

SIFT 特徴点を用いた Point Cloud の位置合わせ手法に関する研究 A study on point cloud registration with SIFT features

山口 裕大[†] 黒木 修隆[†] 廣瀬 哲也[†] 沼 昌宏[†]
Yudai Yamaguchi Nobutaka Kuroki Tetsuya Hirose Masahiro Numa

1. はじめに

画像や映像から被写体の 3 次元モデルを生成する 3 次元復元には、異なる視点から撮影した 3 次元点群 (Point Cloud) をつなぎ合わせる処理 (位置合わせ) が必須である [1]. そのため、高精度な Point Cloud の位置合わせ手法が必要とされている。

1.1 従来の位置合わせ手法

図 1 に Point Cloud の位置合わせの流れを示す。まず、空間中の孤立点などの外れ値除去を行った後、処理の高速化のためにダウンサンプリングを行うことで Point Cloud の密度を削減する。次に、FPFH (Fast Point Feature Histogram) と呼ばれる特徴量のヒストグラムを各点に与えることで Point Cloud の特徴抽出とする。最後に初期位置合わせには SAC-IA (Sample Consensus Initial Alignment) を、高精度な位置合わせには ICP (Iterative Closest Point) アルゴリズムを用いて処理を完了する。

1.2 問題点

複数視点による計測を行う際に、カメラと被写体との距離が変わることから、Point Cloud 上では対象物が拡大/縮小されていることがある。従来手法ではスケールが大きく異なる場合、2 つの Point Cloud 間で点の数や密度が大きく異なり、SAC-IA で点の対応付けが上手くいかず、初期位置合わせに失敗する問題があった。

2. 提案手法

SIFT (Scale-Invariant Feature Transform) 特徴点によるスケール変換を用いた Point Cloud の位置合わせ手法を提案する。これにより、対象物体のスケールが大きく異なる場合でも正常に位置合わせを行うことが期待できる。提案手法は図 1 の点線の枠部分に挿入される。

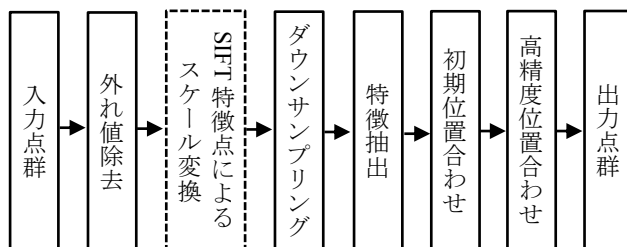


図 1 Point Cloud の位置合わせ手法の流れ
(点線の枠部分は提案手法)

2.1 SIFT 特徴点の抽出

SIFT とは物体上の各点において算出可能な特徴量の一種であり、その物体の回転やスケール変換に対して不変であることが特徴である [2]. イメージモザイク等の画像のマッチングや物体認識に用いられている。通常は物体の平坦部よりもエッジ部を抽出し、SIFT 特徴点とする。特徴点の抽出は DoG (Difference-of-Gaussian) 処理を用いて行われる。

2.1.1 DoG 処理

図 2 に DoG 処理の流れを示す。まず、スケール σ をパラメータとするガウス関数

$$G(x, y, z, \sigma) = \frac{1}{(2\pi\sigma^2)^{\frac{3}{2}}} \exp\left(-\frac{x^2 + y^2 + z^2}{2\sigma^2}\right) \quad (1)$$

を入力点群 $I(u, v, w)$ に畳み込んだ平滑化点群 $L(u, v, w, \sigma)$ を

$$L(u, v, w, \sigma) = G(x, y, z, \sigma) * I(u, v, w) \quad (2)$$

のように求める。同様に σ が k 倍の平滑化点群 $L(u, v, w, k\sigma)$ も求める。両者の差分より DoG 点群 $D(u, v, w, \sigma)$ を

$$\begin{aligned} D(u, v, w, \sigma) &= (G(x, y, z, k\sigma) - G(x, y, z, \sigma)) \\ &\quad * I(u, v, w) \\ &= L(u, v, w, k\sigma) - L(u, v, w, \sigma) \end{aligned} \quad (3)$$

のように求める。 σ を k 倍ずつ大きくしながら同様の処理を行い、複数のスケールについて DoG 点群を求める。

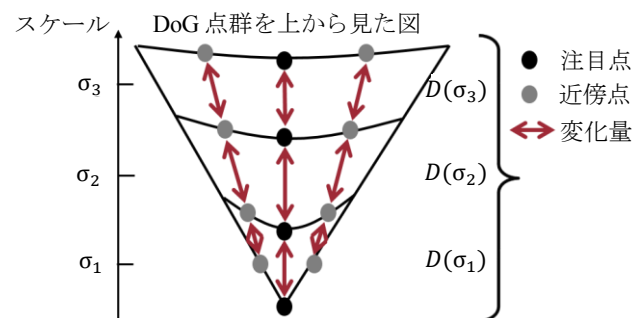


図 2 DoG 処理の流れ

2.1.2 DoG 点群からの極値検出

DoG 点群は 2 つの異なるスケールによる平滑化点群の差分であり、点ごとの変化量を表している。DoG が大きくなることは、そのスケールにおいてエッジ等の情報を多く含んでいることを意味する。そこで、DoG 点群から極値を検出し、特徴点として採用する。

極値の検出は図 3 のように 3 つの DoG 点群を 1 セットにして行う。入力点群におけるある注目点とその周りの 25 近傍の点について、3 つの DoG を比較し極値を検出する。極値が存在する点を SIFT 特徴点として採用する。この極値検出は、 σ の小さい DoG 点群から順に、全てのスケールの DoG 点群に対して行う。

[†] 神戸大学大学院工学研究科,
Graduate School of Engineering, Kobe University

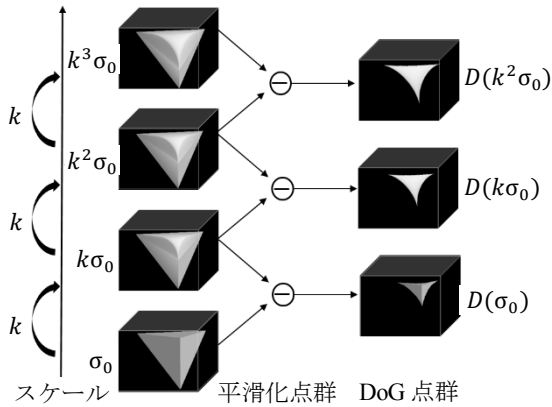


図 3 DoG 点群における極値検出の概要

2.2 スケール変換方法

位置合わせを行う 2 つの Point Cloud の一方をインプット、他方をターゲットとし、それぞれの SIFT 特徴点の座標の分散値 V を

$$V = \frac{1}{N} \sum_{i=1}^N \|p_i - \bar{p}\|_2^2 \quad (4)$$

のように求める。ここで、 N は Point Cloud の SIFT 特徴点の数、 p_i は i 個目の SIFT 特徴点の座標、 \bar{p} はそれらの平均値とする。インプットの分散値を V_{input} 、ターゲットの分散値を V_{target} としたとき、スケール変換パラメータ s を

$$s = \sqrt{V_{target}/V_{input}} \quad (5)$$

のように求める。ターゲットである Point Cloud の全点に $1/s$ 倍の拡大縮小処理を施し、

$$p'_i = \frac{1}{s} p_i \quad (6)$$

とすることでスケールを合わせる。なお、ダウンサンプリング以降の処理は 1 章で述べた従来手法と同じである。

3. 評価実験と考察

従来手法と提案手法を用いて、2 つの Point Cloud 間の位置合わせを行った。10 種類の各 Point Cloud に対して、未処理の Point Cloud をインプットとし、事前にランダムに拡大縮小・回転・平行移動したものをターゲットとして使用した。なお拡大倍率は 1.2 倍から 1.6 倍の範囲である。評価指標は、2 つの Point Cloud の位置合わせの誤差と処理時間である。誤差は平均二乗誤差 (MSE)

$$MSE = \frac{1}{N} \sum_{i=1}^N \|p_i - q_i\|_2^2 \quad (7)$$

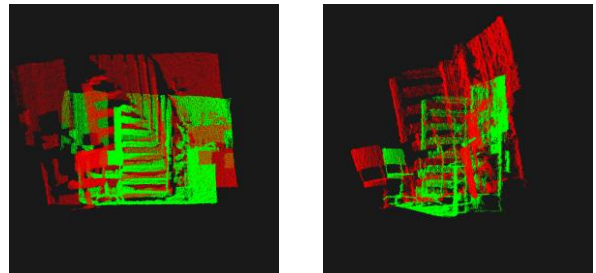
で求める。ここで、インプットおよびターゲットの i 番目の点をそれぞれ p_i, q_i とする。

表 1 に 10 個の Point Cloud の各手法における平均二乗誤差 (MSE) と処理時間の平均値を示す。また、図 4 に従来手法での出力結果の一部を、図 5 に提案手法での出力結果の一部を示す。図 4, 5 において、インプットの Point Cloud を緑色に、ターゲットの Point Cloud を赤色に着色している。表 1 より、従来手法に対して提案手法は平均二乗誤差が約 17 分の 1 に削減されたことが分かる。従来手法ではスケールが異なる Point Cloud 間で同じボクセルサイズのダウンサンプリングを行った結果、削減される点の数が大きく異な

ったため、その後の SAC-IA において点の照合に失敗した。それに対し、提案手法では 2 つの Point Cloud 間のスケールをある程度合わせているため、両者の特徴抽出点がほぼ一致し、位置合わせが正確にできたと考えられる。ただし SIFT 特徴点の算出に時間がかかったため、処理時間が約 6 倍に増加した。

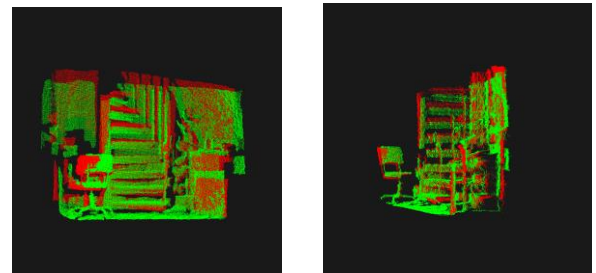
表 1 位置合わせ後の MSE と処理時間の平均値

評価指標	MSE [m ²]	処理時間 [s]
処理前	2.43	
従来手法	0.385	48.3
提案手法	2.32×10^{-2}	2.89×10^2



(a) 視点 1 (b) 視点 2

図 4 従来手法による位置合わせ結果の一部



(a) 視点 1 (b) 視点 2

図 5 提案手法による位置合わせ結果の一部

4. まとめ

本研究では、回転や拡大縮小、平行移動などの位置ずれに対応した Point Cloud の位置合わせを目的として、SIFT 特徴点を用いたスケール変換を提案した。評価実験を行った結果、従来手法に比べ約 6 倍の処理時間を要する結果となったが、平均二乗誤差が約 17 分の 1 に減少した。今後はノイズが多い実測値に対して、提案手法の有効性を確認する予定である。

参考文献

[1] 林 昌希, “コンピュータビジョンのセカイ — 今ここにあるミライ”, マイナビニュース連載, http://news.mynavi.jp/series/computer_vision/059/, 2013.
 [2] 藤吉 弘亙, “Gradient ベースの特徴抽出 — SIFT と HOG —”, 情報処理学会 研究報告, CVIM160, pp.211-224, 2007.