

2018年度 卒業論文

建造物の壁に対する相対姿勢を用いた  
6DoF位置姿勢推定

2019年1月28日

機械工学科

(学籍番号: 153R153031)

尾崎 亮太

明治大学理工学部

## 概要

この論文では，屋内や市街地のような整備された人工環境の走行を想定した，移動ロボットのための自己位置姿勢推定法を提案する．本手法では，移動体の姿勢推定に従来用いられる，内界センサを使ったデッドレコニングによる推定と，外界センサを使ったSLAMによる推定との統合に加え，鉛直に建てられた建造物の壁に対する相対姿勢を用いる．一般的な建造物の壁がおおよそ鉛直に建てられていることを利用し，ロボット座標系での重力ベクトルを推定することで推定ロール，ピッチ角を補正する．デッドレコニング，SLAM，壁面観測，それぞれによる推定は拡張カルマンフィルタで統合される．そして，屋外での実機走行実験によって本手法の有用性を示す．

# 目次

<b>第1章</b>	<b>序論</b>	<b>1</b>
1.1	研究背景 . . . . .	1
1.2	先行研究 . . . . .	1
1.3	研究目的 . . . . .	2
<b>第2章</b>	<b>建造物の壁に対する相対姿勢を用いた 6DoF 位置姿勢推定</b>	<b>3</b>
2.1	はじめに . . . . .	3
2.1.1	座標系の定義 . . . . .	3
2.1.2	拡張カルマンフィルタ . . . . .	4
2.1.3	四元数とオイラー角との変換 . . . . .	5
2.2	IMU を用いた初期姿勢推定 . . . . .	6
2.2.1	センサ出力値の平均 . . . . .	6
2.2.2	初期姿勢推定 . . . . .	6
2.3	壁面を用いた推定姿勢補正 . . . . .	7
2.3.1	主法線抽出 . . . . .	7
2.3.2	重力ベクトル推定 . . . . .	9
2.3.3	推定姿勢補正 . . . . .	10
2.3.4	キャリブレーション . . . . .	11
2.4	拡張カルマンフィルタによる統合 . . . . .	12
<b>第3章</b>	<b>評価実験</b>	<b>14</b>
3.1	実験概要 . . . . .	14
3.2	実験環境 . . . . .	14
3.3	実験結果 . . . . .	18
3.4	追加実験 . . . . .	21
<b>第4章</b>	<b>結論</b>	<b>25</b>

## 目 次

2.1	Extraction steps . . . . .	8
2.2	Clustering in Gaussian sphere . . . . .	9
2.3	Estimation and correction of the gravity vector . . . . .	10
3.1	System architecture . . . . .	15
3.2	Estimation of depth by LSD-SLAM . . . . .	16
3.3	The experimental robot . . . . .	16
3.4	The experimental environment . . . . .	17
3.5	Estimated trajectories of the robot in 1 round . . . . .	19
3.6	Estimated trajectories of the robot in 3 rounds . . . . .	20
3.7	The additional experimental environment . . . . .	22
3.8	Point cloud of the additional experimental environment . . . . .	22
3.9	Point and normal cloud of the additional experiment no.3 . . . . .	23
3.10	Point and normal cloud of the additional experiment no.4 . . . . .	24
3.11	Point and normal cloud of the additional experiment no.5 . . . . .	24



## 表 目 次

2.1	Nomenclature . . . . .	5
3.1	Errors of position and pose estimations in 1 round . . . . .	18
3.2	Detail of obstacles . . . . .	21
3.3	Errors of pose estimations with obstacles which are not vertical planes . . .	22

# 第1章 序論

## 1.1 研究背景

近年，屋内や市街地のような整備された人工環境において活動する自律移動ロボットへの需要が高まっている．また，次世代の移動ロボットは，その適用範囲を広げるため，高い機動性を持つことが要求される．高い機動性を実現するための手段の1つとして，姿勢制御があり，実際に開発されている [1]．移動ロボットの姿勢制御を行うには，走行中における時々刻々の3次元空間におけるロボットの自己姿勢を推定し把握し続ける必要がある．

## 1.2 先行研究

移動体の姿勢は，一般に，ジャイロ스코ープや加速度センサを使って推定することが多い．地面上を走行する移動ロボットの場合，ロボット自身の加速度や地面からの振動が推定精度に大きな影響を与えるため，さらに外界センサを用いたSLAM (Simultaneous Localization and Mapping) [2] との組み合わせが提案されている [3]．SLAMは，以下のように多くの手法が提案されている．ICP スキャンマッチング [4] や NDT スキャンマッチング [5, 6] では，2つの3次元点群に対して共通領域を基に点群の位置合わせを行うことで相対位置姿勢を推定する．Zhangら [7] はこの位置合わせにおいて，平面やエッジの特徴量を活用している．一方，清水ら [8] は，点群そのままではなくクラスタリングした法線をガウス球に投影し，支配的な法線（主平面）で位置合わせを行っている．また，カメラまたはRGB-Dカメラを用いたSLAMによる相対姿勢推定も提案されている [9, 10, 11]．これらの手法は，初期姿勢に対する相対変化を推定するため，蓄積誤差を補正することが難しい．この蓄積誤差を補正する方法として，事前情報と時々刻々のセンサ情報とを対応付けする手法があり，特に事前環境地図を用いる手法が多く提案されている [12]．また，人工環境での特徴を利用した手法も以下のように提案されている．S. Ramalingamら [13] は，街の3次元モデルと，全点球カメラの画像をマッチングを行う．このマッチング

では、空を見上げた時の空と建物の境界線 (skylines) を用いている。後藤ら [14] は同様にモデルと全点球カメラの対応付けを提案しており、天井から床に至る線分および天井と壁、床と壁との線分など、人工物環境に多数存在する線分をマッチングの特徴量として用い6自由度位置姿勢推定を行っている。Alberto Y. Hata ら [15] は自動車のための自己位置推定を提案しており、縁石の情報を含んだ地図と、センサで観測した縁石とのマッチングを行っている。このような環境地図や3次元モデルの使用は、適用可能環境を限定する。また、高精度なGPSと慣性航法装置を利用した車両のピッチ角手法も提案されている [16, 17]。しかしGPSの使用も、同様に適用可能環境を限定する。一方、李ら [18] は、車載単眼カメラを用いた走行道路に対するピッチ角推定を提案している。車両が道路平面上を走行するという点に注目し、前後フレーム間の並進ベクトルが走行道路平面に平行である拘束を用いて累積誤差のない推定を示している。しかしここで推定される姿勢角は、道路に対する姿勢角であり重力座標系での絶対値ではない。

### 1.3 研究目的

本研究では、従来用いられる慣性センサとSLAMの統合に加え、鉛直に建てられた建造物の壁に対する相対姿勢をワールド座標系における絶対姿勢として観測することで、誤差の蓄積を適時補正することができる姿勢推定法を提案する。主に、時々刻々での推定精度を重視しており、ループクローズによる過去の軌跡の補正は本研究の趣旨と一致しない。なお、本手法は、一般的な建造物の壁がおおよそ鉛直に建てられていることを利用し、事前環境地図を必要としない。屋外での実機走行実験によって本手法の有用性を示す。

# 第2章 建造物の壁に対する相対姿勢を用いた6DoF位置姿勢推定

## 2.1 はじめに

本章では建造物の壁に対する相対姿勢を用いた6DoF位置姿勢推定について述べる。ロボットが静止している状態でIMUによって初期姿勢を求め、それに対してデッドレコニングおよびSLAMで推定される相対姿勢変化を積算する。そして、鉛直面を観測した場合は、それらを用いてロボット座標系での重力ベクトルを推定し蓄積誤差を補正する。なお、この推定姿勢はカルマンフィルタの観測として統合される。

### 2.1.1 座標系の定義

座標系を以下に定義する。

- ロボット座標系

ロボットに固定され、進行方向をX軸の正とする右手直交座標系とする。各軸を $X_r$ ,  $Y_r$ ,  $Z_r$ 軸とする。

- ワールド座標系

初期姿勢位置において原点がロボット座標系と一致し、重力方向をZ軸の負とする右手直交座標系とする。各軸を $X_w$ ,  $Y_w$ ,  $Z_w$ 軸とする。

## 2.1.2 拡張カルマンフィルタ

次の第 2.2.2 項および第 2.4 節で拡張カルマンフィルタを用いるため、この節でそれについて記述する。表 2.1 に使用する変数の定義を示す。本論文内での拡張カルマンフィルタの記述ではこれらの変数を共通して用いる。拡張カルマンフィルタは、非線形状態モデルを近似で線形化することで適用できるカルマンフィルタである。カルマンフィルタの状態方程式および観測方程式は離散化されている必要があり、式 (2.1), (2.2) に示す通りである。

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_{k-1}, \quad \mathbf{w}_{k-1} \sim N(0, \mathbf{Q}_k) \quad (2.1)$$

$$\mathbf{z}_k = h_k(\mathbf{x}_k) + \mathbf{v}_{k-1}, \quad \mathbf{v}_{k-1} \sim N(0, \mathbf{R}_k) \quad (2.2)$$

予測ステップでは以下の計算が行われる。状態  $\mathbf{x}$  の予測が計算され、状態方程式のヤコビアンと設定されたノイズによって予測の分散が評価される。

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}), \quad \mathbf{F}_{k-1} \approx \left. \frac{\partial f_k}{\partial \mathbf{x}} \right|_{\mathbf{x}_{k-1}, \mathbf{u}_{k-1}} \quad (2.3)$$

$$\mathbf{P}_k = \mathbf{F}_{k-1} \mathbf{P}_{k-1} \mathbf{F}_{k-1}^\top + \mathbf{Q}_{k-1} \quad (2.4)$$

観測ステップでは以下の計算が行われる。カルマンゲインが計算され、予測ステップでの状態および分散が、観測と統合される。

$$\mathbf{H}_k \approx \left. \frac{\partial h_k}{\partial \mathbf{x}} \right|_{\mathbf{x}_k}, \quad \mathbf{K}_k = \mathbf{P}_k \mathbf{H}_k^\top (\mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^\top + \mathbf{R}_k)^{-1} \quad (2.5)$$

$$\mathbf{x}_k = \mathbf{x}_k + \mathbf{K}_k (\mathbf{z}_k - h_k(\mathbf{x}_k)), \quad \mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k \quad (2.6)$$

表 2.1: Nomenclature

$k$	discrete time step
$f, h$	non-linear process and measurement model function
$\mathbf{x}$	state vector
$\mathbf{P}$	estimation error covariance matrix
$\mathbf{u}$	input vector
$\mathbf{z}$	measurement vector
$\mathbf{w}, \mathbf{v}$	process and measurement noise vector
$\mathbf{Q}, \mathbf{R}$	process and measurement covariance matrix
$N$	normal distribution
$\mathbf{F}, \mathbf{H}$	partial derivative matrices (Jacobians)
$\mathbf{K}$	Kalman gain matrix

### 2.1.3 四元数とオイラー角との変換

四元数（クォータニオン） $\mathbf{q}$  とオイラー角（ロール  $\phi$ ，ピッチ  $\theta$ ，ヨー  $\psi$ ）の関係は，

$$\begin{aligned} \mathbf{q} &= \begin{pmatrix} q_x & q_y & q_z & q_w \end{pmatrix}^T \\ &= \begin{pmatrix} \cos \frac{\phi}{2} \sin \frac{\theta}{2} \cos \frac{\psi}{2} + \sin \frac{\phi}{2} \cos \frac{\theta}{2} \sin \frac{\psi}{2} \\ \cos \frac{\phi}{2} \cos \frac{\theta}{2} \sin \frac{\psi}{2} + \sin \frac{\phi}{2} \sin \frac{\theta}{2} \cos \frac{\psi}{2} \\ \sin \frac{\phi}{2} \cos \frac{\theta}{2} \cos \frac{\psi}{2} + \cos \frac{\phi}{2} \sin \frac{\theta}{2} \sin \frac{\psi}{2} \\ \cos \frac{\phi}{2} \cos \frac{\theta}{2} \cos \frac{\psi}{2} + \sin \frac{\phi}{2} \sin \frac{\theta}{2} \sin \frac{\psi}{2} \end{pmatrix} \end{aligned} \quad (2.7)$$

$$\begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix} = \begin{pmatrix} \tan^{-1} \frac{2(q_w q_x + q_y q_z)}{q_w^2 + q_x^2 - q_y^2 - q_z^2} \\ \sin^{-1} 2(q_w q_y - q_x q_z) \\ \tan^{-1} \frac{2(q_x q_y + q_w q_z)}{q_w^2 - q_x^2 - q_y^2 + q_z^2} \end{pmatrix} \quad (2.8)$$

また，姿勢角の定義域は  $[-\pi, \pi]$  であり，数値的に不連続であることを考慮する必要があるが，具体的な処理の記述は省略する。

## 2.2 IMUを用いた初期姿勢推定

ロボットが静止している状態でのIMUの出力値を用いてワールド座標系におけるロボットの初期姿勢を推定する。また角速度センサのバイアスも推定する。

### 2.2.1 センサ出力値の平均

IMUの出力値（加速度 $\alpha_{\text{IMU},i}$ ，角速度 $\omega_{\text{IMU},i}$ ）を取得するたびに，出力回数 $n$ で平均をとる。加速度の平均値 $\alpha_{\text{ave}}$ は初期姿勢の推定に用いられ，角速度の平均値 $\omega_{\text{ave}}$ は角速度センサのバイアス $\omega_{\text{bias}}$ として用いられる。

$$\alpha_{\text{ave}} = \frac{\sum_{i=0}^{n-1} \alpha_{\text{IMU},i}}{n} \quad (2.9)$$

$$\omega_{\text{ave}} = \frac{\sum_{i=0}^{n-1} \omega_{\text{IMU},i}}{n} \quad (2.10)$$

$$\omega_{\text{bias}} = \omega_{\text{ave}} \quad (2.11)$$

### 2.2.2 初期姿勢推定

拡張カルマンフィルタを用いて初期姿勢を推定する。カルマンフィルタの分散が収束するまでの時間，IMUを静止させておく必要がある。式(2.25)～(2.15)でそれぞれ示される状態ベクトル，観測ベクトル，状態方程式，観測方程式を用いて，IMUの出力値を取得するたびに予測，観測を行う。ワールド座標系の $X_w$ 軸， $Y_w$ 軸， $Z_w$ 軸まわりの姿勢角をそれぞれ $\phi$ ， $\theta$ ， $\psi$ とする。ロボットは静止状態であるため，ここでは $\psi = 0$ で一定とする。回転行列を $\mathbf{Rot}$ ，ワールド座標系での重力ベクトルを $\mathbf{G}_{\text{world}}$ とする。表記に関して，例えば $\alpha_{\text{ave},X_r,k}$ は $k$ ステップでの $X_r$ 軸方向の平均加速度値を表す。

$$\mathbf{x}_k = \begin{pmatrix} \phi_k & \theta_k \end{pmatrix}^T \quad (2.12)$$

$$\mathbf{z}_k = \begin{pmatrix} \alpha_{\text{ave},X_r,k} & \alpha_{\text{ave},Y_r,k} & \alpha_{\text{ave},Z_r,k} \end{pmatrix}^T \quad (2.13)$$

$$f(\mathbf{x}_k) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \phi_{k-1} \\ \theta_{k-1} \end{pmatrix} \quad (2.14)$$

$$h(\mathbf{x}_k) = \mathbf{Rot}_k \mathbf{G}_{\text{world}} = \mathbf{Rot}_k \begin{pmatrix} 0 \\ 0 \\ G_{\text{world}, Z_w} \end{pmatrix} = \begin{pmatrix} -G_{\text{world}, Z_w} \sin \theta \\ G_{\text{world}, Z_w} \sin \phi \cos \theta \\ G_{\text{world}, Z_w} \cos \phi \cos \theta \end{pmatrix} \quad (2.15)$$

$$\mathbf{Rot}_k = \begin{pmatrix} \cos \theta_k \cos \psi_k & \cos \theta_k \sin \psi_k & -\sin \theta_k \\ \sin \phi_k \sin \theta_k \cos \psi_k - \cos \phi_k \sin \psi_k & \sin \phi_k \sin \theta_k \sin \psi_k + \cos \phi_k \cos \psi_k & \sin \phi_k \cos \theta_k \\ \cos \phi_k \sin \theta_k \cos \psi_k + \sin \phi_k \sin \psi_k & \cos \phi_k \sin \theta_k \sin \psi_k - \sin \phi_k \cos \psi_k & \cos \phi_k \cos \theta_k \end{pmatrix} \quad (2.16)$$

## 2.3 壁面を用いた推定姿勢補正

### 2.3.1 主法線抽出

測距センサで得られる点群（図 2.1b）を用いて、各注目点の近傍局所点群に主成分分析 [19] を適用することで法線ベクトルを算出する（図 2.1c）。近傍点群は、kd-tree[20] を用いて指定半径以内の点を探索することで得られる。センサから得られる点群は、センサから遠くなるほど点の密度が粗になるため、注目点とセンサとの距離が大きいほど、探索半径を大きくする。

この法線群から、信頼性の高い鉛直面を持つ法線を以下の条件で抽出する（図 2.1d）。

- 注目点を持つ近傍点の数が十分多い（密度が高い）。
- 式 (2.17) で算出される角度  $\beta$  が十分小さい。これは 1 ステップ前の推定重力ベクトル  $\mathbf{G}'_{\text{robot}}$  をもとに、観測された法線  $\mathbf{N}$  が鉛直面を持つかを判定するものである。

$$\beta = \left| \cos^{-1} \frac{\mathbf{N} \cdot \mathbf{G}'_{\text{robot}}}{\|\mathbf{N}\| \|\mathbf{G}'_{\text{robot}}\|} - \frac{\pi}{2} \right| \quad (2.17)$$

- 法線に垂直な平面と近傍点群の二乗誤差が十分小さい（平面度が高い）。

抽出された法線群でガウス球 [21] を生成し、球内の点群に対してユークリッド距離に基づくクラスタ分析を適用する。ガウス球とは、各点が保有する法線ベクトルの各成分をユークリッド空間内に再配置することで得られる点群であり、この操作は一般にガウス写像と呼ばれる。なお本手法では、適当な法線を反転させガウス球をさらに半球のみに反映



している。つまり、平行で向かい合う平面は同じ法線ベクトルを持つようにする。クラスタリング終了後に、メンバが閾値より少ないクラスタは外れ値として除去する。各クラスタが持つ点群の重心を算出し、それらの位置ベクトルを主法線（支配的な法線）として用いる。図 2.2 にガウス球でのクラスタリングの概要図を示す。

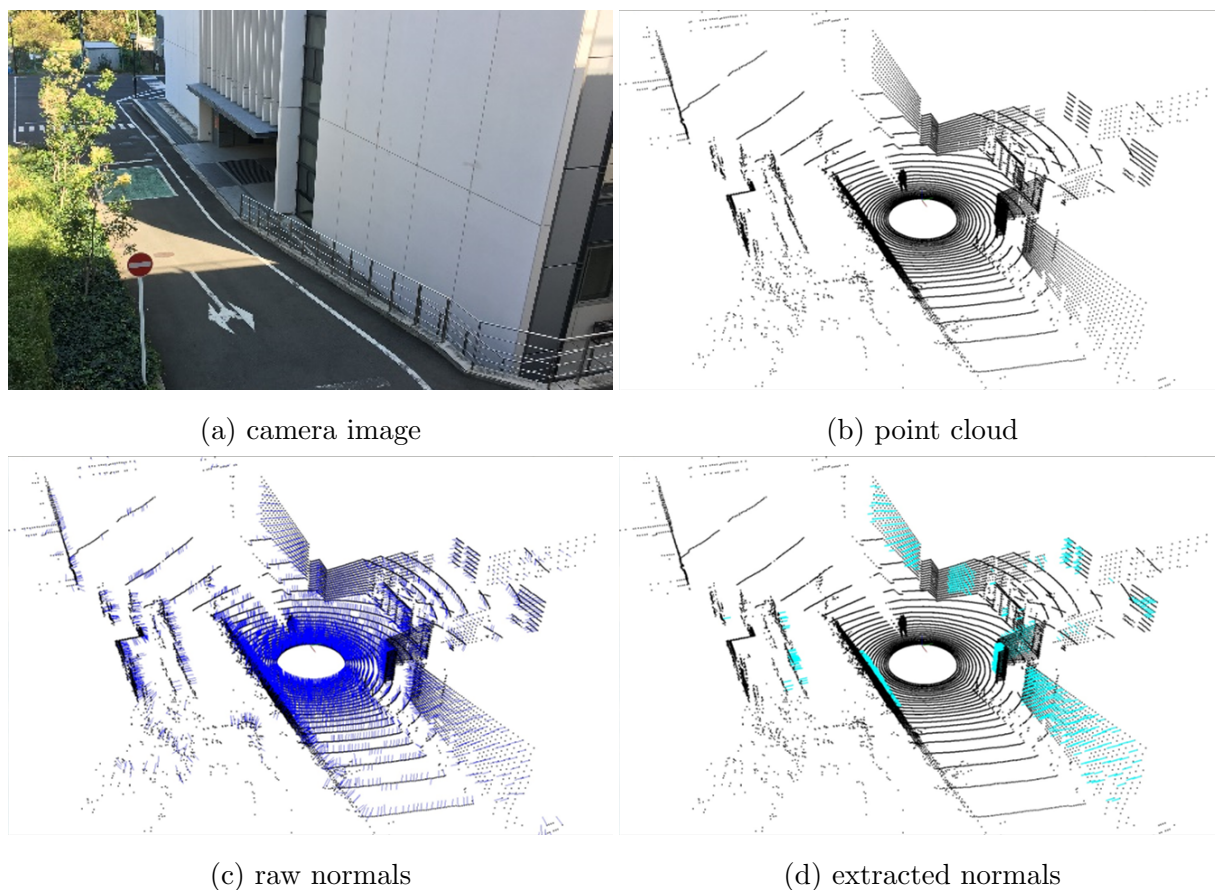


図 2.1: Extraction steps

Extraction of reliable normals which would have vertical planes in the environment (a) is shown here. Point cloud which is obtained with 3D-LiDAR in the environment is shown in (b). By applying analysis of principal component to the point cloud, normals are computed as shown in (c). Normals which meet the conditions for judging whether they would have vertical planes as shown in (d).

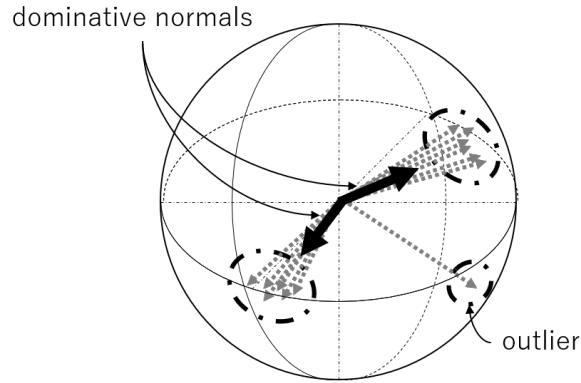


図 2.2: Clustering in Gaussian sphere

Extracted normals which satisfy the conditions are projected into Gaussian sphere as points. The points on Gaussian sphere are clustered here. Clusters which do not have enough numbers of member points are purged as outliers. Position vectors of the centroids clusters are used for estimating the gravity vectore at next step as dominative normals.

### 2.3.2 重力ベクトル推定

クラスタリングされた主法線（ガウス球内での点）を用いてロボット座標系における単位重力ベクトル  $\mathbf{G}_{\text{robot}}$  を推定する．主法線の数に応じて以下のように算出する．ただし，以下のように外積などで算出されるベクトルが，本来の重力ベクトルと方向が逆になる場合もある．そのため，算出後に，1ステップ前の重力方向をもとに，算出したベクトルの向きを判定する必要がある．

- 主法線の数：3 以上

主法線をガウス球における点として扱い，この点群に対する主成分分析で得た法線を重力ベクトルとする．ただし，クラスタが持つ点の数で重み付けを行う．

- 主法線の数：2

2本の主法線ベクトル  $\mathbf{N}_1, \mathbf{N}_2$  の外積を重力方向ベクトルとする．図 2.3a に推定の様子を示す．

$$\mathbf{G}_{\text{robot}} = \mathbf{N}_1 \times \mathbf{N}_2 \quad (2.18)$$

- 主法線の数：1

ワールド座標系における絶対的な姿勢を得るには、平行でない鉛直面が2面以上必要であるが、1ステップ前の推定重力ベクトル  $\mathbf{G}'_{\text{robot}}$  を用いて、式 (2.19) のように部分的に推定重力ベクトルを補正し算出する。観測された主法線ベクトルを  $\mathbf{N}$  とする。図 2.3b に推定の様子を示す。

$$\mathbf{G}_{\text{robot}} = \frac{\mathbf{G}'_{\text{robot}} - (\mathbf{G}'_{\text{robot}} \cdot \mathbf{N})\mathbf{N}}{\|\mathbf{G}'_{\text{robot}} - (\mathbf{G}'_{\text{robot}} \cdot \mathbf{N})\mathbf{N}\|} \quad (2.19)$$

- 主法線の数 : 0

重力方向を推定することが出来ない。

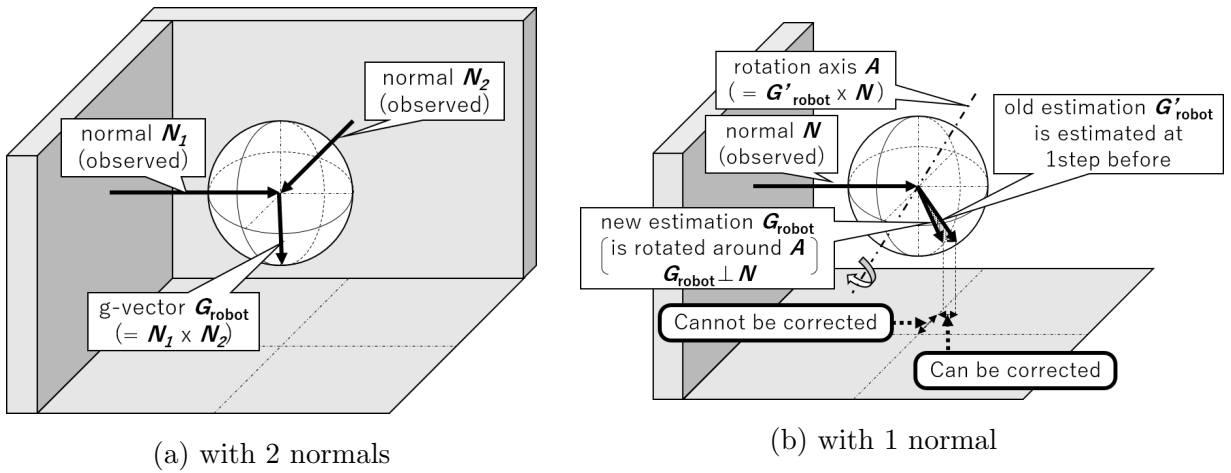


図 2.3: Estimation and correction of the gravity vector

In (a), the gravity vector  $\mathbf{G}_{\text{robot}}$  is estimated by the cross product of the two observed normals  $\mathbf{N}_1, \mathbf{N}_2$ .  $\mathbf{N}_1, \mathbf{N}_2$  are normals which are extracted and clustered, and they cannot be parallel. In (b), estimating the gravity vector  $\mathbf{G}_{\text{robot}}$  is supposed to need two or more vectors which are not parallel to each other. The proposed method partially corrects the estimation of gravity vector by using  $\mathbf{G}'_{\text{robot}}$  which is estimated one step before.

### 2.3.3 推定姿勢補正

重力ベクトルの推定誤差が、推定姿勢の誤差であることを用いて補正する。なお計算は四元数（クォータニオン）を用いて行う。ただし、第 2.2.2 項や第 2.4 節で記述されるカル

マンフィルタでは、姿勢をロール、ピッチ、ヨーで扱っているため、四元数に変換する必要がある。補正後の姿勢を表す四元数  $\mathbf{q}_{\text{pose.wall}}$  の算出を以下に示す。1ステップ前の推定重力ベクトル  $\mathbf{G}'_{\text{robot}}$  と、第2.3.2項で推定した推定重力ベクトル  $\mathbf{G}_{\text{robot}}$  において、この2ベクトル間の回転を表す四元数  $\mathbf{q}_{\text{error}}$  は、

$$\begin{aligned}\mathbf{q}_{\text{error}} &= \left( q_{\text{error},x} \quad q_{\text{error},y} \quad q_{\text{error},z} \quad q_{\text{error},w} \right)^T \\ &= \left( C_{X_r} \sin \gamma/2 \quad C_{Y_r} \sin \gamma/2 \quad C_{Z_r} \sin \gamma/2 \quad \cos \gamma/2 \right)^T\end{aligned}\quad (2.20)$$

$$\gamma = \cos^{-1} \frac{\mathbf{G}'_{\text{robot}} \cdot \mathbf{G}_{\text{robot}}}{\|\mathbf{G}'_{\text{robot}}\| \|\mathbf{G}_{\text{robot}}\|}\quad (2.21)$$

$$\mathbf{C} = \left( C_{X_r} \quad C_{Y_r} \quad C_{Z_r} \right)^T = \frac{\mathbf{G}'_{\text{robot}} \times \mathbf{G}_{\text{robot}}}{\|\mathbf{G}'_{\text{robot}}\| \|\mathbf{G}_{\text{robot}}\|}\quad (2.22)$$

推定重力ベクトル  $\mathbf{G}'_{\text{robot}}$  に対応する推定姿勢を表す四元数を  $\mathbf{q}'_{\text{pose}}$  としたとき、次のように補正される。

$$\mathbf{q}_{\text{pose.wall}} = \mathbf{q}'_{\text{pose}} \mathbf{q}_{\text{error}}\quad (2.23)$$

$\mathbf{q}_{\text{pose.wall}}$  および  $\mathbf{q}'_{\text{pose}}$  はワールド座標系での姿勢であり、 $\mathbf{q}_{\text{error}}$  はロボット座標系での誤差（回転）である。

### 2.3.4 キャリブレーション

距離センサの取り付け誤差があった場合、それに合わせて点群を回転させる必要がある。ロボットが初期位置姿勢に静止している状態でキャリブレーションを行う。第2.2節で推定した初期姿勢を表す四元数  $\mathbf{q}_{\text{pose.ini}}$  に対する、第2.3.3項で推定した姿勢を表す四元数  $\mathbf{q}_{\text{pose.wall}}$  の誤差を取り付け誤差  $\mathbf{q}_{\text{misalignment}}$  とする。

$$\mathbf{q}_{\text{misalignment}} = \mathbf{q}_{\text{pose.wall}} \mathbf{q}_{\text{pose.ini}}^{-1}\quad (2.24)$$

初期姿勢推定後は、点群に対して回転  $\mathbf{q}_{\text{misalignment}}$  を適用させて上記の重力ベクトル推定を行う。

## 2.4 拡張カルマンフィルタによる統合

車輪型ロボットで人工環境を走行することを想定しており、各姿勢角度変化がある程度急激なものではなく、カルマンフィルタで十分扱えると仮定する。IMUを用いたデッドレコニングで推定する姿勢を予測，SLAMで推定する姿勢を観測1，壁面に対する相対姿勢を観測2とし，拡張カルマンフィルタで統合する。

$$\mathbf{x}_k = \begin{pmatrix} \phi_k & \theta_k & \psi_k \end{pmatrix}^T \quad (2.25)$$

- 予測（デッドレコニング）

センサ座標系における角速度を，重力座標系へ座標変換しレート積分を行う。ただし，センサのバイアスを考慮している。

$$f(\mathbf{x}_{k-1}) = \begin{pmatrix} \phi_{k-1} \\ \theta_{k-1} \\ \psi_{k-1} \end{pmatrix} + \begin{pmatrix} 1 & \sin \phi_{k-1} \tan \theta_{k-1} & \cos \phi_{k-1} \tan \theta_{k-1} \\ 0 & \cos \phi_{k-1} & -\sin \phi_{k-1} \\ 0 & \sin \phi_{k-1} \sec \theta_{k-1} & \cos \phi_{k-1} \sec \theta_{k-1} \end{pmatrix} \begin{pmatrix} \omega_{X_r, k-1} - \omega_{\text{bias}, X_r} \\ \omega_{Y_r, k-1} - \omega_{\text{bias}, Y_r} \\ \omega_{Z_r, k-1} - \omega_{\text{bias}, Z_r} \end{pmatrix} dt \quad (2.26)$$

- 観測1（SLAM）

SLAMによって得られる姿勢をそのまま観測するのではなく，1ステップ前にSLAMで推定された姿勢との相対変化を用いることで，観測2を推定結果に反映しやすくする。

$$\mathbf{z}_k = \begin{pmatrix} \phi_{k-1} + (\phi_{\text{slam}, k} - \phi_{\text{slam}, k-1}) \\ \theta_{k-1} + (\theta_{\text{slam}, k} - \theta_{\text{slam}, k-1}) \\ \psi_{k-1} + (\psi_{\text{slam}, k} - \psi_{\text{slam}, k-1}) \end{pmatrix} \quad (2.27)$$

$$h(\mathbf{x}_k) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \phi_k \\ \theta_k \\ \psi_k \end{pmatrix} \quad (2.28)$$

- 観測 2 (壁面観測)

上記2つのステップでは初期姿勢に対する相対姿勢を推定しているのに対し、このステップではワールド座標系での絶対姿勢を直接観測することができる。式 (2.23) で算出される四元数  $\mathbf{q}_{\text{pose.wall}}$  をワールド座標系のロール、ピッチに変換し観測する。第 2.3.3 項で場合分けされるように、主法線の数に応じて共分散  $\mathbf{R}$  を変化させる。

$$\mathbf{z}_k = \begin{pmatrix} \tan^{-1} \frac{2(q_{\text{pose.wall},w}q_{\text{pose.wall},x} + q_{\text{pose.wall},y}q_{\text{pose.wall},z})}{q_{\text{pose.wall},w}^2 - q_{\text{pose.wall},x}^2 - q_{\text{pose.wall},y}^2 + q_{\text{pose.wall},z}^2} \\ \sin^{-1} 2(q_{\text{pose.wall},w}q_{\text{pose.wall},y} - q_{\text{pose.wall},x}q_{\text{pose.wall},z}) \end{pmatrix} \quad (2.29)$$

$$h(\mathbf{x}_k) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \phi_k \\ \theta_k \\ \psi_k \end{pmatrix} \quad (2.30)$$

## 第3章 評価実験

### 3.1 実験概要

起伏のあるコースで移動ロボットを走行させ、提案手法および比較手法で、時々刻々のロボットの自己姿勢を推定する。走行中の推定姿勢を評価することは難しいため、推定姿勢と並進速度を統合することで、走行中の推定姿勢誤差を、走行終了時の推定位置誤差に置き換えて評価する。つまり、同じ並進速度を用いた場合、姿勢の推定誤差が小さいほど、推定位置（軌跡）の誤差が小さくなると仮定する。

### 3.2 実験環境

本実験での提案手法のシステム図を図3.1に示す。図中、Staticは第2.2節の初期姿勢推定のシステム、Dynamicは走行開始後のシステムをそれぞれ表している。カルマンフィルタの観測で用いるSLAMとして、LSD-SLAM[9]を用いる。LSD-SLAMとは単眼カメラSLAMである。推定のレート of the speed and scale does not depend on the pose estimation, which is suitable for pose estimation. Fig. 3.2 shows the pose estimation by LSD-SLAM. In the figure, Integration uses the coordinate transformation matrix  $\mathbf{Rot}$  in Eq. (2.16), and Eq. (3.1) is used to perform the coordinate transformation, integrate the pose and the linear velocity.  $\mathbf{V}_{\text{robot}}$  is the linear velocity vector in the robot coordinate system, and  $\mathbf{L}_{\text{world}}$  is the linear velocity vector of the robot in the world coordinate system.

$$\begin{aligned} \mathbf{L}_{\text{world},k} &= \mathbf{L}_{\text{world},k-1} + \mathbf{Rot}_k^{-1} \mathbf{V}_{\text{robot},k} dt \\ &= \mathbf{L}_{\text{world},k-1} + \mathbf{Rot}_k^T \mathbf{V}_{\text{robot},k} dt \\ &= \begin{pmatrix} L_{X_w,k-1} \\ L_{Y_w,k-1} \\ L_{Z_w,k-1} \end{pmatrix} + \mathbf{Rot}_k^T \begin{pmatrix} V_{\text{robot},X_r,k} \\ 0 \\ 0 \end{pmatrix} dt \end{aligned} \quad (3.1)$$

本実験では、統合される並進速度としてホイールオドメトリで得られる並進速度を用いる。

姿勢推定の比較手法として、ジャイロ스코ープによるデッドレコニングと、単独のLSD-SLAMを用いる。提案手法と同様に、これらの手法で姿勢をそれぞれ推定し、ホイールエンコーダの並進速度と統合する。なお、本研究の趣旨より、LSD-SLAMでのループクローズは行わない。

図 3.3 に実験で使用する移動ロボットを示す。ロボットに搭載し、使用するセンサは、Velodyne HDL-32E, Xsens MTi30, RealSense D435, ホイールエンコーダである。RealSense D435 は RGB-D カメラであるが、LSD-SLAM のために画像のみを利用しているため、深度情報は利用していない。評価に用いる真値として、走行開始時と走行終了時における、モーションキャプチャで得られる位置姿勢を用いる。本実験では Vicon Vero v1.3X を用いる。図 3.4 に、明治大学生田キャンパス内の実験環境の航空写真を示す。走行コースは、一周約 250m, 高低差約 3m である。

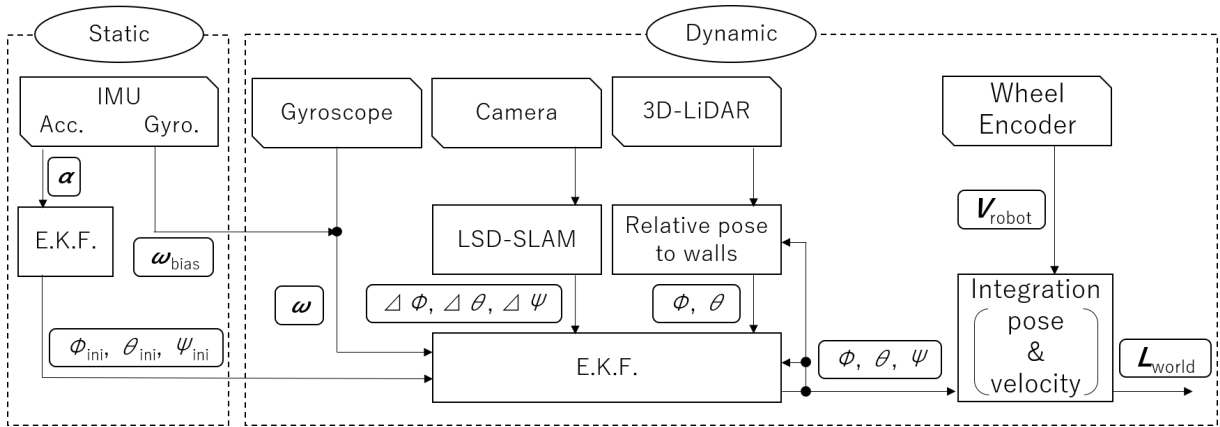


図 3.1: System architecture

A system configuration diagram of the proposed method for the experiment is shown here. “Static” part represents the system which runs when the robot has no move. “Dynamic” part represents the system after the robot starts driving. Estimating pose with vertical walls is referred to as “Relative pose to walls” in this figure, and the pose is merged through E.K.F. with other estimations. In “Integration” part, estimated pose(orientation) and linear velocity are integrated.



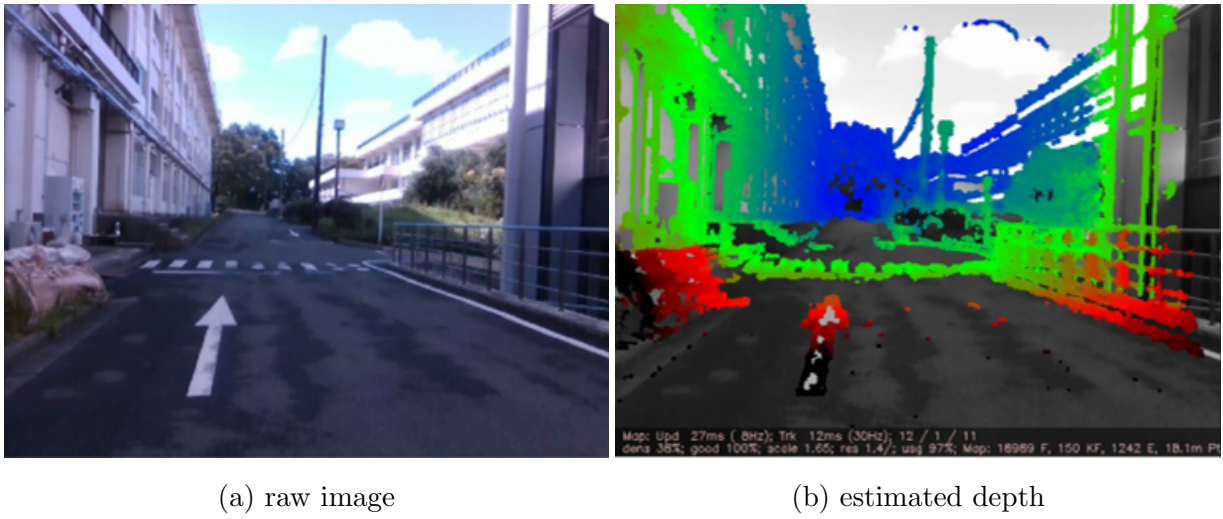


図 3.2: Estimation of depth by LSD-SLAM

LSD-SLAM: Large-Scale Direct SLAM is monocular camera SLAM[9]. In (b), red pixels represent shorter depth, blue pixels represent longer depth. The rate of the camera pose estimation is about 30 Hz.

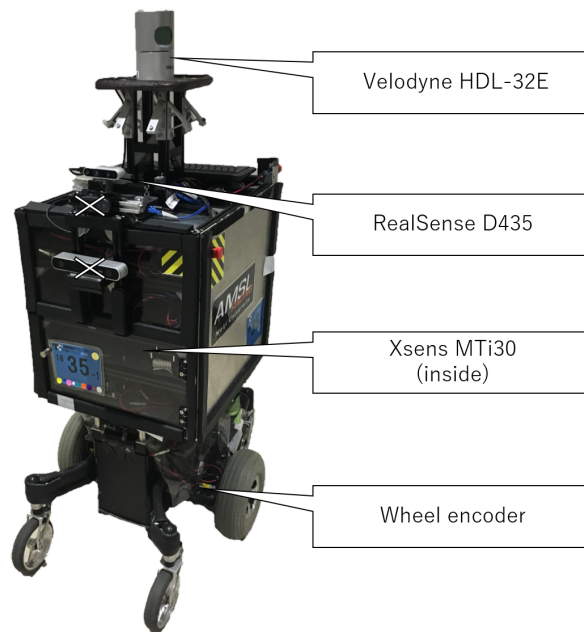
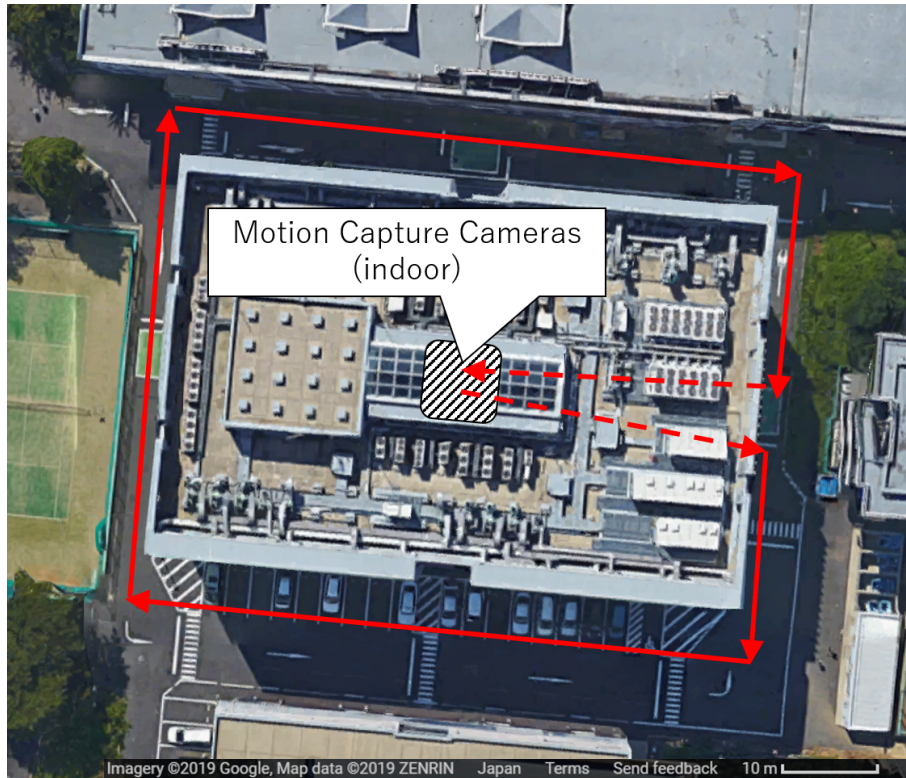


図 3.3: The experimental robot

The experimental mobile robot has 3D-LIDAR, camera, IMU and wheel encoders. RealSense D435 is RGBD camera, however this is just for LSD-SLAM and depth information is not used in the experiment.



(a) aerial shot



(b) motion capture area

☒ 3.4: The experimental environment

An aerial shot of the experimental environment is shown here. Motion capture cameras locate on indoor area of a building. The robot starts driving at that area and comes back there. The course the robot drives has some slopes.

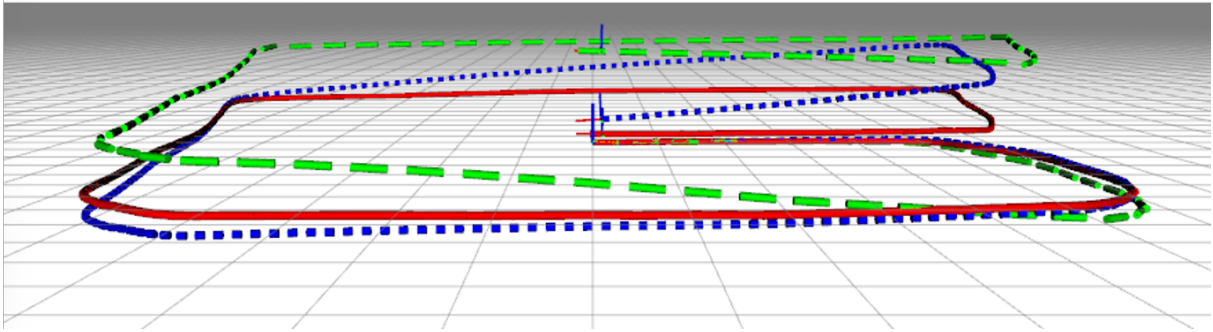
### 3.3 実験結果

表 3.1 に、各手法での推定位置姿勢の誤差を示す。表中の誤差は、ワールド座標系の座標軸を基準とした位置および姿勢角の誤差である。また、Euc.dist. は、ユークリッド距離で表した位置推定誤差である。図 3.5 に、各手法での推定されたロボットの軌跡を示す。ユークリッド距離で表した位置推定誤差に関して、比較手法に比べて、提案手法の値が小さいことから、走行中の姿勢推定の誤差も提案手法の方が小さいと言える。特に、 $Z_w$  軸（鉛直）方向の並進誤差が小さく抑えられており、壁面を用いたロール、ピッチ補正による結果だと考察する。図 3.6 に、図 3.4 と同様のコースを 3 周した場合の推定軌跡を示す。ユークリッド距離で表した位置推定誤差はそれぞれ、提案手法で 2.762m、LSD-SLAM で 10.544m、ジャイロスコープによるデッドレコニングで 3.393m である。走行距離が長くなった分、走行終了後のユークリッド距離誤差が大きくなったものの、 $Z_w$  軸方向の並進誤差が小さく抑えられる傾向が再現されたと言える。

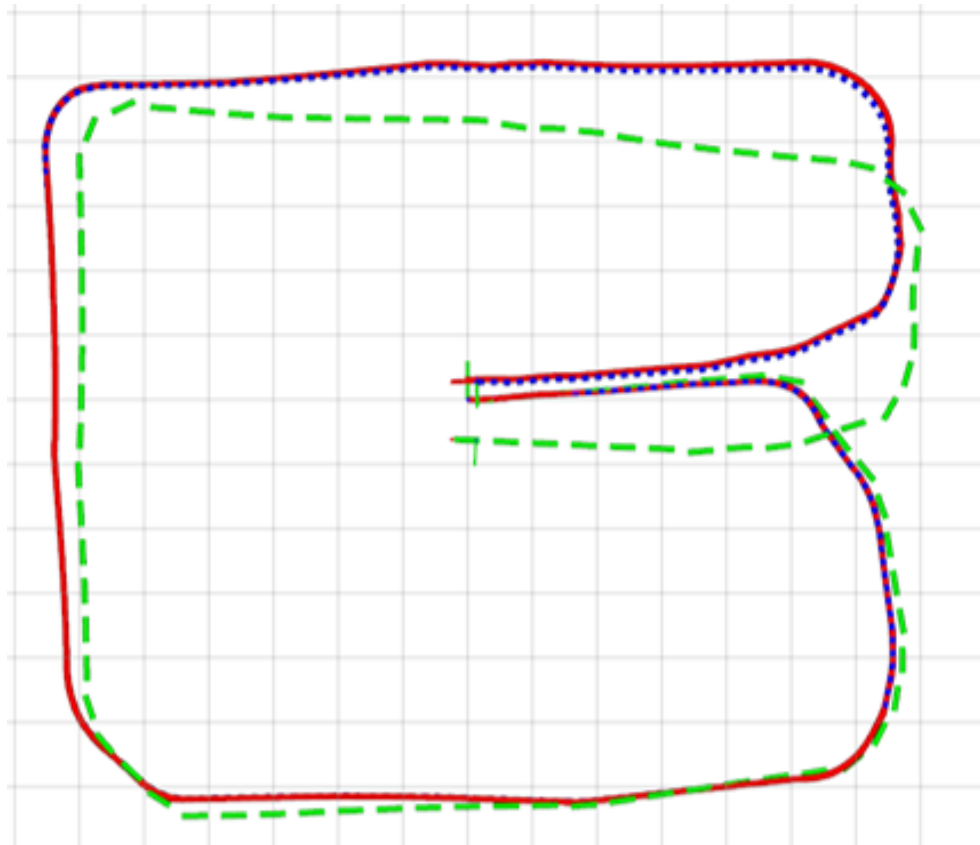
表 3.1: Errors of position and pose estimations in 1 round

error in...	proposed method	LSD-SLAM	Gyrodometry
$X_w$ [m]	+1.1692	+0.9382	+1.1079
$Y_w$ [m]	+1.1338	-3.2837	+1.2586
$Z_w$ [m]	+0.4831	+7.0578	+1.7022
Euc.dist.[m]	1.699	7.841	2.389
roll $_w$ [deg]	-1.471	-1.520	-2.681
pitch $_w$ [deg]	+1.073	+0.168	+3.351
yaw $_w$ [deg]	+2.006	-1.081	+2.331

- Proposed method —
- LSD-SLAM - - -
- Gyrodometry ⋯



(a) perspective view

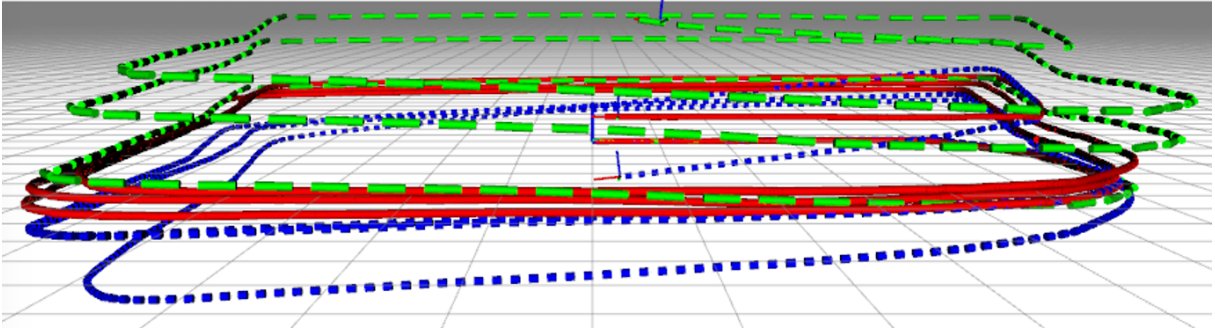


(b) x-y plane

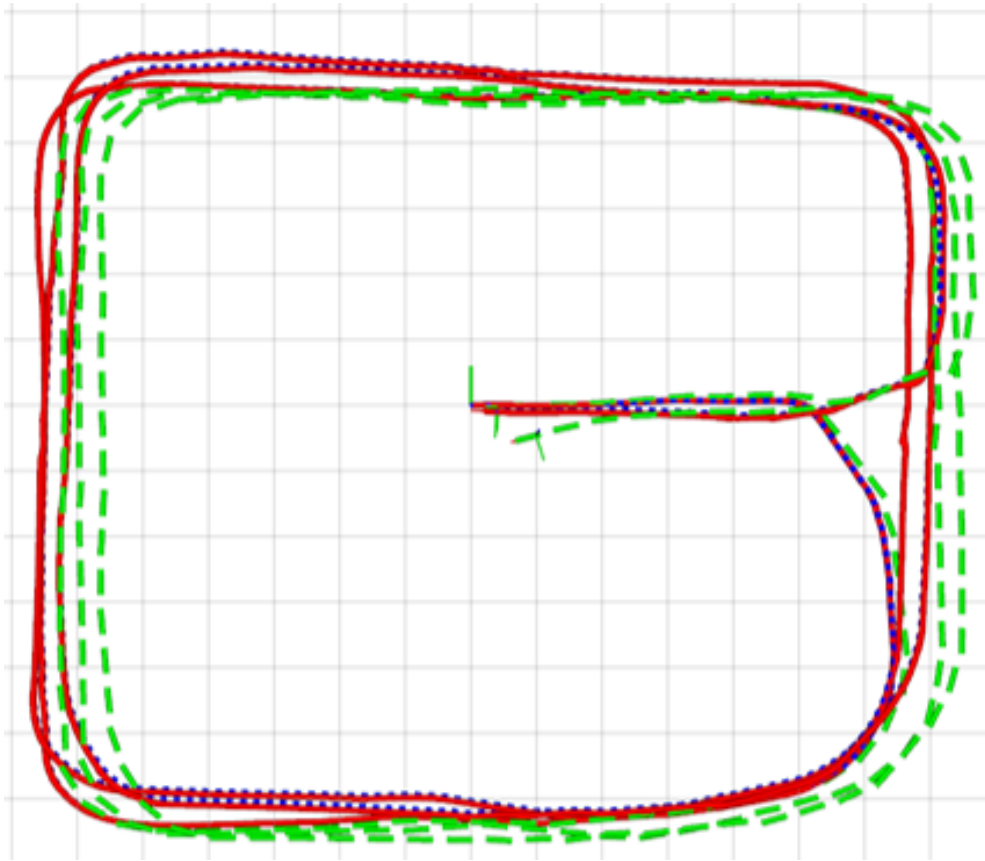
图 3.5: Estimated trajectories of the robot in 1 round

It is visually found that the trajectory estimated by the proposed method have smaller translational errors in  $Z_w$  axis than those estimated by the others.

- Proposed method —
- LSD-SLAM - - -
- Gyrodometry ⋯



(a) perspective view



(b) x-y plane

图 3.6: Estimated trajectories of the robot in 3 rounds

The tendency that the trajectory estimated by the proposed method have smaller translational errors in  $Z_w$  axis can be seen here, similar as the 1-round experiment.

### 3.4 追加実験

鉛直でない平面が提案手法に与える影響を，簡易的な実験によって検証する．鉛直面で囲まれた屋内空間に，鉛直でない平面を持つ障害物を設置し，第2.3節で記述したように鉛直面を観測し姿勢を推定する．ただし，ロボットは静止させ，IMUを使ったデッドレコニングおよびSLAMは行わない．また，第2.3.4項のキャリブレーションは行わない．実験環境が，屋内であり十分整備されているため，ロボットの姿勢角の真値は $\phi_{\text{true}} = 0$ ， $\theta_{\text{true}} = 0$ とする．障害物として，表3.2のような掲示用のボードとアルミ板を使用し，傾斜角や障害物の種類および数を変えて比較した．図3.7に追加実験の様子，図3.8に追加実験環境の点群をそれぞれ示す．

表3.3に追加実験の結果を示す．表内の誤差は，1分間の平均である．また，実験結果の様子为例として，追加実験no.3の3次元点群および法線群を図3.9に示す．なお図中のextracted normalは，第2.3.1項で設定される条件を満たし，信頼性が高いと判定された法線，clustered normalは主法線を表す．障害物の傾きが大きくなるほど，推定姿勢角の誤差は大きくなる．ただし，式(2.17)の $\beta$ が閾値を超えた場合，追加実験no.4のように障害物の影響がなくなる．追加実験no.4の様子を図3.10に示す．また，障害物が持つ平面の面積が大きいほど，推定姿勢角の誤差は大きくなる．第2.3.2項の通り，観測された法線の数が多い場合，クラスタが持つメンバの数で重み付けを行っているからだ．ただし，メンバの数が閾値を下回った場合外れ値とするため，追加実験no.5のように，障害物の影響がなくなる．追加実験no.5の様子を図3.11に示す．信頼性の高い法線であるかの判定や，クラスリングの外れ値であるかの判定の基準はパラメータで調節している．判定基準を厳しくすれば，鉛直面の誤認識を防げるが，本来の目的である補正がかかりづらくなってしまふ．想定される環境に応じて，適当に設定する必要がある．

表 3.2: Detail of obstacles

name	dimension	description
small plane	400 mm × 600 mm	aluminum board
large plane	1530 mm × 900 mm	bulletin board



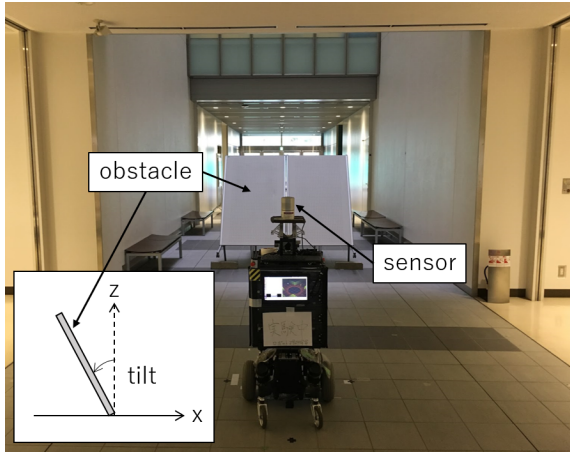


图 3.7: The additional experimental environment

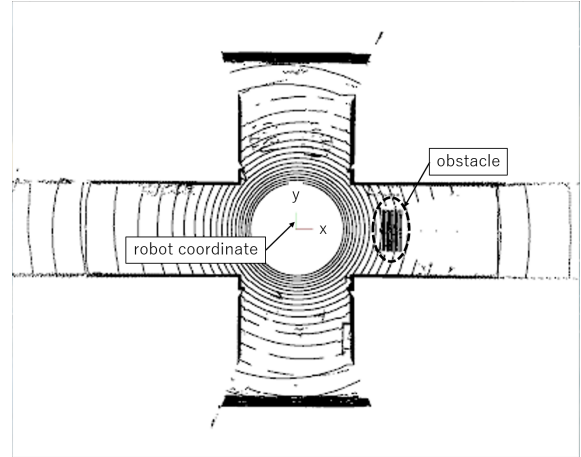


图 3.8: Point cloud of the additional experimental environment

表 3.3: Errors of pose estimations with obstacles which are not vertical planes

no.	obstacle type	tilt[deg]	roll error[deg]	pitch error[deg]	recital
1	no obstacle	-	+0.103	-0.831	-
2	large plane×1	15	+1.24	+1.57	-
3	large plane×1	30	+1.89	+3.01	-
4	large plane×1	45	+0.104	-0.831	out of the angle range
5	small plane×1	30	+0.113	-0.832	outlier
6	large plane×2	30	+3.96	+7.38	-

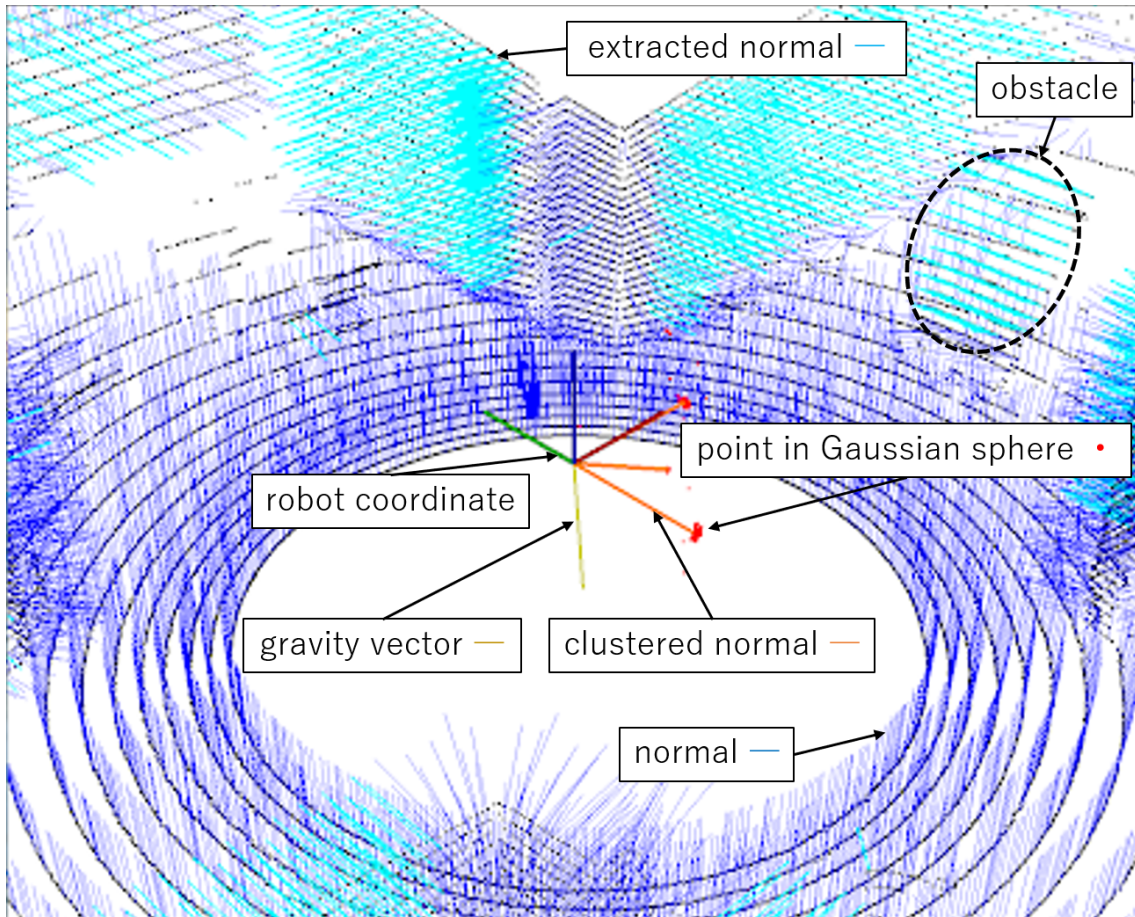


图 3.9: Point and normal cloud of the additional experiment no.3

Extracted normals are picked out from the original normals, with conditions for judging whether a normal has a vertical plane. These extracted normals are projected to a Gaussian sphere. And the points in the Gaussian sphere are clustered. Normals on the obstacle which is not vertical make the estimation of the gravity vector worse.



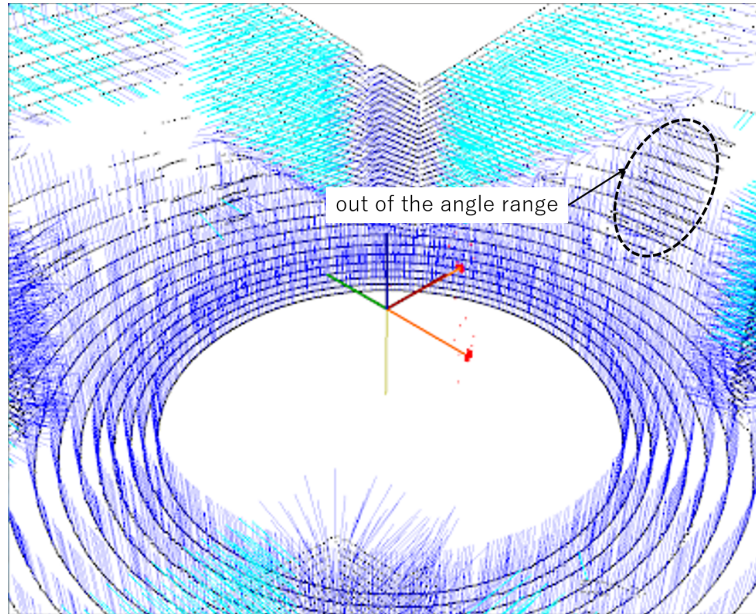


图 3.10: Point and normal cloud of the additional experiment no.4

There is no influence for the pose estimation when experimental obstacles are tilted more than the threshold which the proposed method has.

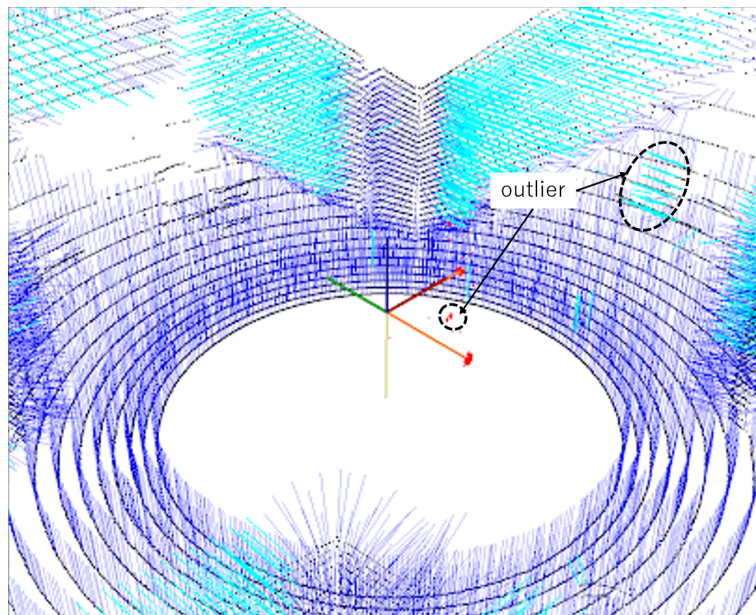


图 3.11: Point and normal cloud of the additional experiment no.5

There is no influence for the pose estimation when experimental obstacles are small enough to be recognized as outliers in the clustering step.

## 第4章 結論

従来の慣性センサと SLAM による移動ロボットの姿勢推定に加え，鉛直に建てられている建造物の壁に対する相対姿勢を利用する姿勢推定法を提案した．実験結果より，本提案手法の，蓄積誤差に対する補正の有用性が示された．

# 謝辞

本研究を進めるにあたり，ご指導を頂いた指導教員の黒田洋司教授に心より感謝申し上げます。また，機械工学科の先生方からは中間審査会等にて貴重なアドバイスをいただきました。そして，ロボット工学研究室の皆様にはたくさんの相談や質問をさせていただき，丁寧に対応していただきました。さらに，本研究の一部は，国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の，次世代人工知能・ロボット中核技術開発事業による支援を受けた。ここに篤く御礼申し上げます。

## 参考文献

- [1] S. Kwon, S. Kim, and J. Yu: “Tilting-Type Balancing Mobile Robot Platform for Enhancing Lateral Stability”, *IEEE/ASME transactions on mechatronics*, Vol.20, No.3, pp.1470–1481, 2015.
- [2] S. Thrun, W. Burgard and D. Fox: “Probabilistic robotics”, *The MIT Press*, pp.309–336, 2005.
- [3] L. von Stumberg, V. Usenko and D. Cremers, “Direct Sparse Visual-Inertial Odometry using Dynamic Marginalization”, *in International Conference on Robotics and Automation*, pp.2510-2517, 2018.
- [4] S. Rusinkiewicz and M. Levoy: “Efficient Variants of the ICP Algorithm”, *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, pp.145-152, 2001.
- [5] P. Biber and W. Straßer: “The Normal Distributions Transform: A New Approach to Laser Scan Matching”, *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, Vol.3, pp.2743–2748, 2003.
- [6] M. Magnusson, A. Lilienthal and T. Duckett: “Scan registration for autonomous mining vehicles using 3D NDT”, *Journal of Field Robotics*, vol.24, no.10, pp.803–827, 2007.
- [7] J. Zhang and S. Singh, “LOAM: Lidar Odometry and Mapping in Real-time”, *in Robotics: Science and Systems Conference*, pp.161-195, 2014.
- [8] 清水尚吾, 黒田洋司: “主平面を用いた点群の高速位置合わせ”, 第19回ロボティクスシンポジウム講演予稿集, pp.453–458, 2014.
- [9] J. Engel, J. Stueckler and D. Cremers: “LSD-SLAM: Large-Scale Direct Monocular SLAM”, *European Conference on Computer Vision*, pp.834–849, 2014.
- [10] C. Kerl, J. Sturm, and D. Cremers: “Dense Visual SLAM for RGB-D Cameras”, *Proc. IEEE/RSJ Intl. Conf. on Intelligent Robot Systems*, pp.2100-2106 2013.

- [11] J. Engel, V. Koltun and D. Cremers: “Direct Sparse Odometry”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.40, No.3, pp.611–625, 2018.
- [12] M. A. Quddus, W. Y. Ochieng, and R. B. Noland: “Current map-matching algorithms for transport applications: State-of-the art and future research directions”, *Transportation Research Part C: Emerging Technologies*, Vol.15, No.5, pp.312–328, 2007.
- [13] S. Ramalingam, S. Bouaziz, P. Sturm and M. Brand: “SKYLINE2GPS: Localization in Urban Canyons Using Omni-Skylines”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.3816-3823, 2010.
- [14] 後藤 翼, S. Pathak, 池 勇勳, 藤井 浩光, 山下 淳, 淺間 一: “人工物環境における全天球カメラの位置姿勢推定のための直線特徴に基づく 3D-2D マッチング”, *精密工学会誌*, Vol.83, No.12, pp.1209–1215, 2017.
- [15] A. Y. Hata, F. S. Osorio, and D. F. Wolf: “Robust Curb Detection and Vehicle Localization in Urban Environments”, *IEEE Intelligent Vehicles Symposium Proceedings*, pp.1257-1262 2014.
- [16] J. Ryu, E. J. Rossetter and J. C. Gerdes: “Vehicle Sideslip and Roll Parameter Estimation using GPS”, *Symposium on Advanced Vehicle Control*, pp.373-380, 2002.
- [17] H.S. Bae, J. Ryu and J.C. Gerdes: “Road Grade and Vehicle Parameter Estimation for Longitudinal Control Using GPS”, *Proceedings of IEEE Conference on Intelligent Transportation Systems*, pp.166–171, 2001.
- [18] 李 博, 張 曉林, 佐藤 誠: “車間距離計測のための車載単眼カメラを用いたピッチ角推定”, *映像情報メディア学会誌*, 69 卷 4 号, pp.169–176, 2015.
- [19] M. Pauly, M. Gross and L.P. Kobbelt: “Efficient simplification of point-sampled surfaces”, *Proceedings of the conference on Visualization*, pp.163–170, 2002.
- [20] J.L. Bentley: “Multidimensional binary search trees used for associative searching”, *Communications of the ACM*, Vol.18, No.9, pp.509-517, 1975.
- [21] B.K.P. Horn: “Extended gaussian images”, *Proceeding of the IEEE*, vol.72, no.12, pp.1671–1686, 1984.