

# 森林作業道における移動ロボットの 自律移動に関する研究

山梨大学大学院  
医工農学総合教育部  
博士課程学位論文

2022年3月

中込 広幸

# 論文概要

本研究は、森林作業道において車両の自律移動を実現するための研究である。森林作業道とは、林業において木材の搬出作業を行うための道路であり、集材作業を行う林業機械や、搬出作業を行うフォワードの通行を目的に使用される。本研究における自律移動とは、車両が与えられた経路情報に従ってスタート地点からゴール地点まで自律的な移動制御により到達することを言う。自律移動を実現するには、

- 走行経路に沿った地図の取得
- 地図に対する現在位置の推定
- 目標経路に対する誘導制御

などの要素技術が統合されなければならない。さらに、これら要素技術は森林作業道という特殊環境において実現されなければならない。

本論文は、これら要素技術についての研究報告、および要素技術を統合したロボットによる自律移動の実験結果について報告し、将来的にフォワードの無人化を目指すものである。

## 目標タスクの設定

森林作業道は自然を含む環境のため多様である。すべての森林作業道の自律移動を確実に行うことは困難である。そこで、本研究では自律移動の対象とする森林作業道をあらかじめ定めることで、環境を限定する。森林作業道として基本的な特徴をもつ環境を、対象とする森林作業道として設定した。将来的に様々な森林作業道の自律移動に応用するための、基礎的な手法を構築することを目的とする。実現目標とするタスクは、森林作業道を走行する移動ロボットの自律移動とする。

## 対象とする森林作業道

本研究において対象とする森林作業道は、森林作業道作設ガイドライン [1][2] を参考に、森林作業道の基本的な特徴を持つ環境を設定した。森林作業道の基本的な特徴とは、

- 木材を搬出する車両が通行することを目的としていること
- 路面は、土砂を転圧したものであること (アスファルト等での舗装はされていないこと)

- 森林作業道の幅員は、通行を想定するフォワーダの幅に対して、最低でも 0.25m の余裕が確保されていること
- 斜面に敷設され、谷側斜面・山側斜面が存在すること

以上を満たす環境を山梨県内に 2 箇所設定し、実験を行う。

## 対象とするロボット

対象とするロボットは差動駆動型の 4 輪移動ロボットとした。フォワーダは主にクローラベルトにより走行し、2 自由度のアクチュエータにより駆動するため、独立 2 輪駆動型の移動機構モデル [3] を適用することができる。差動駆動型の 4 輪移動ロボットも同様に、独立 2 輪駆動形の移動機構モデルを適用できる。このことから、対象とするロボットとして差動駆動型の 4 輪移動ロボットを選択した。将来的にフォワーダの無人化に応用する際、同じ制御方法を適用することができる。

## 目標タスク実現のための基本戦略

屋内および屋外整地における自律移動の基本的な方法は、既に確立されている。しかし、これらに使用されてきた各要素技術は、森林作業道において利用困難である。例として自己位置推定においては、以下の通り。

- GNSS(Global Navigation Satellite System) による自己位置推定：衛星電波が地形や森林により遮蔽
- デッドレコニングによるオドメトリ推定：不整地走行時に、車輪・クローラベルトが路面に対して滑り、オドメトリの誤差が発生
- ビジュアルオドメトリ：環境光の変化が頻繁に発生するためサチュレーションが発生、不整地走行時の振動によりモーションブラーが発生
- スキャンマッチングによる自己位置推定：人工物が少ないため、スキャンマッチング時にローカルミニマムが発生

さらに、路面の推定においては、以下が困難である。

- 未舗装：走行可能な領域の推定が困難
- 学習用のデータが市街地と比較して極めて少ない

このように、これまで使用されていた要素技術の適用は困難である。自律移動を行うためには、森林作業道に対応した新たな要素技術が必要となる。そこで、本研究では従来使用されてきた要素技術について、森林作業道における自律移動に適用するための改良を行う。

## 接地面と車輪との滑りに影響されないスキャンマッチング

ICP アルゴリズムを用いたスキャンマッチング手法は、初期値が解に近い場合にのみ有効な手法である。このため、初期値と解が離れる場合、デッドレコニング・慣性計測装置により求めたオドメトリ推定値を、初期値として使用する。しかし、森林作業道では、急旋回や車輪・クローラベルトと路面との滑りによって、オドメトリの推定が困難である。さらに、森林作業道の周辺形状に起因した局所解が多く存在しうるため、さらにスキャンマッチングによる自己位置推定が困難となる。

これらに対応するため、本研究では、ICP アルゴリズムにおいて位置・姿勢・並進速度を同時に推定する。並進速度を求めることで、デッドレコニングを使用せずに、次スキャン時の初期値を解に近づける方法を提案する。これにより、路面との滑りに影響されないスキャンマッチングが可能となる。さらに、ICP マッチング時、ボクセルを用いた特徴量による重み付けを行うことで、スキャンマッチングの精度を低下させる原因となる、ばらつきの大い形状の影響を軽減した手法を構築する。

## 深層学習を用いた路面推定

森林作業道における路面の推定方法としては、教師データに基づいて、ロボットが自動的に路面を推定する方法を提案する。森林作業道の路面は公道と異なり、舗装されておらず、白線や縁石といった目印となるものが存在しないため、路肩との境界が明確でない。さらに、崩落しないよう、転圧された領域を路面として推定し、転圧された領域のみ通行しなければならない。

これらに対応するため、本研究では、LiDAR-SLAM により得られた TopView 画像を基に、U-Net による路面の推定方法を提案する。TopView 画像を使用することで、後方を含めた広範囲の形状を基に路面の推定を行うことができる。さらに U-Net を採用することで高精度な路面推定を可能とする。

## 森林作業道における自律移動システムの構築

森林作業道の自律移動を実現するために必要な機能として、3D-LiDAR を用いた SLAM による地図作成、自己位置推定、および経路計画手法を構築し、自律移動システムとする。自律移動システムにより、指定した経路に沿った自律移動を実現する。自律移動システムのうち、地図作成と自己位置推定を行う機能は、提案したスキャンマッチング法を用いて構築する。さらに、経路計画機能は、提案した路面検出法、および Dynamic Window Approach を用いて実現する。

## 森林作業道における自律移動の実験とその結果

森林作業道における自律移動の評価方法を定め、対象とする森林作業道において、ロボットの自律移動実験を行った。自律移動システムの評価として、搭載した車両が位置推定誤差により作業道を逸れ、滑落することがないか確認しなければならない。このため、経路に対して左右方向（森林作業道の横断方向）に生じた、自律移動ロボットの位置制御誤差を算出し、評価した。横断方向の位置制御誤差が 0.25 m 以内であることを目標とし、実験結果の解析により、目標とする精度での自律移動が可能であることを確認した。

## まとめ

本研究では、森林作業道における自律移動を目標タスクとし、タスクが実現可能な自律移動システムの構築を行った。森林作業道に対応したスキャンマッチング手法、および路面検出法を提案し、これらを使用した自律移動システムを構築した。目標タスクである森林作業道における自律移動の評価として、自律移動時における横断方向の位置制御誤差を測定し、目標とする精度での自律移動が可能であることを確認した。

# Contents

1	はじめに	8
1.1	概要	8
1.2	本研究の背景	8
1.2.1	森林作業道とは	8
1.2.2	林業における問題とフォワードの自律移動の必要性	8
1.2.3	これまでの森林作業道における自律移動	9
1.2.4	近年の自律移動手法と森林作業道に対する適用	10
1.3	本研究の目的	11
1.4	目的を達成するための方法	11
1.4.1	スキャンマッチングにおける問題と提案手法	11
1.4.2	路面推定における問題と提案手法	12
1.5	森林作業道における自律移動の実験と検証	12
1.5.1	森林作業道の幅員と横断方向の精度	13
1.5.2	本研究成果の活用	14
1.6	論文の構成	14
2	森林作業道において自己位置推定可能なスキャンマッチング	15
2.1	概要	15
2.2	森林作業道におけるスキャンマッチングの要件	15
2.3	関連研究	16
2.3.1	ICP アルゴリズムを使用したスキャンマッチング	16
2.3.2	Motion distortion に対応したスキャンマッチング	17
2.3.3	森林作業道に対応したスキャンマッチングと従来手法との比較	17
2.4	提案手法	18
2.4.1	提案手法の概要	18
2.4.2	センサ入力と前処理	18
2.4.3	歪み補正	19
2.4.4	スキャンマッチングによる位置姿勢, および並進速度の推定	20
2.4.5	点群の蓄積	21
2.5	提案手法の評価	22
2.5.1	実験に用いた装置	22
2.5.2	RTK-GNSS を用いた予備実験	22
2.5.3	横断方向の位置推定誤差の算出方法	23
2.5.4	RTK-GNSS を用いた予備実験の結果	24
2.5.5	予備実験の結果に対する考察	27

2.5.6	森林作業道における実験	28
2.5.7	森林作業道における位置推定誤差の算出方法	28
2.5.8	森林作業道における実験の結果	28
2.5.9	森林作業道の実験結果に対する考察	33
<b>3</b>	<b>深層学習を用いた路面推定</b>	<b>34</b>
3.1	概要	34
3.2	関連研究と森林作業道の路面検出における課題	34
3.2.1	関連研究	34
3.2.2	森林作業道の路面推定における課題	35
3.3	ハードウェア構成	35
3.4	LiDAR-SLAM と U-Net を用いた路面検出	36
3.4.1	LiDAR-SLAM による点群の取得	37
3.4.2	TopView 画像の作成	37
3.4.3	U-Net のアーキテクチャ	39
3.4.4	データセットの作成方法	39
3.5	実験	41
3.5.1	データセットの作成と学習	41
3.5.2	路面検出の精度評価	42
3.5.3	路面輪郭の精度評価	44
3.6	考察	46
3.6.1	IoU の算出結果に対する考察	46
3.6.2	路面輪郭精度の算出結果に対する考察	46
<b>4</b>	<b>森林作業道における自律移動システムの構築</b>	<b>48</b>
4.1	概要	48
4.2	森林作業道の自律移動に必要な機能	48
4.2.1	自動化を行う作業	48
4.2.2	自律移動を実現するための機能	49
4.2.3	自律移動時の環境・条件	50
4.2.4	自律移動時における速度	51
4.3	地図作成機能の構築	51
4.3.1	運用時の要求事項	51
4.3.2	スキャンマッチングによる地図作成の手順	52
4.3.3	スキャンマッチングによる自己位置推定と点群の復元	52
4.3.4	位置姿勢推定値と歪み補正済みの点群の蓄積と統合	52
4.3.5	地図作成機能の処理時間の処理時間	53
4.3.6	森林作業道における地図作成の実行例	53
4.4	自己位置推定機能の構築	55
4.4.1	スキャンマッチングによる自己位置推定の手順	55
4.4.2	スキャンマッチングによる自己位置推定と点群の復元	55
4.4.3	処理時間	56
4.4.4	森林作業道における自己位置推定の実行例	56
4.5	経路計画機能の構築	57
4.5.1	経路計画の手順	57

4.5.2	目標経路の取得	57
4.5.3	走行経路上の点群を入力とした路面推定	57
4.5.4	走行経路上の点群を入力とした障害物判定	57
4.5.5	Dynamic Window Approach によるパスの生成	58
4.5.6	処理時間	58
4.5.7	森林作業道における経路計画機能の実行例	58
<b>5</b>	<b>森林作業道における自律移動の実験と評価</b>	<b>60</b>
5.1	概要	60
5.2	評価方法	60
5.2.1	横断方向の位置制御精度の算出	60
5.2.2	実験に用いたハードウェア・ソフトウェア	61
5.3	実験手順	61
5.3.1	地図と移動経路の取得	61
5.3.2	自律移動の実施	61
5.4	森林作業道 A における実験	61
5.4.1	実験環境	61
5.4.2	実験結果	62
5.5	森林作業道 B における実験	65
5.5.1	実験環境	65
5.5.2	実験結果	67
5.6	考察	69
<b>6</b>	<b>おわりに</b>	<b>70</b>
6.1	本研究の目的	70
6.2	目的を達成するための方法	70
6.2.1	スキャンマッチング手法の提案	70
6.2.2	路面推定手法の提案	70
6.3	森林作業道における自律移動の実験とその結果	71
6.4	本研究の成果と今後の課題	71
	謝辞	72
	参考文献	73
	研究業績	77

# Chapter 1

## はじめに

### 1.1 概要

本章では、本研究の背景として、林業における問題、森林作業道における運搬作業と自動化の必要性、およびこれまでの取り組みについて述べる。さらに、本研究の目的、目的を達成するためのアプローチ方法、目的達成を確認するための検証方法、および研究成果の活用方法について述べる。最後に、本論文の構成について述べる。

### 1.2 本研究の背景

#### 1.2.1 森林作業道とは

林業では造林、間伐、丸太生産の為の伐採で森林に出入りするため、森林作業道と呼ばれる特定の林業従事者が使用する道路が整備される (Fig. 1.1)。森林作業道の特徴は、林業機械 (6~13 トン) や小型トラック (2 トン積程度) の走行を想定して作られたものであり、作設に当たっては、経済性を確保しつつ丈夫で簡易な構造とすることが求められる。森林作業道の敷設される場所は主に山間部であるため、傾斜のある環境に敷設されることとなる。森林作業道は、森林を切り開き、傾斜を切土・盛土し、転圧することで敷設される。このため、森林作業道は勾配があるにもかかわらず、コンクリート等による舗装は無い。さらに、勾配が強くなると切土・盛土が困難となるため、十分な幅員を確保することが難しい。

#### 1.2.2 林業における問題とフォワーダの自律移動の必要性

林業では足場の悪い山の中で重量物を取り扱うため、労働災害の発生率が全産業の中で最も高くなっている。危険作業を伴うため林業就業者が減少し、担い手不足の問題が発生している。労働災害における発生原因の一つとして、森林作業道の通行が挙げられる。森林作業道は、低コストな作設が可能である一方、強度や幅員を得ることが難しいため、車両の走行時に変形や崩落し、滑落による死亡事故が発生している。特に、フォワーダ (Fig. 1.2) と呼ばれる林業機械を使った運搬作業では、木材を搭載するため、本体の重量に加え、木材の積載により重量が増加する。さらに、山間部から木材を切り出すことから、積載時は下りとなり、フォワーダが加速しやすい状態となっている。予想以上の速度が出たために路面との滑りにより減速・停車ができず、滑落するという事故が発生している [4]。さら

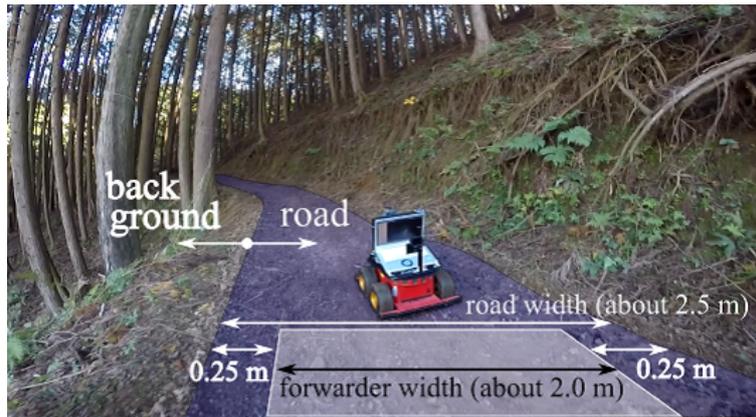


Fig. 1.1: Example of a forest road.



Fig. 1.2: The Forestry vehicle: Forwarder.

に、コスト面においても問題がある。主に勾配のある環境でしか木材を生産ができないことから、平地での生産と比較して搬出コストが高くなる。このため、輸入材に価格競争で勝てないという問題が発生している。

これらの問題を解決する方法の一つとして、フォワーダの自律移動化が挙げられる。フォワーダを自動化し、運用することができれば、無人化により滑落事故の労働災害を低減することが可能となる。さらに、フォワーダの運転に要した人材を削減することができるため、搬出作業のコストダウンを行うことが可能となる。フォワーダを自律移動化することで、フォワーダによる木材の集荷・運搬作業の安全性確保、人材不足の解消、搬出コストの削減が可能である。

### 1.2.3 これまでの森林作業道における自律移動

フォワーダの自律移動に関する取り組みは、既にいくつかの事例が存在している。毛綱ら [5, 6] は、デッドレコニングにより、フォワーダの自己位置推定を行い、指定した

経路を自動走行する方法を提案している。デッドレコニングにおいて、車速計による速度測定、および光ファイバジャイロによる角速度測定を行い、得られた速度、角速度を積分することで自己位置を推定する。デッドレコニングの問題として、センサのノイズ、および車輪・クローラベルトと路面との滑りにより誤差が蓄積する。毛綱らは、路面に永久磁石を設置することでデッドレコニングに生じた誤差を修正している。

毛綱らは、さらに森林作業道に誘導電線を敷設し、磁気センサを用いてライントレースを行う方法を提案している [7]。森林作業道は、敷設コストの低減や勾配の関係から幅員を広くとることが難しく、目標経路に対する道路横断方向の誤差を十分小さくしなければならない。ライントレースによる方法であれば、誘導電線に沿った正確な走行が可能である。

地磁気センサと QR コードを用いた自動走行手法 [8] では、あらかじめ設置した QR コードを読み取ることで直進・停止・旋回を行い、地磁気センサから得られる方位を用いて旋回角度を制御する。旋回角度や並進速度などのパラメータを、事前に調整しなければならないが、誘導電線を設置する必要がないため、林業機械の走行による、誘導電線の切断やズレを考慮する必要が無い。

これら森林作業道における自動走行手法の問題点は、永久磁石、誘導電線、および QR コードの設置が必要とされることである。有人による運用時には必要のなかった、永久磁石、誘導電線、および QR コードの設置作業、運用前の調整、運用時の保全、および撤去作業が必要となる。設置・撤去等に関わる作業は、無人走行によるコスト低減に反するとともに、走行距離が伸びるほど無人化のためのコストが増加していくこととなる。さらに各手法ごとの問題として、誘導電線を使用する場合は、他の林業機械の走行により電線の切断や変形の可能性がある。QR コードを使用する場合、林業機械の走行等による、路面の変形に伴い、路面や路肩に設置した QR コードの設置位置が変化する。さらに、シカなどの野生生物の食害により、設置後の QR コードが、設置位置から移動、もしくは紛失する可能性がある。

#### 1.2.4 近年の自律移動手法と森林作業道に対する適用

近年実用化されている自動運転技術では、電線や QR コードなどの指標（マーカと呼ばれる）が不要な手法が提案されている。自動車の自動運転では全地球即位システム (Global Navigation Satellite System, 以下, GNSS と称する), 慣性計測装置 (以下, IMU と称する), およびデッドレコニングによる自己位置推定を行い、指定された経路を走行することができる。さらに、GNSS を必ずしも必要としない手法として、LiDAR, またはカメラを用いて、SLAM (Simultaneous Localization and Mapping) [9, 10] により事前に地図を作成し、自律移動を行う手法についても提案されている。例として、Autoware [11, 12] と呼ばれる自律移動システムは、主に LiDAR を使用することで自律移動を実現している。LiDAR により収集された 3 次元形状情報を地図として使用し、さらに、自律移動時は LiDAR から得られた 3 次元形状情報と、地図の形状を比較することで自己位置推定が行われる。自己位置推定結果をもとに、地図上の指定した経路を沿うよう操舵することで、自律移動を実現している。

これら近年の自動運転技術をフォワードの自律移動に応用することができれば、森林作業道における、マーカレスな自律移動が実現できる。しかし、森林作業道の特徴が、これら手法の導入を困難にしている。

森林作業道は、採算に見合うよう低コストでの敷設が必要である。さらに、主に山間部などの勾配のある環境に敷設しなければならない。このため、路面の幅員を広くとること

が困難となっている。森林作業道において自律移動を行う場合、狭い路面から逸脱しないよう、高精度な自己位置推定と走行制御が要求される。GNSSを使用した自律移動を行う場合、RTK(Real Time Kinematic) 測量法による高精度な自己位置推定が必要となる。しかし、森林作業道は主に山間部に存在し、かつ森林作業道の周辺は樹木に囲まれる。このことから、衛星信号の遮蔽による自己位置推定誤差が発生し、自律移動が困難であることが報告されている [13]。さらに、デッドレコニングを使用した自己位置推定は、不整地を走行することから、車輪・クローラベルトと路面との滑りが発生し、正しい移動量を求めることができない。カメラを使用した自己位置推定は、不整地走行時の振動に起因するモーションブラーの発生が問題となる。さらに、周辺の木々による環境光の変化を原因としたサチュレーションが発生する。

本研究では、森林作業道の自己位置推定に最も適したセンサはLiDARであるとして、自律移動手法の構築を行った。LiDARはカメラと比較して、環境光の影響が小さく安定したセンシングが可能である。さらに、Autoware[11, 12]等のシステムにおいて、LiDARを用いた自律移動手法はすでに確立されている。自己位置推定においてカメラやデッドレコニングが不要となる方法も存在する [14, 15, 16, 17]。

このように、森林作業道の自律移動では、LiDARを用いた既存の自律移動手法が適用できると考えられるが、要素技術である自己位置推定・路面検出において、森林作業道特有の問題を解決する必要があることを確認した。本研究は、LiDARを使用した自律移動を実現することを最終的な目標とし、要素技術である自己位置推定と路面推定を提案することで、森林作業道特有の問題解決を図る。

## 1.3 本研究の目的

本研究の目的は、目標タスクである森林作業道の自律移動を可能とすることである。そのため、要素技術である、森林作業道の自律移動に応用可能なスキャンマッチング手法・路面推定手法を提案する。その後、本研究にて提案するスキャンマッチング手法を応用し、地図作成機能、地図中の自己位置推定機能を開発する。さらに、本研究にて提案する路面推定手法と既存の経路生成アルゴリズムを組み合わせることで、パスと呼ばれる、ロボットを安全に走行するための目標経路を生成し、走行指令値の出力を行う機能(以下、経路計画機能と称する)を開発する。地図作成機能、地図中の自己位置推定機能、経路計画機能により、自律移動に必要なシステムを構築する。

## 1.4 目的を達成するための方法

### 1.4.1 スキャンマッチングにおける問題と提案手法

スキャンマッチングとは、LiDARにより得られた複数の点群を重ね合わせる技術である。スキャンマッチングにより、重ねあわせに要した距離や回転角度を求めるとともに、複数の点群を統合することができる。これを応用することで、周辺の地図の作成、作成した地図に対する現在位置の取得、および現在位置を利用した自律移動制御が可能となる。しかし、従来手法のスキャンマッチングは形状の重ね合わせを行うという性質上、森林作業道での運用が難しい。その原因は主に2つある。

森林作業道での自律移動が困難である原因の一つとして、不整地の走行により発生する

点群の歪みが挙げられる。LiDARはToF(Time of Flight)方式のレーザセンサを回転させることで測定視野を広げている。LiDARを搭載した車両が走行すると、点群の測定中にLiDAR本体が移動するため、移動量に応じて点群の座標を修正しなければならない。この測定中の移動量は通常、デッドレコニング、およびIMUにより求められるものである。しかし、森林作業道を移動する場合、不整地面を走行するため、正確なデッドレコニングを求めることが困難となる。このため、デッドレコニングによる点群の補正ができず、点群に歪みが生じる。

自律移動を困難とするもう一つの原因として、森林作業道の移動時に取得する周辺形状が、人工物に囲まれた環境と大きく異なることが挙げられる。壁や建物等の人工物を測定する場合、LiDARから発射されたレーザ光は物体に当たり、反射される。一方で森林作業道では、周囲が草や木に覆われている環境であるため、レーザは枝や葉などの細かい物体に当たり反射する場合もあれば、光源の位置の変化により、これら物体に当たらず貫通していく場合もある。森林作業道では、測定時の位置姿勢の変化が微小であっても、得られる点群の形状が大きく変化するため、スキャンマッチングにおける点群の重ね合わせが困難となる。

本研究では、これら2つの問題を解決する方法として、スキャンマッチングに2つのアイデアを組み込む。一つ目のアイデアとして、スキャンマッチングにおいて並進量・回転量に加え、並進速度を同時に推定する手法を導入する。並進速度を求めることで、デッドレコニングを使用することなく、移動時に発生する点群の歪みの補正を可能とする。二つ目のアイデアとして、形状近似時に得られるパラメータを使用した重み付きスキャンマッチングを構築する。形状近似時に得られるパラメータにより、草や木に囲まれた環境においても高精度な点群の位置合わせを可能とする。

#### 1.4.2 路面推定における問題と提案手法

森林作業道の自律移動においては、指定した経路上を走行できたとしても、障害物や崩落が新たに生じる可能性がある。例えば、森林作業道の車道は土砂を転圧することで作設されるため、フォワードのような重量物が走行すると変形や崩落が発生する。安全な自律移動を行うためには、移動時の路面を正確に判断する必要がある。さらに、路面は落ち葉や雑草などの表層物に覆われるため、見た目や形状が日々変化する。

本研究では、LiDAR-SLAMにより得られた点群を使用し、ディープラーニングによる学習結果を使用した路面推定手法を構築する。LiDARは、カメラ画像を使用する場合と比較して、環境光の影響を減少させることができる。さらに、LiDARの1回の測定で得られる点群は少ないが、LiDAR-SLAMにより得られた点群を使用することで、広い範囲の点群を取得できる。さらに、レーザ光であれば、雑草等の表層物を貫通し、路面の形状を測定することが可能である。

### 1.5 森林作業道における自律移動の実験と検証

森林作業道における自律移動の可否は、複数の森林作業道において自律移動を実施することで判断する。評価方法として、指定した経路の自律移動時において、経路の横断方向の位置制御精度を算出する。

### 1.5.1 森林作業道の幅員と横断方向の精度

森林作業道は技術基準などが整備されておらず，路線計画や作設作業の内容は，作業を行うオペレータの判断に委ねられている．一方で，森林作業道の作設にあたっては，作設のための費用を抑えて経済性を確保しつつ，間伐（森林の間隔を作り主伐する木の成長を促すために木を間引く作業）や，主伐（木材となるために成長した木を伐って収穫する作業）による木材の集材・搬出，主伐後の造林・保育等の森林施行において，継続的に用いられる道であることが求められる．このため，適切な森林作業道の作設を促すことを目的に，森林作業道作設指針 [1] という形で，森林作業道の作設における目安が示されている．

森林作業道作設指針のうち，自律移動に関わる要素として，森林作業道における幅員について述べられている．作業システムに最も影響を与えるのは林地の傾斜であるとして，傾斜区分ごとに，森林作業道の幅員の目安，およびその環境で主に使用されるベースマシンの重量を示している．森林作業道作設指針を引用した文献 [2] では，ベースマシンの各クラスごとに車幅の例が示されており，これらから森林作業道の横断方向の余裕（以降，クリアランスと呼ぶ．Fig. 1.3）を算出したところ (Table 1.1)，最小で 0.25 m となった．このことから，自律移動システムの精度は，クリアランス 0.25 m 以内となることが望ましい．

Table 1.1: Clearance in forest work roads.

slope angle [deg]	machine classes [t]	vehicle width [m]	road width [m]	clearance [m]	clearance (one-side) [m]
under 25	9 ~ 1.3	2.49	3.0	0.51	<b>0.25</b>
	6 ~ 8	2.32	3.0	0.68	0.38
25 ~ 35	6 ~ 8	2.43	3.0	0.68	0.38
	3 ~ 4	1.96	2.5	0.54	0.27
over 35	3 ~ 4	1.96	2.5	0.54	0.27

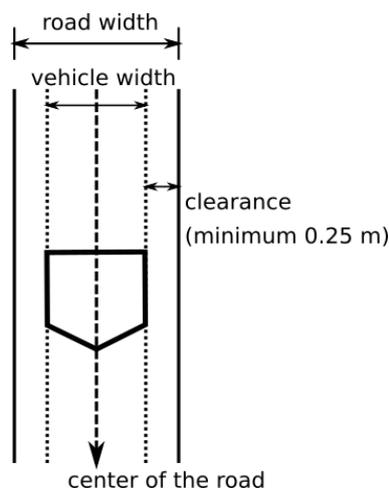


Fig. 1.3: Lateral clearance.

## 1.5.2 本研究成果の活用

本研究は森林作業道における自律移動の実現を目指しているが、舗装されていない路面上の自律移動に適用が可能である。例としては、農業用途の自律移動ロボットのような、不整地において自律移動が必要なケースに応用が可能である。本研究の自律移動システムは、GNSS を用いることが困難な、山間部の農地や、ブドウ棚・ハウス内のような環境における自律移動に活用することができる。その他、不整地において、土木工事を行う車両を自動化することが可能である。このように、本研究成果は広い分野に活用することができる。

## 1.6 論文の構成

本論文の構成は、本章において概要を説明し、要素技術であるスキャンマッチング手法、および路面検出手法を、それぞれ 2 章、3 章において提案する。4 章において、提案要素を使用した自律移動システムの構築方法を述べ、5 章において、森林作業道における横断方向の位置制御誤差を測定し、本研究の目的が達成されていることを確認する。6 章において、本研究の実施内容についてのまとめを述べる。

## Chapter 2

# 森林作業道において自己位置推定可能な スキャンマッチング

### 2.1 概要

本章では，森林作業道における自己位置推定を実現するためのスキャンマッチング手法について述べる．スキャンマッチングとは，LiDAR により得られた複数の点群を重ね合わせる技術である．スキャンマッチングにより，点群の重ねあわせに要した並進量・回転量を求めることができる．さらに，逐次重ね合わせを行った点群は，地図として使用することができる．スキャンマッチングを応用することで，自律移動に必要な，オドメトリの推定，地図作成，および地図中における現在位置の推定を行うことができる．本章で提案するスキャンマッチング手法は，ボクセルによる特徴量抽出，および並進速度の推定を取り入れた位置姿勢最適化により，森林作業道においても精度良く自己位置を推定する手法である．

本章の流れとして，関連する従来研究，および提案手法の構築方法を述べた後，従来手法と提案手法を用いた実験を行い，RTK-GNSS を真値とした自己位置推定精度，および森林作業道における自己位置推定精度を算出する．自己位置推定精度の比較により，提案手法の有効性を確認する．

### 2.2 森林作業道におけるスキャンマッチングの要件

森林作業道の周辺形状は，人工物に囲まれた環境と異なる．人工物に囲まれた環境では，壁や建造物などの，平面や曲面形状が豊富に存在するため，LiDAR から発射されたレーザーは物体を突き抜けずに反射される．これに対して，森林作業道では，周囲の物体は主に草や木に覆われている環境であるため，レーザーは枝や葉などの細かい物体に当たり反射する場合もあれば，これら物体に当たらず貫通していく場合もある．LiDAR の位置姿勢が微小に変化しただけでも，得られる点群の形状が大きく変化するため，このような環境に対応したスキャンマッチング手法が必要となる．

さらに森林作業道では，motion distortion と呼ばれる LiDAR の測定原理に起因した問題が発生する．屋外で利用できる LiDAR の多くは，レーザーを回転体に搭載することで，測定範囲の拡大を行っている．LiDAR が瞬間的に測定できる点群はレーザー 1 本に対して 1 点のみであるため，LiDAR 本体が測定中に移動する場合，測定中の移動量，および回転量を計算に入れた形状の補正を行わなければならない．補正できなかった形状は点群の

歪み (motion distortion と呼ばれる) として残ることとなる。通常、測定中の並進移動量はデッドレコニング、回転量は IMU により測定されるが、森林作業道のような舗装されていない路面では、砂利や表層物が車輪と路面の間に侵入することで車輪の空転が生じるため、デッドレコニングによる並進量の測定が困難となる。森林作業道のスキャンマッチングでは、デッドレコニングが不要な motion distortion の補正を行う新たなアルゴリズムが必要とされる。

## 2.3 関連研究

### 2.3.1 ICP アルゴリズムを使用したスキャンマッチング

代表的なスキャンマッチング手法としては Iterative Closest Point(ICP) アルゴリズムが知られている。Besl ら [18] の ICP アルゴリズムは、最近傍となる点对点のペア (point-to-point) を探索する。探索後、すべての点群ペアに対してペア同士の距離を求めることでコスト関数とし、コスト関数を最小化する回転角度と並進量を求めることでマッチングを行っている。類似した手法として、Chen ら [19] の手法は、点対面のペア (point-to-plane) を使用している。point-to-plane 用いる手法は、点群の各点群を頂点とするメッシュを作成し、新たに入力された点群とメッシュとの距離を最小化する。点群の分布に偏りがある場合や疎な場合において、point-to-point を用いる手法より高精度な重ね合わせを期待することができる。

Biber らの手法は、NDT(Normal Distribution Transform) を使用する手法 [20] を提案している。NDT とは、3次元空間をボクセルと呼ばれる格子状に分割し、ボクセル内に含まれているすべての点群の座標を用いて、共分散行列を取得する。NDT は、共分散行列から正規分布を算出し、これを特徴量として取り扱う手法である。Biber らの手法は、NDT により得られた分布を用いて、点对分布 (point-to-distribution) の対応により重ね合わせを行う。ボクセルを使用することで、対応点探索を高速化し、同時に特徴量抽出による情報量の圧縮、およびノイズによる影響の低減を可能としている。これら利点により、NDT によるスキャンマッチングは、屋外環境の SLAM、および自己位置推定に多く用いられている [21, 22, 23, 24]。

Aleksandr らは、Generalized-ICP[25] と呼ばれる手法を提案している。Generalized-ICP は、最近傍探索により点群を抽出し、抽出した点群から共分散行列を取得する。共分散行列から、point-to-point、point-to-distribution、および分布対分布 (distribution-to-distribution) の対応を取得することができるため、複数の対応付け方法を、同時に実行できる手法となっている。

point-to-point、point-to-plane による手法は、森林作業道のような自然物の多い環境において、正確な点群の位置合わせが難しい。枝や葉など小さい物体が多く存在する環境では、レーザの照射方向の変化や、風による揺れなどにより、レーザが頻繁に物体を透過する。このため、測定する物体が同一のものであっても、取得される形状が常に変化する。point-to-point による手法は、単純に最も距離の近い点群同士を対応点として扱うため、点群ペアの誤対応が発生する。point-to-plane は、点群をメッシュ化することで面を取得するため、草や木の葉など、形状のばらつきが大きい物体に対しては、メッシュ化による形状の平面近似が困難である。

NDT、GICP のような点群の面や分布を推定して使用する手法は、point-to-point や point-to-plane と比較して、測定点の偏りや、取得形状のばらつきに対して有効であると考えられるが、ボクセルの大きさによるトレードオフが存在する。ボクセルを拡大した場

合、複雑な形状の近似ができない。近似精度を高めるため、取得範囲を縮小した場合、点群の不足により、分布を求めるための共分散行列が縮退する。共分散行列が縮退した場合、点群の重ね合わせに必要な逆行列が求められなくなる。共分散行列が縮退した場合の例外処理として、固有値を用いた平面近似を使用することで、縮退の発生時においても点群の位置合わせが実行できる。しかし、3D-LiDAR で取得される点群の多くは、線のような形状が多く含まれるため、平面近似による位置合わせが困難となる。

### 2.3.2 Motion distortion に対応したスキャンマッチング

motion distortion に対応したスキャンマッチング手法として、Bosse ら [14, 15] は、LiDAR から得られた点群を一定時刻ごとに分割し、各時刻の位置姿勢を、複数の位置姿勢 (trajectory と呼ばれる) として取り扱う手法を提案している。ボクセルを用いた特徴量を算出し、参照とする点群と、入力として変形を行う点群の特徴量が一致するように trajectory を推定することで、motion distortion の修正を行っている。

Zhang ら [16, 17] の提案した LOAM(Lidar Odometry and Mapping) は、trajectory の最適化を 2 つの頻度に分解し推定している。短時間に取得した trajectory を最適化することで、motion distortion の補正と点群の蓄積を行う。その後、motion distortion 修正後の点群を用いて位置・姿勢を推定する。さらに、LOAM はマッチングにおいて、局所的な点群の滑らかさを評価することで、点群をエッジ、面として分類している。point-to-point と比較してより正確に点群のペアを求めることができ、かつ点群の間引きを行うことができるため、精度と計算効率のバランスが良い手法である。

LeGO-LOAM[26] は LOAM を基に、草や木の葉などのスキャンマッチングの難易度が上がる環境を想定し、よりロバストかつ計算効率が良い手法を提案している。LOAM と同様に点群をエッジと面に分類するほか、点群のセグメンテーションにより、路面と非路面の分類を行うことで、より正確な点群のペアを求めることを可能としている。

森林作業道での運用を考慮すると、Bosse らの手法は、加速度センサから trajectory の初期値を取得する必要があるため、不整地を走行するフォワードのような、強い振動が加わる環境での運用に適さない。LOAM, LeGO-LOAM による手法は、点群をエッジ特徴量・面特徴量に分類・抽出することで、不要な点群の除去によるペア探索のロバスト性を向上させている。しかし森林作業道では、草や葉のようなもともとのばらつきが大きい形状を対象としなければならない場合があり、これらの形状は明確に分類できる形状でない。このため、森林作業道においては、他の手法と比較して有効な手法とは言えない。

### 2.3.3 森林作業道に対応したスキャンマッチングと従来手法との比較

本研究ではこれら従来手法の森林作業道における問題に対し、次の手法を提案する。

1. 位置姿勢のドリフト量と並進速度の同時推定
2. ボクセル内点群の主成分分析を用いた重み付き最小二乗法によるスキャンマッチング
3. 主成分分析により得られた固有ベクトルを用いた、point-to-line, point-to-plane によるコスト関数の設定

1 については motion distortion の修正のために提案する。LOAM[16, 17] では並進速度の推定により motion distortion の修正を行い、その後位置・姿勢を修正するという手

順を行っているが、並進速度推定時に発生した誤差については、位置・姿勢推定時に修正する機会がない。これに対し提案手法は、位置・姿勢・並進速度を同時に推定することができるため、LOAMと比較して、位置・姿勢・並進速度の推定精度を向上させることが可能となると考えられる。

2については、主成分分析を用いてボクセル内点群のばらつきを評価し、ばらつきの少ない形状ほど、スキャンマッチング時のコスト関数に影響するよう提案する。LOAM, LeGO-LOAMのような面、エッジによる分類は不要であり、さらに、ばらつきの少ない形状を優先したスキャンマッチングが可能となるため、森林作業道における自己位置推定精度を向上させることが期待できる。

3については、点群の近似を行う範囲を狭く設定することで、複雑な形状の近似を可能とする。さらに、少ない点群しか取得できなかった場合においても、形状の近似、およびスキャンマッチングを可能とすることを目的としている。

各スキャンマッチング手法は、Table 2.1のように分類することができる。森林作業道で運用するスキャンマッチングの要件として、motion distortionの修正、およびエッジ、面の分類が必要のない手法が求められる。提案手法は、motion distortionの修正に対応し、かつエッジ、面の分類が必要のない手法に位置付けられる。

Table 2.1: Comparison of scan matching methods.

Method	Correspondence	Motion distortion
Standard ICP [18] [19]	point-to-point, point-to-plane	-
GICP [25]	distribution-to-distribution	-
NDT [20]	point-to-distribution	-
LOAM [16, 17]	semantic feature	✓
LeGO-LOAM [26]	semantic feature	✓
ours	point-to-line, point-to-plane	✓

## 2.4 提案手法

### 2.4.1 提案手法の概要

本システムにおける処理の流れを Fig. 2.1 に示す。本システムは、3D-LiDAR により得られる点群 (以降、入力点群と呼ぶ) と、逐次処理により、事前に補正済みの点群 (以降、参照点群と呼ぶ) を、スキャンマッチングにより重ね合わせる。スキャンマッチングの結果により、位置姿勢、および並進速度の推定を行う。スキャンマッチングは、位置・姿勢・並進速度推定値を用いて入力点群を補正する処理と、補正済み入力点群を用いた重ね合わせにより位置・姿勢・並進速度の推定値を更新する処理を、交互に繰り返し実行する。この処理を繰り返し実行することをイテレーションと呼ぶ。

### 2.4.2 センサ入力と前処理

はじめに、本システムの入力について説明する。本システムは

- 3D-LiDAR から、3D-LiDAR 本体の座標系を基準とした 3 次元点群  $\rho$

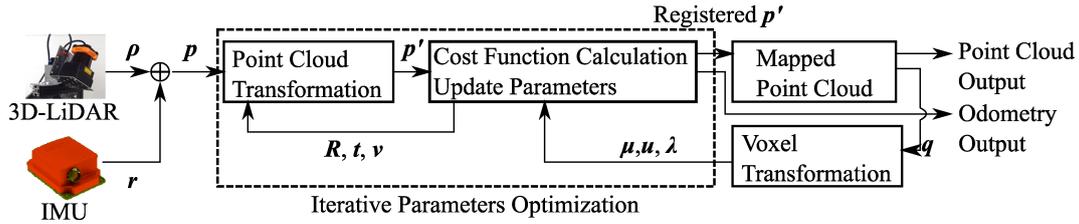


Fig. 2.1: Diagram of the 3D-LiDAR odometry software system.

- IMU から，内蔵されたカルマンフィルタにより得られる 3D-LiDAR 本体の姿勢推定値  $r$

を取得し，これを入力とする．

ただし IMU は起動時に，加速度センサの重力方向検知により得られた水平面，および地磁気センサにより得られた方位を用いて原点座標系を設定する．IMU が起動時に設定した原点座標系を，全体座標系と呼び使用する．さらに，IMU 本体と 3D-LiDAR は固定されており，固定に使用した治具の寸法を用いて，IMU 本体姿勢推定値を 3D-LiDAR の姿勢推定値に変換する．これら  $\rho$ ,  $r$  を用いてスキャンマッチングを行う．

入力の取得後，前処理として，点群  $\rho$  を 3D-LiDAR の姿勢推定値  $r$  により全体座標系に変換する．変換後の 3 次元点群を入力点群  $P = [p_1, \dots, p_N]$  とする． $p_i (i = 1, \dots, N)$  は，3D-LiDAR が 1 回のレーザの走査により， $N$  点の周辺形状を測定し，得られた  $i$  番目のスキャン点の座標である．

### 2.4.3 歪み補正

次に，位置姿勢，および並進速度推定値を用いて入力点群の歪み補正を行う方法を述べる．ただし，初回のスキャンマッチングは，3D-LiDAR 本体は停止しているものとして，位置を原点，姿勢を  $r$ ，並進速度をゼロとする．さらに初回以降は，前スキャンマッチング時に得られた位置姿勢，および並進速度推定値を用いる．

3D-LiDAR より得られた 1 スキャンのセンサデータのうち，点群位置を  $p$ ，全体座標系に対する 3D-LiDAR の姿勢を回転行列  $R$ ，1 スキャン開始時の 3 次元位置を  $t$ ，3 次元並進速度を  $v$  として，式 (2.1) により，点座標  $p$  を，歪み補正済みの点座標  $p'$  に変換する．

$$p'_i(R, t, v) = Rp_i + t + i\delta\tau v \quad (2.1)$$

点群  $P = [p_1, \dots, p_N]$  に対して歪み補正を実行し， $P' = [p'_1, \dots, p'_N]$  を得る． $\delta\tau$  は 3D-LiDAR が 1 点の測定に要する時間である．式 (2.1) を適用した場合の 3D-LiDAR 位置を Fig. 2.2 に示す．すべての  $i$  について，式 (2.1) により，点群座標を変換する．イテレーション開始時， $R$ ,  $v$  の初期値は，前スキャン時の推定値を用いる． $t$  の初期値は，前スキャン時の位置  $t'$ ，および 3 次元並進速度  $v'$  を用いて， $t = t' + N\delta\tau v'$  とする．これにより，前スキャンにおける推定値を利用して，次スキャンの位置姿勢を推測した状態からイテレーションを開始することができる．

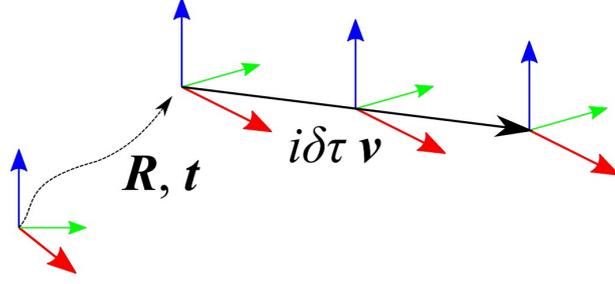


Fig. 2.2: 3D-LiDAR pose.

#### 2.4.4 スキャンマッチングによる位置姿勢，および並進速度の推定

スキャンマッチングによる  $R, t, v$  の推定方法を述べる．参照点群に対してボクセルを設定し，ボクセル内に含まれる点群位置  $q_k$  から，平均値  $\mu$ ，共分散行列  $C$  を式 (2.2)，(2.3) により算出する．

$$\mu = \frac{1}{L} \sum_{k=1}^L q_k \quad (2.2)$$

$$C = \frac{1}{L} \sum_{k=1}^L (q_k - \mu)^T (q_k - \mu) \quad (2.3)$$

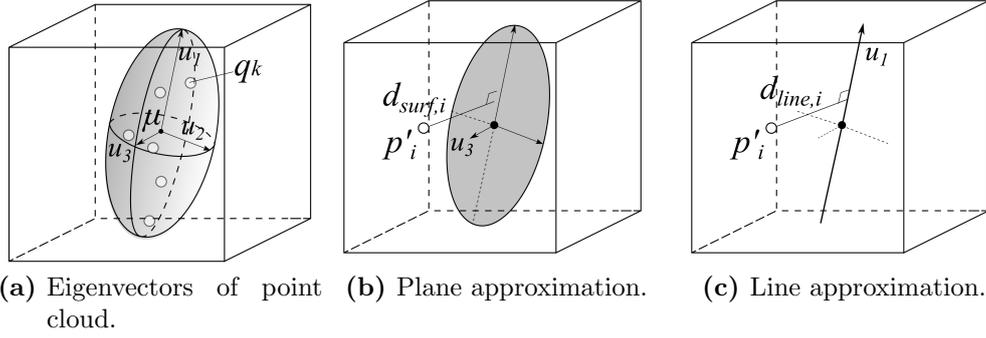
ただし， $k$  はボクセル内点群の番号  $k = 1, \dots, L$  であり， $L$  はボクセル内点群数である．さらに，共分散行列の固有ベクトルを算出することで，固有値  $\lambda$ ，および固有ベクトル  $u$  を得る． $\lambda = (\lambda_1, \lambda_2, \lambda_3)$ ， $u = (u_1, u_2, u_3)$  であり， $\lambda_1 > \lambda_2 > \lambda_3$  となるよう，固有値と対応する固有ベクトルをソートする (Fig. 2.3(a))．入力点群  $p'_i$  に最も近接するボクセルから得られる  $u_3$  を用いて，ボクセル内点群の近似平面に対する入力点群の距離  $d_{surf,i}$  (Fig. 2.3(b)) を式 (2.4) により算出する．さらに，ボクセル内点群の近似直線に対して垂直かつ入力点群を通過するベクトル  $d_{line,i}$  (Fig. 2.3(c)) を式 (2.5) により算出する．

$$d_{surf,i}(R, t, v) = \frac{u_3^T \{p'_i(R, t, v) - \mu\}}{\|u_3\|} \quad (2.4)$$

$$d_{line,i}(R, t, v) = \frac{u_1 \times \{p'_i(R, t, v) - \mu\}}{\|u_1\|} \quad (2.5)$$

すべての入力点群に対して式 (2.4), (2.5) を算出し， $d_{surf,i}$ ， $d_{line,i}$  が最小となる  $R, t, v$  を求めることでスキャンマッチングは終了となる．ただし，実際の森林作業道内では，葉などの点群のばらつきが大きい形状と，路面や樹幹などの密な点群を持つボクセルが存在し，誤った推定を行う可能性がある．このような場合を考慮し，路面や樹幹などの平面近似結果の良いボクセルを優先的にスキャンマッチングに使用するため，固有値を用いた重みパラメータとして式 (2.6) の  $w_{surf}$ ，および式 (2.7) の  $w_{line}$  を設定し，評価関数として式 (2.8) の  $F(R, t, v)$  を使用する．

$$w_{surf} = \exp\left(\frac{|\lambda_3|}{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}}\right) \quad (2.6)$$



**Fig. 2.3:** Plane and line approximation in a point cloud.

$$w_{line} = \exp\left(\frac{|\lambda_2| + |\lambda_3|}{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}}\right) \quad (2.7)$$

$$F(R, t, v) = \sum_i W_i A_i \quad (2.8)$$

ただし,

$$A_i = \begin{pmatrix} d_{surf,i} \\ d_{line,i,x} \\ d_{line,i,y} \\ d_{line,i,z} \end{pmatrix} \quad (2.9)$$

$$W = \begin{pmatrix} w_{surf,i} & 0 & 0 & 0 \\ 0 & w_{line,i} & 0 & 0 \\ 0 & 0 & w_{line,i} & 0 \\ 0 & 0 & 0 & w_{line,i} \end{pmatrix} \quad (2.10)$$

式 (2.8) について,  $R, t, v$  に関する重み付き最小二乗法を実行し, 評価関数の最小化, およびパラメータである  $R, t, v$  の更新を行う.

評価関数の最小化には複数回の最小二乗法の実行が必要となる. イテレーションとして, 次の処理を複数回実施する.

1. パラメータの更新毎に, 式 (2.1) を用いて入力点群の座標を変換
2. 評価関数の算出と最小二乗法の実行によるパラメータの更新

イテレーション終了時のパラメータ  $R, t, v$ , およびパラメータにより座標変換された点群  $p'$  を出力する. なお, 今回は実験的に求めた値を用いて, ボクセルサイズは 1 辺 0.3 m, イテレーションの繰り返し回数は 15 回とした.

#### 2.4.5 点群の蓄積

スキャンマッチングによる位置姿勢, および並進速度の推定が完了した後に, スキャンマッチングによる重ね合わせが完了した点群を, 参照点群に追加する.

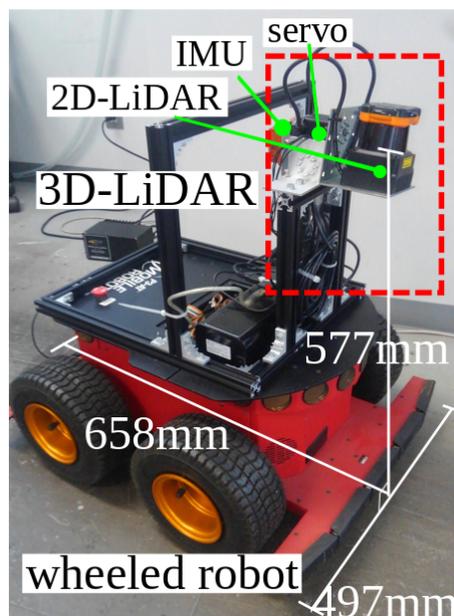


Fig. 2.4: Experimental devices.

## 2.5 提案手法の評価

### 2.5.1 実験に用いた装置

本研究で使用するハードウェアの外観を Fig. 2.4 に示す．3D-LiDAR を移動ロボットに搭載し，周辺の 3 次元点群，および移動ロボットの姿勢が取得できる構成となっている．

3D-LiDAR は，周囲の 3 次元形状を測定するための 2D-LiDAR とサーボ，および姿勢推定のための IMU により構成される．2D-LiDAR は北陽電機株式会社 UTM-30LX，サーボは Robotis Dynamixel RX-64R，IMU は Xsens Technologies B.V. MTi-30-AHRS を使用する．この 3D-LiDAR を，車両型ロボット Adept MobileRobots Pioneer 3-AT に搭載する．3D-LiDAR より取得した情報は，コンピュータ (CPU Intel Core i7-8650U 1.90 GHz，メモリ 16.0 GB) によって処理を行う．コンピュータの O.S. には，Ubuntu 18.04 を用いた．ロボット用ミドルウェアに，ROS(Robot Operating System) を採用した．

2D-LiDAR は，0.025 sec ごとに 270 deg の範囲をレーザによりスキャンする．1 回のスキャンで，角度分解能 0.25 deg ごとに形状の測定を行い，計 1080 点の点群を取得する．さらに，サーボは-90 deg から 90 deg の範囲，180 deg/sec の速度で，2D-LiDAR の正面方向に対して回転する．本実験で使用した 3D-LiDAR は，2D-LiDAR とサーボにより，1.0 sec 間隔で，40 スキャン，計 43,200 点の点群を出力する．

### 2.5.2 RTK-GNSS を用いた予備実験

位置推定精度を求めるため，リアルタイムキネマティック汎地球測位航法衛星システム (以下，RTK-GNSS と称する) と呼ばれる，衛星からの信号を用いた高精度な測位法を使用した．RTK-GNSS による 3 次元位置測定では，誤差数 cm 以内での測定が可能である．ただし，山中では衛星信号の遮蔽により，RTK-GNSS の使用が不可能となるため，山梨

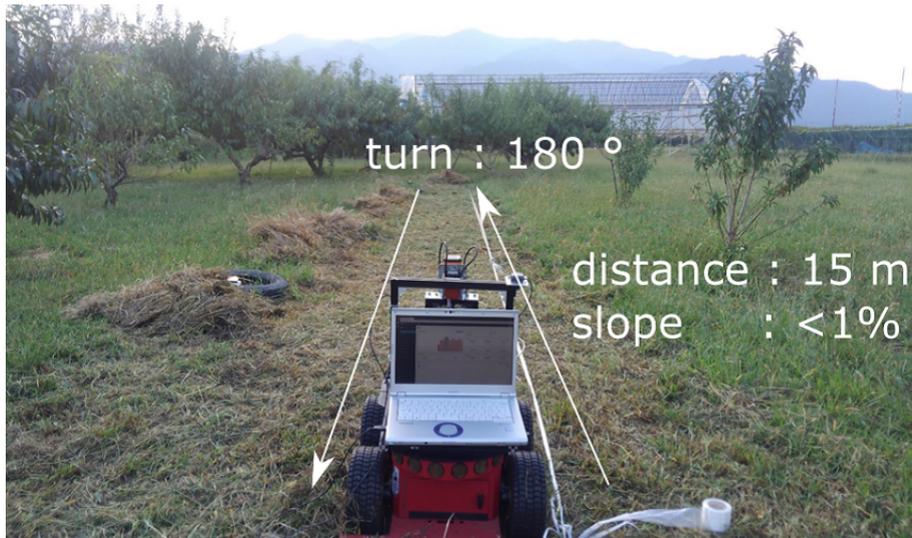


Fig. 2.5: Experimental environment for the comparison of the position error between scan matching methods and the RTK-GNSS.

県内の RTK-GNSS が受信可能な農場において予備実験を行った。森林作業道と同様に周辺に草木が存在し、かつ屋外不整地となる本農場において、従来手法である LOAM、および本研究で提案する手法との位置推定精度を比較した。

実験手順について説明する。3D-LiDAR を移動ロボットに搭載し、実験環境において直進 15 m、旋回 180 deg、直進 15 m となる経路 (Fig. 2.5) を走行した。走行時の前進速度は、ロボットに与える目標値として 0.6 m/sec、0.9 m/sec、旋回速度を 90 deg/sec とした。各条件ごとに 1 回ずつ走行を行い、移動により生じる位置推定誤差を測定した。前進速度の設定根拠は、実験において使用した森林作業道が 2 m から 2.5 m 程度であることに加え、文献 [27] より、森林作業道における幅員と作業車の速度の関係を求めたところ、2 m の幅員の場合は約 0.6 m/s、および 2.5 m の幅員の場合約 0.9 m/s で走行することから決定した。

移動中の 3D-LiDAR を含む各種センサ、および RTK-GNSS の出力を保存した。オフライン処理にて同一のデータを使用し、提案手法、および LOAM の位置推定誤差を算出した。なお、RTK-GNSS の精度は実験時の衛星配置により定まるため、実験の直前に 3 次元位置の推定誤差、標準偏差を測定したところ、水平方向の位置推定誤差は 0.022 m 以内、標準偏差 0.004 m、垂直方向の位置推定誤差は 0.034 m 以内、標準偏差 0.011 m であった。また、実験に使用した RTK-GNSS の更新頻度は 14 Hz である。

### 2.5.3 横断方向の位置推定誤差の算出方法

本研究における位置推定誤差について説明する。森林作業道を走行する車両が本システムを利用する際、位置推定誤差により、道を逸れ滑落することがないかを評価する。このため、経路に沿って走行するロボットの左右方向（森林作業道から逸れる方向）の誤差を、次のような位置推定誤差により設定した (Fig. 2.6)。位置推定誤差の算出は 1) ~ 4) により行う。1) 真値として RTK-GNSS から出力される 3 次元位置情報、および各手法により推定した 3 次元位置情報を取得。2) それぞれの 3 次元位置情報から、高さ方向を表す軸の

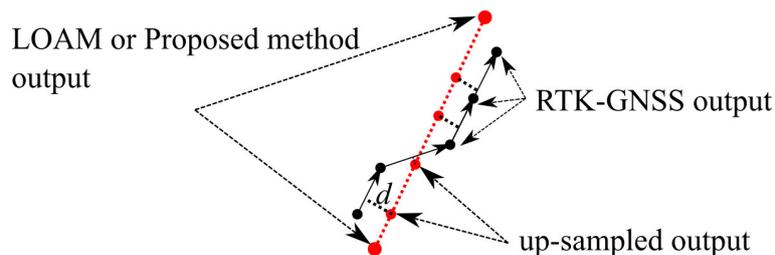


Fig. 2.6: The calculation method of position errors.

Table 2.2: Position error of the proposed method and LOAM calculated from the RTK-GNSS output.

velocity [m/sec]	method	max error [m]	RMS [m]	S.D. [m]
0.6	LOAM	1.03	0.19	0.19
	proposed	<b>0.17</b>	<b>0.04</b>	<b>0.03</b>
0.9	LOAM	0.92	0.24	0.20
	proposed	<b>0.19</b>	<b>0.05</b>	<b>0.04</b>

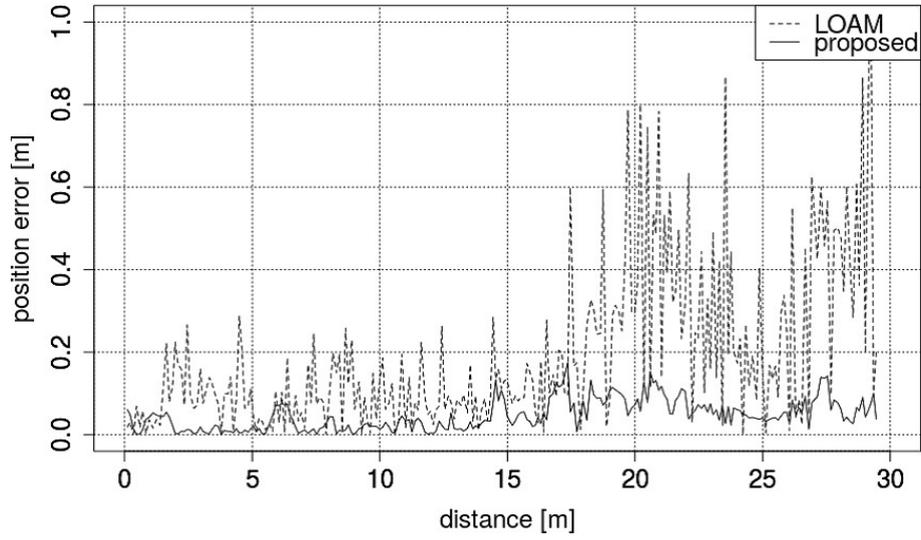
値を 0 に置き換えることで水平面上に変換．3) RTK-GNSS の時間的に前後となる位置出力から，進行方向ベクトルを求め，RTK-GNSS の測定時刻を用いて各手法の出力を内挿．4) 進行方向ベクトルと内挿した点群間の距離 (Fig. 2.6 における  $d$ ) を，位置推定誤差として算出．

#### 2.5.4 RTK-GNSS を用いた予備実験の結果

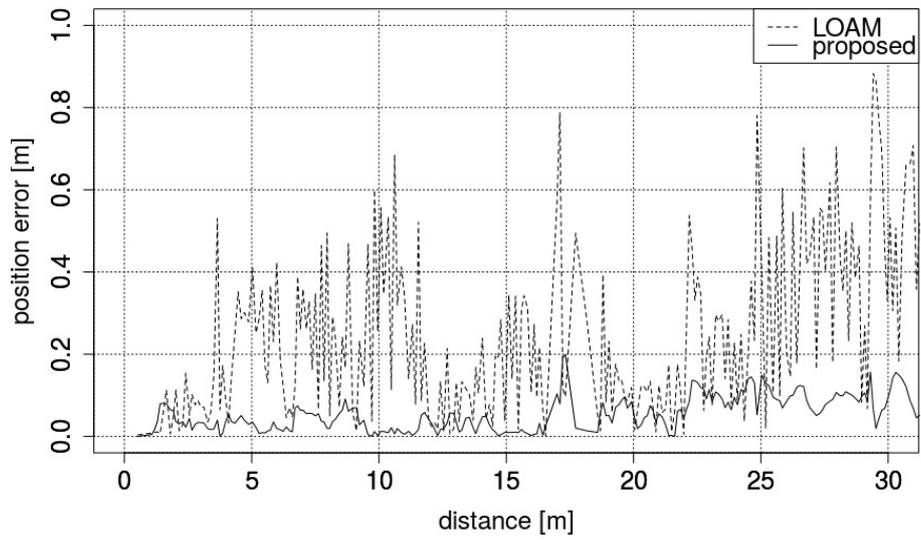
オフライン処理にて同一のデータを使用し，提案手法，および LOAM の位置推定誤差を算出した．横軸を走行距離，縦軸を位置推定誤差として前進速度 0.6 m/sec における算出結果を Fig. 2.7(a)，前進速度 0.9 m/sec における算出結果を Fig. 2.7(b) に示す．さらに，各速度における最大誤差，平均誤差，標準偏差を Table 2.2 に示す．

Table 2.2 の最大誤差を比較すると，前進速度 0.6 m/sec, 0.9 m/sec の両速度において，LOAM を利用した場合，30 m 走行時の位置推定誤差は 0.25 m を超えるものであった．これに対して，提案手法の位置推定誤差は，旋回を含む約 30 m の走行時においても，平均誤差 0.05 m，最大誤差 0.19 m 以下となり，0.25 m を下回る良好なものであった．さらに，Table 2.2 の平均誤差を比較すると，前進速度の増加に伴い LOAM の平均誤差は 0.05 m 増加したが，提案手法の平均誤差は 0.01 m の増加に抑えられた．前進速度の増加に対する平均誤差の増加量からも，提案手法の位置推定誤差が LOAM と比較して低減されていることが確認できた．

さらに，提案手法では並進速度の推定を行っていることから，並進速度の推定結果とデッドレコニングを比較した．3次元並進速度の推定値をロボット前後方向の速度に変換し，デッドレコニングにより得られた前後方向の速度と比較した結果を Fig. 2.8 に示す．提案手法は出力頻度は低いものの，デッドレコニングにより得られた速度と同等の速度推定結果が得られた．

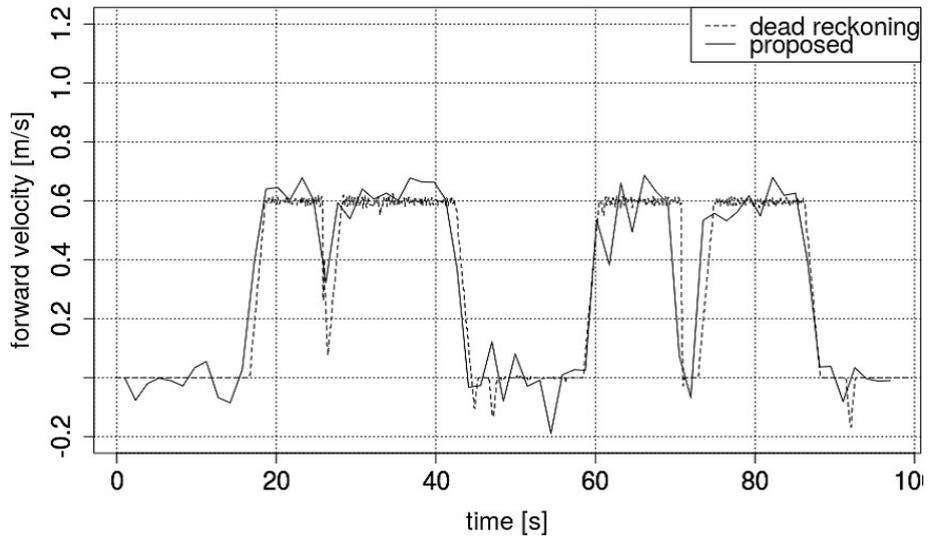


(a) forward velocity: 0.6 m/sec

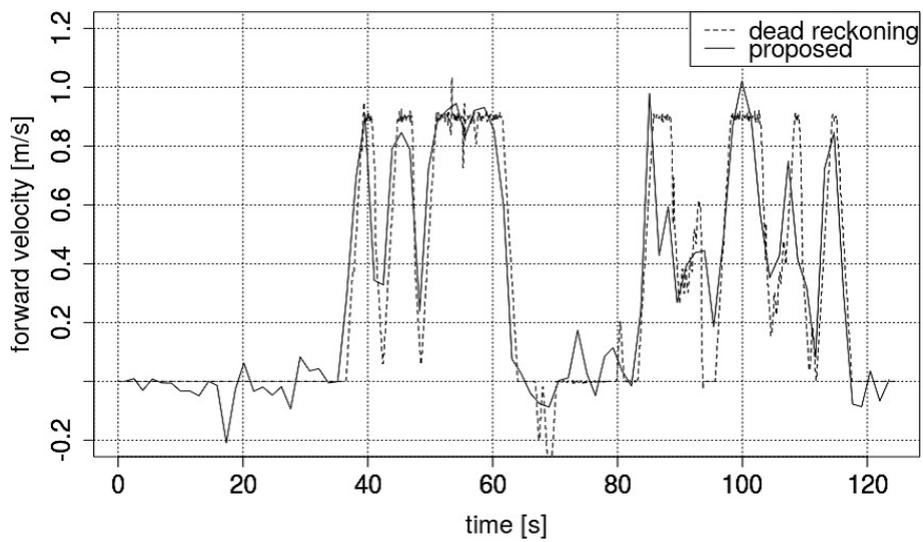


(b) forward velocity: 0.9 m/sec

**Fig. 2.7:** The comparison of position error obtained by the proposed method and LOAM method with reference to the ground truth reported by the RTK-GNSS.



(a) forward velocity: 0.6 m/sec



(b) forward velocity: 0.9 m/sec

**Fig. 2.8:** The comparison between estimated forward velocity with dead reckoning output.

## 2.5.5 予備実験の結果に対する考察

予備実験の結果 (Table 2.2) により, 提案手法の最大誤差・平均誤差・誤差の標準偏差が LOAM と比較して低減されたことを確認した. さらに, 前進速度の増加に伴う位置推定誤差の平均誤差・標準偏差の増加量についても, LOAM と比較して低減されたことを確認した.

スキャンマッチングにおける対応点を求める処理の違いについて考察すると, LOAM では点群の対応を求める際, エッジと面に分類することで, 対応する点群を求める際のフィルタリングを行っている. しかし, 草木に囲まれた環境では, 周辺物体のエッジ・面による分類が困難なため, 入力点群と参照点群における対応点の推定が困難になったと考えられる. これに対して提案手法は, 複雑な形状についても平面近似・直線近似されるため, エッジの誤対応による誤差は発生しない. さらに, 提案手法は, 点群の共分散値から得られる固有値を用いた重み付けにより, 位置推定において, 葉のような, 形状を捉えにくい物体の影響を軽減することが可能である. このことから, 位置推定誤差の低減に有効であったと考えられる.

オドメトリの算出方法の違いについて考察すると, LOAM のオドメトリ算出時は少ない点群数でオドメトリを算出しているが, 提案手法では 1 秒間に蓄積された点群全てを用いてオドメトリを算出している. 点群のマッチングが困難な環境においても, オドメトリの誤差を低減することが可能であったと考えられる.

速度の増加に伴う位置推定誤差の平均誤差・標準偏差の増加については, 路面の起伏等による振動が, 走行速度の上昇により強くなったことが要因と考えられる. LOAM, および提案手法は, ロボットが等速度で移動していると仮定したモデルを用いてオドメトリを推定する. このため, 振動時や加減速時は, モデルと実際の軌道が異なる. 提案手法はスキャンマッチング時に, オドメトリ, および位置姿勢ドリフトの推定を同時に行うため, LOAM と比較してスキャンマッチング時の自由度が高い. このことから, 提案手法は, モデルと実際の軌道との誤差をより低減したと考えられる.

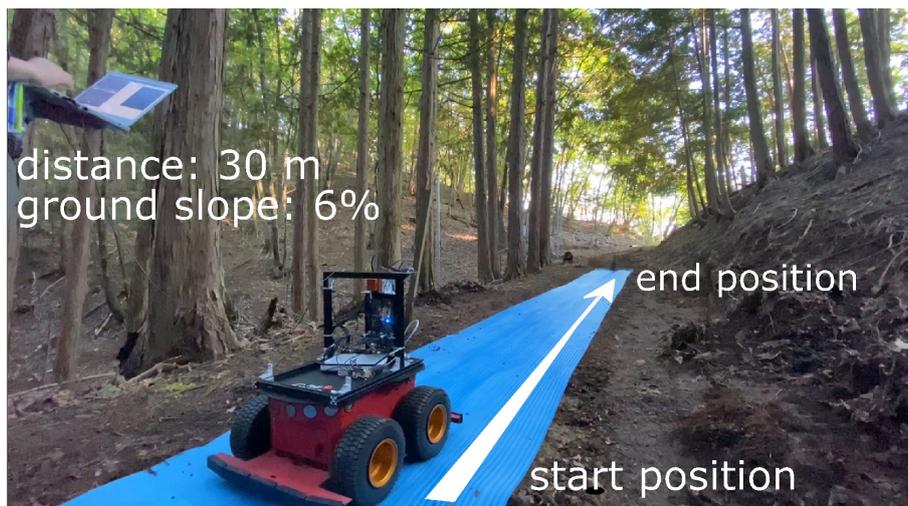


Fig. 2.9: Experimental environment in a forest work road.

## 2.5.6 森林作業道における実験

山梨県甲州市内の森林作業道 (Fig. 2.9) において、従来法と提案手法の比較を行う。

実験手順について説明する。幅員約 2.0 m の直進経路を速度約 0.6 m/sec, 幅員約 2.5 m の直進経路を速度約 0.9 m/sec, 幅員約 2.5 m のカーブを含む経路を速度 0.5 m/sec にて走行し、位置推定誤差を測定する。それぞれの経路において約 30 m の距離を 1 回ずつ走行する。

## 2.5.7 森林作業道における位置推定誤差の算出方法

従来手法、および提案手法の位置推定誤差は次のように算出する。森林作業道は山間部に存在し、衛星からの電波が遮断されるため、RTK-GNSS を用いた高精度な測量は困難である。このため、デッドレコニングによる位置推定値を真値として取得する。その際、不整地走行による車輪の空転を抑制するためのマットを敷設する。デッドレコニング算出のため、並進量推定値として車軸エンコーダの出力、姿勢値推定値として IMU の出力を保存し、同時に車両型ロボットに搭載した 3D-LiDAR の出力を保存する。実験開始時、ロボットを停止状態にし、IMU の角速度値のオフセットを除去することで、姿勢測定時のドリフトの軽減を行う。

## 2.5.8 森林作業道における実験の結果

予備実験と同様に、同一のデータを使用し、オフライン処理にて、提案手法、および LOAM の位置推定誤差を算出した。位置推定誤差を算出するにあたり、デッドレコニングの精度を、RTK-GNSS が使用できる屋外不整地にて確認した。敷設したマット上においてロボットの走行を行い、RTK-GNSS の 3 次元座標値に対するデッドレコニングの誤差を、本研究の位置推定誤差 (Fig. 2.6) を用いて算出した。デッドレコニングの誤差は 1 m

**Table 2.3:** Position error in the forest work road.

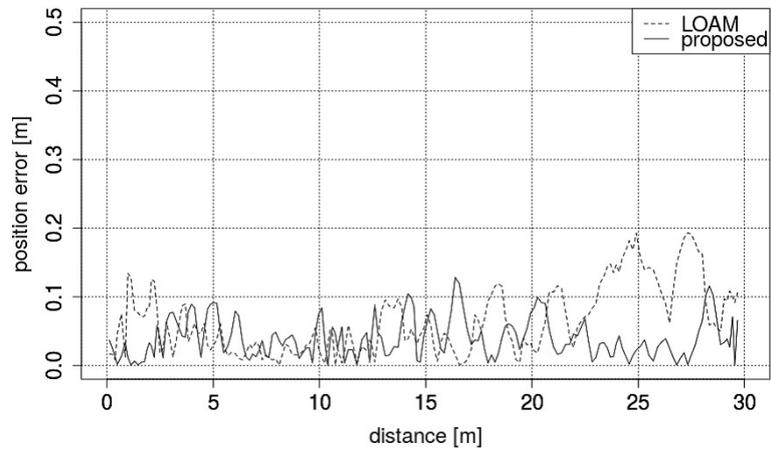
velocity [m/sec]	method	max error [m]	RMS[m]	S.D. [m]
forward 0.6	LOAM	0.19	0.06	0.04
	proposed	<b>0.12</b>	<b>0.04</b>	<b>0.02</b>
forward 0.9	LOAM	0.29	0.07	0.06
	proposed	<b>0.24</b>	<b>0.06</b>	<b>0.05</b>
curve 0.5	LOAM	0.38	0.11	0.08
	proposed	<b>0.26</b>	<b>0.07</b>	<b>0.05</b>

あたり 0.35% の誤差であり，本実験の走行距離 30 m 走行した場合，デッドレコニングの誤差は 0.105 m となる．

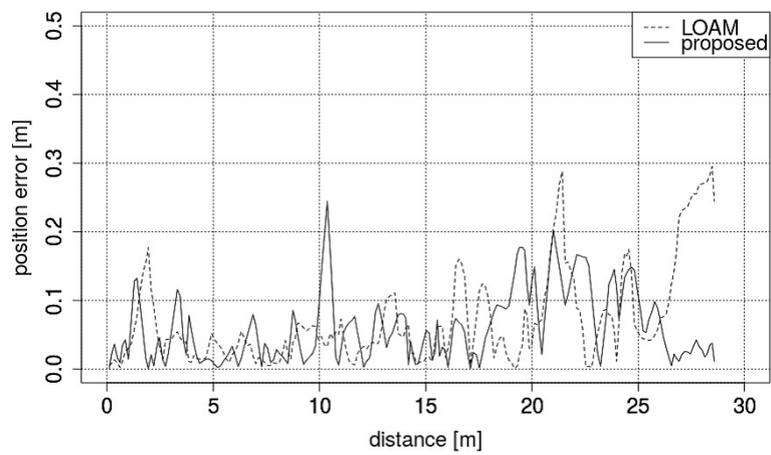
LOAM，および提案手法に対して，デッドレコニングの出力を使用し，位置推定誤差を算出した．横軸を走行距離，縦軸を位置推定誤差として，前進速度 0.6 m/sec における結果を Fig. 2.10(a)，前進速度 0.9 m/sec における結果を Fig. 2.10(b)，カーブを含む経路を Fig. 2.10(c)，各実験における最大誤差，平均誤差，標準偏差を Table 2.3 に示す．森林作業道での実験結果 (Table 2.3) の最大誤差は，LOAM の場合，30 m 走行時の位置推定誤差が前進速度 0.9 m/sec の経路，およびカーブを含む経路において目標とする誤差 0.25 m を越えるものであった．対して，提案手法の位置推定誤差は，前進速度 0.6 m/sec, 0.9 m/sec において 0.25 m 以下となり，カーブを含む経路において約 29 m 走行後，0.26 m となった．

提案手法では並進速度の推定を行っていることから，速度の推定結果をデッドレコニングの結果と比較し確認した．推定された 3 次元並進速度を，ロボット前後方向の速度に変換し，デッドレコニングにより得られた前後方向の速度と比較した結果を Fig. 2.11 に示す．提案手法は出力頻度は低いものの，デッドレコニングにより得られた速度と同等の速度推定結果が得られた．

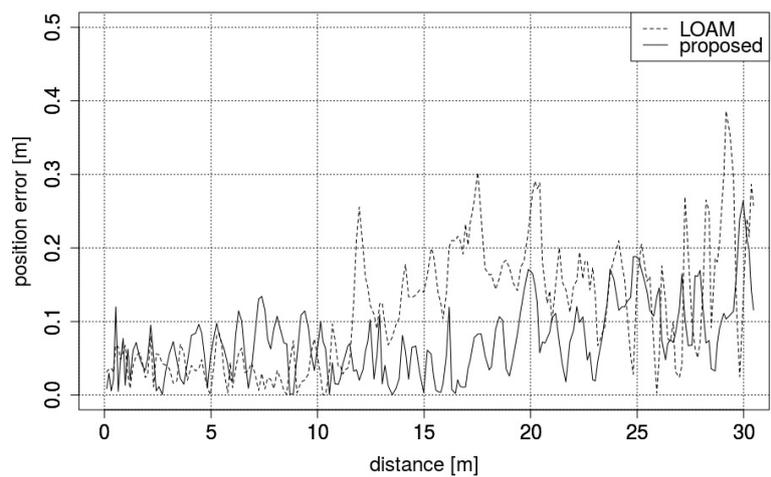
走行後の歪み補正済み点群の出力結果を確認するため，前進速度 0.9 m/sec における歪み補正済み点群の真上からの俯瞰図として表示した結果，および点群の一部を切り出し，森林作業道の横断方向から撮影した図を Fig. 2.12 に示す．提案手法は LOAM と比較して，点群形状のばらつきが抑えられていることが確認された．



(a) forward velocity: 0.6 m/sec

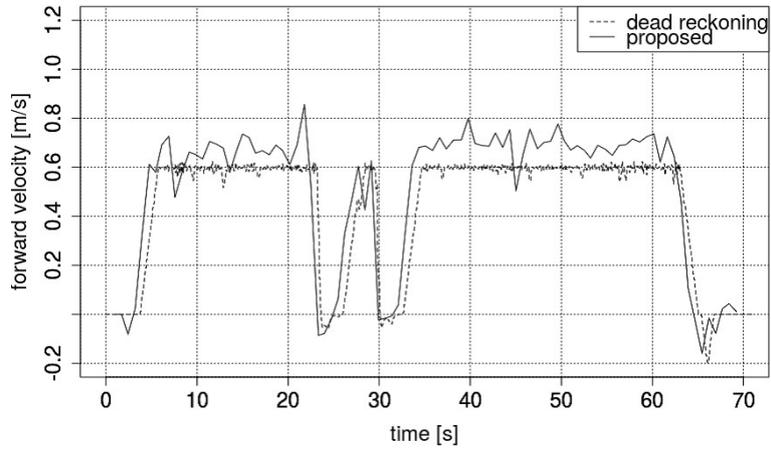


(b) forward velocity: 0.9 m/sec

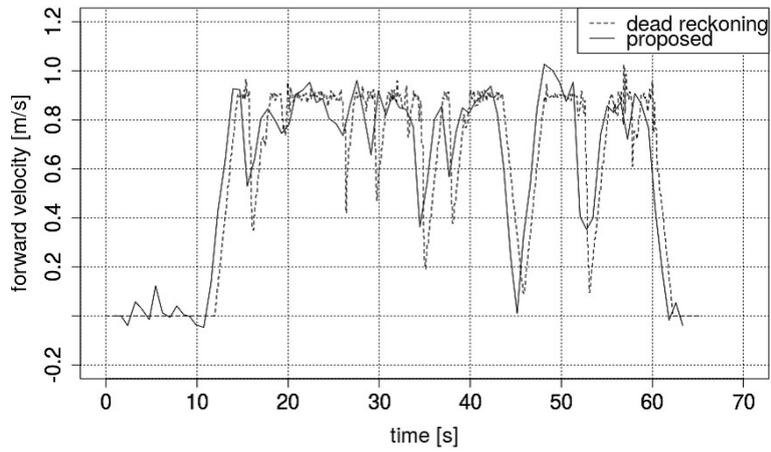


(c) curve velocity: 0.5 m/sec

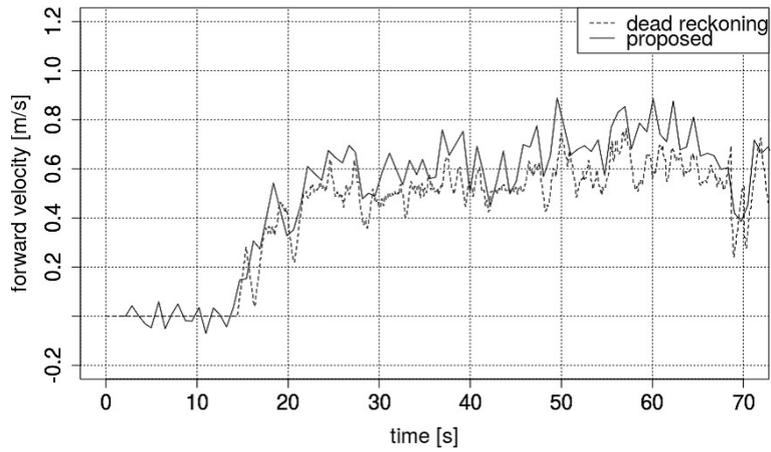
**Fig. 2.10:** The comparison of position errors obtained by the proposed method and LOAM in the forest work road.



(a) forward velocity: 0.6 m/sec

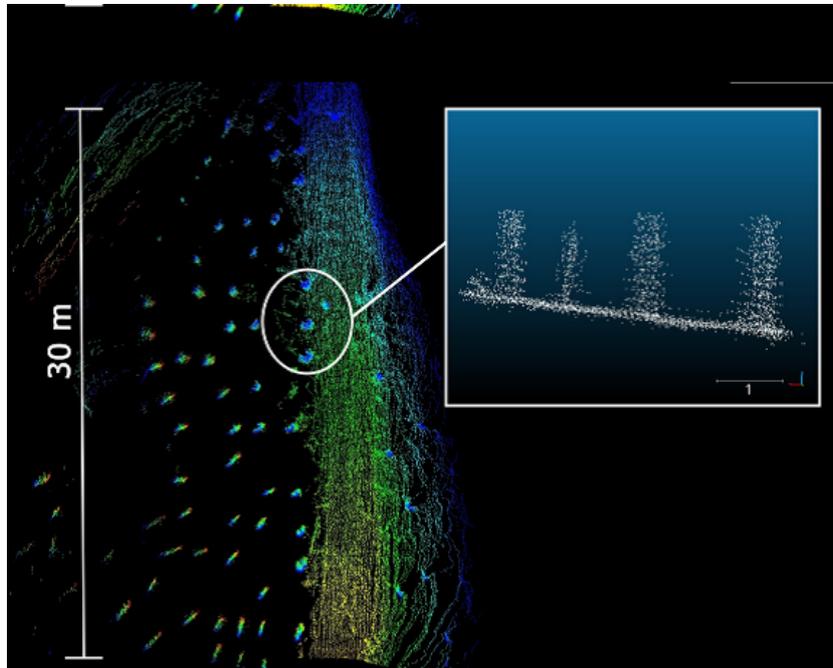


(b) forward velocity: 0.9 m/sec

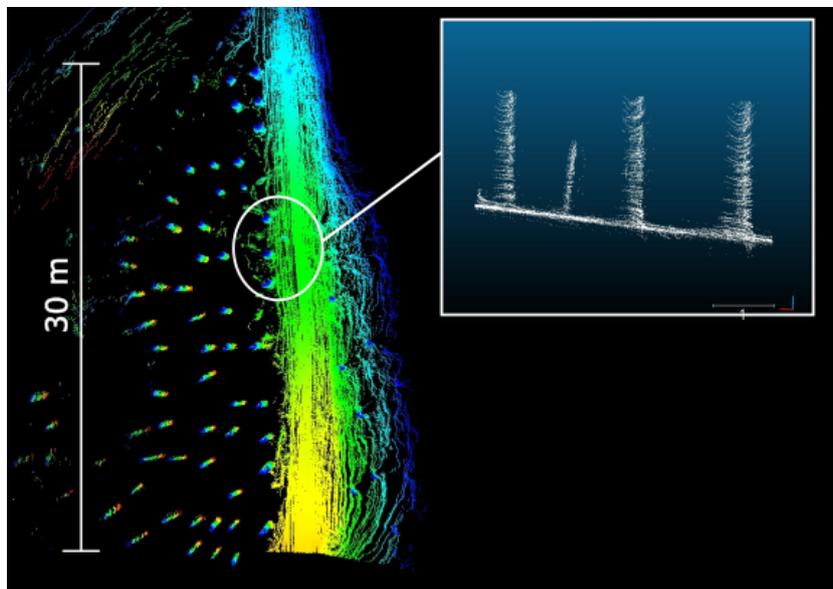


(c) curve velocity: 0.5 m/sec

**Fig. 2.11:** The comparison between estimated forward velocity with the dead reckoning output.



(a) LOAM



(b) proposed method

**Fig. 2.12:** Point cloud generated from LOAM(a), and the proposed method(b).

## 2.5.9 森林作業道の実験結果に対する考察

森林作業道における実験では、森林作業道での実験結果 (Table 2.3) の最大誤差を比較すると、LOAM と比較して、最大誤差が低減された。

平均誤差・標準偏差を比較すると、提案手法、および LOAM では移動速度の増加とともに、位置推定誤差の平均誤差・標準偏差は増加しているが、提案手法における平均誤差・標準偏差は LOAM と比較して全実験において低減されており、良好な結果となった。位置推定誤差が提案手法により改善された要因として、森林作業道では木や斜面などの不定形な物体に囲まれているため、提案手法のような形状近似を使用する手法が有効であったと考えられる。

復元された 3 次元点群の形状を比較すると、LOAM と比較して提案手法はばらつきが少いものとなった。提案手法は、オドメトリの推定と位置姿勢の推定を、スキャンマッチングにおいて同時に実行することで、入力点群の歪み補正を行う手法である。このため、歪み補正による点群位置の復元がより正確に実行されたと考えられる。

なお、提案手法の処理速度は、イテレーションの開始時から完了時までにかかる時間を測定したところ、0.61 sec であった。3D-LiDAR、IMU は 1.0 sec 間の測定情報を蓄積した後に、スキャンマッチングの処理へ入力を行う。新しいセンサ入力の蓄積中にイテレーションが終了するため、処理全体を通して遅延は発生しない。

# Chapter 3

## 深層学習を用いた路面推定

### 3.1 概要

森林作業道の路面は、土砂の転圧により作設されている。フォワーダのように重量のある車両が、森林作業道の谷側の路肩を走行すると、路肩が崩落する可能性が高い。このため森林作業道における路面検出は、十分に転圧された面を、路面として検出する必要がある。

本章では、LiDAR-SLAMにより得られた点群を用いて、学習済みネットワークを用いた路面推定手法を構築する。LiDARを用いることで、環境光の影響を低減可能であり、かつ物体の形状を高分解能・高精度に測定できる。高分解能・高精度な形状情報を用いることで、転圧された面を、路面として検出する。森林作業道において点群を取得し、データセットを作成する。さらに、得られたデータセットを用いて、提案手法の路面推定精度を算出する。

### 3.2 関連研究と森林作業道の路面検出における課題

#### 3.2.1 関連研究

森林作業道を対象とした路面検出手法として、閾値を用いた路面検出手法が、白鳥ら [28]、有水ら [29] により提案されている。Time of Flight カメラにより、路面の形状情報を取得し、平面近似、および閾値処理により、路面の検出を行う。事前に適切な閾値を設定しなければならず、路面の形状変化や、路面・路肩の植生の変化によって、路面検出精度が低下する。検出した路面の輪郭が、誤差 1 m 以上となる場合があり、森林作業道を走行するフォワーダへの適用は困難である。

深層学習を用いた路面検出法として、セマンティック・セグメンテーションによる分類対象に、路面が含まれている [30, 31, 32, 33]。セマンティック・セグメンテーションとは、RGB 画像や深度画像のピクセルごとに、物体の分類を行う手法である。セマンティック・セグメンテーションの主流な研究は、市街地で得られたデータセットを対象としているが、一部の研究では、不整地において得られたデータセットに対して、路面推定を行う手法が提案されている。Valada ら [34] の提案したセマンティック・セグメンテーションでは、草地・地面を分類対象としている。草地・路面として検出した領域のみを走行することで、不整地における、車両ロボットの自律移動を実現している。さらに、RGB 画像・深度画像・マルチスペクトルカメラから得られる EVI(Enhanced Vegetation Index) 画像を入

力とすることで、環境光の変化に対応した、セマンティック・セグメンテーションを提案している。有水ら [35] は、深層学習を、森林作業道の路面検出に導入した。FCN(完全畳み込みニューラルネットワーク)[30]を用いて、RGB 画像に対して、車道・バッファ(路肩)・その他の分類を行う手法を提案している。車道クラスを走行可能な路面、バッファクラスを安全性の低い路肩として、フォワードの自律移動に使用することを検討している。

LiDAR、および深層学習による路面検出法についても、提案されている。Caltagironeら [36] は、LiDAR から得られた 3 次元形状を、TopView 画像と呼ばれる、高さ情報を画素値とした画像に変換し、学習済みの FCN に入力することで、市街地の車道検出を行っている。

### 3.2.2 森林作業道の路面推定における課題

従来研究を、森林作業道の路面推定に適用する場合、次のような課題がある。森林作業道は周囲の森林や地形により、太陽光が遮蔽される。路面を撮影した画像は、影や光量の変化に影響される。さらに、草や落ち葉といった、表層物に覆われるため、時間経過とともに、被写体が変化する。深層学習による路面検出の精度は、学習用データセットに依存するため、RGB 画像を用いた路面推定において、十分なデータセットの確保が困難である。

本研究は、森林作業道における路面の検出において、LiDAR-SLAM と深層学習を用いた路面検出手法を構築する。LiDAR-SLAM より 3 次元点群を取得し、3 次元形状を TopView 画像に変換する。TopView 画像をネットワークに入力することで、路面を検出する。ネットワークは、セマンティック・セグメンテーションに使用されている、U-Net[37]を用いた。LiDAR は、環境光の影響を低減可能であり、かつ、LiDAR-SLAM により、物体の形状を広範囲・高精度に取得できる。このため、環境光の影響を低減し、かつ広範囲の形状情報を使用した、路面検出が可能である。U-Net については、出力された画像において、輪郭の再現性が高いという特徴を持っている。森林作業道の自律移動では、路面の輪郭を正確に検出することが求められるため、路面検出用のネットワークとして採用した。TopView 画像による表現は、自律移動のための経路計画 [38] に利用されている。TopView 画像を用いた路面検出は、自律移動時の経路計画に用いることができる。深層学習に必要な教師データは、取得した TopView 画像に対して、転圧された面、もしくは轍を構成する面を、マニュアルにより指定することで作成する。

森林作業道の検出というタスクにおいて、U-Net の採用が有効であることを確認するため、U-Net と FCN8S に対して、同一のデータセットを用いて比較する。U-Net と FCN8S の評価として、セマンティック・セグメンテーションの評価指標である、IoU(Intersection over Union[39])を用いて、検出精度を算出する。さらに、森林作業道の自律移動では、クリアランス(最低約 0.25 m とされる)範囲内の走行が必要となるため、森林作業道の横断方向における路面輪郭精度を算出した。

## 3.3 ハードウェア構成

本研究で使用するハードウェアは、2 章における Fig. 2.4 と同じ構成となっている。3D-LiDAR を移動ロボットに搭載することで、2 章において構築したスキャンマッチングを行う。スキャンマッチングにより、自己位置推定と点群の蓄積を行うことで、LiDAR-SLAM が可能である。

3D-LiDAR は、周囲の 3 次元形状を測定するための 2D-LiDAR とサーボ、および姿勢

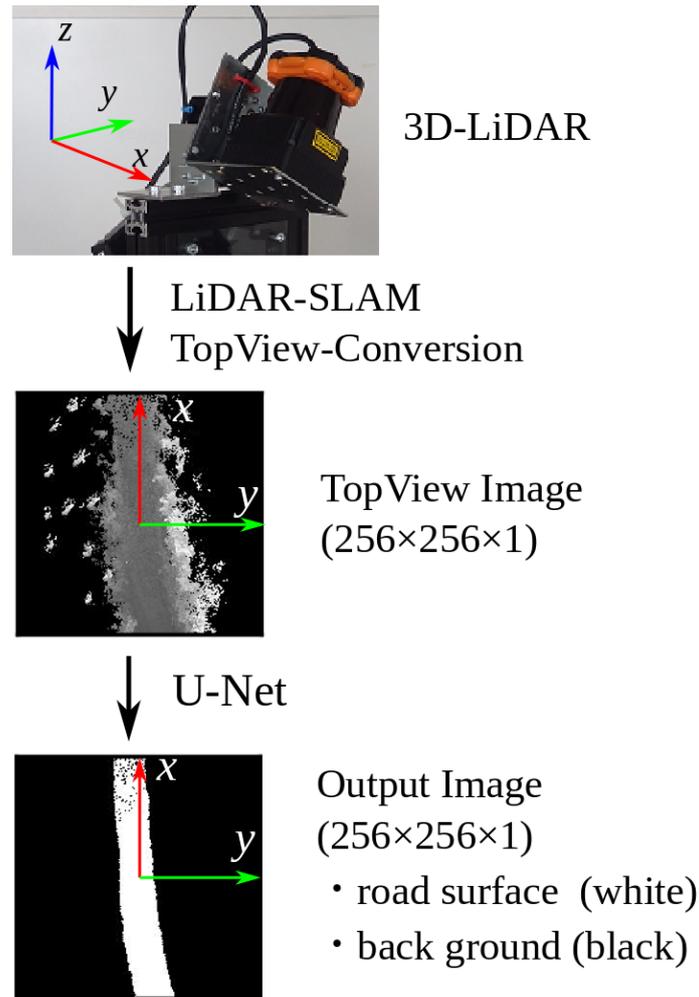


Fig. 3.1: Process of forest road detection using LiDAR-SLAM and U-Net.

推定のためのIMUにより構成される。2D-LiDARは北陽電機株式会社 UTM-30LX，サーボは Robotis Dynamixel RX-64R，IMUは Xsens Technologies B.V. MTi-30-AHRSを使用する。この3D-LiDARを，移動ロボット Adept MobileRobots Pioneer 3-AT に搭載する。3D-LiDARより取得した情報は，コンピュータ (CPU Intel Core i7-8650U 1.90 GHz，メモリ 16.0 GB)，GPU(NVIDIA GeForce GTX 1060，メモリ 6.0 GB) によって処理を行う。コンピュータのO.S.には，Ubuntu 18.04を用いた。ロボット用ミドルウェアに，ROS(Robot Operating System)を採用する。

### 3.4 LiDAR-SLAMとU-Netを用いた路面検出

本研究では，LiDAR-SLAM[40]により得られた点群を，TopView画像に変換し，U-Netにより路面の検出を行う。本手法の処理手順をFig. 3.1に示す。はじめに，LiDAR-SLAMにより点群を逐次取得する。次に，点群をセルごとに分割することで，TopView画像に変換する。最後に，学習済みのU-NetにTopView画像を入力し，路面検出を行う。

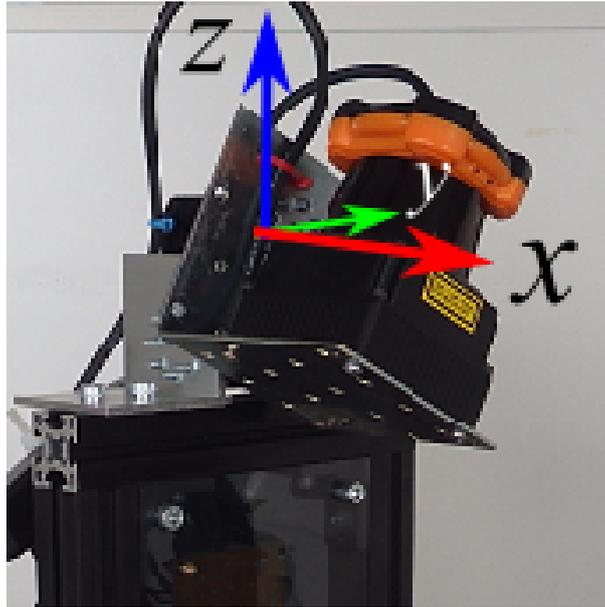


Fig. 3.2: The coordinate system to express the point cloud.

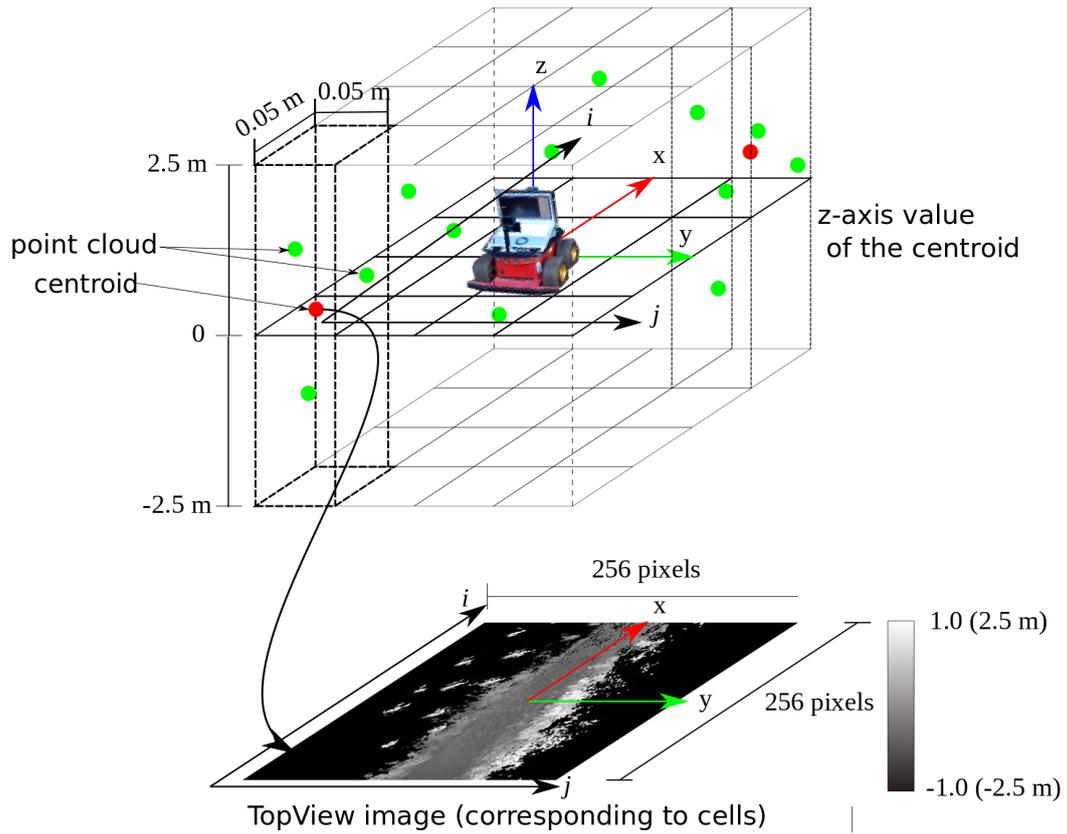
### 3.4.1 LiDAR-SLAM による点群の取得

森林作業道において、移動ロボットによる走行を行い、LiDAR-SLAM を 1 秒ごとに逐次実行する。SLAM により、3D-LiDAR の位置姿勢、および点群を取得する。点群は、3D-LiDAR の最新の位置姿勢を原点とする座標系で表現される。座標系は 3 軸を有し、3D-LiDAR の前後方向を  $x$  軸、左右方向を  $y$  軸、上下方向を  $z$  軸とする (Fig. 3.2)。

### 3.4.2 TopView 画像の作成

TopView 画像は、次のように生成する (Fig. 3.3)。原点を中心として、 $x - y$  軸方向に  $0.05 \text{ m} \times 0.05 \text{ m}$  毎に、セルを設定する。セルの範囲は  $256 \times 256$  とする。セルごとに、3D-LiDAR を搭載する移動ロボットの底面から、 $\pm 2.5 \text{ m}$  の範囲に存在する点群を抽出し、点群の  $z$  軸座標の平均値を求める。すべてのセルにおける  $z$  軸平均値が、 $z \in [-1, 1]$  となるよう正規化する。正規化後の  $z$  軸平均値を、2 次元配列に代入する。2 次元配列を TopView 画像として生成し、深層学習の入力とする。

点群の  $z$  軸抽出範囲を設定する理由を説明する。フォワードの高さ以上に存在する樹木の枝等は、通行の妨げとならないが、路面の高さを求めるためには、路面以外の点群を除去しなければならない。フォワードの標準的な高さを用いて、 $z$  軸抽出範囲を設定することで、フォワードの高さ以上に存在する樹木の枝等の点群を予め除去する。



$$I(i, j) = \begin{cases} -1.0 & (n = 0) \\ C_z(i, j)/2.5 & (otherwise) \end{cases}$$

where  $C_z$  is z-axis value of the centroid,  
 $n$  is the number of points in a cell.

**Fig. 3.3:** Conversion of the point cloud into the TopView image.

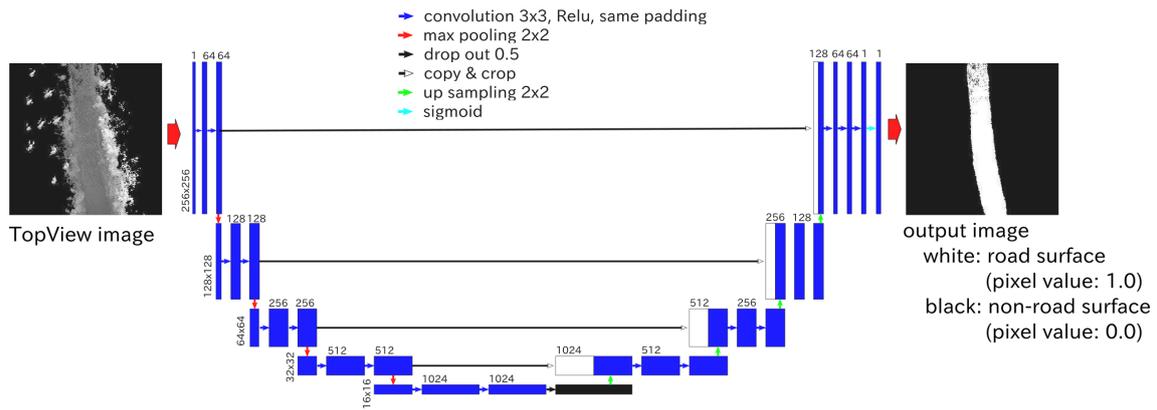


Fig. 3.4: U-Net architecture.

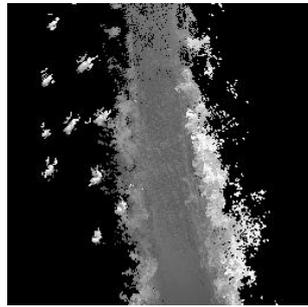
### 3.4.3 U-Net のアーキテクチャ

本研究で使用した U-Net のアーキテクチャを，Fig. 3.4 に示す．U-Net の作成，および学習には，Tensorflow，および keras を使用する．学習後の U-Net に，新しく得られた TopView 画像を入力することで，路面の範囲を検出した画像（以下，出力画像と称する）を得る．

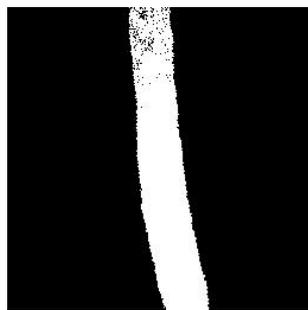
### 3.4.4 データセットの作成方法

TopView 画像 Fig. 3.5(a) と，正解画像 Fig. 3.5(b) のペアを，教師データとする．正解画像の作成方法は，次の手順で行う (Fig. 3.6) ．

- 1 秒ごとに，センサから得られたデータの蓄積を行う．
- 蓄積されたセンサーデータを用いて，LiDAR-SLAM を実行し，3D-LiDAR の位置姿勢と点群を取得する．
- 全経路において得られた，3D-LiDAR の位置姿勢と，それに対応した点群を保存する．
- 3D-LiDAR の各位置ごとに，Point Cloud Library を用いた点群のスージング，点群ごとの法線ベクトルを算出する．
- 3D-LiDAR の各位置ごとに， $x - y$  方向  $0.05 \text{ m} \times 0.05 \text{ m}$  毎にセルを設定する．
- 法線ベクトル  $n = (n_x, n_y, n_z)$  ごとに，勾配を表す数値  $d = (|n_x| + |n_y|) / \sqrt{n_x^2 + n_y^2 + n_z^2}$  を算出する．
- セル内における  $d$  の平均値を画素値とすることで，勾配強調画像を作成する．
- 勾配強調画像において，転圧の形跡，および轍となっている範囲を，目視にて確認し，路面として画素値を 1.0 とする．さらに，路面以外を画素値 0.0 とする．これを正解画像とする．

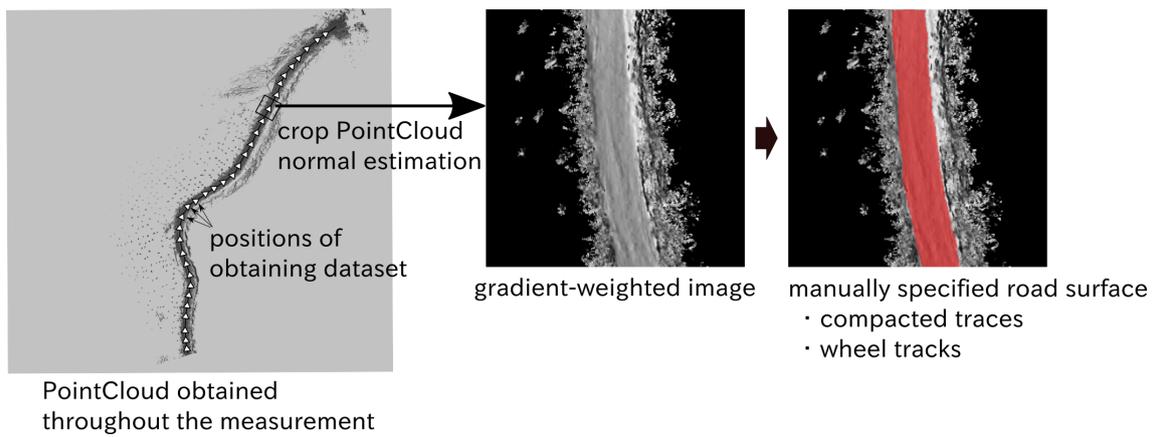


(a)



(b)

**Fig. 3.5:** Training data. (a) TopView image, (b) Ground truth.



**Fig. 3.6:** Obtaining ground truth images.

## 3.5 実験

森林作業道の検出というタスクにおいて，U-Net と FCN8S を，同一のデータセットを用いて比較することで，U-Net が有効であることを確認する．U-Net・FCN8S を用いて路面検出を行い，正解画像を比較することで，森林作業道における路面検出精度を算出する．路面検出精度は IoU (Intersection over Union) を用いて，路面検出精度を数値化する．IoU は，セマンティック・セグメンテーションにおける，検出精度の評価に用いられる指標である．

さらに，路面検出手法が，フォワードの森林作業道に沿った走行制御に使用されることを考慮し，路面の輪郭検出精度を算出する．路面輪郭の精度は，出力画像の輪郭部において，偽陽性のピクセル ( $FP$ )，および偽陰性のピクセル ( $FN$ ) をカウントすることにより，数値化する．

- 偽陽性のピクセル ( $FP$ )：出力画像において路面と判定し，正解画像においては路面でないピクセル
- 偽陰性のピクセル ( $FN$ )：出力画像において路面でないと判定し，正解画像においては路面であるピクセル

偽陽性のピクセル ( $FP$ ) は，路面でない面を走行可能と判断しているため，対応する面の走行は危険である．偽陰性のピクセル ( $FN$ ) は，走行可能な面を走行できないものとして判断している．安全ではあるものの，自律移動時に本来通行できる面を，通行不可と判断することとなる．

### 3.5.1 データセットの作成と学習

山梨県内の森林作業道においてデータセットの取得を行う．

- Fig. 2.4 に示す装置により，森林作業道内を走行
- 総走行距離 117 m 走行し，LiDAR-SLAM の出力点群を，走行距離 3.0 m 毎に取得
- データセットとして，TopView 画像，および正解画像を，計 40 ペア作成
- 走行距離 3.0 m 毎交互に，学習用データセット (20 ペア)，検証用データセット (20 ペア) へ振り分け
- 学習用データセットの TopView 画像，および正解画像に対して，範囲-50 deg から 50 deg，分解能 10 deg ごとに，回転した画像を生成することで，学習用データセットを拡張 (拡張後の学習用データセット数：220 ペア)

U-Net と FCN8s に対して同条件の最適化関数とパラメータで学習を行う．

- 最適化関数：Adam (学習率：0.001)
- バッチサイズ：8
- 学習回数：80 エポック

**Table 3.1:** Results of road surface detection accuracy in validation dataset.

method	max IoU	mid IoU	min IoU	mean IoU	S.D.
U-Net	<b>95.2%</b>	<b>90.6%</b>	<b>86.1%</b>	<b>90.2%</b>	<b>3.0%</b>
FCN-8s	89.0%	83.2%	75.6%	83.0%	3.9%

### 3.5.2 路面検出の精度評価

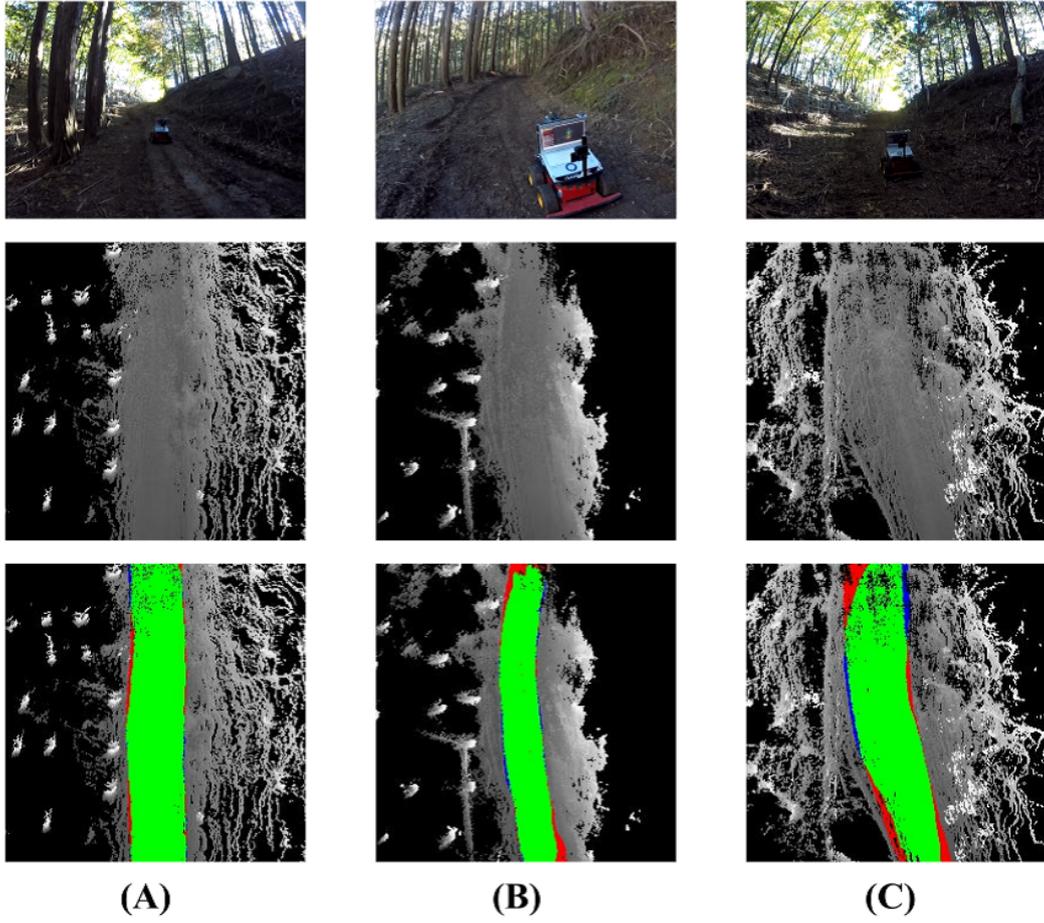
セマンティック・セグメンテーションの評価指標である，IoU を用いて，検証用データセットにおける，路面検出精度 (Table 3.1) を算出する．IoU の定義を以下に示す．

$$IoU = \frac{S_{TP}}{S_{TP} + S_{FP} + S_{FN}} \quad (3.1)$$

ここで， $S_{TP}$ ， $S_{FP}$ ， $S_{FN}$  はそれぞれ，

- $S_{TP}$ ：出力画像，および正解画像ともに，路面と判定されたピクセル ( $TP$ ) の総数
- $S_{FP}$ ：出力画像において路面，正解画像において路面でない，と判定されたピクセル ( $FP$ ) の総数
- $S_{FN}$ ：出力画像において路面でない，正解画像において路面，と判定されたピクセル ( $FN$ ) の総数

とする．路面検出の例として，U-Net の出力から得られた IoU が，最大値・中間値・最小値となる地点の，周辺風景・TopView 画像・比較画像を Fig. 3.7 に示す．Fig. 3.7 における比較画像の各ピクセル値は， $TP$ ：緑色， $FP$ ：青色， $FN$  赤色に着色し，条件に当てはまらないピクセルを，TopView 画像の濃淡値としている．



**Fig. 3.7:** Results of the road surface detection by the U-Net. (A) maximum IoU, (B) mean IoU, (C) minimum IoU. In the bottom images, the green, red, and blue pixels represent  $TP$ ,  $FN$ , and  $FP$ , respectively.

### 3.5.3 路面輪郭の精度評価

森林作業道の横断方向における路面輪郭精度 (以下, 路面輪郭精度と称する) の算出を行う。出力画像における,  $y$  軸方向 (Fig. 3.8 における  $y$  軸) を, 森林作業道の横断方向として, 路面輪郭精度は,  $y$  軸方向に走査した際の,  $FP$  と  $FN$  の長さ (Fig. 3.8 における  $a, b$ ) を測定することで得る。

正解画像における路面の輪郭に対して, 出力画像における路面の輪郭が, 外側に誤推定された場合の長さを  $a$  として表している。同様に, 正解画像における路面の輪郭に対して, 出力画像における路面の輪郭が, 内側に誤推定された場合の長さを  $b$  として表している。以下, 路面輪郭精度における  $a$  を  $FP$  長さ,  $b$  を  $FN$  長さとする。 $x$  軸方向 (Fig. 3.8 における  $x$  軸) は, 進行方向を表している。 $x$  軸方向の範囲  $0\sim 2\text{ m}$ ,  $2\text{ m}\sim 4\text{ m}$ ,  $4\text{ m}\sim 6\text{ m}$  ごとに,  $FP$  長さ・ $FN$  長さの平均値, 最大値を算出し, 路面輪郭精度を評価する (Table 3.2)。

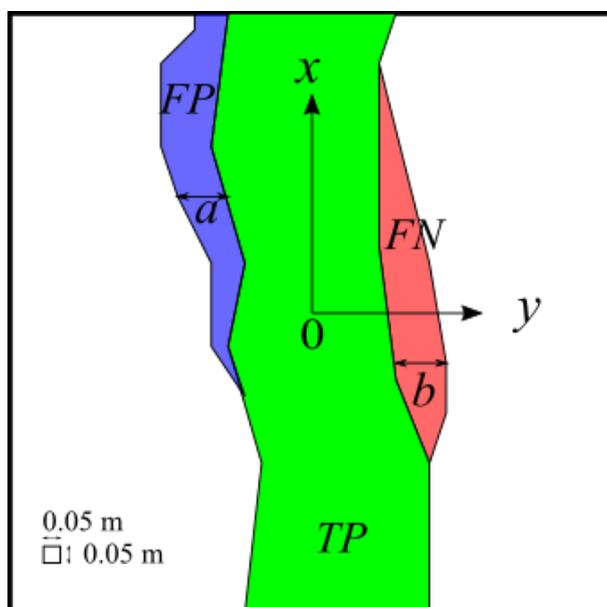


Fig. 3.8: Detection error of road surface contour.

**Table 3.2:** Results of road surface contour detection errors in validation dataset.

method	$x$ -axis	0~2 m	2 m~4 m	4 m~6 m
U-Net	max $a$ m	<b>0.150</b>	<b>0.150</b>	<b>0.250</b>
	mean $a$ m	<b>0.014</b>	<b>0.008</b>	<b>0.030</b>
	S.D. $a$ m	<b>0.025</b>	<b>0.011</b>	<b>0.028</b>
FCN-8s	max $a$ m	0.300	0.600	1.250
	mean $a$ m	0.026	0.044	0.069
	S.D. $a$ m	0.040	0.054	0.067
U-Net	max $b$ m	<b>0.200</b>	<b>0.300</b>	<b>0.550</b>
	mean $b$ m	0.041	<b>0.046</b>	0.072
	S.D. $b$ m	<b>0.033</b>	<b>0.028</b>	0.076
FCN-8s	max $b$ m	0.300	0.450	0.750
	mean $b$ m	<b>0.027</b>	0.047	0.072
	S.D. $b$ m	0.043	0.040	<b>0.065</b>
U-Net	max $(a + b)$ m	<b>0.350</b>	<b>0.450</b>	<b>0.800</b>
	mean $(a + b)$ m	0.055	<b>0.054</b>	<b>0.102</b>
	S.D. $(a + b)$ m	<b>0.058</b>	<b>0.039</b>	<b>0.104</b>
FCN-8s	max $(a + b)$ m	0.600	1.050	2.000
	mean $(a + b)$ m	<b>0.053</b>	0.091	0.142
	S.D. $(a + b)$ m	0.830	0.095	0.132

## 3.6 考察

### 3.6.1 IoU の算出結果に対する考察

IoU の算出結果 (Table 3.1) を考察する。はじめに、IoU 値を用いて U-Net と FCN8s を比較すると、平均値、標準偏差、最大値、中間値、最小値の全てにおいて、U-Net が FCN8s の IoU 値を上回った。FCN8s のネットワークは、畳込みを複数回行った後、逆畳込み込みを行い、路面判定結果を出力する。畳込み込みを行うごとに、より複雑な情報処理が可能となるが、輪郭などの詳細な位置情報は失われていく。U-Net のネットワーク構造は、畳込みを行う階層ごとに得られた特徴量を保存し、逆畳込み込みの際に結合する。U-Net のネットワーク構造は、位置情報を保持した出力が得られるため、森林作業道の路面推定というタスクにおいて、FCN8s と比較して良好な結果が得られたと考えられる。

次に、Fig. 3.7 の (A) 地点において IoU が最大値となった原因を、路面形状から考察すると、(A) 地点のような直線経路では、学習用データセット、および検証用データセットに似た形状のデータが多く含まれていたため、IoU 値が高くなったと思われる。地点 (C) 地点において IoU が最小値となった原因を考察すると、地点 (C) は他の地点と比較して勾配が穏やかな箇所であり、このような箇所は拡張前の学習用データセット 20 セット中、1 セットのみであった。データセット全体に占める割合が少なかったことから、地点 (C) の IoU 値が低下したと考えられる。

### 3.6.2 路面輪郭精度の算出結果に対する考察

評価に用いた各パラメータについて、導入した意図を説明する。 $FP$  長さ  $a$  は、出力画像における路面の輪郭が、正解画像における路面輪郭に対して、外側に誤推定された場合の長さを表している。自律移動の際、路面の外側の走行を避けるためには、 $FP$  長さ  $a$  は 0 となることが望ましい。 $FP$  長さ  $a$  が 0 とならない場合、あらかじめ、 $FP$  長さ  $a$  の最大値を求め、 $FP$  長さ  $a$  の最大値より内側の路面のみ、走行可能な面として出力する必要がある。

$FN$  長さ  $b$  は、出力画像における路面輪郭が、正解画像における路面輪郭に対して、内側に誤推定された場合の長さを表している。自律移動の際、 $FN$  長さ  $b$  が、森林作業道において確保されているクリアランス以上となった場合、幅員が不足しているとして、本来であれば通行可能な道路を、通行不能と判定してしまう。さらに、安全のため、 $FP$  長さ  $a$  の最大値より内側の路面のみ、走行可能な面として出力する場合、 $FN$  長さ  $b$  は、最低限確保されているクリアランスが 0.25 m として、 $(0.25 \text{ m} - a)$  以下となる必要がある。よって、森林作業道の路面検出において、 $(a + b)$  が 0.25 m 以下となる必要がある。

路面輪郭精度の算出結果を考察する。路面輪郭精度の算出結果 (Table 3.2) における、 $(a + b)$  の平均値を参照すると、 $0.102 \pm 0.104 \text{ m}$  以下となり、森林作業道の最低限のクリアランスとしている 0.25 m 以下の誤差であった。 $(a + b)$  の平均値が、クリアランスとしている 0.25 m 以下であったため、実験環境の主な範囲において、フォワードの自律移動に、本路面検出手法が使用可能である。

一方、 $(a + b)$  の最大値は、前方を表す  $x$  軸値の増加に伴い増加し、 $x$  軸値が 0 m から 2 m の間 0.35 m、 $x$  軸値が 4 m から 6 m の間 0.8 m となった。 $x$  軸値の増加に伴い  $(a + b)$  の最大値が増加した原因として、LiDAR により取得された点群が、遠方ほど疎であった。TopView 画像において、形状情報をもつピクセルが減少し、路面推定精度の低下に寄与

したと考えられる。

今後の展望として、 $(a + b)$  の最大値は、0.25 m 以下となることが望ましいことから、より高密度かつ広範囲に、点群の取得が可能な LiDAR を使用し、TopView 画像の解像度、および形状情報をもつピクセル数を向上することで、路面推定精度の向上を図る。さらに、より高精度な路面推定が可能なネットワークの構築、および路面輪郭の曲線近似等による、路面輪郭における外れ値の除去等により、最大誤差を削減することを検討する。

# Chapter 4

## 森林作業道における自律移動システムの構築

### 4.1 概要

本章では、森林作業道の自律移動を実現するために必要な、地図作成機能、自己位置推定機能、および経路計画機能について説明する。地図作成機能、自己位置推定機能、および経路計画機能を統合することで、自律移動に必要となる、自己位置の推定、および目標経路に沿う制御が可能となる。森林作業道の自律移動を実現するためには、クリアランス 0.25 m 以内の位置制御誤差で、自律移動を行う必要がある。SLAM による地図作成機能、および自己位置推定を行う機能は、2 章において提案したスキャンマッチング手法を用いて構築する。さらに、経路計画機能は、3 章において提案した路面検出手法、および Dynamic Window Approach[38] により構築する。地図作成機能、自己位置推定機能、経路計画機能を統合することで、自律移動システムを構築する。

### 4.2 森林作業道の自律移動に必要な機能

#### 4.2.1 自動化を行う作業

本研究で構築する自律移動システムは、将来的にフォワーダに搭載することで、自律移動によるフォワーダの無人化を目標としている。無人化により、フォワーダを用いた搬出作業の、省力化、および危険作業の代替が可能となる。フォワーダによる木材の搬出作業は、伐採に関わる工程において実施される。伐採には、主伐（木材となるために成長した木を伐採し、収穫する作業）や、間伐（森林の間隔を作り、主伐する木の成長を促すために、木を間引く作業）と呼ばれる工程があり、主伐・間伐において、木材の搬出作業が実施される。フォワーダによる搬出作業は、

- 空荷の状態で、先山（切り出した木材の集積地）まで走行
- 木材を積載した状態で、先山から土場（先山から搬出した木材の集積地）まで走行
- 再び空荷の状態で、土場から先山まで走行

の繰り返し作業 (Fig. 4.1) である。自律移動システムを搭載したフォワーダの使用方法としては、先山における積込作業、および土場における積み下ろし作業は、有人により実施

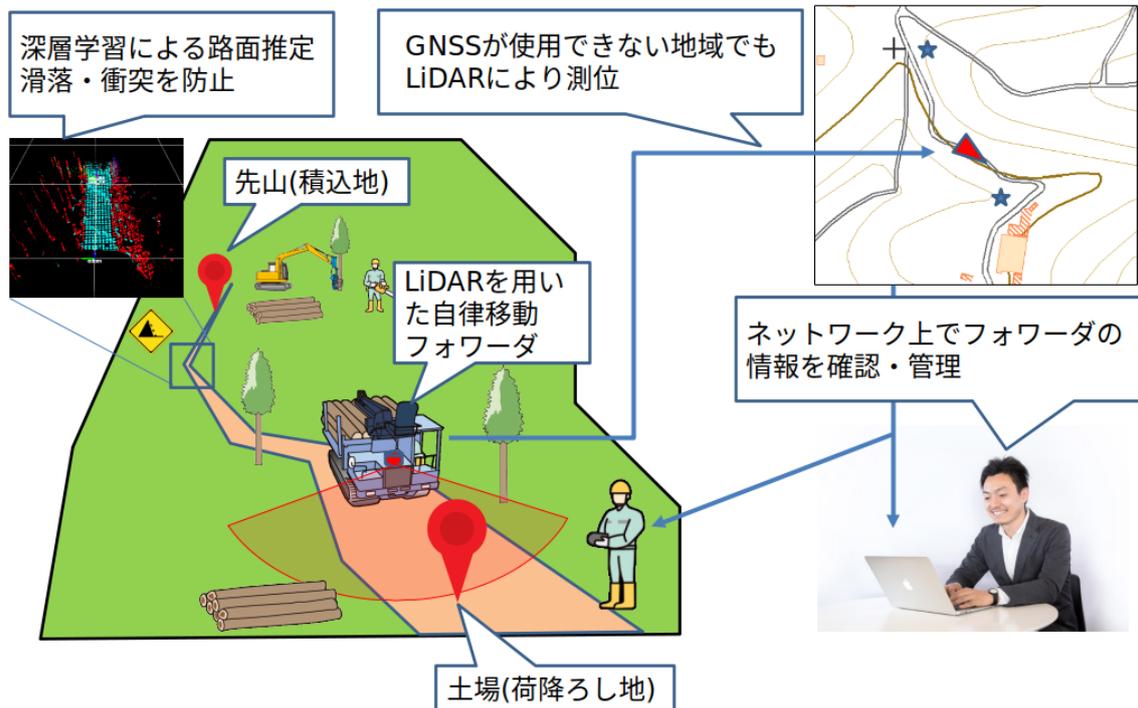


Fig. 4.1: Automation of forwarders with the autonomous mobility system.

し、先山・土場間の走行を、自律移動システムにより自動化する。

自動化したフォワーダを用いた搬出作業は、次の手順により実施することを想定している。

1. 手動運転により、先山と土場を往復し、走行経路の周辺地図を作成（1日に1度、作業開始前に実行。さらに、走行経路上の安全を確認）
2. 空荷の状態では、先山（切り出した木材の集積地）まで自律移動
3. 木材をフォワーダに積載（有人作業）
4. 木材を積載した状態で、先山から土場（先山から搬出した木材の集積地）まで自律移動
5. 木材をフォワーダから搬出（有人作業）
6. 搬出作業を終了するまで2.～5.を繰り返し実施

となる。本研究では、搬出作業時における、フォワーダの自律移動を実現するシステムについて構築する。

#### 4.2.2 自律移動を実現するための機能

搬出作業の自動化を実現するため、自律移動システムとして、次の機能を開発する。

- 地図作成機能: スキャンマッチングを用いた SLAM により, 周辺の 3 次元点群を取得し, 地図を作成する機能である. 運用時は, 1 日の作業開始前に, 先山 (木材の積込先), と土場 (木材の搬出先) 間の経路を手動により走行し, スキャンマッチングを用いた SLAM を行う. SLAM により得られた経路上の点群を, 地図として保存する. さらに, 地図作成時の走行経路を, 目標経路として保存する.
- 自己位置推定機能: 事前作成した地図に対してスキャンマッチングを行うことで, 地図上の現在位置を取得する機能である. さらに, 本機能は, スキャンマッチングにより得た点群を蓄積する機能をもつ. 蓄積した点群は, 経路計画機能に用いる.
- 経路計画機能: 最適なパスを生成した速度・角速度を, 車両に対して出力することで, 目標経路に沿った移動を実現するための機能である. 経路計画機能は, 自己位置推定機能により得た点群を用いて, 深層学習による路面検出を行う. さらに, Dynamic Window Approach[38] により, 現在位置・目標経路・路面検出結果を用いて, 速度・角速度値から複数のパスの生成, および最適なパスの選択を行う. 最適なパスを生成した速度・角速度を, 車両に対して出力することで走行制御を行う.

これら機能は, Autoware[11, 12] 等の自律移動システムにおいて, 既に構築されている. しかし, フォワーダによる木材の運搬作業という, 特殊条件下における自律移動を実現するため, 本研究において構築したスキャンマッチング手法と路面検出手法を用いる.

#### 4.2.3 自律移動時の環境・条件

本研究の自律移動システムは, フォワーダによる木材の搬出作業時に使用されることを想定している. 森林作業道における運搬作業を自動化する上で, 自律移動に関わる環境・条件を次のように想定した.

1. 移動経路は先山と土場を単純に往復するものとする
  - 複数の先山・土場の移動は想定しない
2. 搬出作業日ごとに 1 回, 土場・先山間を往復し, 地図を作成
  - 土砂崩れ・倒木・落石等が発生するため, 運搬作業の開始前に, 森林作業道の状態確認が必要
  - 地形, および周辺物体の変形が発生しうるため, 作業日ごとに地図の更新が必要
  - 準備に要する時間を削減するため, 往復移動と同時に地図を作成し, 往復後 30 分以内に, 地図作成を完了
3. クローラ型のフォワーダを想定
4. 走行路の縦断勾配は 0% ~ 17% となることを想定
5. 森林作業道の斜面勾配は 0% ~ 35% となることを想定

6. 作業道のクリアランスは、最低でも 0.25 m 確保されていることを想定
7. 空車時、0.9 m/sec の走行速度を想定
8. 木材の積載時、0.6 m/sec の走行速度を想定

本研究では、これら条件を想定し、自律移動システムの構築を行う。

#### 4.2.4 自律移動時における速度

自律移動を行うフォワーダの生産性を、有人時と比較して低下させないために、自律移動時の移動速度を、有人によるフォワーダの運転時と同程度となるよう設定する。有人によるフォワーダの走行速度は、

- 車幅と作業道幅員の比率
- 作業道の縦断方向勾配
- 作業道の曲率半径
- 木材の積載量
- フォワーダの性能

により変化することがわかっている [41, 42, 43]。岡勝ら [42] による調査を参考とし、車幅と作業道幅員の比率から算出される有人走行時の速度を求めた。作業道のクリアランスが 0.25 m、フォワーダの幅が 1.9 m、横断勾配が 10 度 (17.6%) 以内の場合、車幅と作業道幅員の比率から算出される有人走行時の速度は、

- 木材の積載時：約 0.6 m/sec
- 空車時：約 0.9 m/sec

であった。本研究では、木材の積載時約 0.6 m/sec、空車時の速度約 0.9 m/sec での移動を可能とする、自律移動システムを開発する。

### 4.3 地図作成機能の構築

#### 4.3.1 運用時の要求事項

自律移動に用いる地図は、自律移動を実施する環境毎に作成する。先山・土場間の作業道を、運搬作業の開始前に 1 度往復することで、走行経路上の安全確認を行う。同時に地図作成機能により、通行する作業道周辺の地図を作成する。さらに往復時の自己位置推定結果を、目標経路として保存する。地図は運搬作業日毎に、削除・新規作成することを想定する。速やかな自律移動運搬作業へと移行するため、先山・土場間の往復走行後、30 分以内に地図作成を完了する。

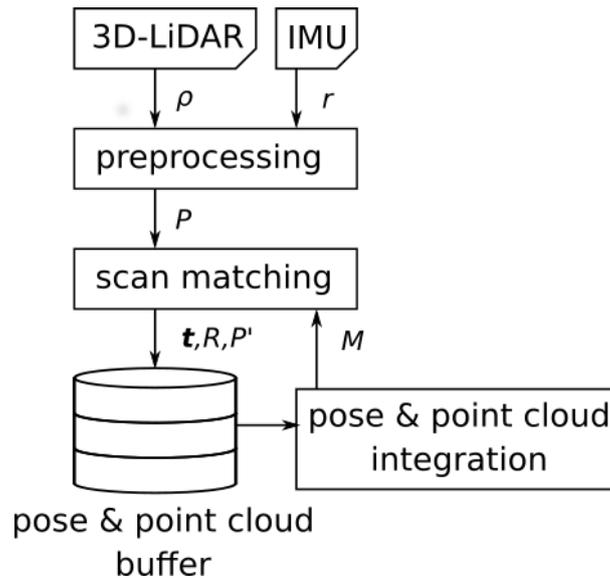


Fig. 4.2: Block diagram of the SLAM software system.

#### 4.3.2 スキャンマッチングによる地図作成の手順

スキャンマッチングを用いた SLAM により，地図作成を行う．SLAM による地図作成は，次の手順により行う (Fig. 4.2) ．

1. 3D-LiDAR，および IMU による点群の取得
2. スキャンマッチングによる自己位置推定，および点群の歪み補正
3. 位置姿勢推定値，および歪み補正済みの点群の蓄積と統合

#### 4.3.3 スキャンマッチングによる自己位置推定と点群の復元

2 章において提案したスキャンマッチング手法を使用する．3D-LiDAR・IMU により得られた点群 (以下，入力点群  $P$  と称する)，逐次処理により事前に補正済みの点群 (以下，参照点群  $M$  と称する) を用いてスキャンマッチングを行う．スキャンマッチングの出力として，位置姿勢推定値  $t, R$ ，および入力点群に対して，歪み補正を行った点群  $P'$  を取得する．地図作成の開始時，参照点群  $M$  は存在しないため，停車状態で入力点群  $P$  を取得し，これを参照点群  $M$  として用いる．

#### 4.3.4 位置姿勢推定値と歪み補正済みの点群の蓄積と統合

スキャンマッチングにより得られた，位置姿勢推定値  $t, R$ ，および歪み補正済みの点群  $P'$  は，逐次データベースに蓄積する．データベースは，Index をスキャンマッチングの実行した回数とし，フィールドとして，Index，位置姿勢推定値  $t, R$ ，歪み補正済みの点群  $P'$  を持つ (Fig. 4.3) ．地図作成のための先山・土場間の往復走行中に，順次スキャンマッチングを実行し，データベースに蓄積する．

参照点群  $M$  は，データベースのすべてのリストから，歪み補正済みの点群座標  $P'$  を，

index	pose ( $\mathbf{t}, R$ )	point cloud $P'$
1	$\mathbf{t}_1, R_1$	$P'_1$
2	$\mathbf{t}_2, R_2$	$P'_2$
$\vdots$	$\vdots$	$\vdots$

Fig. 4.3: Database structure.

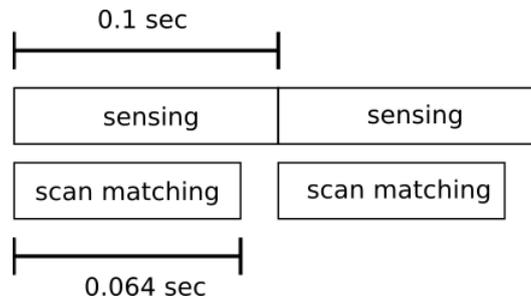


Fig. 4.4: Processing time required for SLAM.

時系列順に順次読み出し，1つの点群  $M$  として結合する．点群  $M$  は，スキャンマッチング時の参照点群として用いる．先山・土場間の往復後，点群  $M$  を地図として保存する．さらに，データベースから，位置情報を時系列順に読み出すことで，地図作成時の走行経路を取得する．得られた走行経路は，自律移動時の目標経路として用いる．

#### 4.3.5 地図作成機能の処理時間の処理時間

地図作成に要する時間を測定した．スキャンマッチングによる SLAM(スキャンマッチング，位置姿勢推定値および歪み補正済み点群の蓄積，歪み補正済み点群の統合を行う処理) に要した時間は，実験に用いたハードウェア・ソフトウェア環境 (5.2.2) において，平均 0.064 sec であった．3D-LiDAR，および IMU による点群の取得は，0.1 sec ごとに出し，スキャンマッチングによる SLAM に対して，別プロセスにて実行される (Fig. 4.4)．3D-LiDAR，および IMU によるセンシング中に，スキャンマッチングによる SLAM に関わる処理が完了するため，地図作成時，スキャンマッチングによる SLAM は，遅延なく実行することが可能である．

地図作成時，走行中のセンシングと同時に，遅延なく SLAM を実行しているため，地図作成のための走行が完了すると同時に，地図を出力し，速やかに自律移動を開始することができる．

#### 4.3.6 森林作業道における地図作成の実行例

森林作業道においてスキャンマッチングによる SLAM を実行し，得られた点群を Fig. 4.5 に示す．Fig. 4.5 は，森林作業道における，上部からの視点により，点群全体を描画した画像，および特定の視点から点群を描画した画像を表示している．

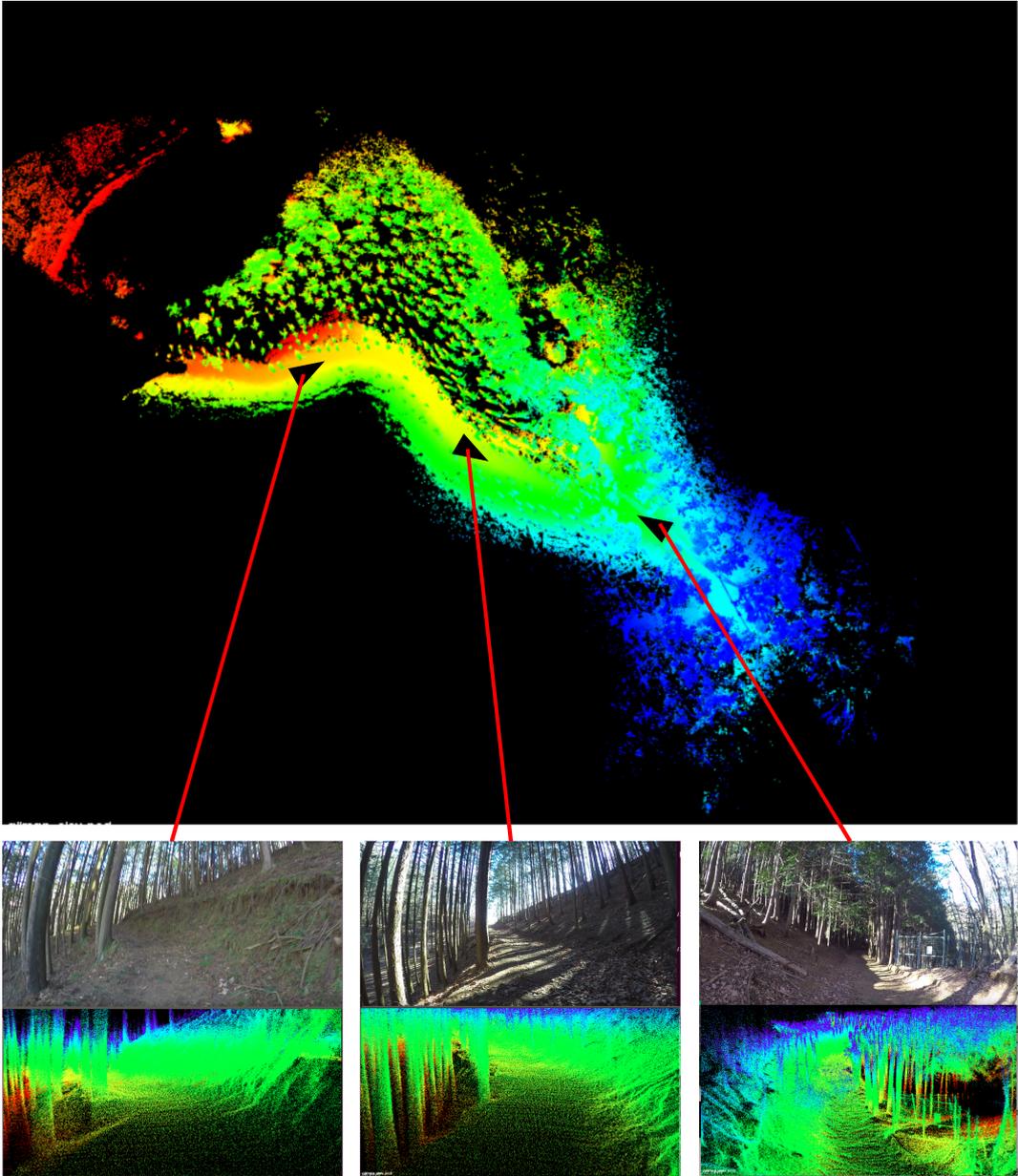


Fig. 4.5: Example of 3D mapping using SLAM.

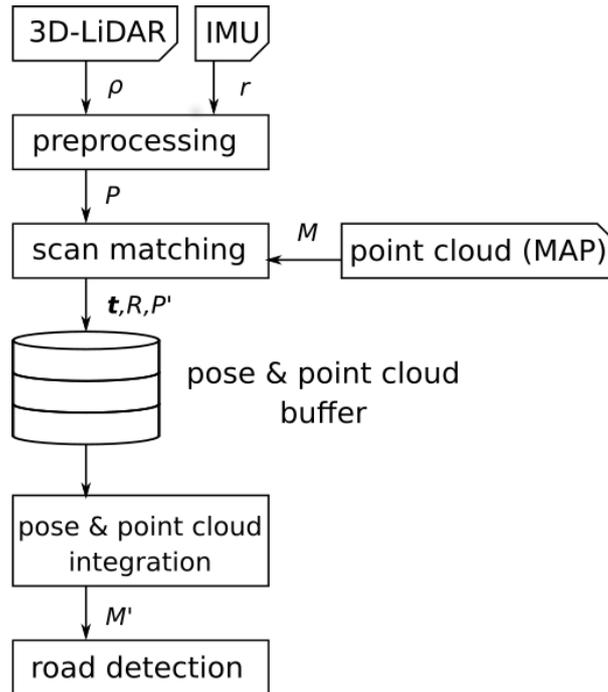


Fig. 4.6: Block diagram of the Localization software system.

## 4.4 自己位置推定機能の構築

### 4.4.1 スキャンマッチングによる自己位置推定の手順

自己位置推定機能 (Fig. 4.6) は、地図作成時と同様に、入力点群をスキャンマッチングにより参照点群に重ね合わせることで、自己位置を推定する。ただし、地図作成時と異なり、参照点群  $M$  は、地図作成機能により得られた点群を読み出すことで取得する。

自己位置推定は次の手順により行う。

1. 地図作成機能により得られた点群の読込
2. 3D-LiDAR、および IMU による点群の取得
3. スキャンマッチングによる自己位置推定、および点群の蓄積・統合

### 4.4.2 スキャンマッチングによる自己位置推定と点群の復元

自律移動の開始前に、地図作成により得られた点群を、参照点群  $M$  として読み込む。3D-LiDAR、および IMU により得られた点群を、入力点群  $P$  として取得する。さらに、参照点群  $M$  に対して、入力点群  $P$  を、2章において提案したスキャンマッチング手法により、歪み補正、および重ね合わせを行う。スキャンマッチングにより得られた位置姿勢推定値  $t, R$  が、地図に対する自己位置推定結果となる。さらに、スキャンマッチングにより得られた  $P'$  を、地図作成時と同様に蓄積、および統合することで、自律移動時における、周辺の形状情報として取得する。自己位置推定時に得られた走行経路上の点群を  $M'$  とし、経路計画機能の入力として用いる。

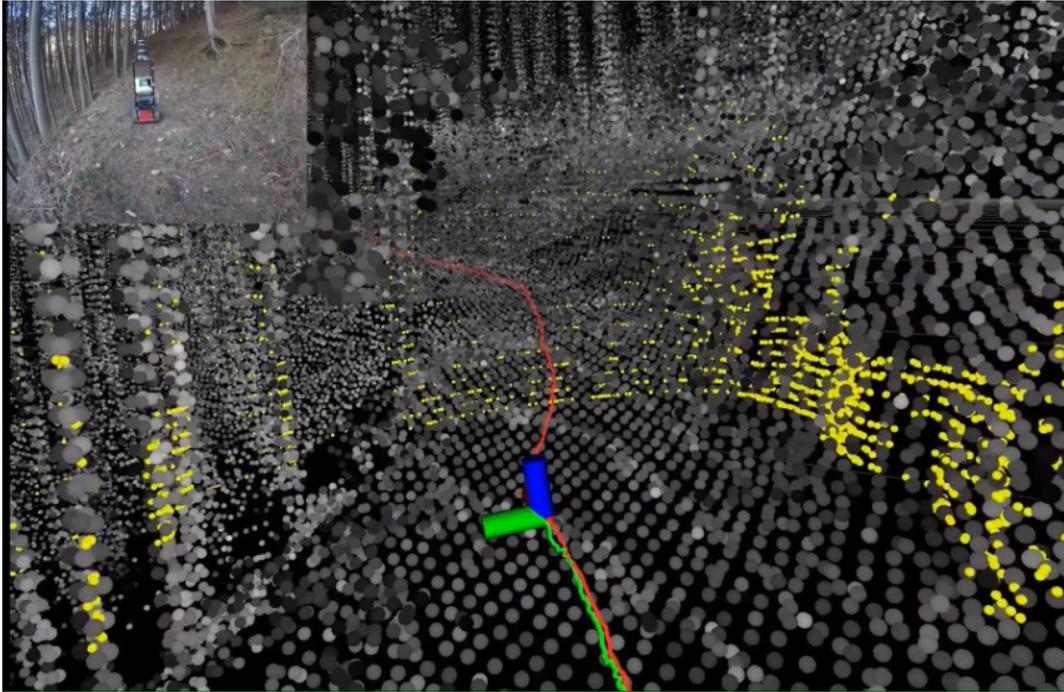


Fig. 4.7: Example of localization.

#### 4.4.3 処理時間

スキャンマッチングによる自己位置推定に必要な処理時間は、実験に用いたハードウェア・ソフトウェア環境 (5.2.2) において、平均 0.064 sec となった。地図作成時と同様に、センシングとは別プロセスにて、スキャンマッチングによる自己位置推定、および点群の蓄積と統合を実行する。LiDAR、および IMU によるセンシングは 0.1sec 毎に実行する。スキャンマッチングを 0.1 sec 以内に実行することから、遅延なく自己位置を行うことができる。

#### 4.4.4 森林作業道における自己位置推定の実行例

自己位置推定の実行例を Fig. 4.7 に示す。Fig. 4.7 における点群 (灰色) は、地図として用いる点群である。点群 (黄色) は、スキャンマッチングによる歪み補正、および重ね合わせ後の入力点群である。

## 4.5 経路計画機能の構築

### 4.5.1 経路計画の手順

経路計画機能は、自己位置推定機能により得られた、走行経路の周辺形状を表す点群  $M'$ 、および Dynamic Window Approach を用いることで、目標経路に追従したパスの生成と、車両に対する速度・角速度の目標値を出力する。

経路計画機能は次の処理を行う。

- 地図作成時に得られた、目標経路の取得
- 自己位置推定機能により得られた、走行経路の周辺形状を表す点群  $M'$  の取得
- 点群  $M'$  を入力とした路面推定
- 点群  $M'$  を入力とした障害物の推定
- Dynamic Window Approach によるパスの生成と評価
- 最適なパスを生成した速度、および角速度の出力

### 4.5.2 目標経路の取得

自律移動は、目標経路に車両が追従するよう制御する。目標経路は、地図作成時に得られた自己位置推定結果を用いる。地図作成後のデータベース (Fig. 4.3) から、Index 順に逐次、位置推定値  $t$  を読み出す。目標経路を表す座標を  $T$  とし、 $T = [t_1, \dots, t_N]$  として、Index 順に逐次読み出された位置推定値を目標経路とする。ただし、 $i$  は Index の値である。 $N$  はデータベースに登録された Index の最大値であり、 $t_1$  は地図作成開始時の位置推定値、 $t_N$  は地図作成終了時の位置推定値を表す。

### 4.5.3 走行経路上の点群を入力とした路面推定

自己位置推定機能により得られた、走行経路周辺の形状を表す点群  $M'$  を入力として、路面推定を行う。3章において提案した、路面推定手法により、

- 点群  $M'$  を TopView 画像に変換
- 学習済みネットワークによる路面検出

を行い、路面検出結果である出力画像を、路面情報  $I_{road}$  として取得する。

### 4.5.4 走行経路上の点群を入力とした障害物判定

TopView 画像を用いて、高さが一定以上の物体を障害物として検出し、障害物の検出結果を表す画像  $I_{collision}$  を出力する。

#### 4.5.5 Dynamic Window Approach によるパスの生成

Dynamic Window Approach[38](以降, DWA と呼ぶ) を使用し, パスの生成を行う。入力として,

- 目標経路の座標  $T$
- スキャンマッチングにより得られた並進速度  $v_{in}$ , 角速度  $w_{in}$
- 路面情報  $I_{road}$
- 障害物情報  $I_{collision}$

を使用する。DWA は次の処理により, 入力から車両に対する命令値を算出する。

1. 入力である速度値  $v_{in}$ , 角速度値  $w_{in}$ , および各種パラメータにより, 車両の目標速度  $v$ , 目標角速度  $w$  の取り得る組み合わせを決定
2.  $v, w$  からパス  $L_{v,w} = [t_1(v, w), \dots, t_N(v, w)]$  を生成
3. 評価関数  $G(v, w) = \alpha * heading(T, L_{v,w}) + \beta * dist(I_{road}, I_{collision}, L_{v,w}) + \gamma * velocity(L_{v,w})$  により, 経路  $L_{v,w}$  ごとに評価値を算出
4. 最も評価値の高いパスを求め, その経路を生成した速度・角速度  $(v, w)$  を車両に対するの命令値として出力

本研究における DWA は参考文献 [38] をもとに構築する。参考文献 [38] では,  $dist(v, w)$  において, パス  $L_{v,w}$  と障害物間の距離を算出するが, 本研究の経路計画機能は, 路面情報  $I_{road}$  において, 非路面である領域についても, 障害物として取り扱う。 $I_{road}, I_{collision}$  それぞれにおいて, パス  $L_{v,w}$  と障害物間の距離を算出し, 評価値として使用する。

#### 4.5.6 処理時間

経路計画の処理時間を測定したところ, 実験に用いたハードウェア・ソフトウェア環境 (5.2.2) において,

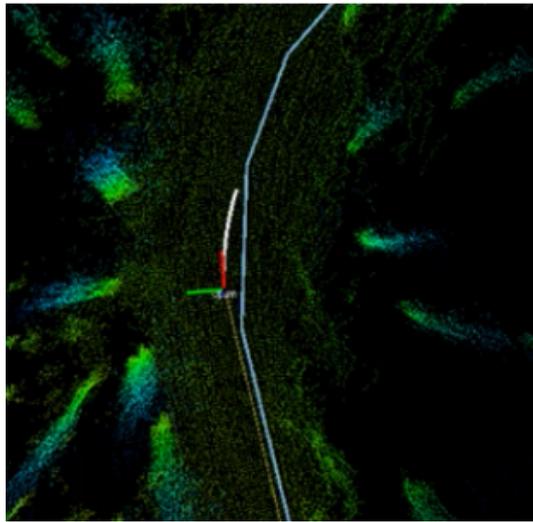
1.  $I_{road}$  の算出: 0.032 sec
2.  $I_{collision}$  の算出: 0.011 sec
3.  $T_{v,w}$  の生成と評価: 0.013 sec

であった。1), 2) は別プロセスとして実行可能なため, 並列処理を行い, 全ての処理に 0.045 sec 必要となる。

センシング結果の取得, 自己位置推定, 経路計画の出力まで, 0.1 sec 以上の処理時間が必要となった。センシング結果は, 0.1 sec 毎に入力される。経路計画機能による処理が終了する前に, 次センシング結果の入力があった場合は, 入力情報は破棄される。このため, 経路計画機能により求められた速度・角速度地は, 0.2 sec ごとに出力される。

#### 4.5.7 森林作業道における経路計画機能の実行例

Fig. 4.8 における白線は, 経路計画機能により生成されたパスのうち最も評価値の高かったパスを表示している。最も評価値の高かったパスを生成した速度  $v$ , 角速度  $w$  を車両に出力することで, 目標経路 (青線) に沿った自律移動が可能となる。



**Fig. 4.8:** Example of path generation.

# Chapter 5

## 森林作業道における自律移動の実験と評価

### 5.1 概要

本章では，森林作業道における，自律移動実験の結果について報告する．4章において構築した自律移動システムを，移動ロボットに搭載し，2箇所の森林作業道において実施した，地図作成，および自律移動の結果を述べる．評価として，自律移動の結果から，目標経路に対する横断方向の位置制御誤差を求め，本研究の目標である，目標経路に対する横断方向の位置制御誤差が，最大 0.25 m 以内となっているか確認する．さらに，森林作業道の各区間における，目標経路に対する横断方向の位置制御誤差を比較し，考察を行う．

### 5.2 評価方法

#### 5.2.1 横断方向の位置制御精度の算出

本研究では，森林作業道のクリアランスが，最低でも約 0.25 m 確保していることを想定している．構築した自律移動システムが自律移動を行う際，目標経路に対して，森林作業道の横断方向に 0.25 m 以内を満たすことを目標とする．ただし，森林作業道の横断方向を厳密に定義することが困難であるため，本研究では，横断方向の位置制御誤差を算出した．

横断方向の位置制御誤差の算出方法は，

1. SLAM 開始時，IMU の加速度ベクトルから，水平面を推定
2. 3次元自己位置推定結果に対して，最も近接する目標経路座標の垂線ベクトル(以下，偏差ベクトルと呼ぶ)を取得
3. 偏差ベクトルを水平面に投影した際の長さを，横断方向の位置制御誤差として取得
4. 車両の進行方向に対して，右側を正值，左側を負値として，横断方向の位置制御誤差を変換

により算出する．横断方向の位置制御誤差における，最大値が 0.25 m 以下であった場合，本システムが，クリアランス 0.25m 以内の森林作業道において，自律移動が可能であると判断する．

## 5.2.2 実験に用いたハードウェア・ソフトウェア

3D-LiDAR を，移動ロボットに搭載する．3D-LiDAR は Velodyne Lidar 社製 VLP-16，3D-LiDAR をチルトさせるサーボとして，Robotis Dynamixel RX-64R，IMU として Xsens Technologies B.V. MTi-30-AHRS を使用する．これらセンサを，車両型ロボット Adept MobileRobots Pioneer 3-AT に搭載する．各種センサより取得した情報は，コンピュータ (CPU Intel Core i7-8650U 1.90 GHz，メモリ 16.0 GB) によって処理を行う．コンピュータの O.S. には，Ubuntu 18.04 を用いた．ロボット用ミドルウェアに，ROS (Robot Operating System) を採用した．

## 5.3 実験手順

### 5.3.1 地図と移動経路の取得

2 箇所の森林作業道において，手動操作により移動ロボットを走行する．土場 (出発地)，先山 (目的地) を想定した地点間を往復する．走行中，本研究において開発した，地図作成機能を用いて，地図作成と目標経路を取得する．

### 5.3.2 自律移動の実施

走行経路を 4 区間に分割し，それぞれの走行経路において，土場 (出発地) から先山 (目的地) までの経路を，登り経路として，先山 (目的地) から土場 (出発地) までの経路を，降り経路として自律移動を行う．森林作業道 A，B において合計 16 回の自律移動実験を行う．

自律移動時，地図作成機能により得られた点群を，地図として使用する．さらに，地図作成時に得られた自己位置推定結果を，目標経路として使用する．得られた地図と目標経路を用いて，自己位置推定機能，経路計画機能により，移動ロボットの自律移動を行う．自律移動時，DWA のパラメータである最高速度を，登り経路の場合 0.9 m/sec，降り経路の場合 0.6 m/sec に設定する．

各経路ごとに 5.2.1 節において説明した方法により評価し，横断方向の位置制御誤差を算出する．

## 5.4 森林作業道 A における実験

### 5.4.1 実験環境

森林作業道 A における自律移動実験は，2021 年 2 月 4 日に実施した．地図作成機能により取得した目標経路を Fig. 5.1 に示す．森林作業道 A における全走行距離は 141 m，標高は，土場 (出発地) が 646.3 m，先山 (目的地) 656.6 m であった．各区間における周辺の様子を Fig. 5.2 に示す．すべての区間においてカーブとなる経路が存在している．さらに，周辺の植生は，左右に杉が植えられており，路面上には落ち葉，および杉の枝が散乱している．

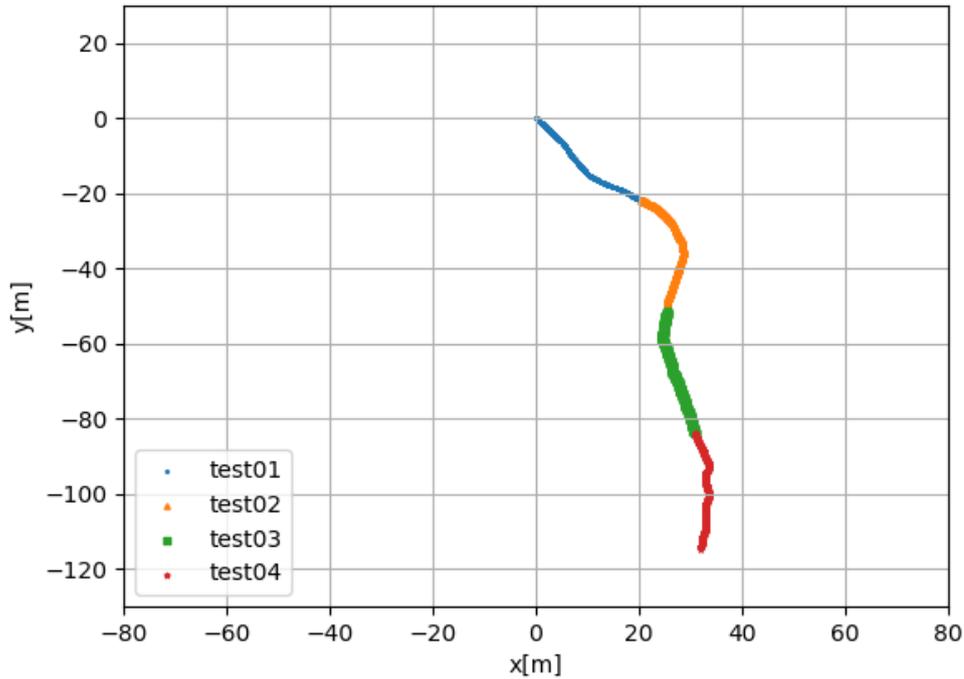


Fig. 5.1: Target path on the forest work road A.

#### 5.4.2 実験結果

森林作業道 A における実験結果を Fig. 5.3 に示す．すべての走行経路において横断方向の位置制御誤差が，目標である 0.25 m 以内となり，森林作業道 A において，目標とする精度での自律移動が可能であることを確認した．

経路ごとに，横断方向の位置制御における，最大誤差に注目すると，登り経路，降り経路にかかわらず，区間 test04 において，他の区間より最大誤差が増加した．さらに，降り経路における test02・test04 区間の最大誤差が，登り経路と比較して，車両の右側方向に増加した．



(a) test01



(b) test02

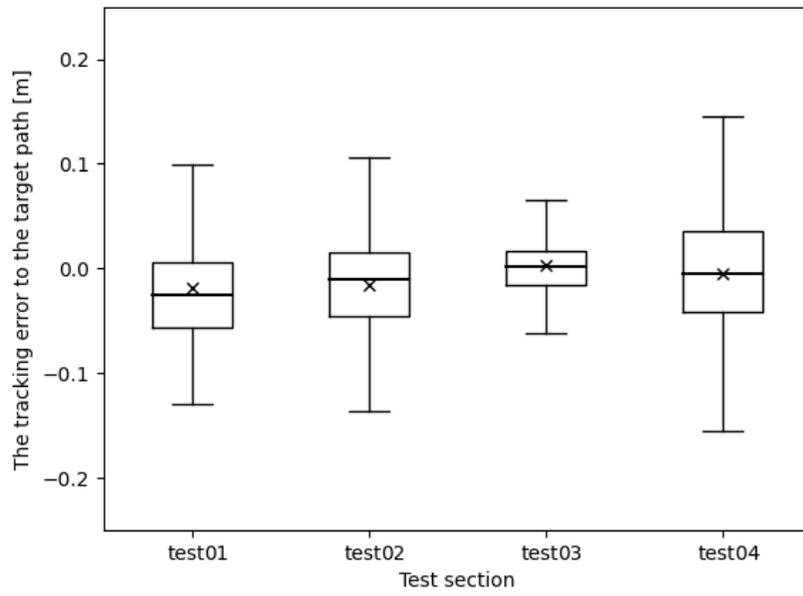


(c) test03

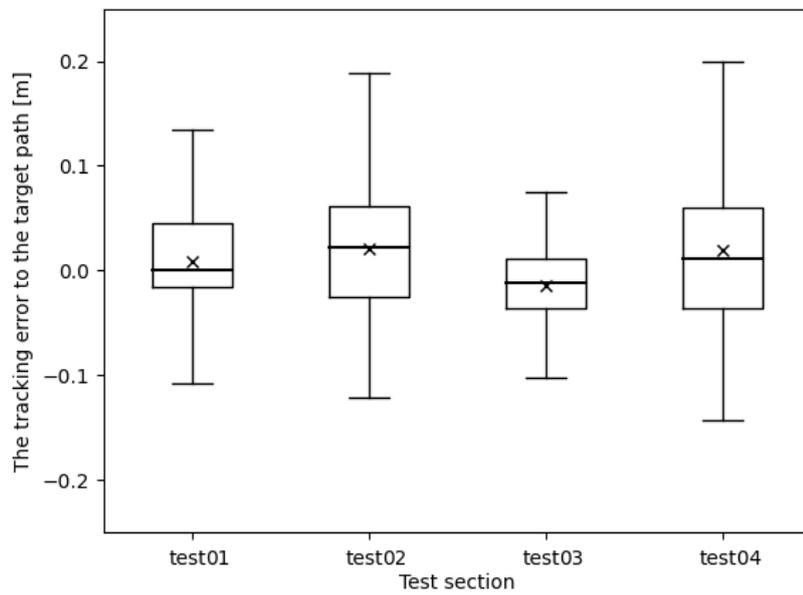


(d) test04

**Fig. 5.2:** Surrounding environment at each test section in the forest work road A.



(a) upward slope



(b) downward slope

**Fig. 5.3:** Position control error in the transverse direction of forest work road A.

## 5.5 森林作業道 B における実験

### 5.5.1 実験環境

森林作業道 B における自律移動実験は、2020 年 1 月 30 日に実施した。地図作成機能により取得した目標経路を Fig. 5.4 に示す。全走行距離は 107 m であり、標高は、土場(出発地)が 336.0 m、先山(目的地)が 345.2 m であった。各区間における周辺の様子を Fig. 5.5 に示す。斜面の勾配は緩やかであり、経路は主に直線となっている。すべての区間において、路面は枯れ葉、枝、砂利等により覆われている状態であり、車輪の空転が容易に発生する。周辺の植生は、左右に樹木が植えられており、斜面は雑草に覆われている。

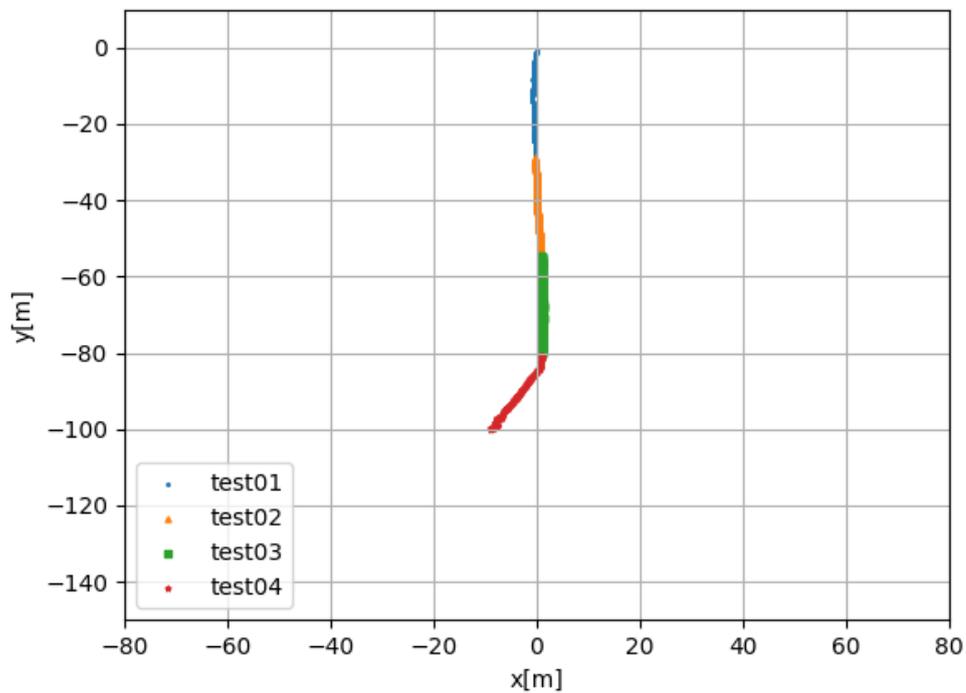


Fig. 5.4: Target path on the forest work road B.



(a) test01



(b) test02



(c) test03



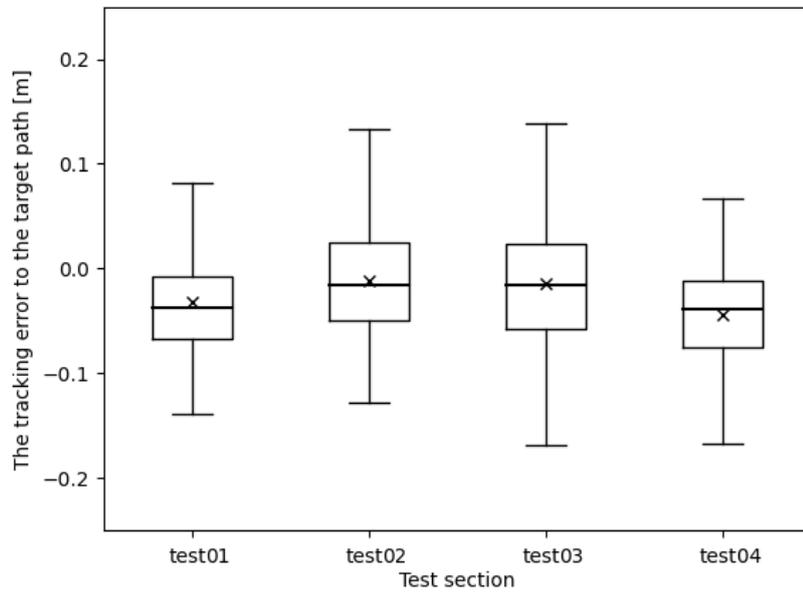
(d) test04

**Fig. 5.5:** Surrounding environment at each test section in the forest work road B.

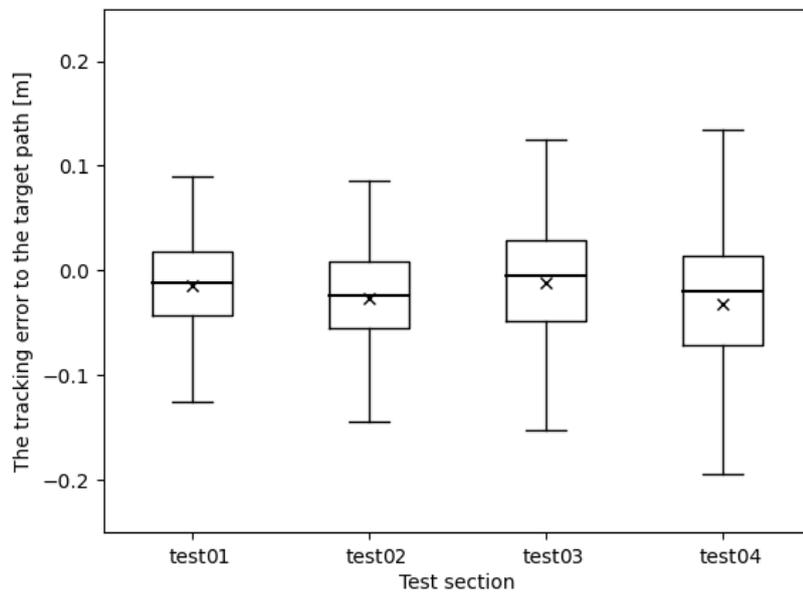
## 5.5.2 実験結果

森林作業道 B における実験結果を Fig. 5.6 に示す．すべての走行経路において横断方向の位置制御誤差が，当初の目標である 0.25 m 以内となり，森林作業道 B において，目標とする精度での自律移動が可能であることを確認した．

区間ごとに，横断方向の位置制御の最大誤差に注目すると，登り経路の場合，区間 test03 において，横断方向の位置制御誤差の最大値が，他の区間と比較して増加した．降り経路の場合，区間 test04 において最大誤差が，他の区間と比較して増加した．



(a) upward slope



(b) downward slope

**Fig. 5.6:** Position control error in the transverse direction of forest work road B.

## 5.6 考察

森林作業道 A において、登り経路、降り経路にかかわらず、区間 test04 における、車両の右側、および左側の最大誤差が、他の区間より増加した。区間 test04 の目標経路を確認したところ、目標経路が、他の区間と比較して蛇行していた。目標経路に対する位置制御が困難となっていたと考えられる。

森林作業道 A の降り経路 test02, test04 の区間において、車両右方向の最大誤差が、他の区間と比較して増加した。原因を調査したところ、森林作業道 A の路面は谷側に傾いており、さらに test02, および test04 区間は左側にカーブする環境であった。カーブを走行する際、重心が谷側に傾いたため、目標経路に対する追従が困難であったと考えられる。

森林作業道 B の登り経路、区間 test03 において、横断方向の位置制御誤差が、車両の右方向、および左方向ともに、他の区間と比較して増加した。test03 は周辺の物体を笹、ススキ、および雑草が占めているため、他の区間と比較して、自己位置推定精度が低下したことが、原因と考えられる。

森林作業道 B の降り経路では、test04 の区間において、誤差の最大値が、左右ともに他の区間より増加した。test04 区間は轍が他の区間と比較して深くなっており、位置制御において、目標経路に対する復帰の妨げとなっていることが考えられる。

本実験結果により、すべての経路において、目標経路に対する横断方向の位置制御誤差が 0.25 m 以内であり、当初の目標精度を達成した。その一方で、位置制御誤差が増加する原因は、路面・目標経路の形状による位置制御精度の低下、および周辺形状に起因した自己位置推定精度の低下が挙げられた。今後、さらに自己位置推定精度の向上を図る。さらに、路面・目標経路の形状、および自己位置推定誤差を考慮した、位置制御手法を開発することで、森林作業道における、位置制御の高精度化を検討する。

# Chapter 6

## おわりに

### 6.1 本研究の目的

本研究の目的は、フォワーダの自動化を行うため、森林作業道における自律移動手法を構築することである。本研究では、森林作業道の自律移動に必要な要素技術として、スキャンマッチング手法、および路面推定手法を構築した。さらに、スキャンマッチング手法、および路面推定手法を用いることで、自律移動に必要な、地図作成機能、地図中の自己位置推定機能、および経路計画機能を構築した。

### 6.2 目的を達成するための方法

#### 6.2.1 スキャンマッチング手法の提案

2章において、森林作業道における自律移動に使用するための、スキャンマッチング手法を提案した。提案したスキャンマッチング手法は、位置、姿勢、並進速度の推定による点群の歪み補正、およびボクセルを用いた形状近似による、点群の重ね合わせを行う。RTK-GNSSによる測位が可能な屋外不整地、および森林作業道において、自己位置推定結果を取得した。従来手法と比較して、提案手法の自己位置推定精度の向上を確認した。

#### 6.2.2 路面推定手法の提案

3章において、LiDAR-SLAMにより得られた点群を入力として、ディープラーニングによる学習結果を用いた路面推定手法を構築した。森林作業道において点群を取得し、データセットを作成した。得られたデータセットを用いて、ネットワークの学習、および検証を行い、提案手法の路面推定精度を確認した。

## 6.3 森林作業道における自律移動の実験とその結果

4章において、自律移動システムの構築方法について述べた。自律移動システムは、地図作成機能、地図中の自己位置推定機能、および経路計画機能により構成され、地図作成機能、地図中の自己位置推定機能は、2章において構築したスキャンマッチング手法を用いた。さらに、経路計画機能は、3章において構築した路面推定手法を用いた。

5章において、構築した自律移動システムを用いて、森林作業道における自律移動を行った結果を述べた。森林作業道における自律移動の可否は、目標経路を自律移動する際、経路の横断方向の位置制御精度を算出することで判断した。自律移動を行ったすべての経路において、横断方向の位置制御誤差が、最低限のクリアランスである 0.25 m 以下となることを確認し、目標精度を達成した。

## 6.4 本研究の成果と今後の課題

本研究の成果は、スキャンマッチング手法を構築することにより、森林作業道において地図作成、および自己位置推定を可能としたことである。さらに、森林作業道における、LiDAR-SLAM を用いた路面推定手法の構築と、経路計画への応用例を示したことである。最後に、これら提案手法により、森林作業道の自律移動が実現できることを示した。

今後は、自律移動システムを電子制御が可能なフォワードに搭載し、多様な森林作業道の形状や目標経路に対して、自律移動の可否を検証する。さらに、路面の勾配や起伏、目標経路の形状、自己位置推定精度の変化に対応した、位置制御手法を開発する。

# 謝辞

本研究は，博士(工学)小谷 信司 山梨大学大学院総合研究部 教授，博士(医工学)渡辺 寛望 山梨大学大学院総合研究部 准教授のご指導の下で行われたものである．両先生には，研究を進めるに際して，ご多忙にもかかわらず，時間を惜しまずに熱心に多大なご指導をいただいた．本論文をまとめるにあたり，博士(工学)鈴木 良弥 山梨大学大学院総合研究部 教授，博士(工学)島 弘幸 山梨大学大学院総合研究部 教授，博士(工学)丹沢 勉 山梨大学大学院総合研究部 准教授，博士(工学)清水 毅 山梨大学大学院総合研究部 准教授，博士(医工学)石田 和義 山梨大学大学院総合研究部 准教授には，適切な助言をいただいた．

本研究を進める上で，博士(工学)神村 明哉 国立研究開発法人産業技術総合研究所 情報・人間工学領域 研究チーム長，博士(工学)横塚将志 国立研究開発法人産業技術総合研究所 情報・人間工学領域 主任研究員よりいただいた助言は非常に有益であった．本研究は，古屋製材株式会社 古屋 清人 取締役専務，古屋 五嗣 取締役，小幡 俊郎 技術員より，研究を実施するきっかけをいただき，実験環境の提供，および実験の補助等の多大なご支援をいただいた．

本研究を遂行するにあたり，山梨県産業技術センター 甲府技術支援センター 電子・システム技術部 河野 裕 部長，および博士(情報科学)布施 嘉裕 電子・システム技術部 システム開発科 主任研究員には，博士論文の執筆に関して深いご理解を頂いた．

最後に，妻 美香は，いつも明るい笑顔で心の支えになってくれ，苦しい時や気持ちが落ち込んだときに，何度も立ち直らせてくれました．母，今は亡き父に深く感謝致します．

上記の皆様をはじめ，ここに書ききれないこれまでお世話になったすべての方々に対して，この場を借りて深い感謝の意を表します．

## 参考文献

- [1] 林野庁. 森林作業道作設指針. 2010. オンライン. <https://www.rinya.maff.go.jp/j/seibi/saisei/pdf/sinrinsagyoudou.pdf>. 2021年11月17日参照.
- [2] 全国林業改良普及協会. 森林総合監理士 (フォレスター) 基本テキスト. 全国林業改良普及協会, 2017.
- [3] 日本ロボット学会. 新版 ロボット工学ハンドブック. 2005.
- [4] 今富 裕樹, 上村 巧, and 加藤 隆. フォワーダ作業の災害要因と災害発生の類型化. 森林利用学会誌, 26(1):21–26, 2011.
- [5] 毛綱 昌弘 and 山口 浩和. 永久磁石とジャイロの組み合わせによる自律走行運材車の開発. 森林利用学会誌, 15(3):197–204, 2000.
- [6] 毛綱 昌弘 and 山口 浩和. 自律走行運材車の走行経路入力方法の開発. 森林利用学会誌, 20(1):3–8, 2005.
- [7] 毛綱 昌弘, 山口 浩和, 鈴木 秀典, 山口 智, 宗岡 寛子, 佐々木 達也, 有水 賢吾, 飯澤 宇雄, 大東 史典, 阿部 慶一, 小長井 信宏, and 辻 浩志. 電磁誘導式自動走行フォワーダによる集材作業の無人化に関する研究. 森林総合研究所研究報告, 20(1):19–28, 2021.
- [8] 経済産業省. 平成 26 年度分中間年報バイオマスエネルギー技術研究開発戦略的次世代バイオマスエネルギー利用技術開発事業 (実用化技術開発) 多形状バイオマスガス化発電・未利用間伐材収集効率化の研究開発. 2014.
- [9] Andreas Nüchter. *3D robotic mapping: the simultaneous localization and mapping problem with six degrees of freedom*, volume 52. Springer, 2008.
- [10] 友納正裕. *SLAM 入門ロボットの自己位置推定と地図構築の技術*. 株式会社 オーム社, 2018.
- [11] Shinpei Kato, Shota Tokunaga, Yuya Maruyama, Seiya Maeda, Manato Hirabayashi, Yuki Kitsukawa, Abraham Monrroy, Tomohito Ando, Yusuke Fujii, and Takuya Azumi. Autoware on board: Enabling autonomous vehicles with embedded systems. In *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, pages 287–296, 2018.
- [12] 竹内 栄二郎. ロボット用ミドルウェアを利用した自律移動ロボット開発. 計測と制御, 57(10):741–744, 2018.

- [13] 毛綱 昌弘, 山口 浩和, 伊藤 崇之, 鈴木 秀典, 干坂 修, 高崎綾信, 草野 兼光, and 北原 成郎. 遠隔操作機能と自動走行機能によるフォワードの操作支援技術の開発. 森林総合研究所研究報告, 15 卷 (4 号):91–102, 12 2016.
- [14] M. Bosse and R. Zlot. Continuous 3d scan-matching with a spinning 2d laser. In *2009 IEEE International Conference on Robotics and Automation*, pages 4312–4319. IEEE, 2009.
- [15] Michael Bosse, Robert Zlot, and Paul Flick. Zebedee: Design of a spring-mounted 3-d range sensor with application to mobile mapping. *IEEE Transactions on Robotics*, 28(5):1104–1119, 2012.
- [16] J. Zhang and S. Singh. Loam: Lidar odometry and mapping in real-time. In *Robotics: Science and Systems*, volume 2, 2014.
- [17] Ji Zhang and Sanjiv Singh. Low-drift and real-time lidar odometry and mapping. *Autonomous Robots*, 41(2):401 – 416, February 2017.
- [18] P.J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.
- [19] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. In *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pages 2724–2729 vol.3, 1991.
- [20] P. Biber and W. Strasser. The normal distributions transform: a new approach to laser scan matching. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, volume 3, pages 2743–2748 vol.3, 2003.
- [21] Martin Magnusson, Achim Lilienthal, and Tom Duckett. Scan registration for autonomous mining vehicles using 3d-ndt. *Journal of Field Robotics*, 24(10):803–827, 2007.
- [22] Eijiro Takeuchi and Takashi Tsubouchi. A 3-d scan matching using improved 3-d normal distributions transform for mobile robotic mapping. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3068–3073, 2006.
- [23] Mark Schadler, Jörg Stückler, and Sven Behnke. Rough terrain 3d mapping and navigation using a continuously rotating 2d laser scanner. *KI-Künstliche Intelligenz*, 28(2):93–99, 2014.
- [24] David Droschel, Max Schwarz, and Sven Behnke. Continuous mapping and localization for autonomous navigation in rough terrain using a 3d laser scanner. *Robotics and Autonomous Systems*, 88:104–115, 2017.
- [25] Aleksandr V. Segal, Dirk Haehnel, and Sebastian Thrun. Generalized-icp.

- [26] Tixiao Shan and Brendan Englot. Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4758–4765, 2018.
- [27] 岐阜県森林研究所. 効率的な機械作業のための作業道規格. <https://www.forest.rd.pref.gifu.lg.jp/rd/kankyoku/mori081101.html> 2021.11.30 アクセス.
- [28] 白鳥 亮介, 森 勇佑, 有賀 一広, and 田坂 聡明. 自走運材車両のための走行経路抽出手法に関する研究. *森林利用学会誌*, 23(4):233–237, 2009.
- [29] 有水 賢吾 and 毛綱 昌弘. Tof カメラを用いた自動走行フォワードのための作業道検知. *森林利用学会誌*, 33(4), 2018.
- [30] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
- [31] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.
- [32] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.
- [33] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, April 2018.
- [34] Abhinav Valada, Rohit Mohan, and Wolfram Burgard. Self-supervised model adaptation for multimodal semantic segmentation. *International Journal of Computer Vision (IJCV)*, jul 2019.
- [35] 有水賢吾 and 毛綱昌弘. 深層畳み込みニューラルネットワークを用いたセマンティックセグメンテーションによる森林作業道抽出. *森林利用学会誌*, 35(1):35–7, 2020.
- [36] L. Caltagirone, S. Scheidegger, L. Svensson, and M. Wahde. Fast lidar-based road detection using fully convolutional neural networks. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1019–1024, June 2017.
- [37] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015.
- [38] Dieter Fox, Wolfram Burgard, and Sebastian Thrun. The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1):23–33, 1997.

- [39] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, January 2015.
- [40] 中込 広幸, 布施 嘉裕, 永田 靖貴, 宮本 博永, 横塚 将志, 神村 明哉, 渡辺 寛望, 丹沢 勉, and 小谷 信司. 森林作業道における並進速度推定を用いた高精度な3次元スキャンマッチング. *電気学会論文誌C (電子・情報・システム部門誌)*, 140(9):1082–1090, 2020.
- [41] 岡 勝, 中澤 昌彦, 田中 良明, 吉田 智佳史, 近藤 耕次, and 加利屋 義広. フォワーダの走行速度に及ぼす路網規格と積載量の影響. *森林利用学会誌*, 22(3):163–170, 2007.
- [42] 岡 勝, 田中 良明, 吉田 智佳史, 近藤 耕次, 佐々木 達也, and 加利屋 義広. フォワーダ走行速度からみた集材路規格に関する検討. *森林利用学会誌*, 21(4):295–298, 2007.
- [43] 熊澤 ゆかり, 藤田 亮, 山崎 篤史, 小山 敢, 市原 恒一, and 岡勝. 安全かつ効率的なフォワーダ集材についての考察. *森林利用学会誌*, 26(3):181–186, 2011.

# 研究業績

## 発表論文

### 1. 学術雑誌

- (a) ( 学術雑誌 (b) の英訳元となる論文誌掲載論文) 中込 広幸, 布施 嘉裕, 保坂 秀彦, 宮本 博永, 中村 卓, 米山 陽, 横塚 将志, 神村 明哉, 渡辺 寛望, 丹沢 勉, 小谷 信司: ”不整地面走行時の振動を考慮した 3D スキャンマッチング”, 電気学会論文誌 C (電子・情報・システム部門誌), Vol.139, No.9, pp.1041-1050 (2019.9), 本文第 2 章に対応
- (b) Hiroyuki Nakagomi, Yoshihiro Fuse, Hidehiko Hosaka, Hironaga Miyamoto, Takashi Nakamura, Akira Yoneyama, Masashi Yokotsuka, Akiya Kamimura, Hiromi Watanabe, Tsutomu Tanzawa and Shinji Kotani: ”3-D Scan Matching for Mobile Robot Localization over Rough Terrain”, Electrical Engineering in Japan, Vol. 209, No.3-4, pp.14-25 (2019.12), 本文第 2 章に対応
- (c) 中込 広幸, 布施 嘉裕, 永田 靖貴, 宮本 博永, 横塚 将志, 神村 明哉, 渡辺 寛望, 丹沢 勉, 小谷 信司: ”森林作業道における並進速度推定を用いた高精度な 3 次元スキャンマッチング”, 電気学会論文誌 C (電子・情報・システム部門誌), Vol.140, No.9, pp.1082-1090 (2020.9), 本文第 2 章に対応

### 2. 査読付き国際学会

- (a) Hiroyuki Nakagomi, Yoshihiro Fuse, Yasuki Nagata, Hidehiko Hosaka, Hironaga Miyamoto, Masashi Yokozuka, Akiya Kamimura, Hiromi Watanabe, Tsutomu Tanzawa and Shinji Kotani: ”Forest Road Surface Detection Using LiDAR-SLAM and U-Net”, Proceedings of the 2021 IEEE/SICE International Symposium on System Integration(SII 2021), pp.727-732 (2021.1), 本文第 3 章に対応

## 国内の学術講演会における口頭発表

- 1. 中込 広幸, 布施 嘉裕, 保坂 秀彦, 宮本 博永, 中村卓, 米山陽, 横塚 将志, 神村 明哉, 渡辺 寛望, 丹沢 勉, 小谷 信司: ”不整地面走行時の振動を考慮した 3D スキャンマッチング”, 第 22 回知能メカトロニクスワークショップ IMEC2017 講演論文集 (IMEC2017), 2A3-1 (2017.8)

2. 中込 広幸, 布施 嘉裕, 永田 靖貴, 宮本 博永, 横塚 将志, 神村 明哉, 渡辺 寛望, 丹沢 勉, 小谷 信司: ”森林作業道における並進速度推定を用いた高精度な 3 次元スキャンマッチング”, 第 24 回知能メカトロニクスワークショップ IMEC2019 講演論文集 (IMEC2019), IMEC-202 (2019.8)
3. 中込 広幸, 布施 嘉裕, 保坂 秀彦, 永田 靖貴, 宮本 博永, 横塚 将志, 神村 明哉, 渡辺 寛望, 丹沢 勉, 小谷 信司: ”森林作業道における U-NET を用いた路面の検出”, 動的画像処理実利用化ワークショップ 2020 講演論文集 (DIA2020), pp.227-230 (2020.3)

## その他 (学位論文と関係のないもの)

### 1. 学術雑誌

- (a) 中込 広幸, 星野 聖: ”結膜血管端検出による高速高精度眼球回旋計測”, 電子情報通信学会論文誌. D, 情報・システム, Vol.96, No.4, pp.876-884 (2013.4)
- (b) 古屋 雅章, 中込 広幸, 布施 嘉裕, 中村 聖名, 山田 博之, 阿部 治, 渡辺 寛望, 小谷 信司: ”対数振幅スペクトル解析による画質評価を用いた自動外観検査用照明の自動調整法”, 精密工学会誌, Vol.80, No.12, pp.1150-1156 (2014.12)

### 2. 査読付き国際学会

- (a) Hiroyuki Nakagomi and Kiyoshi Hoshino: ”Measurement of Rotational Eye Movement Using Conjunctiva Blood Vessel Tracking”, 2012 IEEE/SICE INTERNATIONAL SYMPOSIUM ON SYSTEM INTEGRATION (SII), MP1-B(5), pp.758-763 (2012)
- (b) Hiroyuki Nakagomi and Hoshino Kiyoshi: ”Measurement of Torsional Eye Movement Using Terminal Points of Conjunctival Blood Vessels”, MVA2013 IAPR Intl. Conf. on Machine Vision Applications, Vol.4, No.17, pp.101-104 (2013.7)
- (c) Kiyoshi Hoshino and Nakagomi Hiroyuki: ”Measurement of Rotational Eye Movement under Blue Light Irradiation by Tracking Conjunctival Blood Vessel Ends”, 2013 IEEE/SICE International Symposium on System Integration (SII2013), SP1-K(3), pp.204-209 (2013.12)
- (d) Kiyoshi Hoshino and Hiroyuki Nakagomi: ”High-accuracy measurement of rotational eye movement by tracking of blood vessel images”, 2014 36TH ANNUAL INTERNATIONAL CONFERENCE OF THE IEEE ENGINEERING IN MEDICINE AND BIOLOGY SOCIETY (EMBC), pp.6339-6344 (2014)

### 3. 国内の学術講演会における口頭発表

- (a) 中込 広幸, 布施 嘉裕, 宮本 博永, 油井 誠志, 中村卓, 米山陽, 横塚 将志, 神村 明哉, 渡辺 寛望, 丹沢 勉, 小谷 信司: ”サーマルカメラを用いた人間・ロボット共存環境におけるロバストなビジュアルオドメトリ法”, 第 33 回日本ロボット学会学術講演会 RSJ2015 講演論文集 (RSJ2015), 1K2-03 (2015.9)

# 著者略歴

## 学歴

1. 平成 23 年 3 月 31 日 筑波大学 工学システム学類 卒業
2. 平成 23 年 4 月 1 日 筑波大学大学院 システム情報工学研究科 知能機能システム専攻 入学
3. 平成 25 年 3 月 31 日 同上修了
4. 平成 31 年 4 月 1 日 山梨大学大学院医工農学総合教育部博士課程工学専攻 入学
5. 令和 4 年 3 月 31 日 同上修了見込み

## 職歴

1. 平成 25 年 4 月 1 日 山梨県入庁 山梨県工業技術センター配属
2. 平成 29 年 4 月 1 日 山梨県産業技術センター配置換え 現在に至る

## 所属学会

1. 日本ロボット学会会員
2. 精密工学会会員