

メタ AR 空間構築のための汎用タブレット端末で 実行可能な 3 次元点群差分検出手法について

安江 洸希^{1,a)} 菊地 真人^{1,b)} 大園 忠親^{1,c)}

概要: AR 空間と VR 空間との間で双方向でのインタラクションを可能とする 3 次元点群で構成されたメタ AR 空間の構築において、限られたメモリ使用量の中で特定の点群を現実的な時間で検出することに関して課題がある。本研究の目的は、汎用タブレット端末で実行可能な 3 次元点群の差分による特定の点群検出を現実的な時間で行うための手法を開発することである。本稿では、3 次元点群差分検出手法において、既存手法である RGB 画像を用いた対応点の位置合わせによる変換パラメータの推定における問題点を AR 技術を用いて解決し、差分検出した手法について述べる。

キーワード: 拡張現実感技術 AR, 3 次元点群, 位置合わせ, 物体検出

1. はじめに

近年、現実空間上の様々なデータをデジタル空間上に再現することは活用できる幅が広いことから需要が高まっている。例えば、何らかの実験を行う際に、その場所の環境を忠実に再現することで現実空間上ではなくデジタル空間上でシミュレーションを行うことが可能となる。これによって今までは手間がかかっていたような作業をより手軽に行うことが期待できる。

本研究では、先行研究として AR 技術を用いて現実空間を 3 次元点群として構築したメタ AR 空間を開発している [1]。メタ AR 空間は、iPad Pro に搭載された LiDAR スキャナを用いて現実空間を計測し、現実空間の物体表面を 3 次元点群として取得 (3D スキャン) することで構築しており、現実空間の形状を忠実に再現した空間となっている。メタ AR 空間を用いることで、AR 空間上で行う視覚的な AR オブジェクトの操作と同様の操作を場所に縛られず、どこからでも行うことができる。これによって、現地 (AR 空間) および遠隔地 (メタ AR 空間) との間で AR オブジェクトを介した意思疎通を図り、双方向でのインタラクションを支援することが可能となる。しかし、現状のメタ AR 空間は現実空間との連動性が低く現実空間の状態が分かりにくいという問題があった。ここでの連動性とは、現実空間またはメタ AR 空間上で物体を動かした際にもう

片方の空間でも同様に動作することを指す。現実空間との連動性を高めることは、双方向でのインタラクションを行う上で重要な要素であると考えられる。そこで、本研究の目的は現実空間と高い連動性を持つような空間としてメタ AR 空間を構築することである。そのために、まずは現実空間上の物体を椅子やドアといった動くことが想定される物体 (可動物) とそれ以外の動くことが想定されない物体 (固定物) の 2 つに分け、可動物部分を連動させることができる空間としてメタ AR 空間の構築を目指す。可動物はユーザが定義できるとする。

可動物部分を連動させた空間としてメタ AR 空間を構築するためにはメタ AR 空間上で可動物を検出する必要がある。本研究では、このための手法として 3 次元点群の差分による可動物の検出を行った。本稿では、この手法について詳しく述べていく。また、本研究で使用する AR 技術は、Apple 社が提供する ARKit を用いる。

2. 可動物検出

2.1 可動物検出手法の概要

3 次元点群の差分による可動物検出手法の概要を図 1 に示す。必要なものは、3D スキャンして得られた点群 A と A とは可動物の位置が異なる点群 B である。本研究で扱う点群は、数十万~数百万個の (x, y, z) の集合で構成された順不同で点群間の対応関係がわかっていない点で構成されている。差分化を行うためには、異なる点群間で対応する点同士を削除する必要があるが、対応関係がわからないため初めに位置合わせが必要となる。これによって対応

¹ 名古屋工業大学大学院 情報工学系プログラム

^{a)} ankou@ozlab.org

^{b)} kikuchi@ozlab.org

^{c)} ozono@ozlab.org

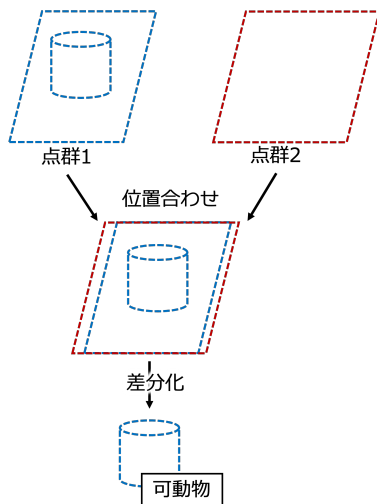


図 1: 可動物検出手法の概要

しているとみなした点を近い位置に持ってくる事ができるため、点同士の対応づけを容易に行うことが可能となる。対応づけができたなら、対応している点（対応しているとみなした点）を削除し可動物に該当する点を検出する。この際に図 1 のように固定物に該当する点が差分として出る可能性がある。これは、3D スキャン時の誤差や撮影範囲の違いなど、可動物以外の差分を含むためである。そこで点群 A 中の可動物を検出したい場合、差分を行う点群 B は点群 A 全体をオーバーラップするように 3D スキャンを行うようにする。図 1 の場合、点群 1 に可動物が含まれているため、点群 2 は点群 1 全体を覆うようにしてスキャンを行う。これによって差分化を行なった際に点群 1 と点群 2 の対応する点と点群 2 を削除することで、残る点は点群 1 の可動物の部分だけになる。このようにして 3 次元点群を差分化して可動物の検出を行う。

2.2 従来の位置合わせ手法と問題点

点群の位置合わせには、一般的に Iterative Closest Point (ICP) アルゴリズムが用いられる [2]。ICP アルゴリズムは、対応点の探索と位置合わせのための変換パラメータ（回転および並進）を推定する手法である。点群中の全ての点を扱うため計算時間はかかるが、位置合わせの精度は高い。予備調査において、iPad Pro 第 3 世代（メモリ容量：7.3GB）ではメモリ不足で実行することができなかった。また、十分なメモリを積んだ環境においても、数十分以上を要するため現実的な時間内での処理が難しい。位置合わせを高速に行うためには、点群のダウンサンプリングやグループ化等により、処理すべき点の個数を減らすことが有効である [3]。位置合わせの精度は、ICP に使用する対応点の誤差に大きく依存する [4]。そのため、適切な点を選ぶことで少ない点でも精度を保つことが可能となる。

ICP アルゴリズムの高速化手法の 1 つに画像を用いた

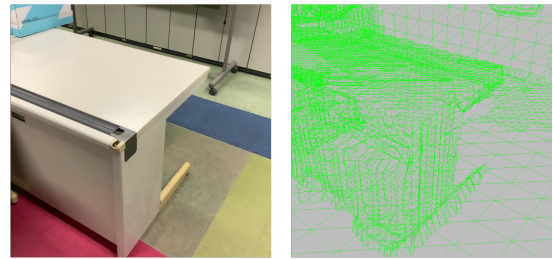


図 2: 現実空間 (左) とその 3D メッシュ (右)

手法が提案されている [5]。これは、位置合わせを行いたい点群と同時に取得した画像から位置合わせのための 2 次元座標である対応点を求め、この点を 3 次元座標に変換して ICP アルゴリズムを行うことで、少ない点から変換パラメータを推定する手法となっている。画像からの対応点は、既存の 2 次元画像からの特徴マッチング手法を用いることで精度の高い対応点を得ることができるため、点群自体の位置合わせの精度を保つことができる。

しかし、この手法は画像から取得した 2 次元座標である対応点を 3 次元座標に変換する際の変換方法に問題がある。通常、2 次元から 3 次元への変換を行うためには深度情報を用いて行うが、汎用的なデバイス上で取得できる深度情報には精度における限界があることや 2 次元空間上で隣接していたピクセルが 3 次元空間上でも隣接しているとは限らないといったことから、3 次元座標に変換を行なった際の変換精度にばらつきが発生する。ここで扱う点の数は少ないため、少しの対応点の誤差が位置合わせを行う際の精度に大きく関係する。そのため、上手く変換を行うことができない場合が発生する。

2.3 提案手法

本研究では、2 次元座標から 3 次元座標への変換方法として AR 技術を利用することで上記の問題を解決する。使用する技術としては以下の 2 つである。

1. デバイスの自己位置検出
2. 現実空間のスキャンによる 3D メッシュの構築

1 は、スキャン中に画像を取得した位置と向きを保存するために用いる。2 の 3D メッシュは、現実空間のスキャン中に収集したデータを用いてピクセル単位で構築しており、スキャン中に最適化され続けるため、精度の高い 3 次元情報として構成される。図 2 は、左の現実空間を 3D メッシュとして構築した例である。これによって、現実世界の形状を可視化することが可能となる。点群を使わなくても、3D メッシュ内には 3 次元情報があるため、これを用いることでも可動物の検出ができそうであるが、本手法で用いている ARKit では、3D メッシュの座標情報を取得できないという制限がある。また、扱えたとしても 3D メッシュは現実空間の細かい形状を再現することが難しいことから、点群を用いる方がより精密な処理を行う上で重要と

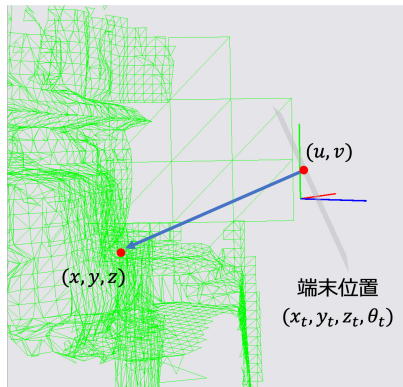


図 3: 3D メッシュによる 3 次元変換の概要

いえる。

3D メッシュを用いた 3 次元変換の概要を図 3 に示す。まず初めに対応点を取得するために使用する画像の位置と向きを復元する。次に画像上の対応点である 2 次元座標 (u, v) から、画像と垂直に伸ばし 3D メッシュと交わった 3 次元座標 (x, y, z) を 2 次元座標 (u, v) を 3 次元変換した座標として使用する。

2.4 差分化

3 次元点群同士の差分化では、2 つの点群を同じボクセル空間に分割し点群間の点を対応づけることで、各ボクセル内に存在する点群の有無を比較し、差分対象の点を把握する手法が広く用いられている [6]。本研究では、VoxelGrid filter^{*1}を用いて点群をボクセル化した。これは点群をダウンサンプリングするためのアルゴリズムで、ボクセル化するためのグリッドを設定し、各格子の中にある点の重心を新しい点として扱うようなアルゴリズムとなっている。本手法では、ボクセル内の点の有無を比較することができれば良いので、格子の中にある点の重心ではなく格子の中心位置を新しい点として扱うようにしている。図 4 は位置合わせ後の点群を VoxelGrid filter に通してボクセル化した時の例である。VoxelGrid filter を用いることでボクセルの状態は以下の 2 つのどちらかとなる。

- ①. 異なる点群の点が含まれている
- ②③. どちらかの点群の点しか含まれていない

これによって点群間の点の対応づけをボクセル内の点の有無を比較することで容易に確認することが可能となる。差分化では、可動物を残すための点群として設定した方の点だけが含まれているボクセルだけを残すことで可動物に該当する点を検出する。図 4 の青色の点群は、可動物の点群を含み、赤色の点群は、可動物の点群を含まない点群とすると、差分化時に①, ②を削除し、③だけを残せば良いことになる。図 5 に本手法を用いての可動物検出の実行例を示す。入力として (a) の 2 つの点群を与えると差分として (b) のような差分部分の点群を得ることができる。

^{*1} <http://pointclouds.org/documentation/tutorials/voxel>

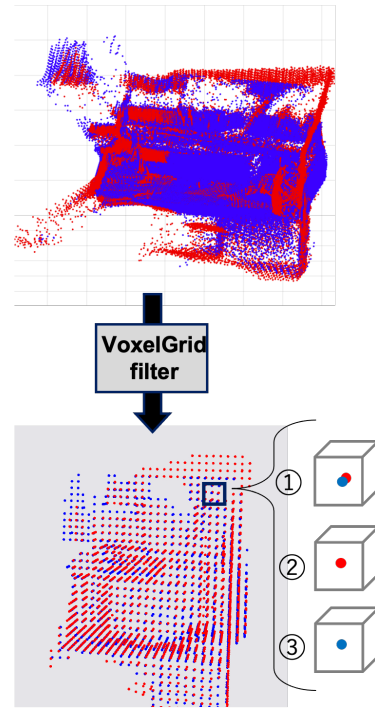
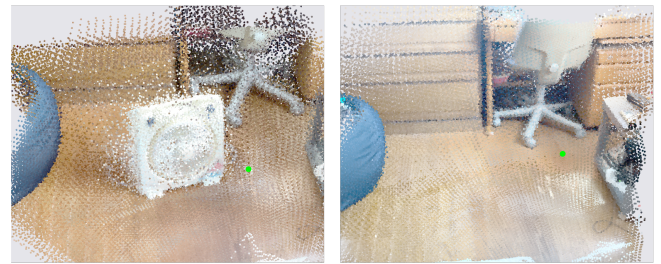


図 4: 位置合わせ後の点群 (上) とボクセル化した点群 (下)



(a) 入力点群：可動物有りの点群 (左), 可動物無しの点群 (右)



(b) 出力点群：検出した可動物部分の点群

図 5: 可動物検出の実行例

3. 実験

3.1 予備実験

本手法では、AKAZE と呼ばれる特徴点マッチング手法を用いて対応点の探索を行う。画像間で取得する対応点を用いて位置合わせを行うため、取得する対応点の数によって位置合わせの精度と時間にどのように影響するかを調べるための予備実験を行う。これによって、画像から取得する適切な対応点の数を見つける。既存手法 [5] では SIFT 特徴点を用いていたが、AKAZE は SIFT や他の手法と比較して処理速度や頑健性の面で優れている [7] ことから本

表 1: 点の数に応じた処理時間と繰り返し回数の平均値

対応点の数	処理時間 [s]	繰り返し回数
10	1.39	3.2
30	1.69	3.8
50	1.71	3.6
100	2.09	4.2
300	6.51	5.8
500	16.00	6.6

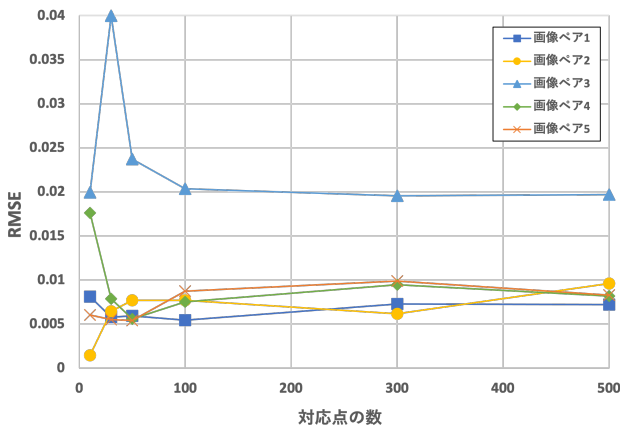


図 6: 対応点の数と RSME

手法では AKAZE を用いた。実験内容としては、画像ペアを 5 種類用意し、本手法で画像から取得する対応点の数を変えて点群の位置合わせまで行った際の処理時間の平均、ICP の繰り返し処理の回数の平均、位置合わせの精度を求め、精度に関しては Root Mean Square Error (RMSE) を採用する。

実験結果を表 1 と図 6 に示す。表 1 より対応点の数が増えるに従って処理時間と繰り返し処理の回数は増加している。50 点以降から繰り返し処理の回数の増加の割合が高くなり、それに応じて処理時間の増加の割合も高くなっている。また、図 6 より対応点の数が 50 点より少ない場合、RSME の値にばらつきがある。このばらつきは 50 点以降で段々と安定しており、100 点と 500 点の間での RSME の変化は少ない。本システムでは、表 1 と図 6 の結果から、使用する対応点の個数を 100 点とした。精度向上が飽和していることが理由である。

3.2 評価実験

点群の点を全て用いて ICP で位置合わせをした際の RSME を正解例とし、本手法での AR 技術を利用した ICP と既存手法での ICP [5] の間で精度の比較を行った。実行環境としては、iPad Pro 第 3 世代 (メモリ容量: 7.3GB) を用いた。点群の点を全て用いた ICP での位置合わせでは、メモリ不足で実行できないため Mac mini 2018 (メモリ容量: 16GB) 上で行った。RMSE¹ は、画像を用いて推定した変換パラメータを点群に適用して位置合わせした後

表 2: 実験結果 (ICP の処理時間は実行環境が異なる点に注意)

	ICP	本手法での ICP	既存手法での ICP
点の数	210,789	100	100
RMSE ¹	0.0166	0.0177	0.0635
RMSE ²	—	0.0135	0.1486
処理時間 [s]	750.88	1.75	1.76

の全点を用いた RSME を指し、RMSE² は画像から取得した対応点による位置合わせでの RSME を指す。また、処理時間の違いも示し、時間と精度の面で比較も行う。

実験結果を表 2 に示す。本手法での ICP と既存手法の ICP を比較すると処理時間がほとんど変わらないのに対し、精度が良好である。したがって、本手法が優れていると考える。点群の点を全て用いた場合の RMSE と比較すると本手法は差が 0.0011、既存手法に関しては差が 0.0469 となり、本手法が点群全て (210,789) を用いた場合と同じくらい高い精度で位置合わせをできていることがわかる。

4. おわりに

本稿では、AR 技術を用いることで既存手法における深度情報の問題を解決し、高い精度で点群の位置合わせを可能とする手法について検討した。評価実験より、本手法での位置合わせは正解例とした点群全てを用いた位置合わせと比較して差が 0.0011 となり、ほぼ同等の精度で位置合わせが可能であることを示すことができた。

謝辞 本研究の一部は JSPS 科研費 JP19K12266 の助成を受けたものです。

参考文献

- [1] 安江 洗希, 大園 忠親, 新谷 虎松, “拡張現実感技術によるナビゲーション支援のための AR オブジェクト共有機構の試作”, 人工知能と知識処理研究会 (AI), pp.30-35, 2021.
- [2] P. J. Besl and N. D. McKay, “A method for registration of 3D shapes”, IEEE Trans. Pattern Anal. Mach. Intell, 14(2), pp.239-256, 1992.
- [3] S. Rusinkiewicz and M. Levoy, “Efficient variants of the ICP algorithm”, Proc. Int’l Conf. 3D Digital Imaging and Modeling, pp.145-152, 2001.
- [4] P. Li, et.al, “Evaluation of the ICP Algorithm in 3D Point Cloud Registration”, IEEE Access, pp.68030-68048, 2020.
- [5] Z. Zheng, Y. Li and W. Jun, “LiDAR point cloud registration based on improved ICP method and SIFT feature”, IEEE International Conference on Progress in Informatics and Computing, 2015.
- [6] K. Liu, J. Boehm and C. Alis, “Change detection of mobile LiDAR data using cloud computing”, ISPRS Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci., XLII-B3, pp.309-313, 2016.
- [7] Kouta Yamada, et.al, “A performance evaluation of key-points detection methods SIFT and AKAZE for 3D reconstruction”, IWAIT 2018, pp.1-4, 2018.