

修士論文

ドローンを利用した樹木の
自動調査に向けた機械学習モデルの
教師データ生成手法

2022年1月27日 提出

指導教員 川原 圭博 教授

東京大学大学院工学系研究科

電気系工学専攻

37-206488 仙田 薫

■ 内容梗概

IoT を活用したセンシング・AI など、先端技術の社会実装が多方面で進む中、農業生産の分野でもこれら最先端の技術が普及し始めている。人間の五感に依らない評価量を用いて対象の植物を定量的に評価・分析することで将来的に生産力の向上やより計画的で安定した農業生産の実現が期待されている。近年では LiDAR に代表されるレーザスキャナなどのセンサ機器や、二次元画像から被写体の三次元構造を再構築する技術などの登場により、樹木の立体構造を定量的に評価できるようになった。特に三次元空間内の位置情報や色情報を持つデータである点群データは、各個体の立体構造を詳細に把握できるため注目されている。しかし分析のため点群を各個体に分割する必要があるが、さまざまな環境に適用できる普遍的な分割アルゴリズムは難しい問題である。またドローンを利用したセンシング手法は広範囲を短時間で測定可能であり、圃場中の生育個体の撮影など農業生産の分野でも利用が進んでいる。特に深層機械学習との組み合わせにより、撮影した画像から目的の各個体を検出する手法が大きな注目を集めているが、学習に必要となる大量の教師データを収集する労力が問題となる。

はじめに樹木を対象に圃場の点群を個体分割するアルゴリズムを提案する。そもそも普遍的に適用できるような分割アルゴリズムは目標とせず、特定の目的に絞って限られた環境下で効果を発揮できる domain-specific なアルゴリズムを考案した。圃場では管理された環境下で樹木が育成されているため、樹木特有の立体構造を仮定し利用できる。面で構成される箱形状の多い人工物とは対照的に、樹木は枝や幹など細長く伸びる柱状が連なるワイヤースケルトン状の構造を持つ。そこで局所的な点群の主成分分析を行い、第一主成分により柱状の枝が伸びる方向を推定する。単純な三次元ユークリッド距離の大小に基づく隣接点定義に方向の概念を追加し、再定義された隣接点に従って走査すれば各樹木の点群をより選択的に探索できる。提案手法を実際の圃場で測定した点群データに適用して評価した結果、 $F_1 \text{ score} = 0.94$ の分割精度を実現した。

次にドローンが撮影する画像から樹木を検出する深層機械学習に必要な教師データを自動生成する手法を提案する。膨大な数の画像に対し手作業によるラベリングは非常に煩雑である。そこでドローンの画像群から三次元再構成によって合成される点群・オルソ画像上で個体検出を行い、検出結果を座標変換し元の各ドローン画像上へ投影して教師データとする。点群またはオルソ画像での検出作業は自動化しかつこの作業は1回で済むため、教師データ生成の作業全体で大幅な労力軽減が実現できる。圃場をドローンで撮影したデータセットを利用して提案手法を評価し、自動生成された教師データで学習した検出モデルは平均精度 $\text{AR}@.5 = 0.80 \sim 0.95$ を達成した。加えてヤフー株式会社との共同研究により、農村周辺をドローンで撮影したデータセットを利用してより実用的な環境への提案手法の適用を実験した。データセットの違いによる評価結果の違いを示しその原因や解決策を議論する。

目次

第 1 章	序論	1
1.1	本論文の背景	2
1.2	本論文の概要	2
1.3	本論文の構成	3
第 2 章	先行研究	5
2.1	点群の測定手法	6
2.2	樹木の点群の個体分割手法	6
2.3	機械学習による画像からの樹木の検出	7
第 3 章	点群の個体分割手法	9
3.1	樹形を考慮した近傍モデルに基づく分割アルゴリズム	10
3.1.1	地面の除去	10
3.1.2	個体の検出	10
3.1.3	点群の個体分割	11
3.2	分割対象の点群データと正解データ	13
3.3	実験と分割性能の評価	13
第 4 章	機械学習による物体検出のためのデータ自動生成手法	17
4.1	三次元点群の個体分割を利用したデータ自動生成	18
4.1.1	入力画像からの三次元再構成	18
4.1.2	三次元点群の個体分割	19
4.1.3	個体分割結果の二次元空間への投影	19
4.1.4	評価方法	19
4.1.5	実験と結果	20
4.2	オルソ画像を利用したデータ自動生成	24
4.2.1	オルソ画像への合成	24
4.2.2	オルソ画像での検出モデル生成	25
4.2.3	オルソ画像からドローン画像への ROI 投影	26
4.2.4	実験と結果	27
4.3	比較と検討	28
第 5 章	実環境への応用可能性の検証	31
5.1	使用するデータ	32
5.2	オルソ画像における ROI の生成	32
5.3	ドローン画像への ROI 投影	34

5.4	検出モデルの学習結果	37
5.5	比較と検討	38
第 6 章	おわりに	41
6.1	議論	42
6.2	結論	43
	謝辞	44
	参考文献	45
	発表文献および受賞	47

目次

1.1	点群の局所的な主成分分析. 赤で示す近傍点 ($k = 100$ 程度) に対し主成分分析を行い得られた第一主成分 v_1 で枝の方向を推定する.	3
1.2	樹木の個体分割における点群の走査の様子. 1本の主幹を始点に枝分かれを経て各末端まで探索が進んでいく.	3
1.3	教師データの自動生成手法. ドローンが撮影した各画像のラベリング作業を自動化する.	3
3.1	樹木の点群の個体分割手法の流れ.	10
3.2	一般的な樹形の模式図. 地面から生える1本の主幹から多数の主枝・垂主枝へと枝分かれを繰り返す.	11
3.3	ボクセル空間における分割結果. 分割された個体ごとに色付けして示すが近接する一部の個体間では正しく分割されていない.	12
3.4	主幹位置の検出結果. 緑色の樹木全体の点群のうち赤色が検出された主幹部位.	12
3.5	実験に使用した点群データ. 高さ方向 z に応じて色付けして示す.	14
3.6	提案手法の個体分割結果.	15
3.7	個体分割された各クラスタの評価結果.	16
4.1	点群の個体分割を利用した検出モデルの教師データ生成手法の流れ.	18
4.2	IoU の計算方法.	20
4.3	精度・感度の曲線. 青色で示した面積が AUC.	20
4.4	東京大学田無キャンパスの圃場の点群. 画像中の点群は高さ方向 z の値に応じて着色している.	21
4.5	ステップ (b) で点群の個体分割から得られた ROI を描画して真上から見た画像. 画像は圃場全体を1枚に合成したオルソ画像であり, 後続のステップ (c) で ROI は各ドローン画像上へ投影される.	21
4.6	学習中の Loss 関数の変化.	22
4.7	点群の個体分割を基にした教師データで学習したモデルが推定した ROI (バウンディングボックス) を可視化した一例. 赤: テストデータ, 青: 推定値.	23
4.8	オルソ画像を利用した検出モデルの教師データ生成手法の流れ.	24
4.9	左: 圃場全体のオルソ画像. 右: グリッド分割の様子.	24
4.10	学習中の Loss 関数の変化.	25
4.11	圃場のオルソ画像上で推定された ROI (バウンディングボックス). 左: 分割された各画像で推定された ROI をすべて分割前のオルソ画像上に描画したもの. 右: 重複する ROI を合算して処理した ROI を描画したもの.	26

4.12	オルソ画像上のピクセル座標から Geo 座標に変換するときの高さ方向 z 座標の曖昧さ.	27
4.13	オルソ画像上の推定結果を元のドローン画像上へ座標変換した ROI (バウンディングボックス) を可視化した一例. 赤: 比較のため 4.1 章の点群個体分割を基に生成した ROI, 黄: 座標変換された ROI.	27
4.14	学習中の Loss 関数の変化.	28
4.16	提案したふたつの手法で生成された ROI の比較. 赤: 点群の個体分割, 黄: オルソ画像上での検出を基に生成されている.	28
4.15	オルソ画像から投影した ROI で学習したモデルの推定値を描画. 黄: 教師データ ROI, 青: 推定された ROI.	29
5.1	農村周辺をドローンで撮影した画像から再構成した点群データ.	33
5.2	農村周辺のオルソ画像.	33
5.3	学習中の Loss 関数の変化.	34
5.5	高さ方向 z 座標の測定を点群データ中に描画した様子. 左: Geo 座標へ変換された ROI が図中の赤線で囲まれた領域となり, 領域内部の点群を高さ方向 z に応じて色付けしている. 右: 領域内部の点群の z 座標の分布.	34
5.4	農村周辺のオルソ画像上で推定された ROI (バウンディングボックス). 上: 分割された各画像で推定された ROI をすべて分割前のオルソ画像上に描画したもの. 下: 重複する ROI を合算して処理した ROI を描画したもの.	35
5.6	オルソ画像上の推定結果を元のドローン画像上へ座標変換した ROI (バウンディングボックス) を可視化した一例. 赤: 比較のため手動による正解データを基に座標変換した ROI, 黄: オルソ画像上の推定値を座標変換した ROI.	36
5.7	学習中の Loss 関数の変化.	37
5.8	DeepForest の樹冠検出モデルの学習手法.	38
5.9	オルソ画像から投影した ROI で学習したモデルの推定値を描画. 黄: テストデータ ROI, 青: 推定された ROI.	39
5.10	ROI の座標変換過程における高さ z の差による投影位置のずれ. 鉛直下を向くカメラ視線軸から離れている部分のみ示す. 高さの違い z_0, z_1 により同一の ROI がドローンの画像上では異なる位置に投影される.	40

■ 表 目 次

3.1	クラスタ数に基づく個体分割の評価.	16
3.2	点群の数に基づく個体分割の評価.	16
4.1	検出モデルの混同行列. P:positive, N:negative	20
4.2	利用する物体検出モデルと学習の詳細.	20
4.3	点群の個体分割を基に生成した ROI で学習させた検出モデルの評価.	22
4.4	オルソ画像上での学習結果.	25
4.5	オルソ画像から投影した ROI で学習させた検出モデルの評価.	28
5.1	オルソ画像上での学習結果. 比較のため前 4.2 章で示した圃場データでの場合の値も載せる.	34
5.2	利用する物体検出モデルと学習の詳細.	37
5.3	オルソ画像から投影した ROI で学習させた結果と比較.	38

■ 第1章

序論

1.1 本論文の背景

IoT を活用したセンシング・AI など、先端技術の社会実装が多方面で進む中、農業生産の分野でもこれら最先端の技術が普及し始めている。特に従来の人間の五感を主とする観察や経験的知識に基づく植物の生育状態の評価に代わり、測定データから定量的に評価する技術であるフェノタイプングが注目されている。フェノタイプングとは、葉の数・葉の色・枝の太さといった従来の素朴な評価量だけでなく、ドローンによる圃場の空撮画像・枝や葉の三次元的な位置情報など、先進的なセンシング技術で測定可能となった評価量を用いて対象の植物を定量的に評価する。フェノタイプングにより対象の植物が数値として表現できると、植物を人間の主観的な判断に依存せずモデル化する解析手法が可能となる。こうした成果を現場の圃場へ反映することで、将来的に生産力の向上やより計画的で安定した農業生産の実現が期待されている。

近年では LiDAR に代表されるレーザスキャナなどのセンサ機器や、二次元画像から被写体の三次元構造を再構築する技術などの登場により、植物の立体構造を定量的に評価できるようになった。森林資源の管理に関する研究は特に盛んで、樹木の生育状況を評価する様々な手法が以前より提案されている [1]。従来は森林の木の高さ・幹の太さ・空撮画像などから樹木のバイオマス量を統計的に推測する手法が多かったが、近年では一本一本の木を三次元空間でモデリングして取り扱う例が増えている [2, 3]。農業生産の観点においては、果樹を対象としたフェノタイプングとして樹木の水分ストレスの評価 [4] や突然変異育種法における変異種を分類する手法 [5]、果樹の構造を測定して生育評価する手法 [6] などがある。その多くが三次元空間内の位置情報や色情報を持つ三次元データである点群データを用いて立体的な構造を分析している [7]。生育評価に限らず点群データを分析するには各個体に分割する必要があるが、普遍的に適用できる点群の個体分割手法は難しい。

またドローンを利用したセンシング手法は広範囲を短時間で測定可能であり、圃場中の生育個体の撮影など農業生産の分野でも利用が進んでいる。特に深層機械学習との組み合わせにより撮影した画像から目的の各個体を検出する手法が大きな注目を集めており、樹木の樹冠を検出する手法 [8] などがある。物体検出の技術では Faster-RCNN [9] など高精度なモデルが次々と提案されており様々な用途への応用が期待されているが、物体検出に限らず深層機械学習に必要な大量の教師データを収集する労力が問題となる。

1.2 本論文の概要

まず樹木を対象とした点群の新たな個体分割手法を提案する。圃場という人の手により管理された樹木が並ぶ環境を仮定することで、樹木特有の立体構造を利用する。具体的には樹木を構成する幹や枝を部分的に円柱状と見なし、図 1.1 のように点群の (x, y, z) 座標に対し局所的な主成分分析を行い第一主成分により枝の伸びる方向を推定できる。純粋な点群の個体分割アルゴリズムでは、三次元ユークリッド距離の大小のみに従って定義される隣接点を辿りながら点群を走査するが、提案手法ではユークリッド距離に加え主成分分析で推定した枝の方向を加味することで図 1.2 のように各個体を選択的に探索できる。

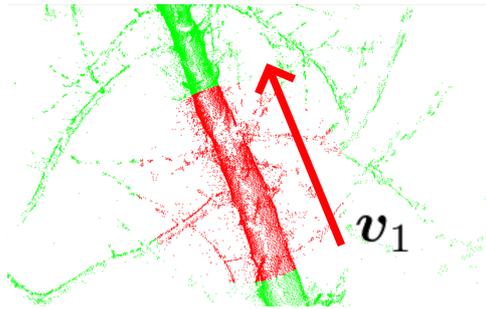


図 1.1: 点群の局所的な主成分分析. 赤で示す近傍点 ($k = 100$ 程度) に対し主成分分析を行い得られた第一主成分 v_1 で枝の方向を推定する.

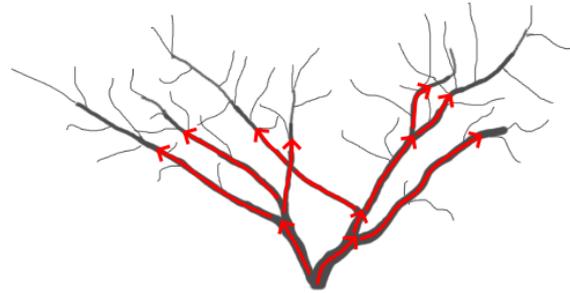


図 1.2: 樹木の個体分割における点群の走査の様子. 1本の主幹を始点に枝分かれを経て各末端まで探索が進んでいく.

次にドローンによる空撮と深層機械学習の組み合わせによる樹木の検出手法において、検出モデルの学習に必要な膨大な教師データの生成を自動化する手法を提案する。愚直に教師データを用意しようとするれば、ドローンが撮影した膨大な数の画像に対し1枚ずつ対象の物体の位置をバウンディングボックスとしてラベリングする必要がある。近年は様々なラベリングを支援するツールが実用化されているものの、手作業を介するラベリング作業は煩雑である。そこで提案手法では三次元再構成技術に注目し、ドローンが撮影した大量の画像を合成して測定範囲全体をひとつの点群・オルソ画像として表現する。この点群またはオルソ画像に対し個体位置を検出し、各ドローンの画像上へ座標変換を行い教師データを生成する。点群の個体分割には前述の提案手法を適用し、オルソ画像中の個体検出には別途検出モデルを学習することで自動化する。結果として図 1.3 のように教師データ生成の作業全体を可能な限り自動化できる。

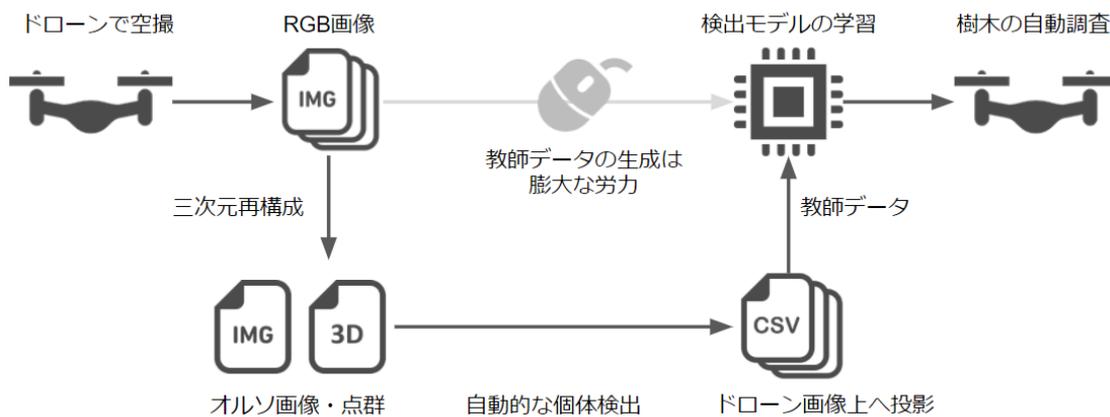


図 1.3: 教師データの自動生成手法. ドローンが撮影した各画像のラベリング作業を自動化する.

1.3 本論文の構成

本稿では、はじめに2章で点群データの測定と個体分割・ドローンと深層機械学習によるセンシング技術に関する先行研究を紹介し、次に3章で樹形を考慮したヒューリスティックな近傍モデル

を利用したより高精度な点群の個体分割手法を提案する。加えて4章では、深層機械学習による物体検出で問題となる教師データの生成を自動化する手法を提案する。5章ではより実用的なデータセットへ提案手法を適用し応用可能の可否を検討する。最後に6章で各評価実験の結果を議論し結論とする。

■ 第2章

先行研究

樹木に限らず物体の三次元的な構造を直接表現できるデータとして三次元点群の利用がある。また深層機械学習により RGB 画像から物体の位置・種類を検出する手法も大きな注目を集めている。ここでは点群の測定・分析と画像からの物体検出に関して先行研究を紹介する。

2.1 点群の測定手法

近年ではレーザスキャナを用いて樹木表面の座標点を容易に収集できるようになった [10]。ただし 1 回の測定ではレーザが直接届く表面上の点しか観測できず同じ樹木を囲むように何回も測定する必要があるため、全体の測定には時間を要する場合もある。また風などの外乱で測定中に対象の位置が変化する、地上に設置された測定機器から離れた上方の点群の密度が低下するなど、留意すべき点がある [11, 12]。植物を測定する場合には葉が生い茂っていると表面しか測定できない問題 [10] があるため、落葉している期間に測定するなどの対策が必要である。

一方で平面画像から三次元空間中の位置を推定する手法もある。同じ対象物を異なる位置から撮影した複数の平面画像を入力とし、各画像中の特徴量を画像間でマッチングさせ三次元再構成を行う。ドローンによる空撮画像を利用する手法は屋外の大きな対象物の点群データも容易に取得できるため近年大きな注目を集めている [13]。樹木を含め一般に植物を対象とする場合は、細い枝や茎・薄い葉・曲がりくねった幹や枝が互いに交差したり重なった複雑な立体構造のため、従来の手法をそのまま適用するのは難しく工夫が必要になる [14]。

2.2 樹木の点群の個体分割手法

樹木をドローンで撮影して点群を測定するとき、圃場全体の点群から個体分割する必要がある。そこで樹木の点群を対象に個体分割する既存手法をいくつか紹介する。

Yu Jiang ら [6] は低木の果樹（ブルーベリー）を対象として矩形範囲で領域を分割する手法を用いている。圃場の地面は平面である、圃場中の各個体は一直線上に整列している、個体どうしの範囲は重複していない、という仮定の下で適用できる手法である。Kai Wang ら [15] も圃場ゆえの仮定を利用したヒューリスティックなアルゴリズムを提案している。はじめに三次元空間のユークリッド距離のみ考慮した隣接点定義に従い単純な個体分割を行い、おおまかな個体ごとに点群を分割している。ただ近接する個体どうしは点群が連続的に分布しており正しく分割できないため、次に分割された点群の分布は正規分布に従う仮定のもと分散を計算し閾値を定め、各クラスターの幾何中心から離れた点を削除することで隣接する個体の点群も分割できる。最後に近傍探索により余計に削除してしまった点を拾って最終的な個体分割が完了する。

また点群の一般的な分割アルゴリズムとして Density-Based Spatial Clustering of Application with Noise (DBSCAN) と呼ばれる手法もある。同じクラスタに属する点群どうしの距離は短くなる点に注目したアルゴリズムであり [16]、単純であるが応用範囲は広い。例えば森林の樹木を対象に LiDAR で測定した点群データを高さ方向に階層的に DBSCAN で分割する手法が提案されている [17]。主に針葉樹で構成される森林中の各個体は鉛直方向へ長く伸びている場合が多いため、水平方向の切断面を見れば各個体の位置が容易に検出できる点に注目した手法である。

別の手法として点群をボクセルに分割して三次元配列としてデータを扱う手法もある。ボクセルとは x, y, z 軸平行に並ぶ同一サイズの立方体で三次元空間全体を離散化した単位（平面画像におけ

るピクセルに相当する)であり、データサイズを大幅に削減できると同時に平面画像と同様な探索アルゴリズムを適用できる利点がある。Fred Westling [18] らはアボカド農園を測定した点群データに対しボクセル (0.1m) により点群を分割し、各ボクセル内部の幾何重心の座標点を頂点に持つようなグラフ構造を定義して探索アルゴリズムを適用している。各頂点には対応するボクセル内部の点群から計算される各種特徴量が付与されるが、探索手法自体は既存のグラフ理論同様に適用できるのが大きな特徴である。

近年、機械学習を用いた物体の識別や位置検出が大きな注目を集めている。特に Convolutional Neural Network (CNN) は二次元画像の処理で大きな成果を上げている。三次元データでも “Point-Net” [19] など点群のクラス判別・領域分割問題を解く手法が提案されている。特に三次元点群の場合には次のような問題がある [20]。

- 教師データの不足。機械学習にとって教師データの量は極めて重要であるが、二次元画像に比べて三次元点群は測定に時間・費用がかかる上に普及も進んでおらず、学習に使える量が圧倒的に少ない。もしくはすべての教師データを自前で用意する必要があるが、それは大きな負担になる。
- 点群は非構造化データである。ピクセルの二次元配列を扱う従来の CNN を直接適用できない。
- 点群の持つ情報の制限。CNN が扱う二次元画像では各ピクセルの画像中の位置情報に加え RGB チャンネルなどの色情報をもつ。しかし点群では、測定方法によっては xyz 座標のみの場合もある。レーザスキャナの測定では反射強度、ドローンの空撮画像と三次元再構成では色情報を含む場合もある。

植物の点群を対象とした個体分割に関しても機械学習を用いる試みがなされている。Shichao Jin ら [21] は平面画像の Region-based CNN (R-CNN) で個体検出を行い、Region-Growth アルゴリズムで植物の点群を個体分割する手法を提案している。LiDAR で測定した三次元点群を 32 の異なる視点から見た二次元画像へ変換し機械学習の入力として用い、R-CNN は画像中の目的物の位置 (region of interest, ROI) として各個体の茎の部分を検出する。次に検出された茎は元の三次元空間上の位置に逆変換され、これらの位置を開始点として Region-Growth アルゴリズムが点群の分割を行う。開始点から点群の近傍探索を進めていき、各点と異なる個体の茎の位置との距離を比較し、どの個体に属するか判断しながら個体分割を行う。

またボクセル空間において点群を CNN を用いて葉・茎の構成要素に分割する手法も提案されている [20]。三次元点群をボクセルの三次元配列として扱えば、構造化データとして処理できるため CNN の利用が可能となる。畳み込み層がボクセル単位で $3 \times 3 \times 3$ の大きさの空間フィルターと ReLU 活性化関数から成るの特徴であり、入力データは元の点群をボクセルの三次元配列で表現したもので、各ボクセルは内包する点群の有無 (教師データでは分割されたクラスターのラベル) の 1 チャンネルの情報を持つ。Auto-encoder によって圧縮された特徴量マッピングを利用して茎・葉の判別と要素ごとの点群分割を行う。

2.3 機械学習による画像からの樹木の検出

物体検出は深層機械学習でもっとも研究されている分野のひとつであり、近年では Faster-RCNN [9] や YOLOv2 [22] に代表されるような高精度な予測・高速な学習が可能なモデルが提案されている。

農業生産や森林資源管理の分野においてもドローンで撮影した画像から植物の位置を検出する手法が多く提案されているので紹介する。

樹木の検出を目的とした研究では、森林を空撮した RGB 画像から樹木の樹冠を検出する DeepForest [8] が開発されている。DeepForest が使用した教師データは米国の National Ecological Observation Network's Airborne Observation Platform [23] によって収集され、37 地点 1 億本以上の樹木を LiDAR で測定した点群と空撮した RGB 画像が含まれている。提案手法ではまず LiDAR の点群を既存手法 (lidR ライブラリ [24]) で個体分割し、おおまかな樹木の位置を推定して学習の教師データとして利用する。空撮した RGB 画像から樹冠を検出するモデルは Retinanet [25] を基に構成され、点群の個体分割からの推定値に加え手作業によるラベリングを教師データとして学習させる。手作業によるラベリングだけでなく、点群の分割により自動生成した教師データで部分的に学習させる点が大きな特徴である。

Juntao Xiong ら [26] はマンゴー園をドローンで撮影した画像データから未熟なマンゴー果実を検出する手法を提案している。YOLOv2 [22] の検出モデルを学習させ入力画像に対し対象物の ROI (バウンディングボックス) を推定する。この検出モデルを利用して各画像から果実の数を検出し、園場全体での果実数を線形近似で推定し収量予測に役立てる目的である。なおモデルの学習に使用する教師データは手作業によるラベリングである。

RGB カメラに限らずマルチスペクトルカメラを利用する手法も提案されている。マルチスペクトルカメラの入力画像は可視光 3 チャンルに加え近赤外帯域を含む 4 チャンルを持つため、可視光だけでは捉えられなかった特徴量も抽出できると期待されている。Ovidiu Csillik ら [27] は園場の柑橘果樹をドローンに搭載したマルチスペクトルカメラで撮影し、4 チャンルの画像を CNN で処理して樹木・地面・雑草の各クラス確率をピクセル値に持つような確率マップを出力している。最後に確率マップを Simple Linear Iterative Clustering (SLIC) アルゴリズムで superpixel (位置・画素値が近いピクセルの集合) にクラスタリングして各クラスの物体の位置を検出する。SLIC は画像を superpixel に分割するアルゴリズムのひとつであり、各ピクセルの座標 x, y と画素値 (Lab 色空間) からなるパラメータ空間に変換しユークリッド距離に従って k-mean 法の要領でクラスタリングする。検出精度の評価に用いる正解データは手作業によるラベリングである。

Lucas Prado ら [28] も柑橘果樹園をマルチスペクトルカメラで撮影した画像から樹木の位置を検出する手法を提案している。CNN により潜在する画像特徴量を抽出し、各ピクセル位置での対象物の存在確率を表した信頼性マップとして出力する。信頼性マップの局所的な最大値を探索して最終的な検出位置を決定するため、学習モデルは CNN のみで構成され複雑な最終層を持たない特徴がある。提案手法の評価に使用する正解データは手作業によりラベリングされた個体位置を基にガウシアンで信頼性マップを生成している。

■ 第3章

点群の個体分割手法

ドローンによる撮影と三次元再構成の組み合わせは広範囲の点群データをまとめて測定できるため、高速・高効率な点群取得方法として注目されている。しかし得られるデータには複数の個体を含むため、各個体に分割する必要がある。従来の点群クラスタリング手法では、圃場に個体が混み合っていると近接する個体どうしの境界をうまく見つけられない欠点がある。ここでは果樹の樹形を考慮したヒューリスティックな近傍モデルを定義することで、点群をより高精度に個体分割する手法を提案する。

3.1 樹形を考慮した近傍モデルに基づく分割アルゴリズム

提案手法を図 3.1 で示した過程ごとに説明する。各過程において点群の走査・特徴量推定には各点の近傍を定めておく必要があるが、今回は実装の簡単と処理速度のため kd 木を用いる。このとき最近傍の点の数を指定するパラメータ k は点群の密度などに影響されるので、各過程ごとに実験的に決めた。



図 3.1: 樹木の点群の個体分割手法の流れ。

3.1.1 地面の除去

まず点群から地面の部分除去する。測定された点群データの大半は地面に属するため、はじめに除去することで後の処理量を軽減できる。地面は局所的には平面と見なせるため各点の法線を推定し ($k = 50$)、鉛直方向へ向いた法線を持つ点群を抽出する。

3.1.2 個体の検出

次に個体分割の基準とするために各個体の位置をボクセル空間において調べる。図 3.2 のように、樹木は一番太い 1 本の主幹から枝分かれし地面から離れながら伸びる形状を持つため、各個体の点群を探索するとき主幹位置を始点とすると都合がよい。

まず圃場全体の点群を 0.2 m ごとに離散化してボクセル空間を用意する。次にクローリング処理をボクセル単位で行った後、互いに隣接するボクセルを探索してクラスタリングする。ノイズを除去するために、各クラスタを成すボクセルの個数 (50) とそのボクセルに含まれる点群の総数 (2000) に下限を設けておく。こうしてボクセル単位で分割された様子が図 3.3 であるが、互いに接近している個体どうしは枝や幹が重なっているため、ひとつのクラスタに複数の個体が含まれる場合がある。そこで各個体は地面から生える 1 本の主幹から始まり枝が伸びる構造に注目し、各ク

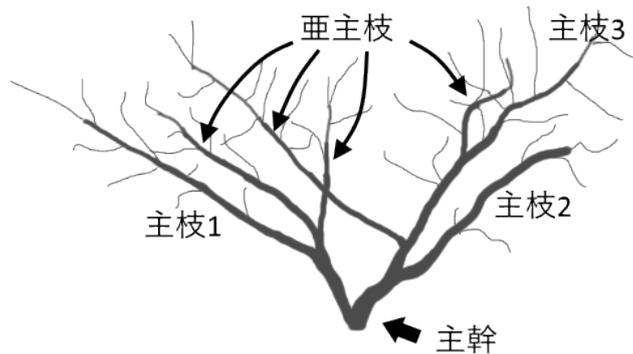


図 3.2: 一般的な樹形の模式図. 地面から生える 1 本の主幹から多数の主枝・亜主枝へと枝分かれを繰り返す.

ラスタに対し高さ方向 2m までのボクセルを抽出して再度クラスタリングすると、図 3.4 のように主幹の部分を検出できる.

3.1.3 点群の個体分割

最後に点群を個体分割する. 図 3.2 のように果樹の立体構造は平面部分が少なく、細く直線上に伸びる幹や枝から構成される. そこで果樹の点群の空間座標 x, y, z に対し主成分分析を行い第一主成分に注目すると、局所的に点群が一直線上に並んでいる方向として枝の伸びる方向を推定できる. 単純なユークリッド距離の大小に加えて、推定された枝の伸びる方向に位置する点へより優先的に探索が進むように隣接点定義を設計することで、樹木の幹や枝を選択的に探索し各個体の点群を検出できる.

具体的には点群の探索時に kd 木 ($k = 50$) を用いて隣接点を探すとき、各点の主成分分析 ($k = 100$) を行い第一主成分の分散が十分に大きいことを条件に加える. 先ほど得られた主幹位置を始点として探索すると、根本から先端の枝までを辿りひとつの個体を探索できる. 多くの場合はこれで個体を判別できるが、個体どうしが接近していると複数の始点から同じ点へ到達可能な場合がある. 圃場で計画的に栽培されている樹木を対象としているため隣合う個体の樹形や大きさに極端な差はないと考え、そのような場合は主幹位置からの最短経路長が短い方へ分類する. これを疑似コードで表現したのがアルゴリズム 1 である.

点群 P と 3.1.2 章で得られた主幹位置を入力として、関数 `Segment_All` (1 行) が個体分割を行う. P の各点 p に関して、あらかじめ計算した主成分分析の結果を分散 (共分散行列の固有値) が大きい順に収めた配列を $p.v$ (固有ベクトル), $p.\lambda$ (固有値) とする. 変数の初期化を終えたら関数 `Segment_One` (13 行) を呼び出し主幹位置を始点として各個体の点群を探索する. 幅優先で探索しながら三次元空間中のユークリッド距離 (20 行) を計算し、始点からの最短経路長を記録しておく. このとき関数 `Get_Neighbors` (25 行) が定義する点の近傍を利用するが、指定された点 c から距離が k 番目に短い近傍点までを kd 木で単純に探索する (27 行) だけでなく、主成分分析の結果も近傍の条件に加えている.

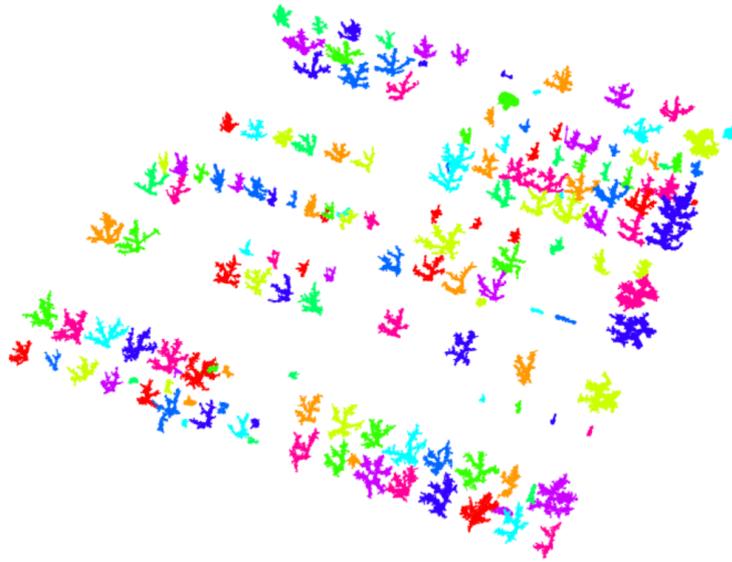


図 3.3: ボクセル空間における分割結果. 分割された個体ごとに色付けして示すが近接する一部の個体間では正しく分割されていない.

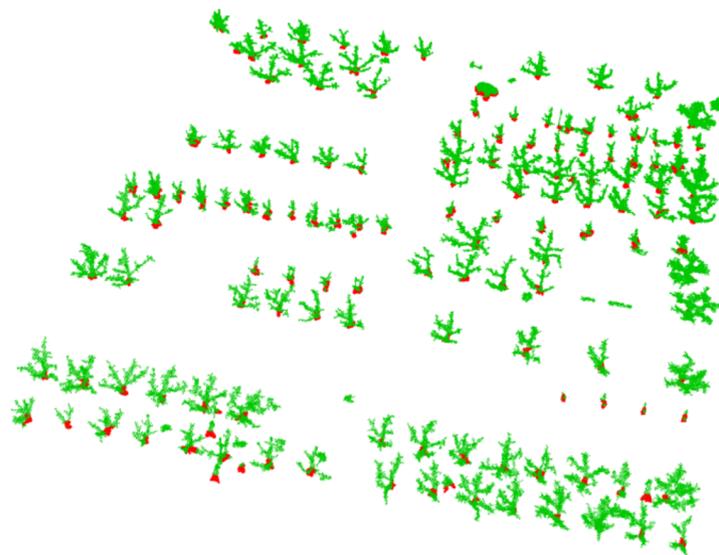


図 3.4: 主幹位置の検出結果. 緑色の樹木全体の点群のうち赤色が検出された主幹部位.

Algorithm 1 個体分割

Input: P : すべての個体の点群, S : 各個体の主幹位置
Output: T_{all} : 分割された各個体の点群のリスト

```

1: function SEGMENT_ALL( $P, S$ )
2:   for  $\forall p \in P$  do
3:      $p$ .label := null
4:      $p$ .dist := FLT_MAX
5:    $T_{\text{all}} := []$ 
6:   for  $l = 1 \dots S$ .length do
7:     SEGMENT_ONE( $P, S[l], l$ )
8:      $T_{\text{all}}$ .add([ ])
9:   for  $\forall p \in P$  do
10:    if  $p$ .label  $\neq$  null then
11:       $T_{\text{all}}[p$ .label].add( $p$ )
12:   return  $T_{\text{all}}$ 
13: function SEGMENT_ONE( $P, s, l$ )
14:    $Q := [s]$ 
15:    $s$ .label =  $l$ 
16:    $s$ .dist = 0
17:   while  $Q$ .length > 0 do
18:      $q := Q$ .pop()
19:     for  $\forall p \in \text{GET\_NEIGHBORS}(P, q)$  do
20:        $d = |p - q|$ 
21:       if  $q$ .dist +  $d < p$ .dist then
22:          $p$ .dist =  $q$ .dist +  $d$ 
23:          $p$ .label =  $l$ 
24:          $Q$ .push( $p$ )
25: function GET_NEIGHBORS( $P, c$ )
26:    $L := []$ 
27:   for  $\forall p \in \text{KD\_SEARCH}(P, c, k)$  do
28:     if  $p$ . $\lambda$ [1]/ $p$ . $\lambda$ [2] > 1.5 then
29:        $L$ .add( $p$ )
30:   return  $L$ 

```

3.2 分割対象の点群データと正解データ

岡山県赤磐市の農業試験場においてモモの圃場を測定した。DJI 社製のドローンを用いて圃場全体を空撮して、撮影された RGB 画像を基に三次元再構成した点群データが図 3.5 である。また分割性能の評価のため手作業による分割結果（計 135 本）を正解データとして用意した。

3.3 実験と分割性能の評価

提案手法の実装には Point Cloud Library (PCL) [29] を利用する。PCL の提供する基本的な点群処理アルゴリズムと可視化ツールを活用できるため採用した。提案手法を適用し、図 3.6 のような分割結果を得られた。

分割アルゴリズムの評価方法として先行研究 [30] で言及されている方法を用いる。まず基準 [31] に従い正解データと提案手法による分割をマッチングしてクラス分けする。クラス分けを説明するため以下の変数を定義しておく。

P 入力として与えられたすべての点の集合。

s 正解データのクラス数 (135)。

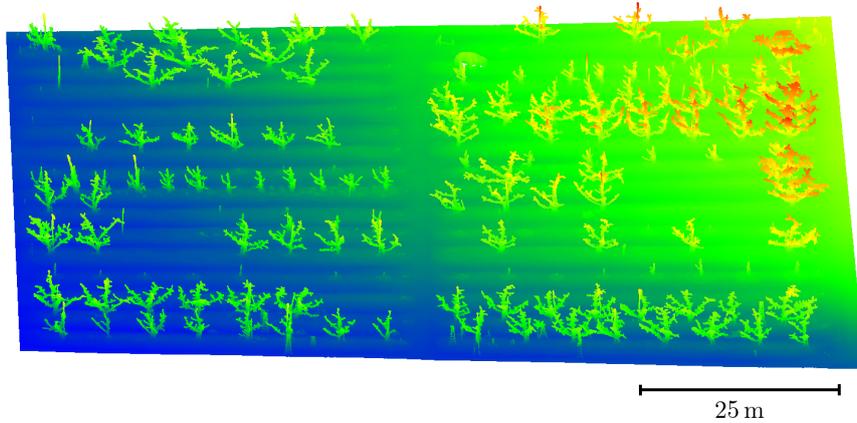


図 3.5: 実験に使用した点群データ. 高さ方向 z に応じて色付けして示す.

S_i 正解データにおいて, ラベル i ($i = 1 \dots s$) に属するクラスタ. 各クラスタは排他的と仮定する.

$$1 \leq \forall i < \forall j \leq s, S_i \cap S_j = \emptyset$$

なお P には地面など, どのクラスタにも含まれない点群が存在する.

$$S_1 \cup S_2 \cup \dots \cup S_s \neq P$$

t 提案アルゴリズムが分割したクラスタの数.

T_j 提案アルゴリズムが分割したクラスタ ($j = 1 \dots t$). 互いに排他的と仮定する.

r クラス分けに用いるパラメータ. 値が 1 に近いほどクラスタのマッチングが厳しくなり, クラスタの一致判定で許される誤差が小さくなる.

文献 [31] には $0.5 < r < 1.0$ が適切とある.

次の定義に従って各クラスタを 5 クラスのうちどれかに分類する.

1. correct detection

正解データのクラスタ S_i と提案アルゴリズムが分割したクラスタ T_j がパラメータ r が許す誤差の範囲で一致しているとき, 正しい分割と判断する. すなわち

$$|S_i \cap T_j| \geq r|S_i| \wedge |S_i \cap T_j| \geq r|T_j|$$

を満たすクラスタ.

2. over-segmentation

正解データにおいてひとつのクラスタを提案アルゴリズムが複数のクラスタに分割した場合は過剰な分割と判断する. すなわち正解データのクラスタ S_i と提案アルゴリズムが分割したクラスタ $T_{j_1}, T_{j_2}, \dots, T_{j_M}$ ($2 \leq M \leq t$) で

$$1 \leq \forall x \leq M,$$

$$|S_i \cap T_{j_x}| \geq r|T_{j_x}| \wedge \sum_{1 \leq x \leq M} |S_i \cap T_{j_x}| \geq r|S_i|$$

を満たすクラスタ.



図 3.6: 提案手法の個体分割結果.

3. under-segmentation

正解データにおいて複数のクラスを提案アルゴリズムがひとつのクラスにしか分割できなかった場合は過少な分割と判断する. すなわち正解データのクラス $S_{i_1}, S_{i_2}, \dots, S_{i_N}$ ($2 \leq N \leq s$) と提案アルゴリズムが分割したクラス T_j で

$$1 \leq \forall x \leq N,$$

$$|S_{i_x} \cap T_j| \geq r|S_{i_x}| \wedge \sum_{1 \leq x \leq N} |S_{i_x} \cap T_j| \geq r|T_j|$$

を満たすクラス.

4. miss

正解データのクラスに対し提案アルゴリズムが分割したクラス中にマッチングするものが存在しない場合は失敗と判断する. すなわち正解データのクラス S_i で “correct detection”, “over-segmentation”, “under-segmentation” のどれにも分類されていないクラス.

5. noise

正解データに対応するクラスが存在しないものの提案アルゴリズムが検出したクラスは雑音と判断する. すなわち提案アルゴリズムが分割したクラス T_j で “correct detection”, “over-segmentation”, “under-segmentation” のどれにも分類されていないクラス.

クラスをマッチングする際に許容する誤差の範囲を指定するパラメータは $r = 0.6$ を用いた. $r \neq 1$ より上記に示した定義では複数のクラスに分類可能なクラスが存在も考えられる. そこで $1, 2, \dots, 5$ の順にクラスをマッチングしていき, 一度クラスに分類されたら以降のマッチングの対象から外すことでクラス分けの一意性を保証する. こうして得られた結果が図 3.7 であり, 各クラス数を集計したのが表 3.1 である. (“correct detection” のクラス数) / (すべてのクラス数) で計算される精度は 83.3% である.

以上はクラスに基づく評価であり, 次は点群の数に基づいた評価を行う. 正解データ・提案アルゴリズムの分割において木々のクラスに属する点群を Positive, 地面などそれ以外の点群を

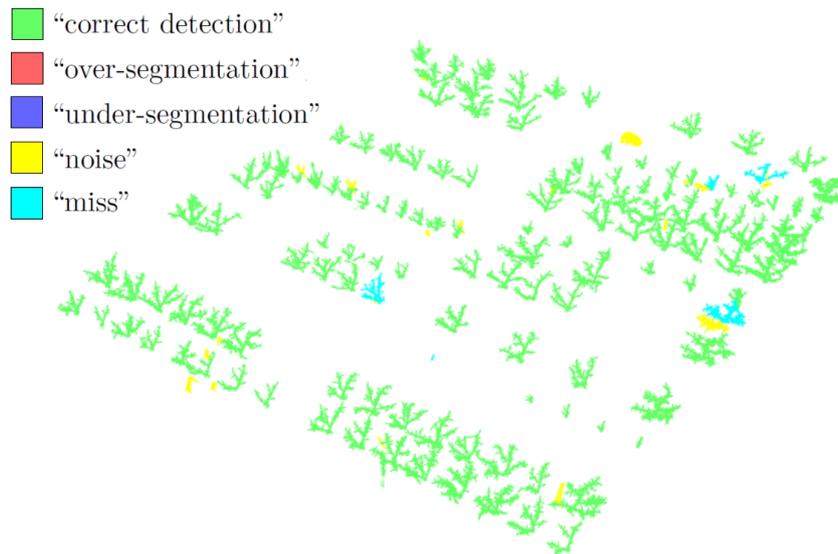


図 3.7: 個体分割された各クラスタの評価結果.

表 3.1: クラスタ数に基づく個体分割の評価.

class	count
correct detection	130
over-segmentation	0
under-segmentation	0
miss	5
noise	21

表 3.2: 点群の数に基づく個体分割の評価.

		Predicted	
		P	N
Actual	P	4.2%	0.18%
	N	0.36%	95%

Negative と定義すれば、表 3.2 のような混同行列を得る。感度 (recall) $TP/(TP + FN) = 95.9\%$ 、精度 (precision) $TP/(TP + FP) = 92.2\%$ である。 F_1 score $= 2 \frac{\text{precision-recall}}{\text{precision+recall}}$ の値を計算すると 0.94 である。

圃場の樹木を対象とした他の点群の個体分割手法と比較すると、LiDAR による高精度点群を利用した Kai Wang らの実験 [15] での分割精度が 95.4%、FredWestling ら [18] の低精度な点群に適用した結果が平均精度 91.5% など、他手法と比べても同程度によい数字が得られた。提案手法では従来のユークリッド距離のみ考慮した隣接点定義に局所的な主成分分析による方向の特徴量を追加したもの、探索の方法は従来通りでアルゴリズム全体がシンプルであり実装も容易である点で他手法より優れている。

■ 第4章

機械学習による物体検出のためのデータ 自動生成手法

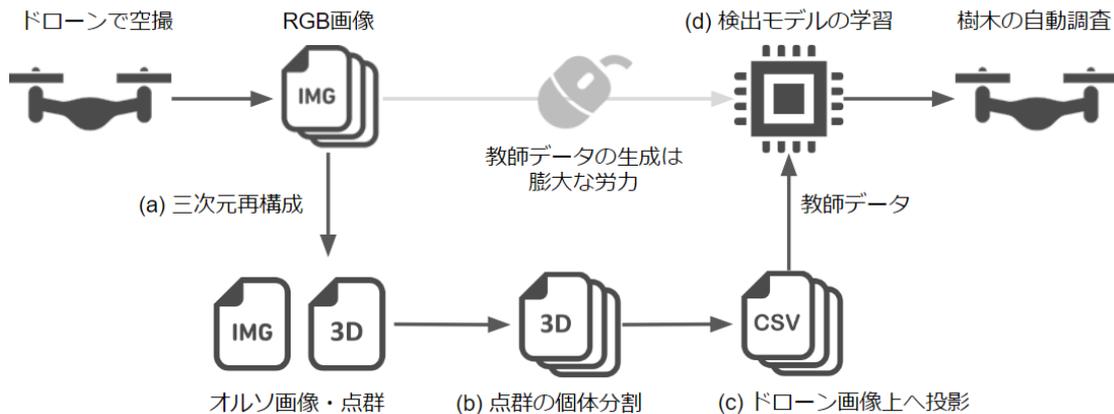


図 4.1: 点群の個体分割を利用した検出モデルの教師データ生成手法の流れ。

近年，深層機械学習による画像処理が大きな成果を挙げており，物体検出とドローンと組み合わせ空撮した圃場の画像から各個体の位置を検出する手法が研究されている．しかし機械学習には大量の教師データが必要であり，物体検出においては個体の位置（ROI, Region of Interest）を示すバウンディングボックスや物体のクラスを表すラベルを画像と共に学習させる．これまで多くのツールが開発されてきたものの，ほとんどが人の手で ROI（バウンディングボックス）やラベルを用意する必要があり，非常に煩雑な作業である．こうした大量の教師データの用意の難しさが問題となり，機械学習のさらなる利活用の妨げとなっている．

類似研究として，Weinstein らはアメリカ合衆国の 37 地点で測定された 1 億を超える木々を撮影した画像データを含む NEON Tree Crowns Dataset [23] を利用し，樹木の樹冠を画像中から検出するモデル DeepForest [8] を開発している．DeepForest が対象とするのは森林を上空から撮影した画像であり，高い検出精度が報告されている．LiDAR で測定した点群を個体分割して教師データを一部自動生成しているものの，手作業によるラベリングが最終的には必要になっている．そこで本稿ではドローンで撮影した画像から三次元再構成で得られる点群とオルソ画像を利用して教師データをより容易にかつ大量に生成する手法を提案する．

4.1 三次元点群の個体分割を利用したデータ自動生成

提案手法では図 4.1 のように三次元空間中での処理を挟むことで自動的に個体位置の ROI（バウンディングボックス）を生成する．大量の入力画像を一旦点群データに変化し，三次元空間中で個体分割を行う．分割結果を二次元空間に投影することで元のドローン画像上の ROI が得られる．なおここでは画像中から検出する個体は圃場中の樹木に限定するため，物体のクラスはひとつのみでラベリングの必要はない．図 4.1 に示した提案手法を段階ごとに説明する．

4.1.1 入力画像からの三次元再構成

図 4.1(a)：入力画像はドローンで空撮した圃場の大量の画像である．通常は学習機にそのまま入力として与えるだけだが，提案手法では大量の画像を三次元再構成してひとつの点群データに変換

する。三次元再構成は SfM (Structure from Motion) と呼ばれる手法であり、画像中の特徴量を計算し画像間の対応する特徴量のピクセル座標から被写体の三次元座標を計算する。そのため、画像は互いに被写体が重複するように少しずつずらしながら撮影する必要がある。三次元再構成には商用ライブラリの PIX4Dmapper を利用する。

4.1.2 三次元点群の個体分割

図 4.1(b)：検出対象以外のバックグラウンドに該当する点群を除去したうえで、検出したい個体を三次元空間中で分割する。元の入力画像は非常に数が多いが、三次元再構成された点群はひとつだけであり、この個体分割の作業は1回で済む。点群の個体分割に関しては先行研究が多くのアルゴリズムを提案している。ここでは前3章で示した手法を利用する。

4.1.3 個体分割結果の二次元空間への投影

図 4.1(c)：個体分割により得られた各樹木の点群がドローンで撮影した画像の中ではどこに位置するか計算する。まず点群 (x, y, z) を EasyIDP [32] により二次元空間に投影する。EasyIDP は PIX4Dmapper による三次元再構成において点群の三次元座標と元の画像中のピクセル座標を任意に変換するためのライブラリである。点群は画像上のピクセル座標 (h, v) の集合に変換され、次のような画像の横・縦方向に平行な四辺で囲まれた矩形範囲を ROI として定める。

$$(x, y, z) \in \mathbb{R}^3 \xrightarrow{\text{EasyIDP}} (h, v) \in \mathbb{R}^2 \xrightarrow{\text{Extract ROI}} \begin{cases} h_{\min} \leq h \leq h_{\max} \\ v_{\min} \leq v \leq v_{\max} \end{cases}$$

4.1.4 評価方法

物体検出モデルの評価方法を簡単に説明する。物体検出では対象の物体の位置 (ROI) をバウンディングボックスとして扱うが、まずテストデータとモデルが推定した ROI が一致するかどうかの判定基準が必要となる。一般に ROI の一致判定には IoU (Intersection over Union) の値が使用される。図 4.2 に示すように比較するふたつの ROI に対し積集合と和集合の面積を計算しその比率を IoU と定義する。適当な閾値 ($0 < \text{IoU} < 1$) を指定して、IoU が閾値を超えたらふたつの ROI は一致する・閾値を下回ったら不一致と判定する。検出モデルの性能評価指標として物体検出のベンチマークとして広く利用されているデータセット COCO (Common Objects in Context)[33] で使用されている評価方法・ソフトウェア [34] を使用する。IoU 閾値を与えるとテストデータの各画像において推定値の ROI と比較して図 4.1 のような混同行列を考えられる。

TP：テストデータと推定値で一致した ROI の数

FP：推定されたものの一致するテストデータがない ROI の数

TN：定義なし

FN：テストデータのうち一致する推定値がない ROI の数

精度 (Precision) と感度 (Recall) は次のように計算される。

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

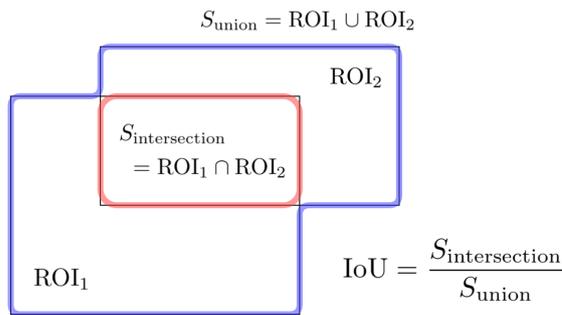


図 4.2: IoU の計算方法.

表 4.1: 検出モデルの混同行列.

P:positive, N:negative

	推定値	
	P	N
テストデータ P	TP	FN
テストデータ N	FP	TN

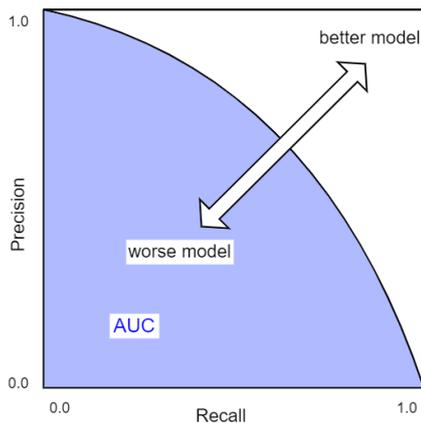


図 4.3: 精度・感度の曲線. 青色で示した面積が AUC.

表 4.2: 利用する物体検出モデルと学習の詳細.

説明	値
学習モデル	Faster-RCNN (pretrained)
データセット	画像 112 枚
テストデータ	22 枚
学習データ	450 枚
学習時間	10 epoch

4.1.5 実験と結果

提案手法を実際の測定データに適用してその有効性を確認する。実験では東京大学田無キャンパスの圃場をドローンで空撮した画像を利用する。図 4.4 に三次元再構成された点群を示す。データ中には 9 種 100 本以上の果樹が存在しており、最終的には学習したモデルでドローンの画像から果樹の位置を検出することを目標にする。

3 章で示した手法で点群を個体分割し、得られた点群を EasyIDP [32] で元の画像上へ投影して図 4.5 のような ROI (バウンディングボックス) 群が得られる。図中では一枚に合成した圃場全体の画像上に ROI を示しているが、実際には各ドローン画像上に ROI が得られる。

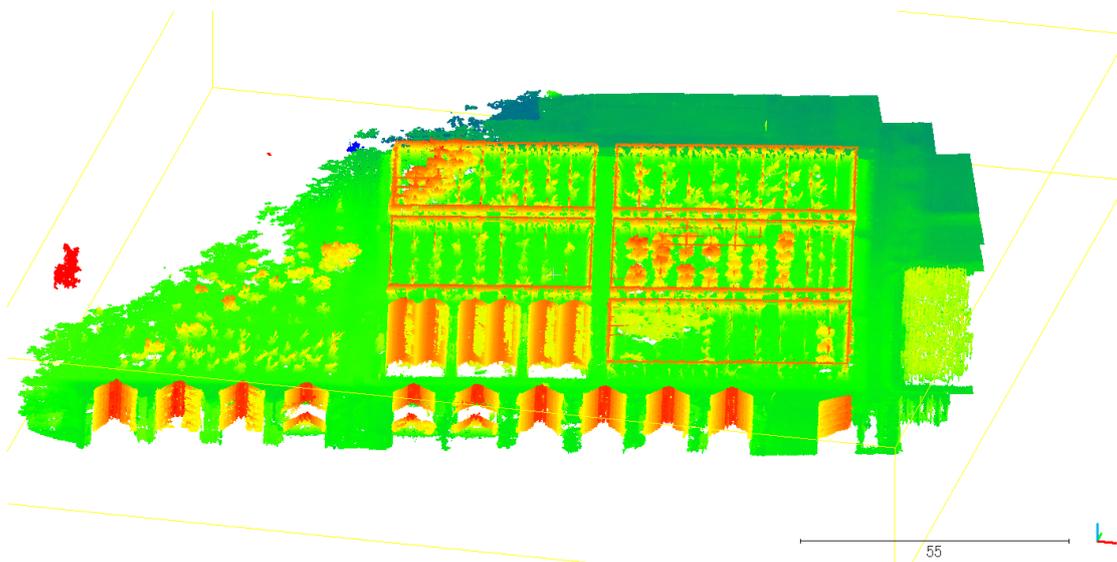


図 4.4: 東京大学田無キャンパスの圃場の点群. 画像中の点群は高さ方向 z の値に応じて着色している.

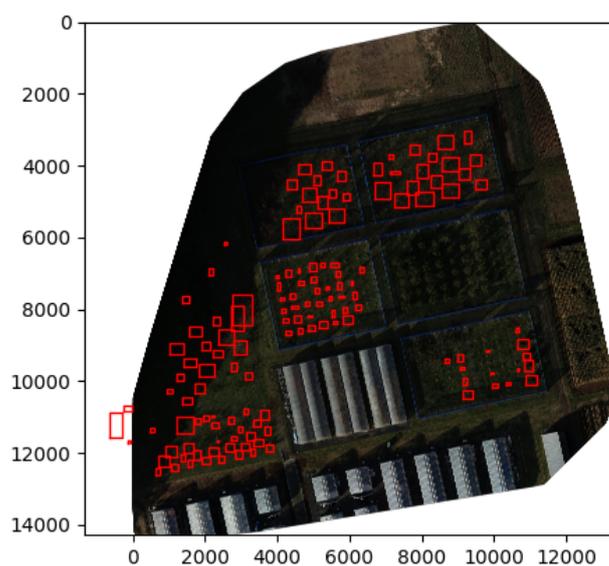


図 4.5: ステップ (b) で点群の個体分割から得られた ROI を描画して真上から見た画像. 画像は圃場全体を1枚に合成したオルソ画像であり, 後続のステップ (c) で ROI は各ドローン画像上へ投影される.

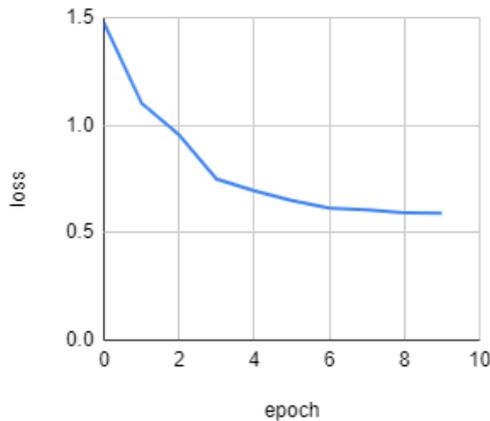


図 4.6: 学習中の Loss 関数の変化.

表 4.3: 点群の個体分割を基に生成した ROI で学習させた検出モデルの評価.

IoU	AP	AR
0.2	0.935	0.981
0.5	0.806	0.866
0.2:0.5	0.891	0.945

次に物体検出の機械学習モデルを用意・学習させる (表 4.2). ここでは学習済み Faster-RCNN [9] を基に転移学習 (ファインチューニング) を行う. 入力画像は 8:2 の比率でランダムに学習・テストデータセットに分割し, 学習データの方は左右反転・Random Crop・Random Zoom Out などの操作により Data Augmentation を行いデータ数を 5 倍に増やした. 学習の結果を図 4.6 に示す. 最終的な検出の精度は表 4.3 のとおりである. ここでは COCO (Common Objects in Context) [33] の定義に従い, 平均精度 (AP, Average Precision) と平均感度 (AR, Average Recall) をいくつかの IoU (Intersection over Union) の値の場合で示している. ただしオブジェクトの大きさに制限は課していない (area = all). 通常の物体検出モデルの評価では $0.5 \leq \text{IoU} \leq 0.9$ 程度の閾値で計算するのが普通である. ただし IoU による検出精度の計算は ROI の重なる面積を基に行われるため, 検出された位置が少しずれただけでも正しい検出と見なされない厳しい計算方法となっている. 今回の樹木の位置を画像から検出する場合は, 樹木の存在の有無と大まかな位置が分かれば十分である. そこで検出位置の 1~2 m 程度のずれは許容できると考え, 一般より低い閾値 $\text{IoU} < 0.5$ でも評価計算を行っている. IoU を閾値とした評価方法は定量的であるものの, 樹木の位置の検出が目的どおり達成されているか直感的な把握は難しい. そこでテストデータに使用した各ドローン画像上に推定された ROI を描画して可視化したのが図 4.7 である. 表 4.3 で計算された平均精度の値と可視化した図から見て樹木の大まかな位置が正しく検出されており, 今回の目的は達成されていると判断できる.

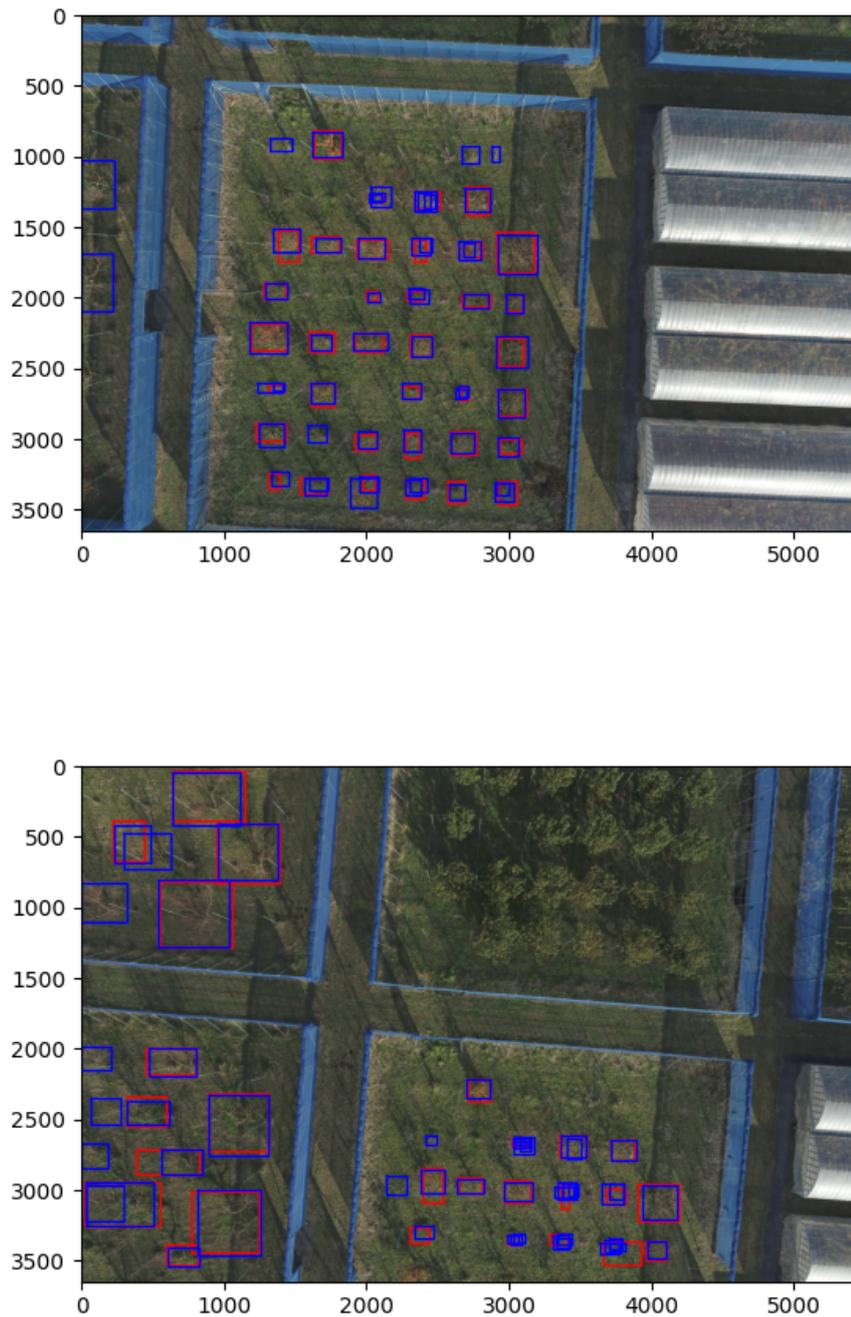


図 4.7: 点群の個体分割を基にした教師データで学習したモデルが推定した ROI (バウンディングボックス) を可視化した一例. 赤: テストデータ, 青: 推定値.

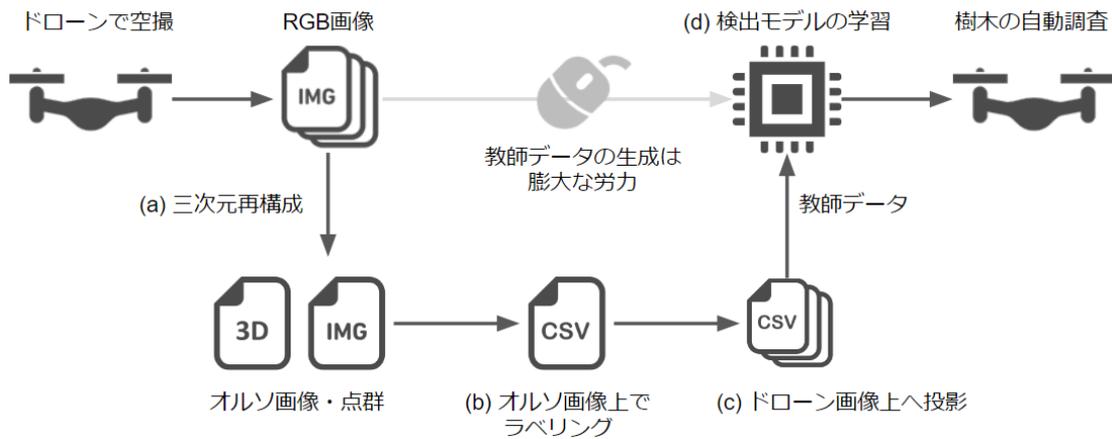


図 4.8: オルソ画像を利用した検出モデルの教師データ生成手法の流れ.

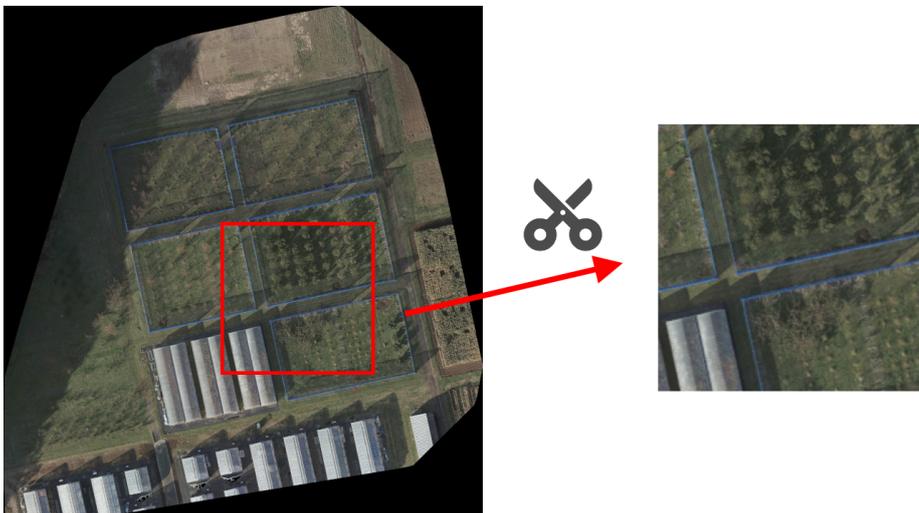


図 4.9: 左：圃場全体のオルソ画像. 右：グリッド分割の様子.

4.2 オルソ画像を利用したデータ自動生成

通常、ドローンで空撮した画像中の物体はカメラからの距離に応じて位置がずれて写る。物体が地面から高く、また画像中の外周部に近いほどずれは大きくなる。この位置ずれを補正して地図のようにちょうど真上から見て物体の大きさと位置を正しく表現した画像をオルソ画像という。今回は点群ではなくオルソ画像を利用して教師データを容易に大量生成する手法（図 4.8）を提案する。

4.2.1 オルソ画像への合成

図 4.8(a)：ドローンで撮影した多数の画像を合成し歪みを補正する。ここでは三次元再構成を行う商用ライブラリ PIX4Dmapper を利用して図 4.9 のような 1 枚のオルソ画像に合成する。

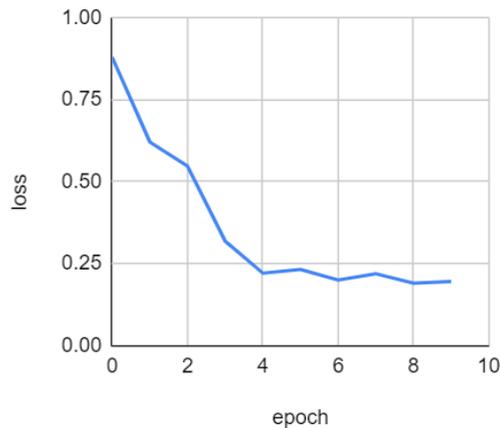


図 4.10: 学習中の Loss 関数の変化.

表 4.4: オルソ画像上での学習結果.

IoU	AP	AR
0.2	0.956	0.988
0.5	0.901	0.926
0.2:0.5	0.941	0.970

4.2.2 オルソ画像での検出モデル生成

図 4.8(b): オルソ画像上で目的の樹木の位置 (ROI) をバウンディングボックスとしてラベリングする. オルソ画像 1 枚のみとは言え画像サイズが非常に大きいため, すべて手作業によるラベリングは煩雑である. ここでは, いくつか樹木のラベリングのみ手入力してオルソ画像上での検出モデルを学習させる.

オルソ画像の分割

測定範囲が大きくなるとオルソ画像も巨大な画像データとなる. そのままでは学習機の入力データとしてサイズが大きすぎるため, 適宜分割する必要がある (図 4.9). ただし, ただグリッド状に分割すると分割の境界線上やその付近に位置する樹木の ROI を学習機に上手く与えられないため, 適当なオーバーラップを設けて分割を行なった. また分割された画像のうち対象の ROI をひとつも含まないような教師データとして不適当な画像は除去する.

- 元のサイズ: 13343×14280
- 分割サイズ: 4096×4096
- 分割のオーバーラップ: サイズの 50%
- 分割後の枚数: 36 (うち ROI を含む有効な画像は 28 枚)

オルソ画像での学習

オルソ画像上での検出モデルを学習するため, 教師データとして使う ROI をラベリングする. ここでは前 4.1 章でも利用した三次元点群の個体分割結果を基にした ROI を画像と一緒に入力する. 学習は 4.1.5 章で述べた方法と同様に行う. ただし画像が 28 枚とさらに少ないためデータ拡張により学習データは 7 倍の 154 枚, テストデータは 6 枚となっている. 学習の様子と結果を図 4.10 と表 4.4 に示す.

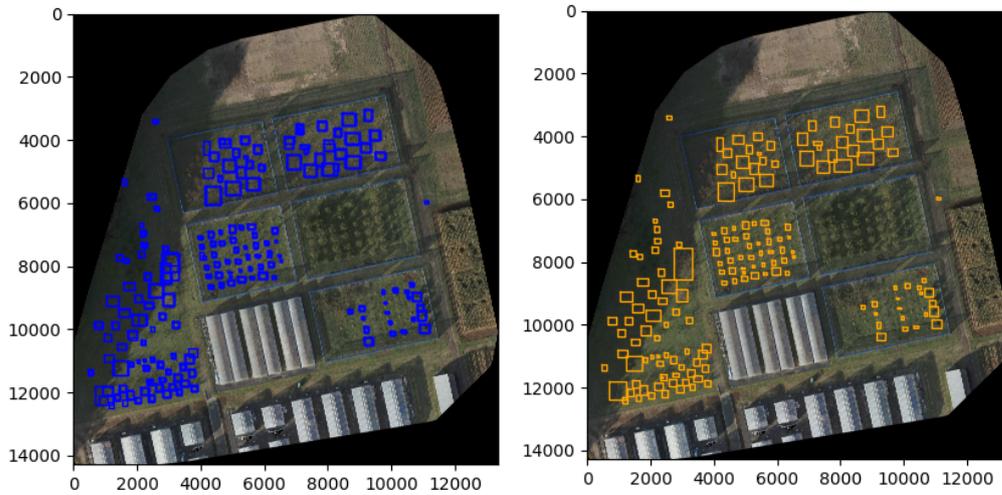


図 4.11: 圃場のオルソ画像上で推定された ROI (バウンディングボックス). 左: 分割された各画像で推定された ROI をすべて分割前のオルソ画像上に描画したもの. 右: 重複する ROI を合算して処理した ROI を描画したもの.

オルソ画像上のラベリング結果

次に全 36 枚の分割画像から推定された ROI を統合し、オルソ画像上に描画すると図 4.11 のようになる. このとき ROI の重複は $\text{IoU} = 0.1$ の基準で決定し、重複する各矩形範囲は合算している. 矩形範囲の合算方法は次のとおりである.

$$\begin{aligned} S_1 &= \{(h, v) \mid h_0 \leq h \leq h_1, v_0 \leq v \leq v_1\} \\ S_2 &= \{(h, v) \mid h_2 \leq h \leq h_3, v_2 \leq v \leq v_3\} \\ S_1 + S_2 &= \{(h, v) \mid \min(h_0, h_2) \leq h \leq \max(h_1, h_3), \min(v_0, v_2) \leq v \leq \max(v_1, v_3)\} \end{aligned}$$

4.2.3 オルソ画像からドローン画像への ROI 投影

図 4.8(c): オルソ画像上の ROI を元のドローン画像へ座標変換し、後続のステップ (d) で物体検出モデルの教師データとして使う. ドローン画像と合成されたオルソ画像の互いの位置関係は既知であり、両者間のピクセル座標は次式のように変換が可能である.

$$\text{オルソ画像 } (h, v) \rightarrow \text{Geo 座標 } (x, y, z) \rightarrow \text{ドローン画像 } i (h_i, v_i)$$

ただし図 4.12 のように変換途中の Geo 座標では高さ方向の z 値がオルソ画像上の ROI のみでは決定できない. Geo 座標から各ドローン画像上のピクセル座標への変換では視点と ROI の三次元的な位置関係が影響するため、 z 座標の値によって投影される ROI の位置がずれる場合がある. 本実験では地面の高さの z_0 を測定して与えているが、例えば図中の高さ z_1 を与えると赤色で示すような異なる座標へ投影されるため z 座標の指定には注意が必要である. 図 4.13 に変換された ROI をいくつかのドローン画像上で可視化して示す.

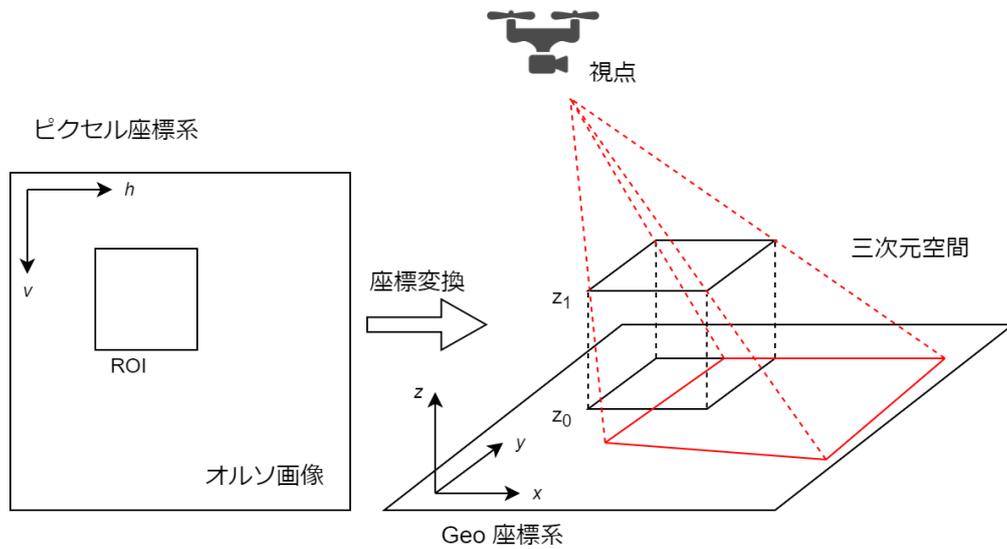


図 4.12: オルソ画像上のピクセル座標から Geo 座標に変換するときの高さ方向 z 座標の曖昧さ.

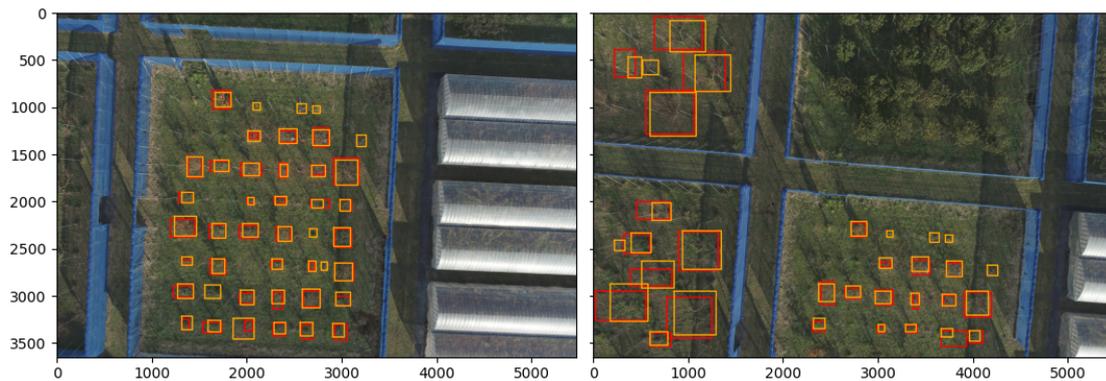


図 4.13: オルソ画像上の推定結果を元のドローン画像上へ座標変換した ROI (バウンディングボックス) を可視化した一例. 赤: 比較のため 4.1 章の点群個体分割を基に生成した ROI, 黄: 座標変換された ROI.

4.2.4 実験と結果

提案手法を実際のデータに適用し評価する. 前 4.1 章と同じく東京大学田無キャンパス内の圃場をドローンで撮影した画像を利用し, まず (a) 図 4.11 のようなオルソ画像を得た. (b) 三次元点群を利用して一部の ROI を決定し, オルソ画像上の検出モデルを学習して全体の ROI を推定した (図 4.11). (c) 次にライブラリ EasyIDP [32] を利用して各ドローン画像上へ座標変換を行い, (d) ドローン画像での検出モデルを学習させる. 学習の詳細は 4.1.5 章で述べたとおりで, 学習の様子と結果を図 4.14 と表 4.5 に示す. 平均精度による評価だけでは樹木の位置が正しく検出できているか直感的な把握は難しい. そこで図 4.15 に推定結果をドローンの画像上に描画した例を一部示す. 前 4.1 章で提案した点群の個体分割を利用する場合の推定結果 (図 4.7) と比べても, テ

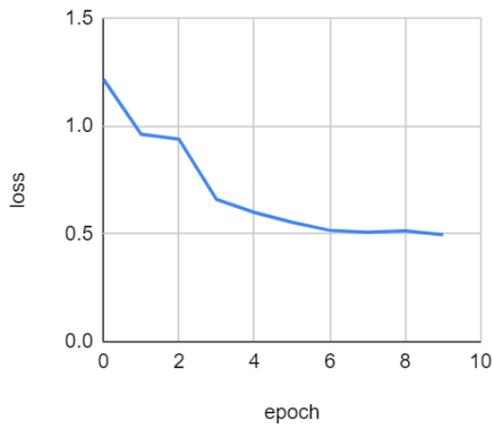


図 4.14: 学習中の Loss 関数の変化.

表 4.5: オルソ画像から投影した ROI で学習させた検出モデルの評価.

	IoU	AP	AR
	0.2	0.962	0.983
	0.5	0.939	0.963
	0.2:0.5	0.955	0.977

ストデータと推定値の ROI がより一致しており高い精度と感度を実現していると視覚的にも確認できる。

4.3 比較と検討

点群の個体分割を利用した手法の評価結果（表 4.3） $AP@.5 = 0.806$ と比べると、オルソ画像の利用した評価結果（表 4.4）の方が $AP@.5 = 0.939$ と良い精度が出ている。検出モデルと学習の方法は同じであるから、自動生成した ROI の精度の差が原因と考えられる。図 4.13 に示した学習に使用した ROI の様子を見ると、黄色の ROI（オルソ画像を利用して生成）に対応する赤色の ROI（点群の個体分割を基に生成）が無い部分が散見される。黄色のみが示す ROI の部分を直接目視で確認すると、樹木の位置として間違っていないと判断できる場合が多かった。赤色の ROI の基になった点群の個体分割の感度が不十分でいくつかの樹木の位置を検出できておらず、3章で使用した評価方法に従えば“miss”に該当するクラスが分割結果に多く含まれていたと言える。3章と本章で使用したデータセットは別の圃場で測定したものであり、本章のデータセットの方が果樹の種類が多くまた小さい果樹も比較的多かったため、分割結果が悪くなっていたと考えられる。



図 4.16: 提案したふたつの手法で生成された ROI の比較。赤：点群の個体分割，黄：オルソ画像上での検出を基に生成されている。

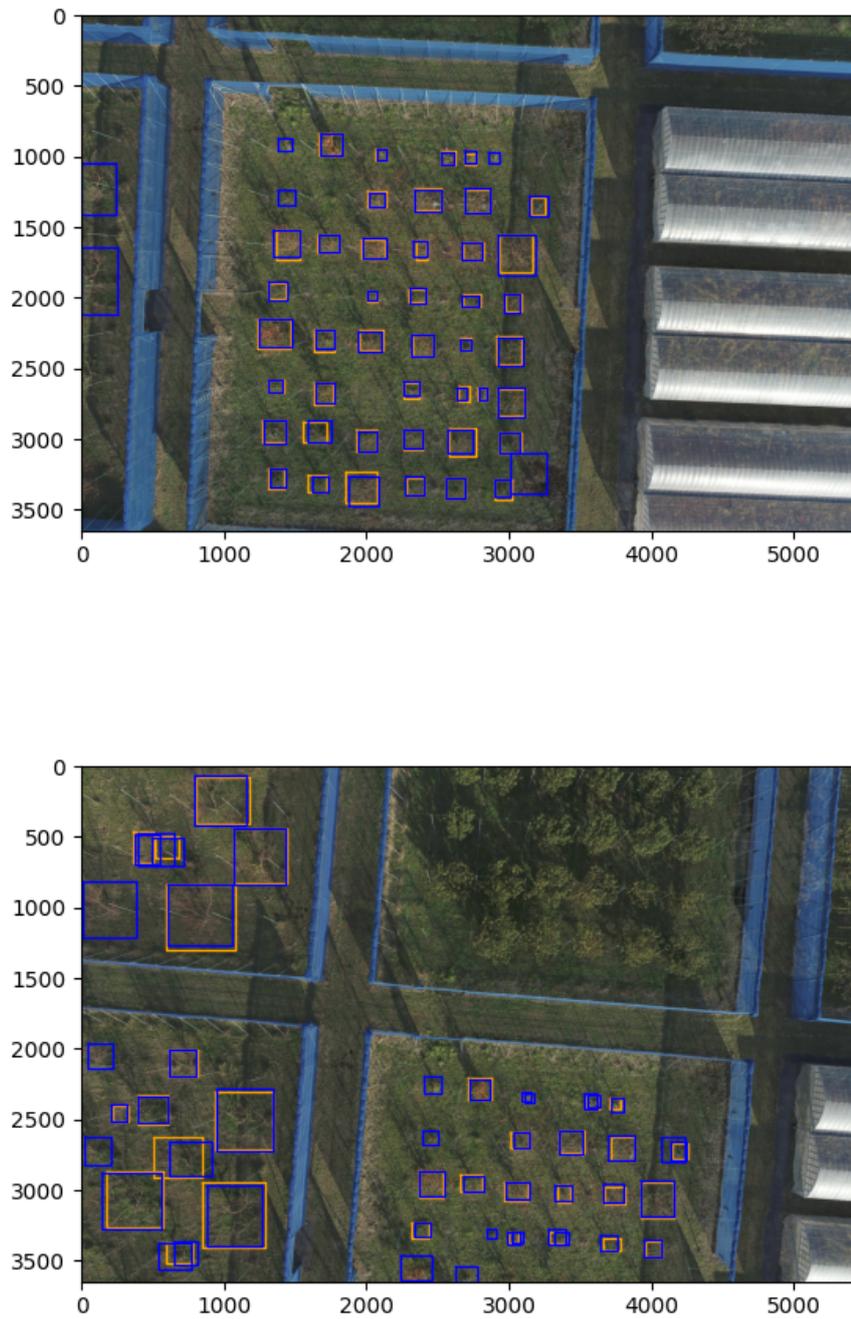


図 4.15: オルソ画像から投影した ROI で学習したモデルの推定値を描画. 黄: 教師データ ROI, 青: 推定された ROI.

実際に小さな果樹と大きな果樹が乱雑に存在する圃場外周部の様子を可視化して確認すると（図4.16），点群の個体分割で捉えられていない個体のROIが多く見られる．この結果より，3章で提案した点群の個体分割手法には仮定「圃場中の樹木のサイズはすべてが同程度」が厳しく要求されることが確かに分かった．次5章では応用として圃場以外のより複雑なデータセットで樹木の検出モデルの学習を試みるが，本章の比較結果より点群の個体分割を利用した手法の適用は難しいと判断し，オルソ画像を利用した手法のみ適用することにした．

■ 第5章

実環境への応用可能性の検証

近年少子高齢化の影響により耕作放棄が拡大しイノシシやサルによる獣害が拡大している [35]. 山端ら [36] が指摘するように獣害の要因としてエサ資源が考えられ、農村の豊かなエサ資源に惹かれて野生生物が人間の生活圏まで出没するようになる。このエサ資源には農作物だけでなく、放置された果樹・二番穂・収穫残渣・廃棄作物など普通人間が被害と認識しない対象も含まれる。こうした「無意識の餌付け」と呼ばれる行為が獣害の原因にもなっている。その中で放置された果樹（放任果樹）に関しては人口減少や空き家の増加に伴い人家・圃場脇に植樹された果樹が放置される場合が増えており、実際横山ら [37] が行った調査では捕獲された個体の分析により柿の実など人間の生活圏に由来する食性が確認されている。ドローンを利用したセンシング手法は広範囲を短時間で測定可能であり、また深層機械学習との組み合わせによりドローンが撮影する RGB 画像から目的の物体を検出できるため、放任果樹の新らたな調査手法として考えられている。しかし一般に機械学習には膨大な教師データが必要であり、その収集と前処理には多大な労力を費やすため調査手法の実現を阻害している。そこで本章では前 4.2 章で提案したオルソ画像を使用して教師データを生成する手法を適用しその有効性を確認する。

5.1 使用するデータ

本章で扱う実験はヤフー株式会社との共同研究として実施した。データは福井県の農村周辺をドローンで撮影した画像である。PIX4Dmapper で三次元再構成した様子を図 5.1 に示す。前 4.2 章で扱った圃場のデータと異なり、農村周辺のデータには樹木だけでなく家屋・道路・橋梁など様々な人工物も含まれる。また樹木も圃場とは対照的に様々な種類・大きさのものが含まれている。圃場という管理された環境下で均一化された樹木の検出に比べ、この雑多なデータからの検出はより難しい問題だと考えらるため、提案手法のより実用的な利活用の検討例として相応しいと思われる。点群データには手動による各樹木の個体分割結果がラベリングされており比較のための正解データとして扱う。

5.2 オルソ画像における ROI の生成

まずは図 5.2 のようなオルソ画像上で ROI（樹木のバウンディングボックス）を生成する。4.2.2 章で提案したとおり 1 枚のオルソ画像を分割して検出モデルを学習させる。

- 元のサイズ：28117 × 22096
- 分割サイズ：4096 × 4096
- 分割のオーバーラップ：サイズの 50%
- 分割後の枚数：130（うち ROI を含む有効な画像は 81 枚）

分割した画像を入力に手作業による点群の個体分割を基にした ROI を教師データとして学習する。全 81 枚を 8:2 の比率で学習データ 61 枚・テストデータ 16 枚に分割し学習データの方は 5 倍の 325 枚に拡張した。学習の様子と結果が図 5.3 と表 5.1 である。最後に全 81 枚から推定された ROI を統合して図 5.4 のような結果を得た。



図 5.1: 農村周辺をドローンで撮影した画像から再構成した点群データ.

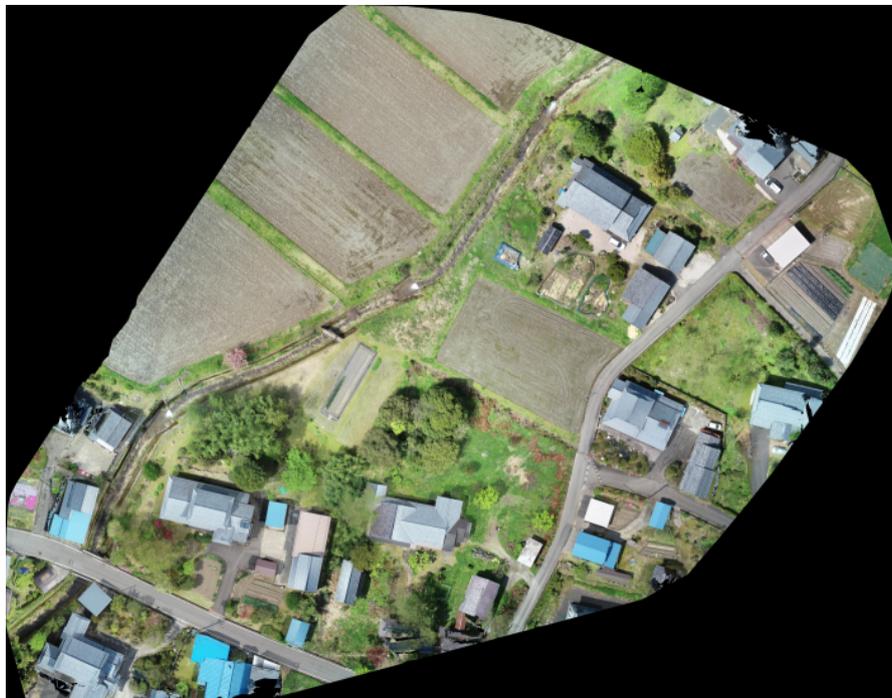


図 5.2: 農村周辺のオルソ画像.

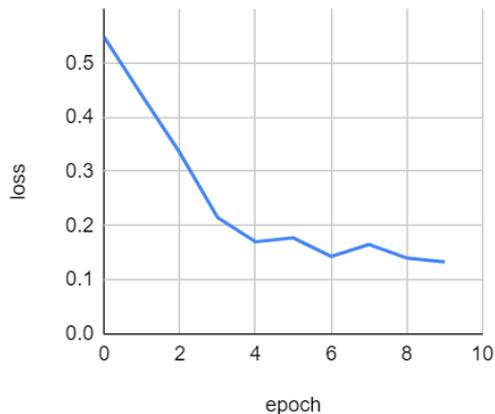


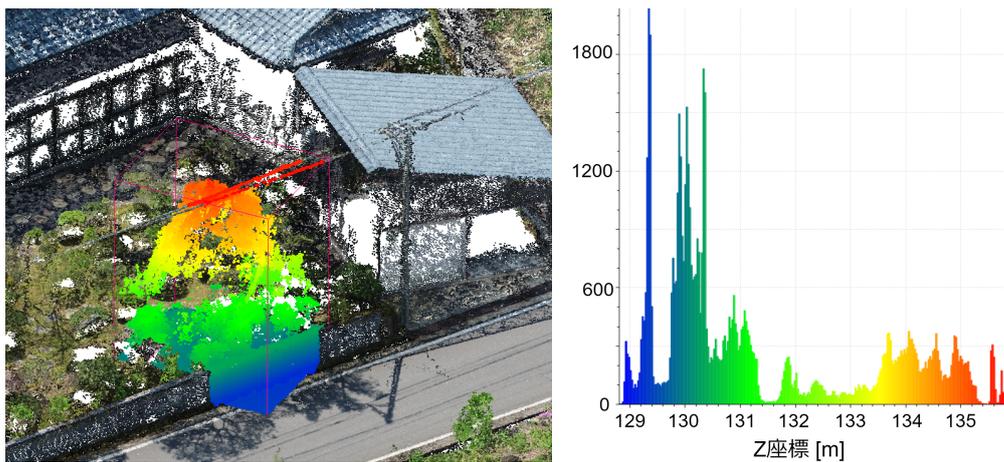
図 5.3: 学習中の Loss 関数の変化.

表 5.1: オルソ画像上での学習結果. 比較のため前 4.2 章で示した圃場データでの場合の値も載せる.

IoU	圃場		農村周辺	
	AP	AR	AP	AR
0.2	0.956	0.988	0.917	0.952
0.5	0.901	0.926	0.892	0.929
0.2:0.5	0.941	0.970	0.911	0.947

5.3 ドローン画像への ROI 投影

オルソ画像上の ROI を各ドローン画像上へ座標変換する. 基本的な処理は 4.2.3 章で説明したとおりであるが 1 点注意が必要である. オルソ画像上のピクセル座標から三次元空間中の Geo 座標へ変換するとき, 高さ方向 z 座標の値を外部から与える必要がある. 前 4.2 章の圃場データでは地面の高さを指定したが, これは圃場の地面が高さ一定と見なせると仮定できたためである. 一方で本章で扱う農村周辺では樹木が様々な場所に位置している. 圃場だけでなく家屋の庭先・道路脇・河川脇など場所は多様で各場所の高さ z の値を一定と見なすのは難しい. そこで今回は投影する各 ROI ごとに地面の高さを計測して z 座標を与える. 図 5.5 に示すように点群データの大部分は地面付近が最も密であるから, 投影対象の ROI (オルソ画像上のバウンディングボックスの 4 頂点) を Geo 座標へ変換して得られるポリゴンで切り取り, 領域内部の高さ z の値の平均値を計算して用いる. 投影された ROI の一部を図 5.6 に示す.

図 5.5: 高さ方向 z 座標の測定を点群データ中に描画した様子. 左: Geo 座標へ変換された ROI が図中の赤線で囲まれた領域となり, 領域内部の点群を高さ方向 z に応じて色付けしている. 右: 領域内部の点群の z 座標の分布.

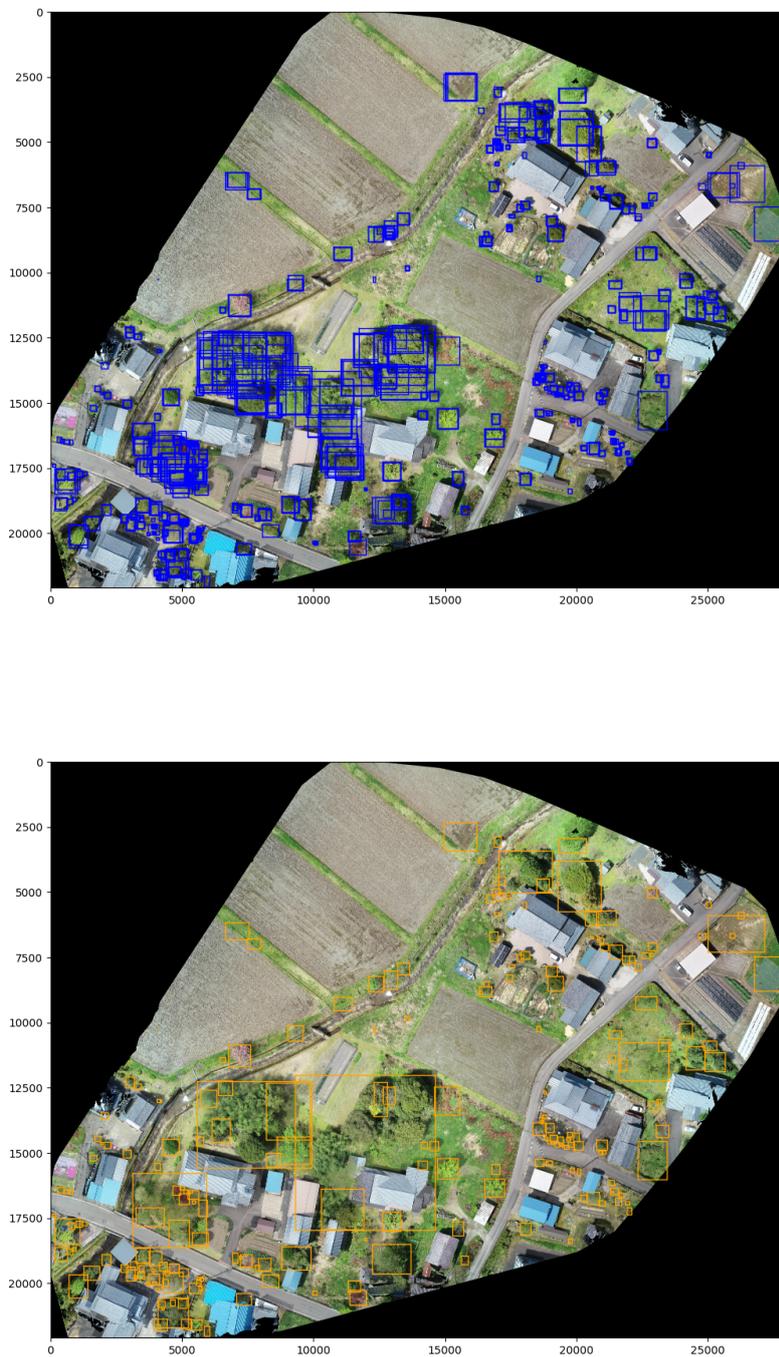


図 5.4: 農村周辺のオルソ画像上で推定された ROI (バウンディングボックス). 上: 分割された各画像で推定された ROI をすべて分割前のオルソ画像上に描画したもの. 下: 重複する ROI を合算して処理した ROI を描画したもの.



図 5.6: オルソ画像上の推定結果を元のドローン画像上へ座標変換した ROI (バウンディングボックス) を可視化した一例. 赤: 比較のため手動による正解データを基に座標変換した ROI, 黄: オルソ画像上の推定値を座標変換した ROI.

表 5.2: 利用する物体検出モデルと学習の詳細.

説明	値
学習モデル	FasterRCNN (pretrained)
データセット	画像 1189 枚
テストデータ	238 枚
学習データ	951 枚 (データ拡張なし)
学習時間	10 epoch

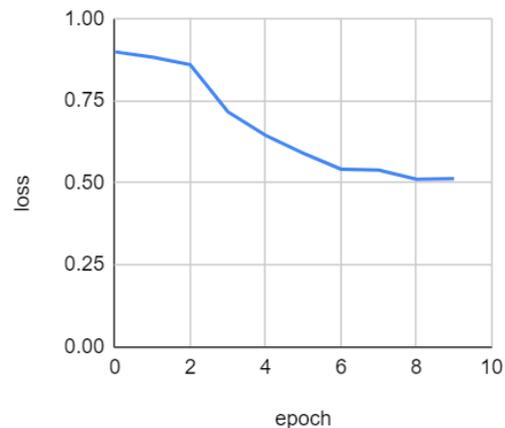


図 5.7: 学習中の Loss 関数の変化.

5.4 検出モデルの学習結果

最終的に各ドローン画像上の ROI を教師データとして学習する。学習の詳細は 4.1.5 章での実験と同様に行いデータ枚数は表 5.2 のとおり。学習の結果を図 5.7 と表 5.3 に示す。今回の放任果樹を調査する目的を考えると、樹木の有無と大まかな位置が検出できれば十分であり、検出位置のずれは大きな問題とならない。そこで推定された ROI とテストデータの ROI が一部でも重複を持てば一致と判定するような閾値 $\text{IoU} = 0.0$ での評価指標も追加した。また比較のため、いくつか別の手法による結果も示す。

- 圃場：前 4.2 章で示した圃場データに同様の提案手法を適用した結果
- 農村周辺（正解データ）：同一の農村周辺を測定したデータに対し、手作業で行った点群の個体分割の正解データを基に座標変換した ROI で学習した結果。オルソ画像上での ROI 推定を介さない分だけ投影される ROI の精度は高いと考えられる。
- DeepForest[8]：National Ecological Observation Network's Airborne Observation Platform [23] が収集する森林のデータセットに対し、樹冠の検出モデルを学習させた結果。Retinanet [25] を使用している・最終的な検出モデルをオルソ画像の入力に対し学習させている・手動によるラベリングも教師データとして使用しているなど提案手法と異なる点はあるが、図 5.8 のように LiDAR で測定した点群の個体分割を基に教師データを自動生成している点や、同じく自然環境中の樹木の検出を目的にしている点、使用する検出モデルの畳み込み層には ImageNet で学習済みの ResNet-50 を用いている点など類似点も多く比較に適切だと考え選択した。

画像から樹木の位置が検出できているか直感的に確認するため、モデルの推定値をドローンの画像上に描画したものを図 5.9 に示す。

表 5.3: オルソ画像から投影した ROI で学習させた結果と比較.

IoU	圃場		農村周辺		農村周辺 (正解データ)		DeepForest	
	AP	AR	AP	AR	AP	AR	AP	AR
0.0	0.962	0.984	0.936	0.999	0.969	1.000	-	-
0.1	0.962	0.972	0.704	0.860	0.862	0.951	-	-
0.2	0.962	0.983	0.658	0.813	0.840	0.937	-	-
0.5	0.939	0.963	0.498	0.645	0.727	0.832	0.61	0.69

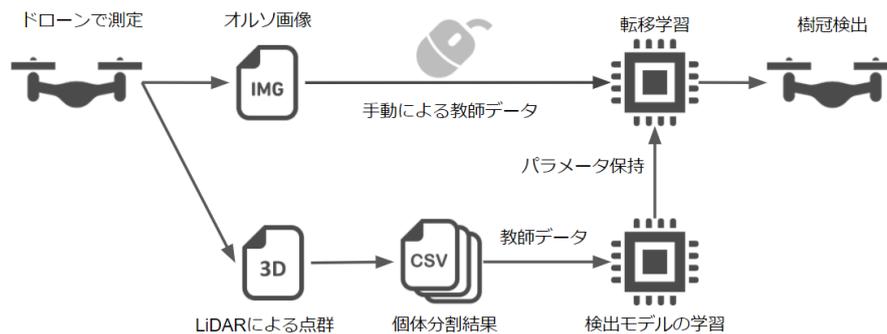


図 5.8: DeepForest の樹冠検出モデルの学習手法.

5.5 比較と検討

表 5.3 を見ると圃場データに比べ精度・感度ともに低い数字となっているが、農村周辺のデータは圃場とは異なり複雑で様々な種類や大きさの樹木を含んでいるため低い数字になるのは想定通りである。手作業による正解データを基にした教師データで学習した場合と比べると、同一データセットにもかかわらず精度・感度ともに低い数字であり ROI の自動生成に関して何か問題があったと考えるのが自然だが、詳しくは後述する。DeepForest の評価結果と比べると、両者ともに自然環境中のより複雑なデータを扱っており低い評価指標になっている。提案手法ではオルソ画像上で推定した ROI を投影して教師データを生成する点や、ドローンが直接撮影する RGB 画像を入力とする検出モデルを学習している点においてより実用的なアプリケーションにも応用可能だと言える。ただし両者ともにオルソ画像上で最低限の手動によるラベリング作業が必要であるため、完全な教師データ生成の自動化は達成できていない。

また図 4.15 で可視化したモデルの推定を見ると、多くの樹木の位置を正しく検出できているが気になる ROI も多い。

- FP: False Positive, テストデータには含まれないが推定された ROI
(単独で位置する青色のバウンディングボックス)
- FN: False Negative, テストデータとして与えたものの推定されていない ROI
(単独で位置する黄色のバウンディングボックス)

FP については扱うデータセットの性質に依るところが大きいと考えられる。農村周辺には種類も大きさも多種多様な樹木が散見されるため 5.2 章ですべての樹木を ROI として拾い切れていない。FP に分類される推定 ROI を可視化して見ても樹木の位置として間違っていないと判断できる

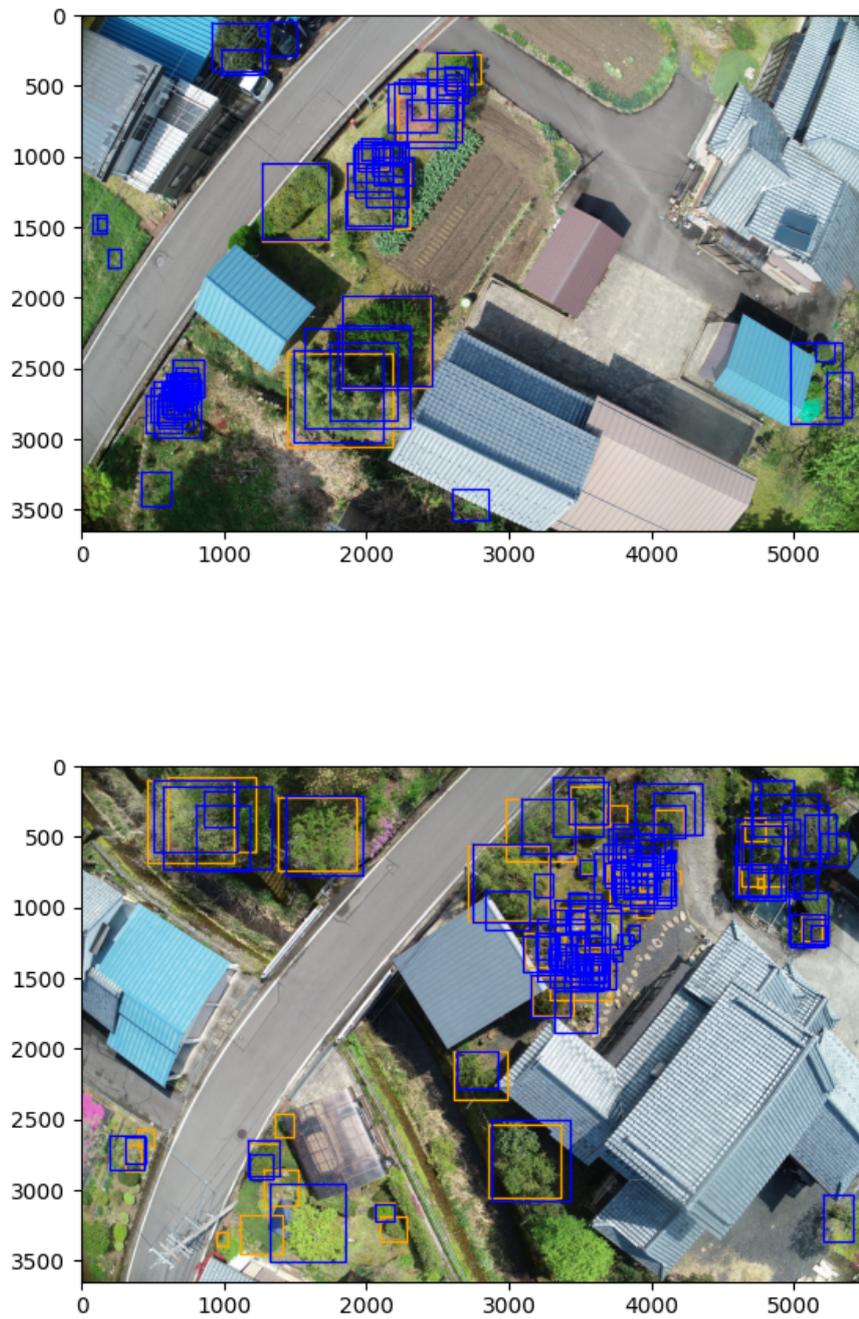


図 5.9: オルソ画像から投影した ROI で学習したモデルの推定値を描画. 黄: テストデータ ROI, 青: 推定された ROI.

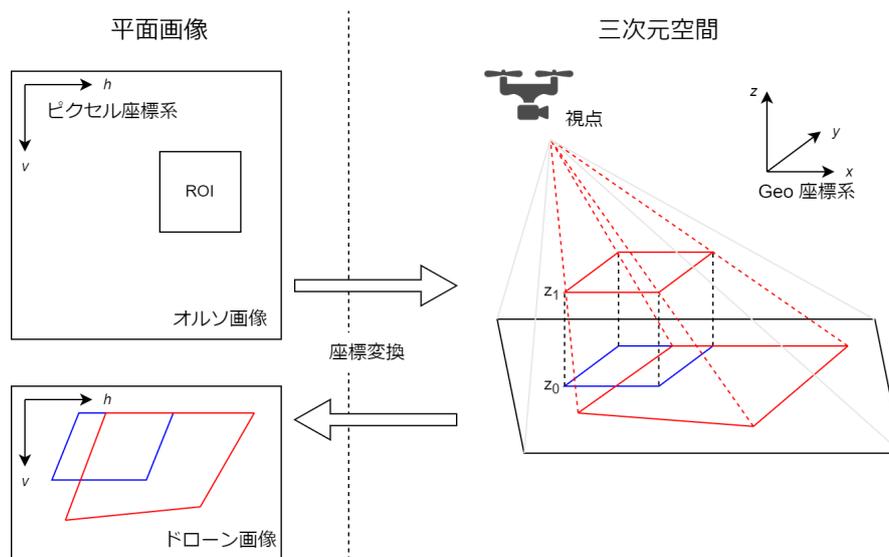


図 5.10: ROI の座標変換過程における高さ z の差による投影位置のずれ. 鉛直下を向くカメラ視線軸から離れている部分のみ示す. 高さの違い z_0, z_1 により同一の ROI がドローンの画像上では異なる位置に投影される.

ケースが多かった. また放任果樹の調査という最終的な目的に照らし合わせれば, 精度が多少低くとも樹木らしき物体の位置が分かれば調査の手助けとして十分に機能するため, FP に当たる ROI はそこまで大きな問題ではないと考えられる.

一方で FN について本来は検出して欲しい樹木の位置をモデルが推定できなかったと意味する. 調査したい放任果樹は圃場中よりは柿など庭先に単独で位置するケースが多いため, 調査対象を見逃さない点は重要であり, モデルの感度を低下させる FN の存在は大きな問題である. 原因はモデルの学習に用いた ROI の精度の低さであろうが, 特に 5.3 章で行った座標変換に問題があると考えている. 4.2.3 章でも言及したとおり, 座標変換ではオルソ画像上で検出した ROI を各ドローンの画像上へ投影する過程で三次元空間中の高さ z 座標の値を補間する必要がある. 提案手法では地面の高さを測定して補間したが, 果たして地面の高さが妥当かどうかは定かでない. 図 5.10 のようにカメラの視線軸 (ドローンのカメラは鉛直下方向に向けたため画像の中心付近に該当) から離れる画像の外周部ほど, 高さ z の値による投影のずれが現れる. 表 5.3 で比較した手作業による点群の個体分割結果を基に座標変換した場合は, 点群が高さ z の値を持つため補間は不要であり, ROI の自動生成に投影のずれの問題は発生せず, その結果検出モデルの評価結果に大きな差が生じたと考えられる.

加えてカメラの歪みが大きいとこの投影のずれはさらに大きくなるが, 圃場の撮影と農村周辺の撮影には異なるドローンとカメラを使用しており, カメラが有する歪みの影響も異なる. 図 5.4 で例示したドローンの画像を見ると, 本来真っすぐなはずの道路が曲がって写っているとおり, 農村周辺のドローン画像の方が大きな歪みを持っているのが分かる. そのため座標変換された ROI (図中の黄色) も歪みが大きい画像外周部でずれが大きくなっている. そのため圃場データへ上手く適用できた手法も農村周辺のデータへの適用には問題が発生したと考えられる. 問題を解決するにはカメラの歪みも考慮した ROI の座標変換手法の検討が必要であり, そのためにはカメラの歪みを定量的に表したカメラ関数 (内部パラメータ) の利用が考えられる.

■ 第6章

おわりに

6.1 議論

3章では樹木の点群から各個体を分割する手法を提案し、実験でその分割性能を評価した。樹形に注目して各個体の主幹位置を検出し、点群のクラスタリングの探索始点と利用することで各個体を検出し分割できる。個体ごとにクラスタリングするとき、点群の局所的な主成分分析で幹や枝が伸びる方向を推定し近傍探索の条件に加えたため、異なる個体の点群が混在する空間内でも適切に分割できた。しかし実験で用いた点群データは圃場で管理された樹木を測定したものであり、主幹位置の検出には全個体が水平な同一平面と見なせる地面から生えている仮定を利用している。提案手法の適用できるデータの性質には制限があるものの、圃場のような特定の環境下では優れた分割性能を実現できた。

次の4章ではドローンと深層機械学習による物体検出のセンシング技術に注目し、学習に必要な大量の教師データをより容易に短時間で生成する手法を提案した。まず前3章で提案した点群の個体分割を基に教師データを自動生成する手法を4.1章で示した。ドローンで撮影した画像から三次元再構成を行い測定範囲全体を点群データで表現し、樹木1本ずつ個体分割する。分割された点群を三次元座標から各ドローン画像上のピクセル座標へ変換し、得られたROIを教師データとして検出モデルを学習させた。点群の個体分割・座標変換によるROI生成ともに手作業を必要とせず教師データを完全に自動生成できる。しかしこの手法は点群の分割精度に大きく依存しており、個体分割手法の適用限界がそのまま課せられる。既に議論したとおり、3章で示した個体分割手法は圃場という管理された環境下の点群データを想定するため、適用できるデータセットには限りがある。ゆえに点群の個体分割を基にするこのROI生成手法は、5章で扱うより実用的なデータセットには適用できなかった。

また点群の個体分割とは別に、オルソ画像を利用した教師データの生成手法も4.2章で提案した。ドローンが撮影した画像を合成して測定範囲全体を1枚の画像で表現したオルソ画像には位置ずれや歪みを取り除かれる。一旦このオルソ画像上で樹木の位置をラベリングして、オルソ画像上のROIを各ドローン画像上のピクセル座標へ変換することで教師データを生成し、検出モデルを学習させる。オルソ画像上のラベリングをすべて手作業で行うのは煩雑であるから、適宜分割したオルソ画像でも検出モデルを学習させ、結果としてオルソ画像上・ドローン画像上のふたつの検出モデルを学習させることになる。ただし分割したオルソ画像上での学習にはどうしても教師データを外部から与える必要があり完全な自動化は難しいものの、ドローンが撮影した大量の画像をラベリングする労力に比べれば大幅な作業軽減を実現できた。また点群の個体分割を基にしたROI生成手法と比べて適用できるデータセットは幅広く、5章ではより実用的なデータセットとして農村周辺を撮影した雑多な画像データに適用して提案手法の性能を評価した。

提案手法の評価実験には圃場をドローンで撮影したデータを利用した。圃場内では樹木が管理・育成されているため樹木の種類・大きさは比較的均一であり検出が容易な例と言える。しかし獣害対策としての放任果樹の自動調査など最終的な目的に照らすと、より複雑で検出の難しいデータが想定される。そこで5章ではより実用的なデータセットとして農村周辺をドローンで撮影したデータに提案手法を適用して評価を行った。圃場とは異なり樹木の種類・大きさは幅広く人工物も含め非常に雑多な画像データであり、当然得られた検出モデルの精度は低い数字になった。考えられるいくつかの原因の中でも、特に撮影に用いたカメラに依存する画像の歪みの影響が大きいと考えられる。座標変換によりROIが自動生成される過程で、ROIの三次元空間中の位置を決定する必要があるが、図5.10のように高さ方向 z の曖昧さが投影されたドローン画像上で位置ずれとして現れ

る。このずれはカメラの歪みが大きく画像の外周部ほど影響が大きく、学習に使用する ROI の精度が低下する。

4.2章で提案した手法では、オルソ画像上で検出した ROI をドローン画像上へ座標変換する過程でのカメラの歪みの影響を正しく考慮する必要があると新たに分かった。カメラ関数（内部パラメータ）を利用して歪みを定量的に扱う改善が考えれる。また本稿では一貫して樹木を検出するモデルのみ扱っており、樹木の種類の識別は扱っていない。ただ5章で想定している放任果樹の自動調査など、最終的な目的を考えれば樹木の検出だけでなく、樹種を判別して特定の樹木のみを選択的に検出できる方が望ましい。提案手法により学習させた検出モデルを基に、転移学習で識別タスクも追加したモデルの学習が新たに期待される。

6.2 結論

本稿では立体的な構造解析が可能になる三次元点群データに注目し、ドローンによる植物センシングを農業生産に利用するとき問題となる対象物の点群の検出と分割に対し解決手法を提案し、実験を行い評価した。加えて三次元点群の個体分割の応用やオルソ画像の利用により、深層機械学習による物体検出で問題となる教師データの生成を自動化する手法を提案した。提案手法を圃場のデータセットに適用して評価を行い十分な検出精度を確認した。さらに実用的な例として農村周辺の複雑なデータセットにも適用し、提案手法の応用可能性を示した。一方で6.1章で議論したとおり解決すべき問題は依然残っており、さらなる改良が望まれる。

謝辞

まずはじめに、本研究や修士論文の執筆に際し、熱心に面倒を見てくださった川原圭博教授に深く感謝致します。毎週のミーティングや壁にぶつかったタイミングなどでの確かなアドバイスをくださり、さらに、研究方針や進め方に関して自主性を重んじていただきました。お陰様で、卒業論文を含めたこの3年間、楽しみながらも挑戦しがいのある日々を過ごすことができました。さらに、常に新しい分野に挑戦なさる先生の姿勢には大変刺激を受けました。

また、研究室の教職員・同期を含めた学生の皆様・職員の皆様には、日頃の議論から日常生活に至るまで本当にお世話になりました。お陰様で楽しい研究室生活を送ることができました。心からお礼申し上げます。最後に、精神的に大きな支えであった家族への感謝を述べ、本論文の結びとさせていただきます。

参考文献

- [1] H. Temesgen, D. Affleck, K. Poudel, A. Gray, and J. Sessions, "A review of the challenges and opportunities in estimating above ground forest biomass using tree-level models," *Scandinavian Journal of Forest Research*, vol.30, pp.326–335, 2015.
- [2] K. Calders, G. Newnham, A. Burt, S. Murphy, P. Raunonen, M. Herold, D. Culvenor, V. Avitabile, M. Disney, J. Armston, and M. Kaasalainen, "Nondestructive estimates of above-ground biomass using terrestrial laser scanning," *Methods in Ecology and Evolution*, vol.6, no.2, pp.198–208, 2015.
- [3] J. Gonzalez de Tanago, A. Lau, H. Bartholomeus, M. Herold, V. Avitabile, P. Raunonen, C. Martius, R.C. Goodman, M. Disney, S. Manuri, A. Burt, and K. Calders, "Estimation of above-ground biomass of large tropical trees with terrestrial LiDAR," *Methods in Ecology and Evolution*, vol.9, no.2, pp.223–234, 2018.
- [4] D. Gómez-Candón, N. Virlet, S. Labbé, A. Jolivot, and J.L. Regnard, "Field phenotyping of water stress at tree scale by UAV-sensed imagery: new insights for thermal acquisition and calibration," *Precision Agriculture*, vol.17, no.6, pp.786–800, 2016.
- [5] A.N. Atay, E. Atay, P.E. Lauri, B. Kunter, and K.Y. Kantoglu, "Phenotyping gamma-ray-induced mutant population of ' amasya ' apple for architectural traits, precocity, floral phenology and fruit characteristics," *Scientia Horticulturae*, vol.233, pp.195–203, 2018.
- [6] Y. Jiang, C. Li, F. Takeda, E.A. Kramer, H. Ashrafi, and J. Hunter, "3D point cloud data to quantitatively characterize size and shape of shrub crops," *Horticulture Research*, vol.6, no.43, 2019.
- [7] S. Paulus, "Measuring crops in 3D: using geometry for plant phenotyping," *Plant Methods*, vol.15, no.103, 2019.
- [8] B.G. Weinstein, S. Marconi, S. Bohlman, A. Zare, and E. White, "Individual tree-crown detection in rgb imagery using semi-supervised deep learning neural networks," *Remote Sensing*, vol.11, no.11, 2019.
- [9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.39, no.6, pp.1137–1149, 2017.
- [10] X. Liang, V. Kankare, J. Hyypä, Y. Wang, A. Kukko, H. Haggrén, X. Yu, H. Kaartinen, A. Jaakkola, F. Guan, M. Holopainen, and M. Vastaranta, "Terrestrial laser scanning in forest inventories," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol.115, pp.63–77, 2016.
- [11] V. Kankare, E. Puttonen, M. Holopainen, and J. Hyypä, "The effect of tls point cloud sampling on tree detection and diameter measurement accuracy," *Remote Sensing Letters*, vol.7, no.5, pp.495–502, 2016.

- [12] M. Dassot, T. Constant, and M. Fournier, "The use of terrestrial LiDAR technology in forest science: application fields, benefits and challenges," *Annals of Forest Science*, vol.68, no.5, pp.959–974, 2011.
- [13] D. Mader, R. Blaskow, P. Westfeld, and H.G. Maas, "UAV-based acquisition of 3D point cloud – a comparison of a low-cost laser scanner and SfM-tools," *ISPRS*, vol.XL-3/W3, pp.335–341, 2015.
- [14] F. Hui, J. Zhu, P. Hu, L. Meng, B. Zhu, Y. Guo, B. Li, and Y. Ma, "Image-based dynamic quantification and high-accuracy 3D evaluation of canopy structure of plant populations," *Annals of Botany*, vol.121, no.5, pp.1079–1088, 2018.
- [15] K. Wang, J. Zhou, W. Zhang, and B. Zhang, "Mobile LiDAR scanning system combined with canopy morphology extracting methods for tree crown parameters evaluation in orchards," *Sensors*, vol.21, no.2, 2021.
- [16] J. Sander, "Density-based clustering," in *Encyclopedia of Machine Learning*, eds. C. Sammut, and G.I. Webb, pp.270–273, Springer US, 2010.
- [17] E. Ayrey, S. Fraver, J.A.K. Jr., L.S. Kenefic, D. Hayes, A.R. Weiskittel, and B.E. Roth, "Layer stacking: A novel algorithm for individual forest tree segmentation from LiDAR point clouds," *Canadian Journal of Remote Sensing*, vol.43, no.1, pp.16–27, 2017.
- [18] F. Westling, J. Underwood, and M. Bryson, "Graph-based methods for analyzing orchard tree structure using noisy point cloud data," *Computers and Electronics in Agriculture*, vol.187, 2021.
- [19] R.Q. Charles, H. Su, M. Kaichun, and L.J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," *CVPR - Computer Vision and Pattern Recognition*, pp.77–85, 2017.
- [20] S. Jin, Y. Su, S. Gao, F. Wu, Q. Ma, K. Xu, Q. Ma, T. Hu, J. Liu, S. Pang, H. Guan, J. Zhang, and Q. Guo, "Separating the structural components of maize for field phenotyping using terrestrial LiDAR data and deep convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol.58, no.4, pp.2644–2658, 2020.
- [21] S. Jin, Y. Su, S. Gao, F. Wu, T. Hu, J. Liu, W. Li, D. Wang, S. Chen, Y. Jiang, S. Pang, and Q. Guo, "Deep learning: Individual maize segmentation from terrestrial lidar data using Faster R-CNN and regional growth algorithms," *Frontiers in plant science*, vol.9, 2018.
- [22] J. Redmon, and A. Farhadi, "Yolo9000: Better, faster, stronger," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- [23] B.G. Weinstein, S. Marconi, S. Bohlman, A. Zare, A. Singh, S.J. Graves, and E. White, "NEON Crowns: a remote sensing derived dataset of 100 million individual tree crowns," *bioRxiv*, 2020.
- [24] J.R. Roussel, D. Auty, F.D. Boissieu, A.S. Meador, B.J. François, G. Demetrios, L. Steinmeier, and S. Adaszewski, "lidR: Airborne LiDAR data manipulation and visualization for forestry applications," <https://rdr.io/cran/lidR/>.
- [25] T.Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
- [26] J. Xiong, Z. Liu, S. Chen, B. Liu, Z. Zheng, Z. Zhong, Z. Yang, and H. Peng, "Visual detection of green mangoes by an unmanned aerial vehicle in orchards based on a deep learning method," *Biosystems Engineering*, vol.194, pp.261–272, 2020.

- [27] O. Csillik, J. Cherbini, R. Johnson, A. Lyons, and M. Kelly, "Identification of citrus trees from unmanned aerial vehicle imagery using convolutional neural networks," *Drones*, vol.2, no.4, 2018.
- [28] L.P. Osco, M. dos Santos de Arruda, J. Marcato Junior, N.B. da Silva, A.P.M. Ramos, Érika Akemi Saito Moryia, N.N. Imai, D.R. Pereira, J.E. Creste, E.T. Matsubara, J. Li, and W.N. Gonçalves, "A convolutional neural network approach for counting and geolocating citrus-trees in UAV multispectral imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol.160, pp.97–106, 2020.
- [29] R.B. Rusu, and S. Cousins, "3D is here: Point Cloud Library (PCL)," *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.
- [30] C. Wang, M. Ji, J. Wang, W. Wen, T. Li, and Y. Sun, "An improved dbSCAN method for LiDAR data segmentation with automatic eps estimation," *Sensors*, vol.19, no.1, 2019.
- [31] A.W. Hoover, G. Jean-Baptiste, X. Jiang, P.J. Flynn, H. Bunke, D.B. Goldgof, K.W. Bowyer, D.W. Eggert, A.W. Fitzgibbon, and R.B. Fisher, "An experimental comparison of range image segmentation algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.18, no.7, pp.673–689, 1996.
- [32] H. Wang, Y. Duan, Y. Shi, Y. Kato, S. Ninomiya, and W. Guo, "EasyIDP: A Python package for intermediate data processing in UAV-based plant phenotyping," *Remote Sensing*, vol.13, no.13, 2021.
- [33] T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C.L. Zitnick, "Microsoft COCO: Common Objects in Context," *Computer Vision – ECCV 2014*, eds. D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, pp.740–755, Cham, 2014, Springer International Publishing.
- [34] COCO Consortium, "COCO Common Objects in Context – detection evaluation," <https://cocodataset.org/#detection-eval>, 2015.
- [35] 武山絵美, 九鬼康彰, 松村広太, 三宅康成, "山間農業集落における水田団地への有害獣侵入経路," *農業土木学会論文集*, vol.2006, no.241, pp.59-65, 2006.
- [36] 山端直人, "獣害と農村のマネジメント," *農村計画学会誌*, vol.34, no.3, pp.357-360, 2015.
- [37] 横山真弓, 木下裕美子, "捕獲個体の分析～年齢・繁殖・食性," *兵庫県におけるアライグマの現状*, 第1巻, 第3章, pp.19–28, 兵庫県森林動物研究センター, 2009.

■ 発表文献および受賞

本論文に関する発表文献

- [P1] 仙田薫, 繁田亮, 郭威, 川原圭博, “三次元点群を拡張したボクセル空間における果樹の生育特徴量の抽出方法の検討”. 電子情報通信学会技術研究報告, vol.119, no.226, pp.145–150, 2019.
- [P2] 仙田薫, 繁田亮, 郭威, 川原圭博, “果樹の生育把握に向けた三次元点群からの樹幹構造に基づく個体分割方法”. 情報処理学会全国大会, no.1, pp.325–326, 2020.
- [P3] 仙田薫, 繁田亮, 郭威, 川原圭博, “樹木の立体構造把握のための樹形に基づく点群の個体分割手法”. マルチメディア, 分散, 協調とモバイルシンポジウム (DICOMO 2020), pp.1223–1229, 2020.
- [P4] 仙田薫, 坪内孝太, 郭威, 川原圭博, “ドローンを利用した放任果樹の自動調査に向けた物体検出モデルの教師データ半自動生成手法”. 電子情報通信学会総合大会, 2022 (予定).

受賞

- [A1] SeMI 研究会 2019 年度優秀発表賞
仙田薫, 繁田亮, 郭威, 川原圭博, “三次元点群を拡張したボクセル空間における果樹の生育特徴量の抽出方法の検討”. 電子情報通信学会技術研究報告, vol.119, no.226, pp.145–150, 2019.
- [A2] 学生奨励賞
仙田薫, 繁田亮, 郭威, 川原圭博, “果樹の生育把握に向けた三次元点群からの樹幹構造に基づく個体分割方法”. 情報処理学会全国大会, no.1, pp.325–326, 2020.
- [A3] 大会優秀賞
仙田薫, 繁田亮, 郭威, 川原圭博, “果樹の生育把握に向けた三次元点群からの樹幹構造に基づく個体分割方法”. 情報処理学会全国大会, no.1, pp.325–326, 2020.