

# LiDAR 搭載ドローンに適した不時着時の着陸地点の平坦性評価に関する研究

坂井 美月<sup>†</sup> 馬淵 浩司<sup>†</sup> 塚田 義典<sup>‡</sup>

岩手県立大学大学院ソフトウェア情報学研究科<sup>†</sup> 摂南大学経営学部<sup>‡</sup>

## 1. はじめに

ドローンが抱える問題の1つに、例えば飛行中、電波障害などのトラブルが発生し、操縦者側からの操作を受信しなくなった時、ドローンの墜落は不可避であることが挙げられる。この問題を解決するためには、パラシュートなどで自律的に不時着する際、障害物にぶつからないことと、着陸する場所が安全かつ平坦であることが重要である。本研究では、着陸する場所が安全かつ平坦であることを評価する手法を提案する。具体的には、検出した着陸場所が平坦であるか否かを数値で評価することや、地面からセンサまでの高度に応じて平坦推定にかかわる閾値を自動で設定するよう、パラメータの自動化を行うことである。

## 2. 処理フロー

具体的な処理フローを図1に示す。入力データは、Velodyne 社製の VLP-16 Hi-Res<sup>1)</sup> により取得した、3次元の点群データである。本研究では平坦推定を行う際、1フレームを読み込んで処理を行う。データ読込後の処理は、まず、着陸に適している平坦な地面を決定するために平坦推定処理を行う。この時、平坦推定のための閾値を、高度から得られる後述の式(1)を利用して求める。次に、平坦推定された点の色を変え、結果を表示させるために平坦推定結果表示処理を行う。最後に、推定された平坦が安全かどうかを評価するために、RMS 値算出処理を行う。

## 3. 閾値自動設定処理

点群データから周辺環境の形状やオブジェクトを識別する研究<sup>2)</sup>は数多く行われている。こうした研究では、オープンソースソフトウェアのPCL(Point Cloud Library)<sup>3)</sup>が利用されていることが多い。本研究でも同様に、PCLを利用する。そこで、PCLを用いた平坦推定を行う場合、点群データに応じて閾値を

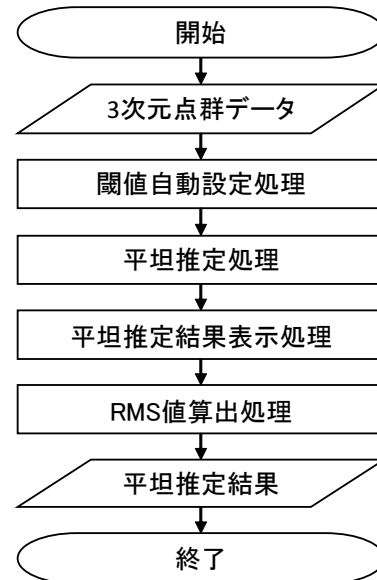


図1: 処理フロー

手動で設定しなくてはならない。リアルタイム処理への移行を考慮し、閾値を自動で求めることのできる式が得られないかどうかの検証を行った。本学1階(約1.5m)、2階(約6m)、3階(約10m)で取得した点群データから平坦推定を行い、結果を目視で確認した。これを繰り返し、目視で最も良い結果を示した時の平坦推定の閾値をおよそ最適な閾値としている。高度と閾値の推移を図2に示す。また、この推移から、近似式(1)を得ることができた。

$$y = 0.0059x + 0.0407 \quad (1)$$

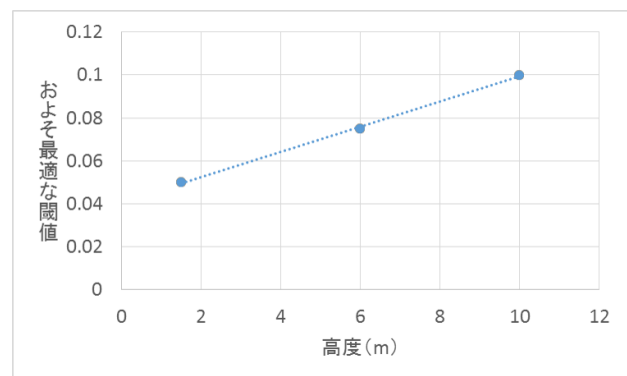


図2: 高度と閾値の推移

Evaluating the Flatness of Emergency Landing Places using a LiDAR to be mounted on drones

Mizuki SAKAI<sup>†</sup>, Hiroshi MABUCHI<sup>†</sup>, Yoshinori TSUKADA<sup>‡</sup>

<sup>†</sup>Graduate School of Software and Information Science, Iwate Prefectural University, <sup>‡</sup>Faculty of Business Administration, Setsunan University

## 4. 平坦推定処理

### 4.1. モデルとメソッドタイプの設定

本処理では、PCL の `pcl::SACSegmentation` クラスを利用する。本研究では、点群データ中の面積が最も大きい平面を安全な着陸地点候補とする。理由は、面積が大きい平面は実世界においてドローンの安全な着陸に適している地面である可能性が高いからである。まず、Segmentation オブジェクトの生成のため、モデルには、点群の平面モデルを抽出可能な `setModelType(pcl::SACMODEL_PLANE)` を設定する。これにより、平面モデル抽出時に3次元空間上の平坦を表す方程式 (2) の各係数  $a, b, c, d$  も同時に得ることができる。

$$ax + by + cz + d = 0 \quad (2)$$

次に、平面の抽出は、RANSAC 法<sup>4)</sup>を用いて平面中で最大の点群が存在する平面を探索するのが一般的である。したがって本研究でも同様にメソッドタイプを RANSAC と設定した。

### 4.2. 閾値の指定

ここで、前述の近似式 (1) を用いて閾値の指定を行う。

## 5. 平坦推定結果表示処理

4. 章の処理により抽出した平坦の点を、そうでない点と別の色に変え、Visualizer で表示を行う。

## 6. RMS 値算出処理

最後に、平坦性評価のため、方程式 (2) と各係数  $a, b, c, d$  を元に、平坦推定された各点が平面モデルとどれだけ離れているかを評価する。評価の値には RMS (Root Mean Square) を用いる。RMS 値は式 (3) で表される。

$$RMS = \sqrt{\frac{X_1^2 + X_2^2 + \dots + X_{n-1}^2 + X_n^2}{n}} \quad (3)$$

## 7. 実験結果

本実験では、VLP-16 Hi-Res を用いて3次元点群データを取得した。VLP-16 Hi-Res と同じ位置から撮影した画像と、取得した3次元点群データを平坦推定した結果の点を赤色に塗り Visualizer で表示した結果を図3に示す。本実験では、壁や天井など、平坦推定の妨げになる点はあらかじめ手で削除している。図3からわかるように、目視で平らであるとわかる床部分を正しく平坦推定していることが確認できた。また、この時の RMS 値は 0.031 (31mm) であり、安全に着陸できる平坦な地面であることを正しく評価している。一

方、地面にテーブルや人などの障害物があり、地面と障害物を同一の平面と推定された場合、RMS 値は 0.216 (216mm) とばらつきが生じる。



図3: 点群データの取得位置 (左) と平坦推定結果 (右)

## 8. おわりに

本研究では、平坦推定後の平坦性評価が正しく行えることと、平坦推定にかかる閾値の自動化ができることを検討した。その結果、リアルタイムに平坦性評価を行うことができる可能性が示唆されたと考える。今後は、近似式 (1) が高度 10m 以上の場合においても適用可能であるかどうかの確認や、点群データを取得する範囲を高度によって増減させることで平坦推定の妨げになる点群を減らす工夫などさらに実用化を考慮したプログラムの改善が課題となる。また、今後はセンサを搭載したドローンから取得した点群データを用いて、本研究の有用性を確認することを予定している。

## 謝辞

本研究にあたりご協力いただいた、東亜非破壊検査株式会社様に感謝する。

## 参考文献

- 1) Velodyne Lidar, “VLP-16 Hi-Res”, <https://velodynelidar.com/products/puck-hi-res/>, (最終閲覧日 2020-06-25) .
- 2) 高田智, 榎田修一, “LIDAR を用いた歩行者検出のための RealAdaBoost に基づく特徴選択”, 自動車技術会論文集, Vol.49, no.4, pp.799-805, 2018.
- 3) “Point Cloud Library”, <https://pointclouds.org/>, (最終閲覧日 2021-01-06) .
- 4) M. A. Fischler, “R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography.” Comm. of the ACM, Vol.6, no.24, pp.381-395, 1981.