

# 車載 LiDAR を用いた既知の経路における障害物検知

鈴木雄弥<sup>1,2</sup> 加藤晋<sup>1</sup> 伊丹誠<sup>2</sup>

**概要:** 自動運転車両があらかじめ決められた経路を走行する場合、システム構築時に走行経路情報を得ることができる。そこで、車両の停止位置などにおける障害物検知手法を提案し、障害物検知システムの開発および評価を行う。

**キーワード:** 点群処理, LiDAR, 環境認識, ITS, 自動運転

## Obstacle detection on known routes using in-vehicle LiDAR

YUYA SUZUKI<sup>†1 †2</sup> SHIN KATO<sup>†1</sup>  
MAKOTO ITAMI<sup>†2</sup>

**Abstract:** When an autonomous vehicle travels only on a predetermined route, the track information can be known at the time of constructing the system. Therefore, we propose and evaluate a system that simplifies obstacle detection when the vehicle is stopped.

**Keywords:** Point Cloud Processing, LiDAR, Environment Detection, ITS, Autonomous

### 1. はじめに

#### 1.1 研究背景

現在我が国では、少子高齢化や都市部への人口集中等の社会構造の変化から、道路交通を巡る様々な社会的な課題の深刻化が懸念されている[1]。特に、地方部における人口減少・過疎化が進むことによる地方交通事業者や地方公共団体の財政逼迫、運転者不足により地域公共交通ネットワークの維持が一層困難となり、交通空白地帯が拡大する恐れがある[1]。

そこで、産業技術総合研究所（以下、産総研）では、経済産業省および国土交通省の平成 30 年度「高度な自動走行システムの社会実装に向けた研究開発・実証事業：専用空間における自動走行などを活用した端末交通システムの社会実装に向けた実証」を幹事機関として受託し、端末交通システムの研究開発と実証を進めている[2]。端末交通システムとは、鉄道などの基幹交通システムと自宅や目的地との間や、地域内といった短中距離の移動を補完するラストワンマイルモビリティとも呼ばれる交通システムのことである[2]。

このような端末交通システムの実現にあたり、自動運転車両の周辺環境の認識技術を行うシステムの構築は必要不可欠である。現在実証実験で運用されている端末交通システムにおける LiDAR から取得した点群の処理は、走行時に前方の決まった範囲に障害物がないか監視を行っている。しかし、実際の道路では車両前方の障害物検知だけでは不

十分な場合がある。例えば、車両が交差点等で停止した際に、走行環境によっては、車両から見て前方のみを見るだけではなく、道路の形状に合わせて柔軟に障害物を検知する必要がある。

そこで、拡張が容易で高速な障害物検知手法を提案し、それを基に障害物検知システムを構築した。そして、端末交通システムが運用されている道路で収集した点群データを用いて、構築した障害物検知システムの動作および評価を行った。

### 2. 想定車両および環境

#### 2.1 端末交通システムと実験車両

本研究で想定している自動運転車両は、前述した実証事業で用いているヤマハ発動機のゴルフカーをベースに開発された図 1 に示す実験車両とした。



図 1 実験車両

Figure 1 Experimental vehicle

<sup>1</sup> 産業技術総合研究所  
National Institute of Advanced Industrial Science and Technology  
<sup>2</sup> 東京理科大学  
Tokyo University of Science

実験車両には、車両制御用 PC ・カメラ ・LiDAR (Light Detection and Ranging) ・GPS ・RFID リーダなど多数センサが搭載されており、環境認識などのデータ処理のための PC も車両制御用とは別に搭載されている。本研究で用いる LiDAR は、図 1 に示した位置に固定されて動作しており、車両制御用 PC やデータ処理用 PC と LAN 接続されている。

端末交通システムにおいて実験車両は、あらかじめ定められた経路に埋め込まれた電磁誘導線に沿って自動操舵をしながら既知の経路上を低速（最高 20[km/h]未満）で走行する。また、走行経路における急カーブや交差点や停止線などの要所には、車両が制御信号を受け取るための RFID (Radio frequency identification) タグが配置されている。

RFID タグによる車両への制御信号とは具体的に、急カーブ前の RFID タグは「減速」、急カーブ後の RFID タグは「加速」、交差点や停止線の前の RFID タグは「停止」といったものである。本研究内で用いる RFID タグの制御信号は「停止」信号を持つものとし、以降は簡単のため、「停止」信号を持つ RFID タグのことを単に RFID と呼ぶ。

本研究では RFID での停止時の障害物検知手法を提案し、提案手法を基に構築した障害物検知システムを構築して動作および評価を行う。

## 2.2 LiDAR から得られるデータについて

LiDAR とは、光を用いたリモートセンシング技術およびその技術を用いたセンサのことであり、センサ内部から放射したレーザ光が物体に当たった際の散乱光を受信し、その到達時間と光速度から、周囲の物体が存在するまでの距離がわかるものである。そして、レーザ光の放射角と合わせて 3次元の点の集合が得られ、それらの点の集合は点群と呼ばれる。本研究では、点群を直交座標系で扱うものとする。

## 2.3 実験車両における点群処理

現在に実証実験で運用されている端末交通システムにおける実験車両での点群処理は、図 1 に示した位置に搭載された LiDAR から取得した点群を 3次元座標のまま処理するのではなく、実験車両を中心とした 2次元平面上のグリッドに割り当てたものを扱っている。そこでは、前方の衝突を回避する事に焦点を当て、グリッドのうち車両前方の、特定の範囲に障害物が存在するかを常に監視しながら走行している。ここで、特定の範囲とは、実験車両の速度や舵角から算出されたおおよその進行経路である。

それに対して本研究では、点群をグリッドに割り当てずに計算に用い、速度や舵角ではなく、実験車両の RFID における停止時に地形に合わせた環境認識を行う。

## 3. 提案手法

### 3.1 前提

本論文内で扱う直交座標系の軸について、実験車両の運転席から見て、X 軸正は右方、Y 軸正は前方、Z 軸正は上方とした。

本研究では、車両が RFID で停止する位置およびその先の走行経路は既知であり、事前に把握することができるものとした。

### 3.2 事前準備

各 RFID による停止位置周辺のうち、障害物検知の対象としたい探索領域をあらかじめ確認しておき、ある点を中心とした球内を障害物検知の対象とするような探索点をデータとして保存する。ここで、探索点の例として、十字路における左折の走行経路を探索領域としたい場合は図 2 のような探索点を準備する。



図 2 探索点の例（走行経路のみ）

Figure 2 Example of search point (only waypoints)

図 2 において、水色の領域が探索領域で、左折時の走行経路を障害物検知の対象としたい場合は、黒色の点のように探索点を事前に保存しておく。

また、探索点は走行経路に必ずしも沿っていなくてもよく、道路への合流時のような、走行経路以外にも障害物を検知する必要がある場合は、図 3 のように探索点を準備する。



図 3 探索点の例（走行経路外を含む）

Figure 3 Example of search points (including outside the route)

図 3 では、図 2 で示した走行経路に加えて、左折時に物体の存在を検知しておくべき領域を探索点に含んでいる。この例では、水色の領域は走行経路を、黄色の領域は右方からの直進車両を探索領域とする。

### 3.3 前処理

本研究で扱う LiDAR は、実験車両の走行中は常に図 1 に示した位置に固定されているため、地面からの高さや傾きは常に一定である。また、走行経路内には急な傾斜が存在しない。それらを踏まえて、処理の簡略化や高速化のため、LiDAR から取得した点群のうち、地面を表す点はあらかじめ除去する。そこでは、各点が地面である条件は、Z 座標が LiDAR の地面からの距離より小さいものとした。

また、車両制御用 PC の起動時にデータ処理用 PC も起動し、各 RFID に対する探索点を読み込んで、車両制御用 PC から受信する障害物検知要求を待機するものとする。

### 3.4 障害物検知の流れ

データ処理 PC 内部で行われる、探索点を用いた障害物検知の流れを図 4 に示す。

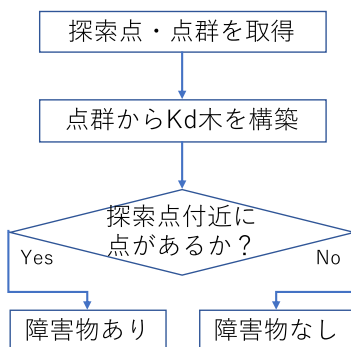


図 4 探索点を用いた障害物検知の流れ

Figure 4 Process flow of obstacle detection with search points

障害物検知としては、探索点および点群を取得した後、点群から Kd 木を構築し、探索点の一定距離内に点が存在するかを探索することで障害物の有無を確認した。ここで、Kd 木とは、任意の次元の点の集合を扱う木構造の事で、本研究では PCL (Point Cloud Library) 上で実装されている pcl\_kdtree ライブラリを用いた。

### 3.5 車両内での処理の流れ

走行している実験車両が RFID で停止してから走行を再開するまでの車両制御用 PC およびデータ処理用 PC の全体の流れを図 5 に示す。

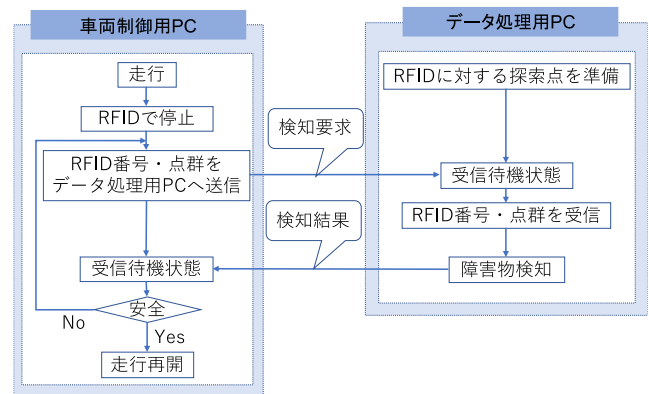


図 5 車両全体の処理の流れ

Figure 5 Process flow of the vehicle

ここで、図 5 の左部は車両制御用 PC 内部の処理を、右部はデータ処理用 PC 内部の処理を示している。まず、車両走行中はデータ処理用 PC をあらかじめ障害物検知のための探索点を読み込んだ状態で、検知要求待機状態しておく。

そして、走行している車両が RFID からの信号を受信して停止した時、データ処理用 PC に対して障害物検知要求を行う。障害物検知要求を受けたデータ処理用 PC は、図 4 に示した障害物検知を行い、その結果を車両制御用 PC に送信する。

### 3.6 障害物検知要求で送信するデータ

障害物検知要求で車両制御用 PC が送信するデータは、データ処理用 PC で必要なデータである、RFID 番号および点群とし、それらを文字列にしたものとした。

## 4. 実験環境

### 4.1 実験装置

LiDAR は Velodyne 社製の VLP-16 を用いた。VLP-16 は、垂直方向 16 本のレーザを水平方向に 360 度回転させ、±3cm の精度で距離 100m まで測定可能なセンサである。

データ処理用 PC は、エッジデバイスである NVIDIA 製の Jetson AGX Xavier を用いた。

### 4.2 単純な曲がり角

第一に環境認識の評価を行う車両の停止位置は、沖縄県北谷町の海沿いの道路である図 6 に示す単純な曲がり角とした。



図 6 曲がり角

Figure 6 Corner of the road

この曲がり角において実験車両は、曲がり角に差し掛かる前に RFID による信号を受信し、図 7 の位置で停止する。



図 7 曲がり角における停止位置

Figure 7 Stop position at the corner

ここで実験車両は、図 7 の位置で停止した後、周囲の安全確認した後、発進して左折する。

#### 4.3 複雑な三叉路

第二に環境認識の評価を行う車両の停止位置は、沖縄県北谷町の公道である図 8 に示す三叉路とした。



図 8 三叉路

Figure 8 Three-way junction

この三叉路には停止線があり、それに合わせて停止する様に RFID が設置されている。そこでの実験車両の停止位置を図 9 に示す。



図 9 三叉路における停止位置

Figure 9 Stop position at the three-way junction

ここで実験車両は、図 9 の位置で停止して周囲の安全確認をした後、発進して道路に合流する。その時、合流後の道路は直線ではなく、若干湾曲した道路となっている。

## 5. 障害物検知の評価

提案手法を基に開発した障害物検知システムを構築し、評価対象とした各停止位置に対する障害物検知の評価を行う。探索点については、各道路について探索が必要な領域に対し、地面に沿った点を準備した。

### 5.1 単純な曲がり角

障害物検知システムの、図 6 に示した曲がり角における、図 7 の停止位置での動作を評価する。ここでは、探索点は走行経路のみとした。この経路においては、走行経路において障害物がなければ進行することができるので、障害物の有無をそのまま安全か危険かの判断とした。結果の表示方法は、曲がり角における停止位置で LiDAR から取得した点群を俯瞰図で表示した。まずは、走行経路内に人や車両などの障害物が存在しない場合の障害物検知の結果を図 10 に示す。

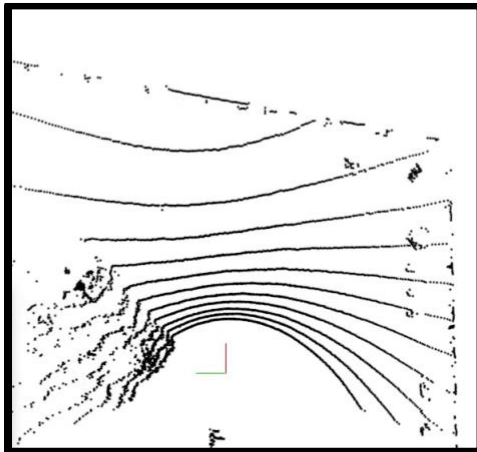


図 10 曲がり角で障害物が存在しない場合  
Figure 10 There are no obstacles at the corner

結果として、障害物検知システムでは、探索領域内に障害物が存在しないと判断された。次に、走行経路内に人が存在する場合の結果を図 11 に示す。

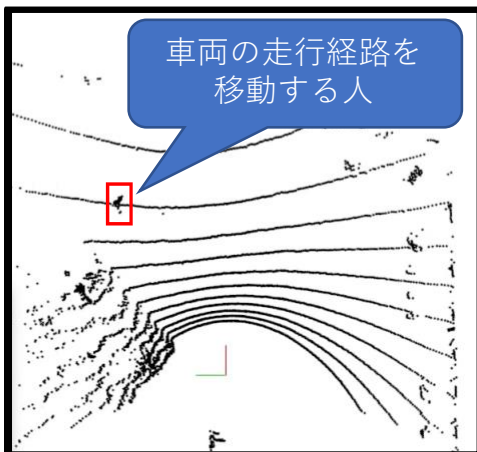


図 11 経路内に障害物が存在する場合  
Figure 11 There are obstacles at the corner

結果として障害物検知システムでは、探索領域内に障害物の存在が検知された。

## 5.2 複雑な三叉路

障害物検知システムの、図 8 に示した三叉路における、図 9 の停止位置での動作を評価する。ここでは、道路への合流を判断するためには右方からの車両の接近を検知する必要があるため、走行経路だけでなく右方の道路も探索点とした。また、道路の形状が湾曲しているため、探索点にも道路の形状に合わせて湾曲させた。ここで、結果の表示方法は、道路や探索点が複雑になっているため、LiDAR から取得した点群から地面を削除した点を黒点、探索点を青点、探索領域に存在するとされた障害物を赤点とし、車両周辺の俯瞰図として表示した。まずは、探索領域内に車両

や歩行者が存在しない場合の障害物検知の結果を図 12 に示す。

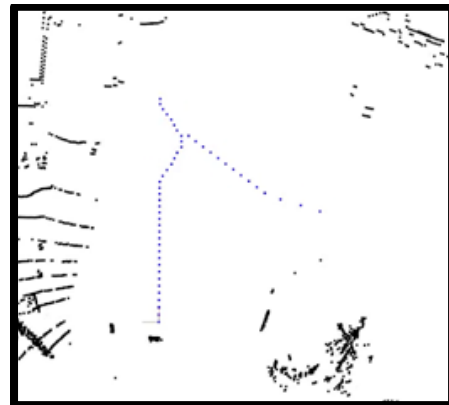


図 12 三叉路で障害物がない場合  
Figure 12 There are no obstacles at the three-way junction

ここで、赤く表示されている点は存在せず、障害物が存在しないと判断された。次に、探索領域内に車両が走行している場合の障害物検知の結果を図 13 に示す。

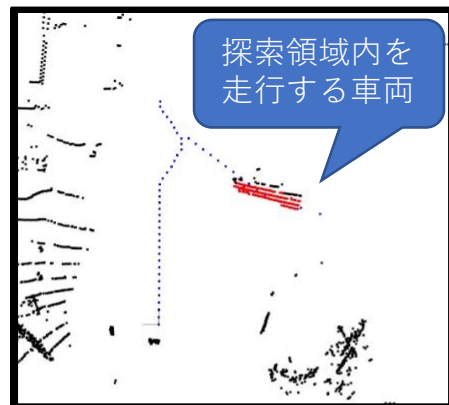


図 13 三叉路で障害物がある場合  
Figure 13 There are obstacles at the three-way junction

結果としては、探索領域内を走行している一般車両の、探索点周辺の点が赤く表示されており、障害物として検知されていた。この時、走行車両の上部は障害物としては検知されなかったのは、事前に準備した探索点が地面に沿っており、車両上部が探索点から一定距離内に存在しなかったため、これを避けるためには探索点を増やすなどの対策方法が考えられる。

## 6. 総括

既知の経路における障害物検知システムの開発・評価を行なった。端末交通システムのように、予め走行経路を知ることができる状況においては、システム構築時に探索点

を用意することで、車両の走行経路外の障害物が検知でき、複雑な形状の道路であっても柔軟に障害物検知を行えることがわかった。

今後の方向性としては、開発した環境認識システムを、実環境で動作して評価することを検討しており、そのためには本論文内で述べた2つの停止位置以外の探索点や各停止位置の安全判定条件を準備する必要がある。さらに、本研究で開発した環境認識システムを、停止位置のみならず走行中にも応用することも検討しており、そのためには走行経路全体の地図および車両の走行経路をデータとして扱える形にした上で、走行中に注意を払う必要がある場所においては経路以外の探索点を準備する必要があると考えられる。

**謝辞** この研究開発は、平成30年度の経済産業省・国土交通省の「高度な自動走行システムの社会実装に向けた研究開発・実証事業：専用空間における自動走行などを活用した端末交通システムの社会実装に向けた実証」の一環として行われたものである。

## 参考文献

- [1] “官民ITS構想・ロードマップ2019” 高度情報通信ネットワーク社会推進戦略本部・官民データ活用推進戦略会議 2019年6月7日 pp.2~3
- [2] “ラストマイル自動走行の実証評価（永平寺町）に係る新たな実証を開始 -遠隔ドライバー1名が2台の自動運転車両を運用する遠隔型自動運転の実証” 国立研究開発法人産業技術総合研究所 平成30年11月14日

## 付録

### 付録 A.1 探索点の保存に用いたデータ形式

システム構築時に予め保存しておく経路データについて、実際に用いた形式の例である。ファイルはJSON形式とし、各RFID番号に対する探索点のXYZ座標を保存した。

```
{
  "all_searchpoints": [
    {
      "rfid": 0,
      "searchpoints": [
        { "x": 0.0, "y": 0.0, "z": 0.0 },
        { "x": 0.5, "y": 0.0, "z": 0.0 }
      ]
    },
    {
      "rfid": 1,
      "searchpoints": [
        { "x": 0.0, "y": 0.0, "z": 0.0 },
        { "x": 0.5, "y": 0.0, "z": 0.0 },
        { "x": 1.0, "y": 0.5, "z": 0.0 }
      ]
    }
  ]
}
```

