心座標の推定値  $\hat{n}_t = [\hat{u}_t, \hat{v}_t]^T$  を求める。そして  $\hat{u}_t$  を利用してマッチングの候補領域を予測することで、対応する構造線の探索領域を絞り込み誤認識の低減および計算速度負荷の低減を可能とする。図3は特徴点を中心としたテンプレートマッチングの範囲予測結果を示している。

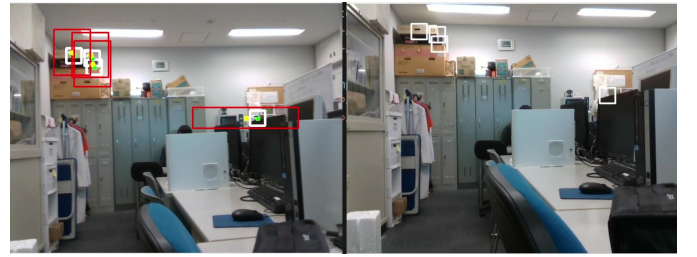


図3 カルマンフィルタを利用したマッチング範囲予測 (赤枠: マッチング予測範囲, 白枠: テンプレート)

## 4. 消失点を利用した環境位置推定精度改善

従来の Struct SLAM では構造線情報の選択に消失点を使用している。一方多くの建築物は直方体の組み合わせで構成されていると仮定できるため、建築物を二次元空間に射影したとき環境地図データである環境位置は長方形の組み合わせによって構成される。そこで消失点の方向を利用して構造線を長方形になるように修正をする事で自己位置推定精度の向上を行う。

### 4.1 消失点の推定

消失点は画像データ上の構造線の交点で与えられることから、構造線を利用して推定することができる。観測データ上には輝度外乱や内部パラメータ誤差などが含まれるため、最小二乗法を用いて求めた交点を消失点とする。図4の直線L上の任意の点座標を  $P_{i0} = [x_{i0} \ y_{i0}]^T$ 、交点  $P_c$  の座標を  $P_c = [x_c \ y_c]^T$  とする。直線Lの法線ベクトルを  $n_i = [a_i \ b_i]^T$  とすると、構造線である直線は次の方程式で与えられる。ここで  $i = 1, 2, \dots, n$  は取得した直線個数を示す。

$$n_i^T (P_{i0} - P_c) = 0 \quad (4)$$

(4) 式より

$$a_i x_c + b_i y_c = a_i x_{i0} + b_i y_{i0} \quad (5)$$

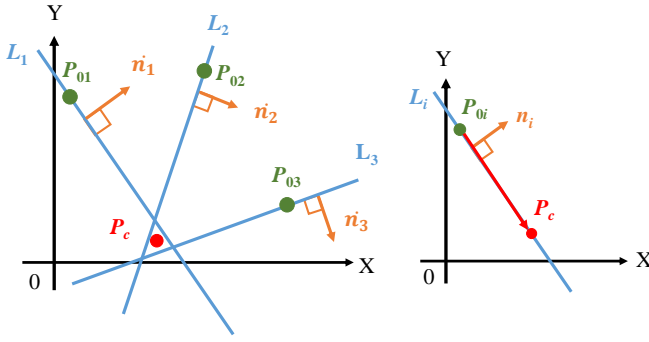


図4 最小二乗法を用いた消失点推定

(5) 式右辺は既知であることに注意し，全観測値の(4)式は以下に示す行列で表現できる．

$$\mathbf{N}\mathbf{P}_c = \mathbf{C} \quad (6)$$

$$\mathbf{N} = [n_1, \dots, n_n]^T, \mathbf{P}_c = [x_c \ y_c]^T$$

$$\mathbf{C} = [a_1x_{10} + b_1y_{10}, \dots, a_nx_{n0} + b_ny_{n0}]^T$$

(6) 式から消失点  $P_c$  は最小二乗法により

$P_c = (\mathbf{N}^T \mathbf{N})^{-1} \mathbf{N}^T \mathbf{C}$  で求まる．精度評価のため直線上の点である  $p_i$  と  $p_c$  に正規分布のノイズを加えたシミュレーションを行い，図5，表1の結果を得た．図5は観測値に  $N(300, 50)$  のノイズを含む結果である．表1は5回の実行での  $x$  座標， $y$  座標の推定値の標準偏差を示す．結果よりノイズ低減のために，法線方向による観測データの分類など前処理を追加が必要であることが分かった．

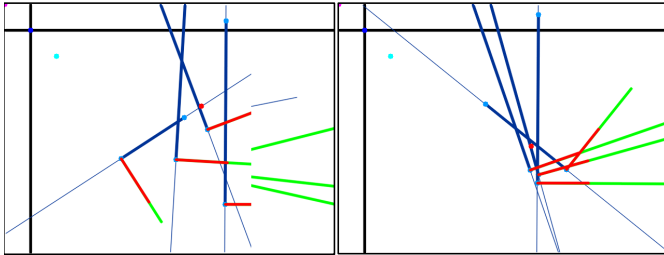


図5 最小二乗法を用いた交点推定結果  
左：誤差が大きい場合，右：誤差が小さい場合

表1 各座標に対する標準偏差

回数	x 座標標準偏差	y 座標標準偏差
1	63.843	68.158
2	48.122	80.187
3	79.331	280.974
4	36.780	143.274
5	35.324	157.828

## 4.2 環境位置推定精度改善法

4.1 で求めた消失点の位置は3次元空間において各構造線の方向を示している．消失点は構造線の延長線上から構成される点より，常に空間の中心に位置してい

る．この性質を利用し常に消失点と世界座標形原点を結んだ線と等距離になるように構造線を修正する．また構造線はカメラ画像上で1点に交わるが，3次元空間ではそれぞれが並行の関係にある．そこでこの並行関係の性質を利用し，各構造線を消失点の座標方向に延長して空間情報である地図上に描写して長方形を形成する事ができる．地図は長方形の組み合わせにより構成されているため，構造線を消失点まで延長した線を基準に近似する事でより精度の良い環境位置推定を行う．

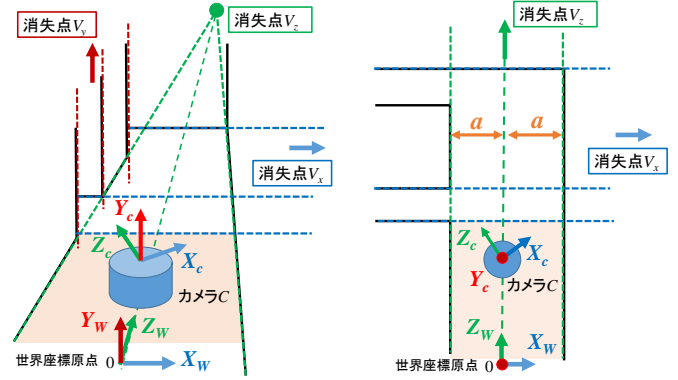


図6 消失点を用いた環境地図作成法

## 5. おわりに

本研究では，消失点を用いた環境位置推定精度改善法，改善法の検討法に必要なテンプレートマッチングの誤認識に対する改良法および消失点の推定方法について説明を行った．消失点推定の実行結果から現在の方法だけでは十分な精度ができない問題点がわかった．そこで今後は消失点推定精度の向上法につて改善を行っていく．さらにマッチングの誤認識に対する改良法も併せて実施を行う．

## 参考文献

- [1] 鶴田 杜広，宮口 幹太，藤井 中，“自走式墨出しロボットののための高効率作業計画”，日本ロボット学会誌，Vol.39，no.1，pp.87-97，2021.
- [2] 原 祥亮，“ROSを用いた自律走行，日本ロボット学会誌”，Vol.35，no.4，pp.286-290，2017.
- [3] 加藤正浩，岩井智昭，小山和紀，鈴 雅美，“MEMS ミラー型 LiDAR を用いた自己位置推定技術の開発”，自動車技術会論文集，Vol.51，no.1，pp.232 - 237，2020.
- [4] Raúl Mur-Artal, J. M. M. Montiel and Juan D. Tardós, “ORB-SLAM: A Versatile and Accurate Monocular SLAM System”, IEEE Transactions on Robotics, Vol.31, no.5, pp.1147-1163, Oct. 2015.
- [5] A. J. Davison, I. D. Reid, N. D. Molton and O. Stasse, “MonoSLAM: Real-time single camera SLAM”, IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 6, pp. 1052-1067, Jun. 2007.
- [6] E. Perdices, L. M. López and J. M. Cañas, “LineSLAM: Visual real time localization using lines and UKF”, Proc. 1st ROBOT Iberian Robot. Conf., pp. 663-678, 2014.
- [7] Huizhong Zhou, Danping Zou, Ling Pei, Rendong Ying, Peilin Liu and Wenxian Yu, “StructSLAM: Visual SLAM With Building Structure Lines”, IEEE Transactions on Vehicular Technology, Vol.64, no.4, pp.1364-1375, April. 2015.