

信州大学審査学位論文

特徴点を用いた
3次元点群処理に関する研究

2021年9月

植西 一馬

(intentionally blank)

論文の内容の要旨

近年、センサや計算機の高性能化により、3次元形状情報処理の研究が活発となり、そのなかでもセンサから取得したばかりの生データである3次元点群の処理技術は特に注目されている。点群は、オブジェクトの表面形状を疎な点の集まりで表現するデータ形式であり、未加工でも形状観察が可能である。この点群の中から、顕著な形状を構成する点分布を見つけ出し、似た形状の部分と比較可能な特徴ベクトルを設定する特徴点抽出は、点群間の対応付けを容易にすることから、各種の応用技術に用いられており、重要な中間処理に位置づけられている。

一方で、未加工の点群には、オクルージョンによる点群の部分欠損、ノイズによる点位置の変動等の正しい特徴点对応の取得を阻害する課題が多くある。これらの解決法としては、複数の点群を重ね合わせて統合するレジストレーションがある。特徴点对応を用いて低計算量のうちにレジストレーションすることも可能であるが、先述の課題のため、高精度に位置合わせすることは困難である。一方で、特徴点对応を用いない方式は座標系変換を何度も繰り返して少しずつ位置合わせしていくアプローチであるため、計算量が大きいという課題をもつ。そこで、本論文では、特徴点抽出とレジストレーションに対し、従来とは異なる枠組みの方式を検討した。特徴点抽出は、位置を決定するキーポイント検出に焦点を絞り、点群内の安定的な形状を活用することとした。レジストレーションは、特徴点对応を用いない方式の計算量低減のため、特徴点の性質を活用することとした。本論文では、この2つで大きな成果を得た。

成果の1つ目は、新たな特徴点の提案である。従来の特徴点のように弁別性を重視するのではなく、再現性を重視するため点群内の安定的な幾何形状を複数抽出し、仮想的な位置に特徴点を見出す方式を提案した。実在の点は、センシング処理において点の位置変動や消失が生起しており、いわば最初から再現性が低い状態にある。そのなかでも、比較的安定してセンシング可能な平面形状に着目し、複数平面の交点位置に特徴点を配置する、VKOP (Virtual Keypoint Of Polyhedron) を提案した。屋内や市街部といった平面を多く含む環境において効果的といえ、シミュレーション実験では従来の特徴点と同等の処理時間で約2倍程度の再現性を示すことを確認した。

VKOPの高い性能が確認された一方で、これに用いる平面抽出法とパラメータに性能が強く依存するという課題が残った。すなわち、点群の環境に合わせて抽出法の選択とパラメータの微調整をしなければ、良いVKOPを得られない。この課題に対しては、平面

の安定性を指標化した尤度を，平面抽出と同時，または事後的に測定し，平面やVKOPそのものに付与する手法を提案した．この尤度の導入により，平面そのものを除去したり，検出するVKOPを選択したりすることで，再現性が高いと見込まれるVKOPのみを残すことが可能となった．さらに，この尤度の測定に適したセンサパラメータを事前に求めておく手法を提案した．これにより，従来のVKOPよりも再現性が向上し，従来の特徴点と比較して，短い処理時間で約4倍以上の再現性を示すことをシミュレーション実験により確認した．

成果の2つ目は，特徴点を用いた新たな枠組みのレジストレーションの提案である．これは，特徴点对応を取らずに，特徴点の位置を活用して計算量を大きく低減する方式である．進化計算によるレジストレーションは，局所解を回避して全体最適解を探索可能であるが，前述のとおり座標系変換を多く繰り返すので，莫大な計算量が必要となる．そこで，これにKPP (KeyPoint Patches)を用いることを提案した．KPPは弁別性の高い形状を有しており，この部分だけを処理に組み込むことで精度を維持しつつ，低計算量を企図したものである．また，KPPの位置，抽出する個数，KPPを成す点数を考慮することで，更なる高能率化を目指した．シミュレーション実験により，従来の進化計算レジストレーションと比較して，提案方式は同等の精度を維持し，約40倍高速に処理できることを確認した．さらに，KPPの位置とKPPを成す点数を調整することで，従来よりも約100倍高速に処理できることを確認した．

目 次

	頁
第 1 章 序 論	1
1.1 3次元形状情報 (3D データ) 処理技術の社会的意義	1
1.2 3次元点群	2
1.3 3次元特徴点	4
1.3.1 特徴点抽出と対応付け	4
1.3.2 特徴点抽出の課題	6
(1) オクルージョン	6
(2) センサノイズ	6
(3) 解像度	7
(4) 課題解決の方針	8
1.4 レジストレーション	9
1.4.1 複数の点群の位置合わせ	9
1.4.2 レジストレーションの課題	11
(1) 点の個数	11
(2) 点と点の関係性	11
1.4.3 特徴点によるレジストレーション	11
1.5 研究の範囲と目的	12
1.6 論文の構成	13
第 2 章 特徴点の従来方式とその活用	15
2.1 はじめに	15
2.2 特徴点抽出	15

2.2.1	特徴点の機能	15
2.2.2	キーポイント	16
	(1) キーポイント検出	16
	(2) Intrinsic Shape Signatures (ISS)	18
	(3) Scale Invariant Feature Transform (SIFT)	18
2.2.3	特徴量	19
	(1) 特徴量記述	19
	(2) Fast Point Feature Histogram (FPFH)	19
	(3) Signature of Histograms of Orientations (SHOT)	20
2.3	特徴点によるレジストレーション	21
2.3.1	レジストレーションの定式化	22
2.3.2	特徴点マッチング	25
2.3.3	誤対応除去	26
2.3.4	変換パラメータの推定	28
2.4	従来の特徴点抽出の課題	29
第3章	特徴点对応によらないレジストレーション	31
3.1	はじめに	31
3.2	近傍点探索による座標系変換を繰り返す方式	31
	3.2.1 初期位置合わせ	31
	3.2.2 近傍点探索	32
	3.2.3 変換パラメータの推定	33
	3.2.4 適応度の測定	33
	3.2.5 終了判定	34
	3.2.6 ICP の課題	34
3.3	進化計算により変換パラメータを探索する方式	35

3.3.1	ECR の処理フロー	35
(1)	個体の生成	35
(2)	個体の進化	36
(3)	適応度の測定と選択	36
(4)	終了判定	37
3.3.2	ECR に適当な進化計算アルゴリズム	37
(1)	Differential Evolution	37
(2)	Self-Adaptive Differential Evolution	38
3.3.3	ECR の課題	39
3.4	従来方式の課題整理	39
第 4 章	仮想特徴点抽出法	41
4.1	本章の概要	41
4.2	従来法の課題と解決方針	41
4.3	平面方程式による仮想特徴点 (VKOP) 抽出	42
4.4	VKOP 抽出の処理フロー	44
4.4.1	平面方程式の推定	45
4.4.2	VKOP によるキーポイント検出	47
4.4.3	VKOP 特徴量記述	47
4.5	評価実験	48
4.5.1	実験環境	49
(1)	ハードウェア及びソフトウェア	49
(2)	データセット	49
(3)	評価指標	54
(4)	使用方式とパラメータ	55
4.5.2	VKOP の性能評価実験	57

4.5.3	検出された特徴点位置に関する考察	61
4.6	まとめ	63
第5章	平面尤度を用いた VKOP	65
5.1	はじめに	65
5.2	従来法の課題と解決方針	65
5.3	平面の尤度を活用した VKOP 検出	66
5.3.1	平面のフィルタリング法	66
5.3.2	VKOP のランキング法	66
5.4	平面の尤度測定法	67
5.5	センサパラメータ決定法	69
5.6	評価実験	71
5.6.1	実験環境	72
(1)	ハードウェア及びソフトウェア	72
(2)	データセット	72
(3)	評価指標	73
(4)	事前実験	73
(5)	使用方式とパラメータ	76
5.6.2	センサパラメータ決定法の効果確認	78
5.6.3	尤度を用いた VKOP の性能評価実験	81
5.7	まとめ	85
第6章	キーポイントパッチを用いた進化計算レジストレーション	87
6.1	本章の概要	87
6.2	従来法の課題と解決方針	87

6.3	KPP 抽出法	88
6.4	KPP 抽出の主要素	88
6.4.1	KPP の位置	89
6.4.2	KPP の個数	91
6.4.3	KPP を成す点数	93
6.5	KPP を用いた ECR	93
6.6	評価実験	94
6.6.1	実験環境	94
(1)	ハードウェア及びソフトウェア	94
(2)	データセット	94
(3)	評価指標	95
(4)	使用方式とパラメータ	97
6.6.2	KPP の性能評価実験	99
6.6.3	KPP-BR の性能評価実験	102
6.6.4	KPP の個数と KPP を構成する点数の評価実験	104
6.6.5	ノイズ耐性の評価実験	106
6.6.6	まとめ	108
第 7 章	結論	111
(1)	再現性を重視した仮想特徴点と尤度評価による高性能化	111
(2)	キーポイントパッチを用いた進化計算レジストレーション	112
	謝 辞	114
	参考文献	115
	研究業績	127

(intentionally blank)

第1章

序 論

3D データ処理の中核技術である点群の特徴点抽出に関し、その重要性和課題について述べ、本研究の範囲と目的を定める。

1.1 3次元形状情報（3D データ）処理技術の社会的意義

いま、我が国が抱える種々の問題解決のために、情報通信技術にかかる社会からの期待はかつてないほど高まっている。この潮流は、新型コロナウイルス感染症の流行により急速化されたのはいうまでもなく、実現の具体例としてはマイナンバーカードの普及 [1]、行政手続の押印廃止 [2]、テレワークの推進 [3] 等があげられる [4]。このように身近なところで実現された例が多くあるため、いわゆるDX (Digital Transformation)[5] は、コンピュータ・ネットワークシステムの活用を大前提とした業務変革（レガシーシステムの刷新）が主眼とされることが多い [6]。しかしながら、これはあくまで余力の確保や、手続きの迅速化のための手段であり、DX の目指すべき姿としているデジタル技術による新たな価値創出に直結するものではない。

翻って、コンピュータビジョンの研究分野においては、コンピュータグラフィックスや人工知能等と連携しながら、新たな価値創出のためのデジタル技術を提供・発展させ続けてきた。旧来、これは2次元の画像・動画により実現されてきたが、計算機とセンサの高性能化・低価格化により、近年は物体の3次元形状を表現するための情報（以下「3D データ」という。）の処理技術が大いに注目されている [7]。これを活用した身近な具体例は、車両の自律走行であろう。自動ブレーキ等の運転支援機能を搭載した自動車 [8] はすでに普及しており、人間が手を触れずとも走行する実証実験が各地で実施されている [9]。現在位置の把握にはGPS (Global Positioning System) といった異なる技術の連携が求められるも

のの、リアルタイムに周囲の環境を把握・分析するとともに、自身の位置と姿勢を推定するには、3D データの活用が必須である [10].

もう1つの具体例として、我が国の防災と3D データ処理技術の関係を挙げる。2011年3月の東日本大震災では、情報収集にはそのほとんどが空撮・衛星画像が用いられ、3D データの活用事例は極めて限定的であった [11, 12]. しかしながら、時は進み2016年4月の熊本地震においては、発災直後から国土地理院ほか複数の測量会社が航空機によるレーザー計測を実施し、DSM (Digital Surface Model)[13]による被害の把握に用いた等の活用例が多数存在する [14], 近年はこれを先に推し進め、平時にセンシングした3D データを備蓄・公開することで、発災後の3D データとの比較を容易にするとともに、広域な3D 都市モデル [15] を構築することにより減災対策シミュレーションの実施やハザードマップの公開等が実現されている [16].

この他にも、顔認証 [17], 医療用データ [18], 拡張現実 [19], 文化財のデジタルアーカイブ [20, 21] 等, 多くの先進的な3D データ処理技術の研究が進められている。これらは既に実用化されているものも多くあり, まさに新たな価値を創出した具体例といえる。すなわち, 3D データ処理技術の発展は我が国にとって大きな社会的意義を有し, ひいてはDXの推進により, その重要性は加速度的に増すことになるであろう。

1.2 3次元点群

先に挙げた3D データの活用事例に多く用いられている自己位置・姿勢の推定や, 3次元形状のモデリングを行うには, まずセンサでオブジェクトの3D データを取得する必要がある。このときに得られる未加工の3D データは, 2次元の距離画像 (Range image) と3次元の点群 (Point cloud) の2つに分けられる。

距離画像は, 画像平面上の各画素にセンサからの距離を保存し, 濃淡で奥行きを表現する形式である [22]. 図 1.1 に, Stanford Bunny[23] の正面からみた外観



(a) 色付きの外観



(b) 距離画像 (淡色: 近距離, 濃色: 遠距離)

図 1.1 : Stanford Bunny の画像

と, この距離画像を示す. 距離画像はデータが密であり, 旧来の画像処理技術も適用できるといったメリットがある. 一方で, 視点を変えて形状を観察したり, ある箇所から別の箇所への距離を計測するといった場合には, センサの内部パラメータを用いて次に示す点群へ変換する必要がある. そのため, 距離画像をそのまま単独で用いる意義は薄いといえる.

3次元点群は, センサの位置と姿勢を基準とした座標系に, 観測した表面の3次元位置座標をそのまま保存し, 疎な点の集まりで形状を表現する3Dデータ形式の1つである [22]. 図 1.2 は, Stanford Bunny を正面からセンシングした点群を, 正面からと, 上から見た外観を示す. 距離画像と異なり, 視点を変えて別方向からの形状の観察が可能であり, ある点から別の点への距離の計測も容易である. また, 位置座標を取得すると同時に, 別の情報も取得して各点へ付与することも可能である. 例えば, 3Dデータの取得に2次元画像を用いるステレオビジョン [24] や SfM (Structure from Motion)[25], 通常の RGB カメラを 3D センサと並列配置した複合センサ (Microsoft Kinect 等)[26] は, 個々の点に RGB の色情報を付与することが可能である. これとは別に, 光の反射を用いて距離情報を得る LIDAR (LIght Detection And Ranging) のような 3D センサでは, 反射してきた

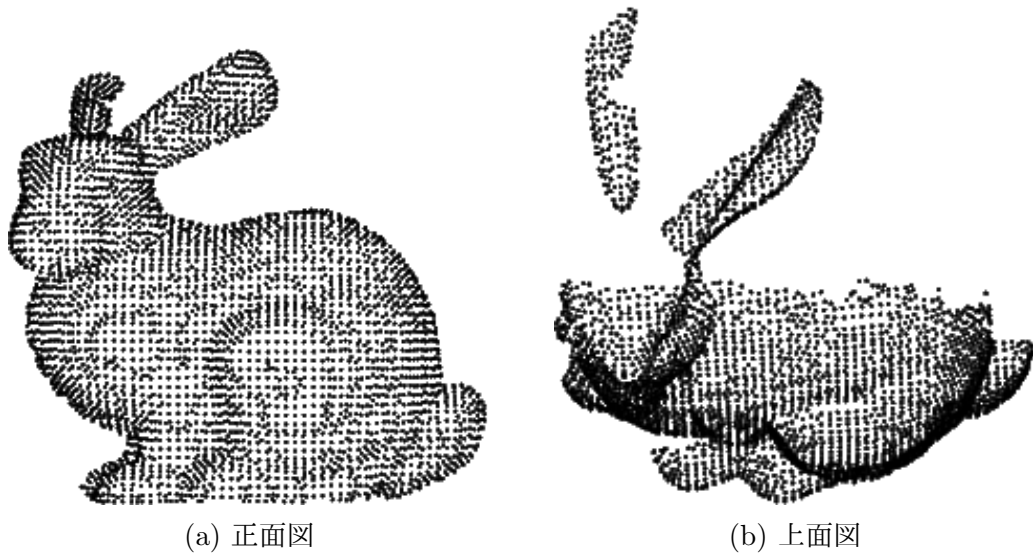


図 1.2 : Stanford Bunny の点群

光の強度を点に付与することもできる [27].

センシングしたばかりの点群は、点の情報が無作為に羅列されているだけの状態となっており、点と点の関係性を有さない。このような状態にある点群を未整理の点群 (unorganized/unstructured point cloud) という。人間の目では、表面の形状がどのようなものなのかを認識できる。例えば図 1.2 をみて、耳先が尖っていることや、背中がなだらかな曲面になっているということは、一瞥で理解できる。しかしながら、計算機は点の位置座標が並んでいるだけの情報を見て、このような形状をもっていることを理解することはできない。そこで、次に述べる特徴点を抽出することで、計算機にも形状を理解させることが、点群処理には重要となる。

1.3 3次元特徴点

1.3.1 特徴点抽出と対応付け

特徴点抽出は、顕著な形状を構成する点分布を見つけ出し、他の部分と容易に比較可能な特徴ベクトルを設定する処理である [28]。これは、2次元画像におけ

る代表的な特徴点である Scale Invariant Feature Transform (SIFT)[29]と同様に、注目した少数個の点から、その近傍にある点分布を分析するのが一般的であり、局所特徴点とも呼ばれる。少数個の特徴点だけを対象とすることで計算量の低減に貢献するとともに、特徴ベクトルだけの比較で似たような形状同士を対応付けできる。このような効果を有することから、モデリング、オブジェクト認識、形状検索、形状分類といった各種の応用技術に用いられており、特徴点抽出は重要な中間処理に位置づけられている [7, 30].

異なる位置からセンシングした2つの Stanford Bunny の点群について、特徴点を抽出し、対応付けした例を図 1.3 に示す。赤と緑の2つの点群上にある大きい丸は抽出した特徴点であり、対応の取れた特徴点を黒線で結んでいる。対応の取れた特徴点の位置は、元のオブジェクトからみて概ね同じ位置にあることがわかる。このような対応の数をカウントすることでオブジェクト認識や形状検索を実現したり、対応の取れる特徴点の位置関係から後に示すレジストレーションを

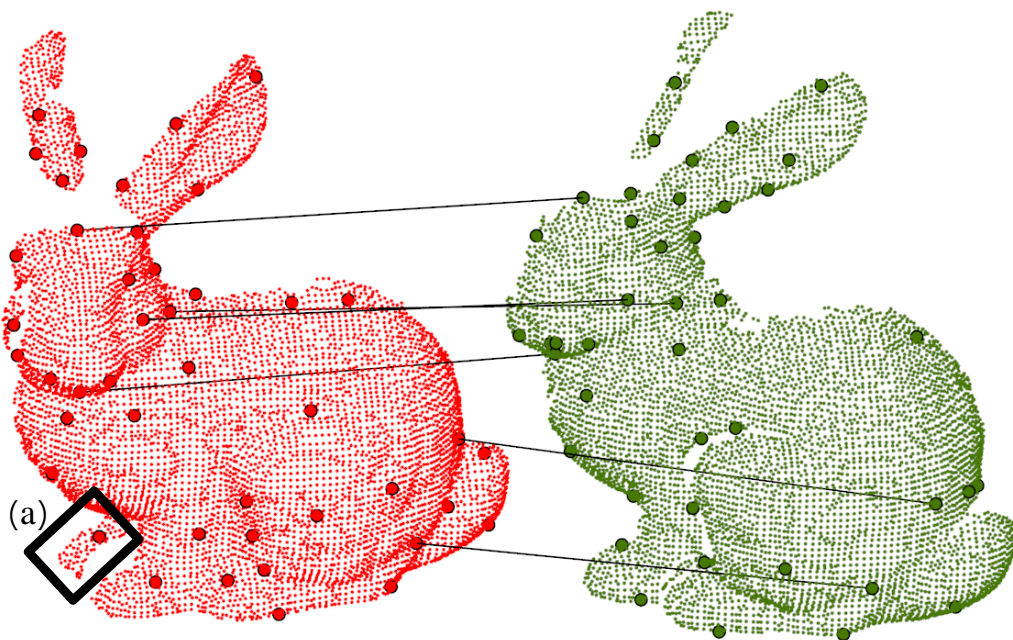


図 1.3 : 特徴点マッチングの例

実現する。すなわち、正しい対応の取れる特徴点を数多く抽出することが、この処理の命題といえる。

1.3.2 特徴点抽出の課題

前項で特徴点の重要性と活用について述べたが、未整理の点群を対象としたとき、特徴点の誤対応を誘発する課題が多く存在する。この解決すべき課題である、オクルージョン、センサノイズ、解像度について述べる。

(1) オクルージョン

オクルージョンは、センサに露出していない部分の点群が欠損することをいう。例えば、図 1.2 ではオブジェクトの正面からセンシングしているため、図 1.2(b) からわかるようにオブジェクトの裏側は点群を取得できず、必ず欠損が生じて形状がオープンとなる。オクルージョンは、オブジェクトの裏側だけでなく、障害物がセンサとオブジェクトの間に入ることで容易に生じる。

特徴点を抽出するとき、オクルージョンの影響を除外することは、大きな課題といえる。例えば、図 1.3(a) に示す奥の足の点は、緑の点群にはオクルージョンによって欠損している状態にある。そのため、このような位置に特徴点を抽出しても、正しい対応が取れないばかりか、誤対応を誘発することとなる。

(2) センサノイズ

ノイズは、一般的なガウシアンノイズと、距離画像から点群に変換した場合の特有なノイズの 2 つがある。前者は正規分布に従う点の位置座標の変動であり、センサの性能に依存するものである。また、センサの計測方式によっては変動の方向が異なり、例えば Microsoft Kinect 等の距離画像から点群に変換する方式のものは、センサから見て画像平面上の変動量は小さいが、奥行き方向の変動量が大きくなる傾向にある [31, 32]。また、Velodyne HDL-64E のような LIDAR は、計測した表面をそのまま点にしているので Kinect のような傾向はなく、単純にセンサからの距離に変動があるとされている [33]。

後者は、ガウシアンノイズによって生じた位置座標の変動が、量子化の影響を

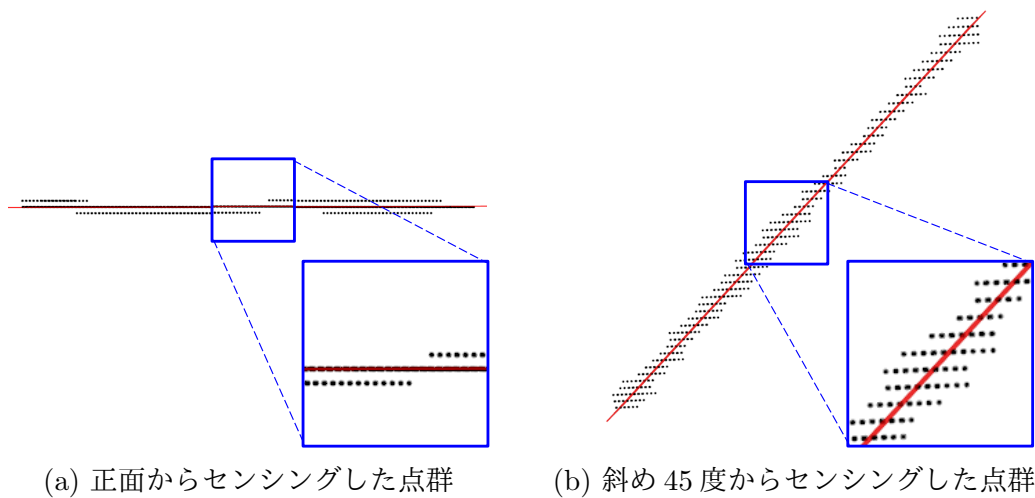


図 1.4 : 量子化ノイズの例

受けて点群がスライス状になるものである [34, 35]. 図 1.4 は, このノイズの発生例を示す. 平面をセンサから 1m 離れたところに置き, 正面からセンシングしたものが図 1.4(a), 入射角が 45 度になるように角度をつけてセンシングしたものが図 1.4(b) である. この図は平面を上から見ており, 赤い線が平面, 黒い点が取得された点群を示している.

これらのノイズの影響により表面形状の正確さが失われ, 特徴のない位置に特徴点を抽出したり, 特徴ベクトルが変動したりすることで, 誤対応が生起することとなる.

(3) 解像度

解像度の高低は, センサの性能以外に, センサとオブジェクトとの距離に依存することは自明である. センシング可能な範囲が狭い場合はその影響は小さいが, 広い場合には同一点群内で解像度の変化が顕著に現れることとなる.

図 1.5 に, 全周 LIDAR で取得した点群の例を示す. これは, Barcelona Robot Lab Dataset[36, 37] の大学構内を模した 3D モデルにおいて, Blensor[38] のシミュレーションセンシングで得た点群である. (a) のセンサの周りに点がないのは, セ

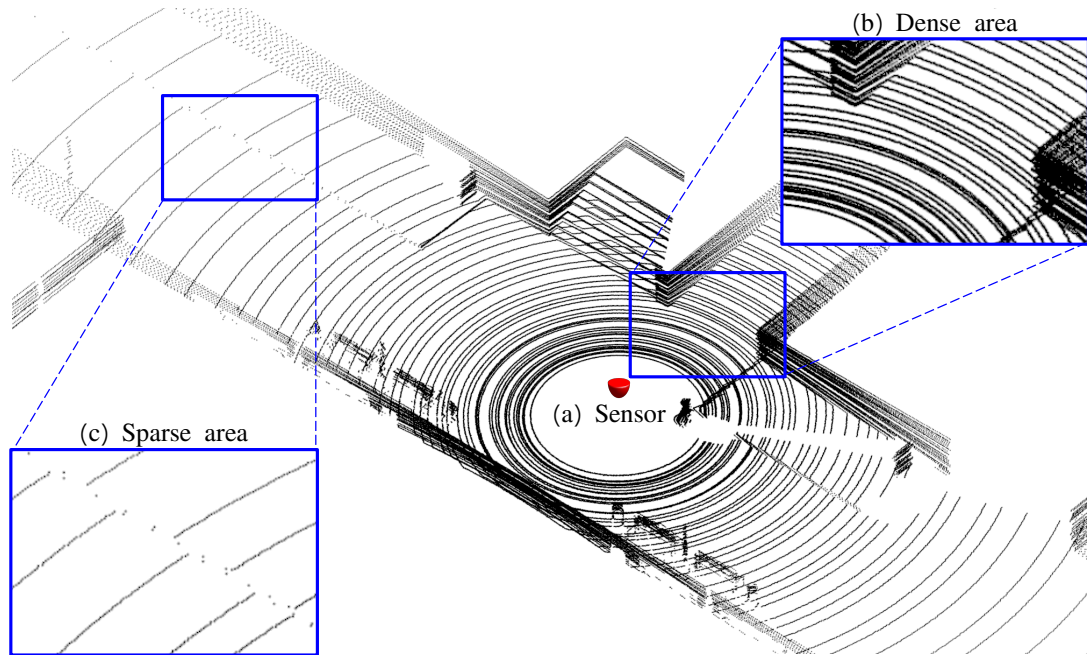


図 1.5 : LIDAR 点群の例

ンシング可能な最小距離よりも近いためである。(b)の領域はセンサに近いので解像度が高く、逆に(c)は遠いためスキャンライン間の解像度が低くなっていることがわかる。

解像度が低い場合は表面形状の精緻さが失われ、オクルージョンのように対応の取れる領域ごと欠損したり、ノイズのように特徴点の性能を低下させたりといった、悪影響を与えることとなる。

(4) 課題解決の方針

先に述べた課題は、次に述べるレジストレーションを行うことである程度解決が可能である。しかしながら、レジストレーションできない場合を想定したとき、特徴点そのものの性能を高める必要がある。例として、オクルージョンの位置に特徴点を抽出せず、またノイズや解像度の影響で形状に違いがあっても正しい対応が取れるように設計しなければならない。

1.4 レジストレーション

1.4.1 複数の点群の位置合わせ

レジストレーションは、複数の点群を正しく重ね合わせて、ある1つの点群の座標系に統合する処理である [39]。このとき、センサの位置と姿勢を変えながら点群を取得しておくことで、先に述べた課題であるオクルージョンによる欠損を補間する、または点群の解像度を増加することが可能である。更に、重なり合う領域を増加させることは、すなわち観測した標本数の増加と同義であり、ノイズ低減手法の効果を高めることも期待できる。このような効果を有することから、レジストレーションは特徴点抽出と同様に、点群処理の重要な中間処理に位置づけられている [7]。

レジストレーションの例を、図 1.6 と図 1.7 に示す。図 1.6 は Stanford Bunny を複数の視点から取得した点群を、正面からみたものである。なお、これらの点群の色は、各点群を識別するために着色したものであり、前述したような色情報を持っているわけではない。図 1.7 は、図 1.6(a) の点群へ他の全てを座標系変換して統合した、レジストレーション後の点群である。図 1.7(b) からわかるように、オクルージョンによる欠損が埋め合わされ、1つのクローズな点群となっている。また、オクルージョン以外の重なりがある領域は点の解像度が増し、元の点群よりも精緻に形状が表現できている。

正確にレジストレーションするためには、基準となる点群の座標系への変換パラメータ、すなわち3軸回転と3軸並進を含む6次元のパラメータを高精度に得る必要がある。前述のとおり、各点群はセンシング時におけるそれぞれの座標系に属しているため、点群の変換パラメータの推定はセンサの位置と姿勢の変化量を取得することに等しい。GPS や INS (Inertial Navigation System) といった外部システム、またはホイールオドメトリといった外部センサがない場合には、点群間の関係性から変換パラメータを推定する必要がある。図 1.7 の例では外部システム等を使用せずに、代表的なレジストレーション方式の ICP (Iterative Closest

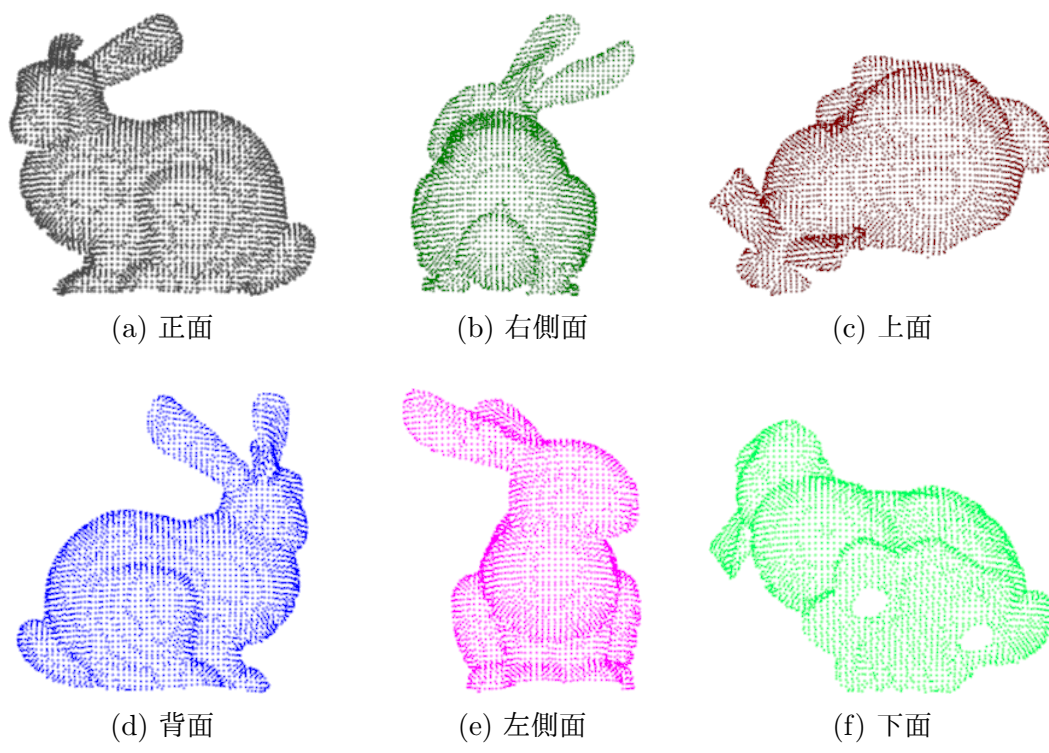


図 1.6 : レジストレーション前の複数点群

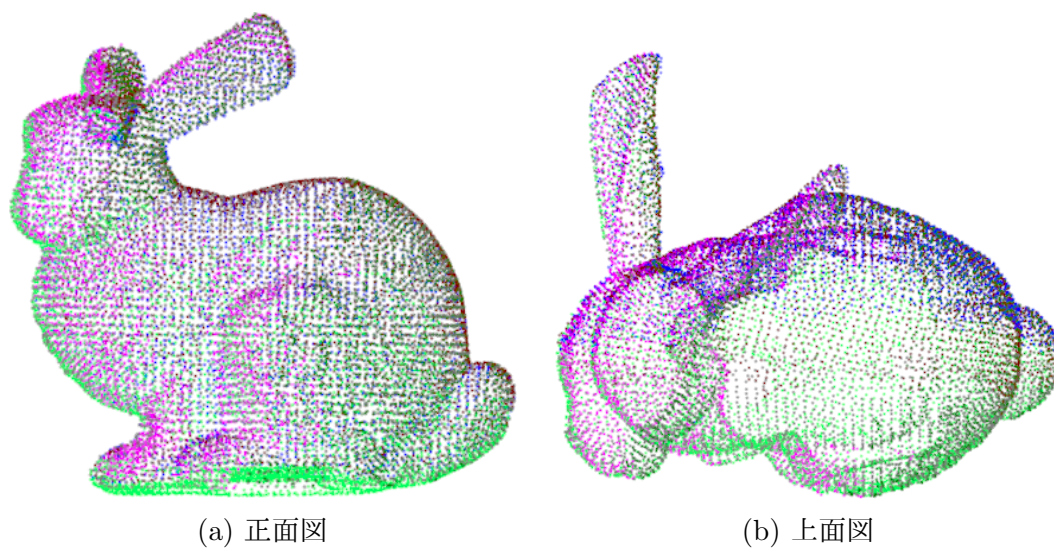


図 1.7 : レジストレーション後の点群

Point) アルゴリズム [40] を拡張した方式で変換パラメータを推定し、それぞれの点群を座標系変換している [41].

1.4.2 レジストレーションの課題

前項でレジストレーションの有効性を述べたが、特徴点抽出で挙げた各種課題の影響は、当然レジストレーションにも影響を与える。また、次に述べる点の個数の多さと、点と点の関係性がないという課題は、前述した ICP のように点群の座標系変換を繰り返す方式の計算量をより増大させる。

(1) 点の個数

センサーの性能にもよるが、取得できる点の個数が多いため、点群処理の計算量は大きくなる傾向がある。例えば、代表的な 3D センサである Microsoft Kinect ver.1 では約 30 万点、ver.2 では約 21 万点の点群を 30fps で取得できる [42]。また、Velodyne HDL-64E は、約 220 万点の全周をセンシングした点群を 1fps で取得できる [33]。計算量を削減するには点数を削減する必要があるが、単純なダウンサンプリングでは形状の精緻さが失われる可能性がある。

(2) 点と点の関係性

未整理の点群を対象とした場合、先に述べた点の個数の多さとともに、点と点の関係性がないことは、点群処理の計算量を増大させる要因である。この解決には、点の位置座標をデータ構造化する必要がある。データ構造化手法にはポリゴン三角形 [43] の構築がよく用いられるが、これは表示に適した形式であって、処理や保存に適しているとはいえない [44]。このため、kd-tree [45] や Octree [46] を用いて構造化することが多い。

1.4.3 特徴点によるレジストレーション

レジストレーションの課題である計算量の大きさは、特徴点对応の活用によりある程度解決できる。すなわち、特徴点抽出とレジストレーションは、課題と解決法を相互に補い合う関係にあるといえる [28, 30, 39].

図 1.8 は、図 1.3 の特徴点对応から変換パラメータを求め、レジストレーショ

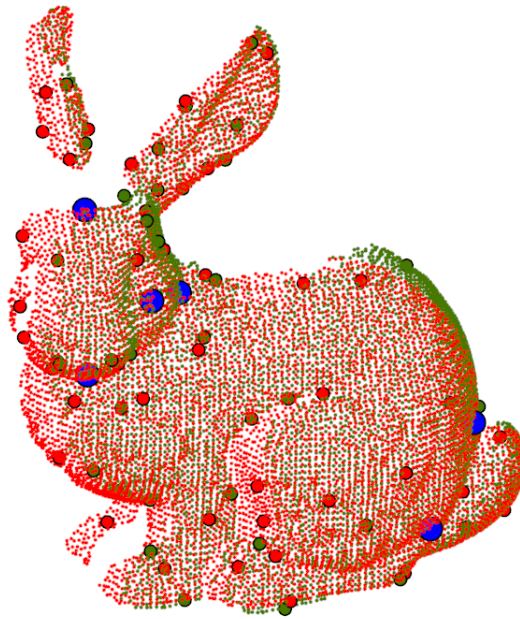


図 1.8 : 特徴点レジストレーションの例

ンした例を示す。点群上の大きく青い丸は、対応の取れた特徴点を示している。この例のように、正しく対応の取れた特徴点の位置関係を用いることで、1度だけの座標系変換で高精度にレジストレーションが可能である。

しかしながら、誤対応が含まれる場合には、特徴点が少数個であるがゆえに悪影響を強く受けてレジストレーションの精度は大きく低下することとなる。この課題を解決するには、前述した特徴点の性能を高めることが考えられる。一方で、特徴点对応を用いない場合の課題である計算量の大きさに着目したとき、形状の精緻さを失わずに点の個数を削減することも解決法の1つといえる。この点の個数の削減に、特徴点の性質を用いることとする。

1.5 研究の範囲と目的

3D データ処理技術の社会的意義、そのうち特徴点抽出の重要性について述べた。この技術は、冒頭で述べた各種の応用技術に用いられており、重要な中間処

理としての位置づけを確立している。一方で、特徴点抽出そのものにも、特徴点の代表的な応用技術であるレジストレーションにも、各種の課題が残ることについて述べた。すなわち、これらの課題を克服した特徴点抽出とレジストレーションの実現は、冒頭に述べたような最新の応用技術の高精度化・低計算量化につながり、我が国全体、ひいては世界的な貢献が大きいといえる。また、色や反射強度等の情報を持たない未整理の点群を処理対象とすることで、より一般的に用いることが可能な技術を目指す。

以上から、位置座標の情報のみを有する未整理の点群を対象とし、これからの特徴点抽出と、特徴点を用いたレジストレーションを本研究の範囲とした。そして本研究の目的は、この2つの処理を阻害する各種課題に対し、特徴点抽出については特徴点の位置の安定性を向上することを、レジストレーションについては特徴点を用いて点群をなす点の個数を削減することで解決を目指す。

1.6 論文の構成

本論文の構成の概略を図 1.9 に示す。

第1章は前述のとおり、研究の範囲と目的について述べた。

第2章は、特徴点抽出の従来方式の詳細を述べるとともに、特徴点对応によるレジストレーション方式について述べ、従来の特徴点の課題を整理する。

第3章は、特徴点对応を用いないレジストレーション方式について、近傍点探索による座標系変換を繰り返す方式、進化計算により変換パラメータを探索する方式の詳細を述べ、解決すべき課題を整理する。

第4章は、特徴点そのものの性能に着目し、点群内に含まれる複数の平面部分から平面方程式を求め、実在点の有無に関わらず特徴点を配置する仮想特徴点抽出法を提案する。車を模したクローズな点群、距離画像センサから得たリビングを模した点群、大学の構内を模した全周の点群を対象に比較評価し、この提案は従来の特徴点と同等の処理時間で、最大2倍程度高性能であることを確認した。

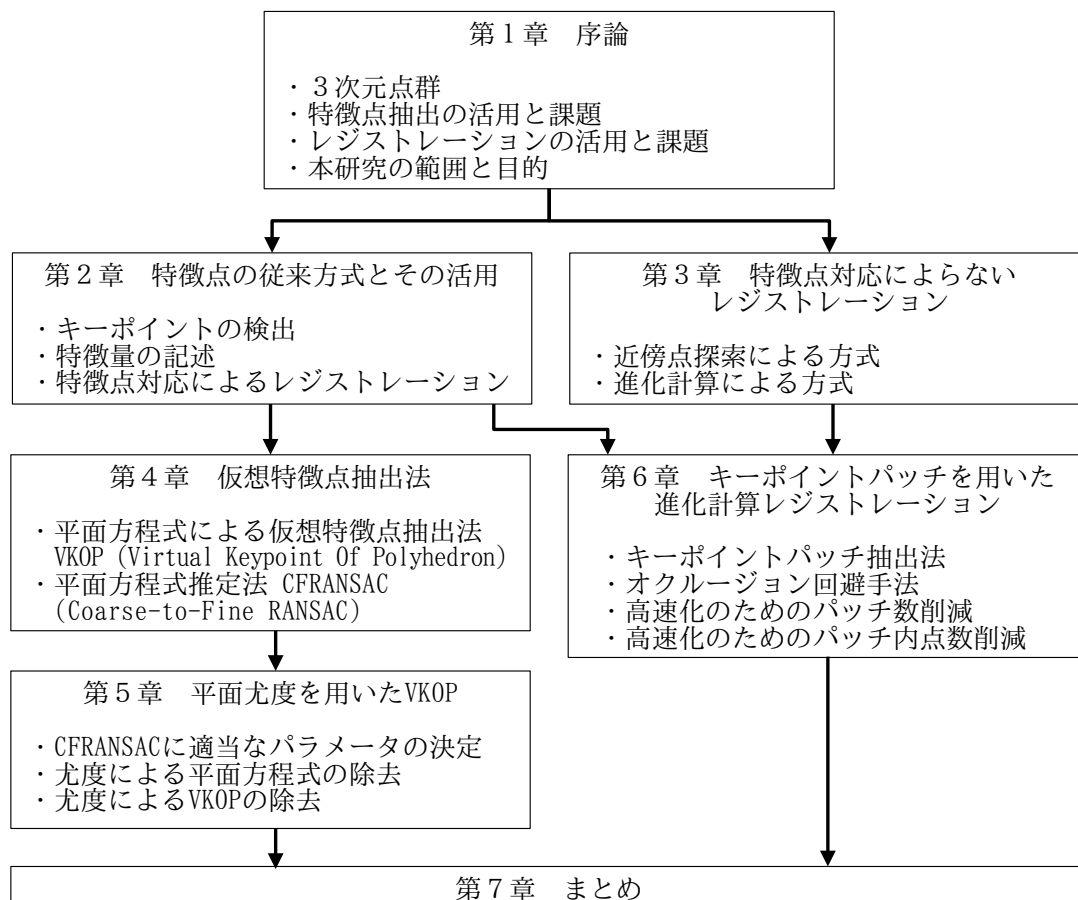


図 1.9 : 本論文の構成

第5章は、第4章で提案した仮想特徴点の性能について、平面方程式の推定法やパラメータの依存度低減のため、最適なパラメータを決定する手法と、平面方程式の尤度測定を用いて適切な仮想特徴点を残す手法を提案する。第4章でも用いたリビングを模した点群により比較評価し、この提案は従来の特徴点よりも短い処理時間で4倍以上高性能であることを確認した。

第6章は、進化計算によるレジストレーションに着目し、特徴点を活用することでレジストレーションを高効率化するキーポイントパッチ抽出法を提案する。Stanford Bunny等のデータセットを用いて性能を評価し、この提案は従来方式と比較して精度を維持しつつ、最大で約100倍以上高速処理できることを確認した。

最後に、本研究成果を第7章にまとめる。

第2章

特徴点の従来方式とその活用

従来の特徴点抽出法と活用法について述べ、課題の解決法を考察する。

2.1 はじめに

1.3節で特徴点の活用法と課題を述べたが、ここではその処理を細分化して詳細に述べる。次に、活用例である特徴点对応によるレジストレーションについて述べ、課題の解決に最も貢献する処理を導出するとともに、課題の解決法を考察する。

2.2 特徴点抽出

2.2.1 特徴点の機能

文献 [28] によると、特徴点は図 2.1 に示すような再現性 (Repeatability) と弁別性 (Distinctiveness) を備えるべきとされている。再現性とは、同じオブジェクトからセンシングした複数の点群において、1.3.2 項で挙げたような各種課題があったとしても、相対的に同一箇所で特徴点を得られる機能をいう。図 2.1(1) に

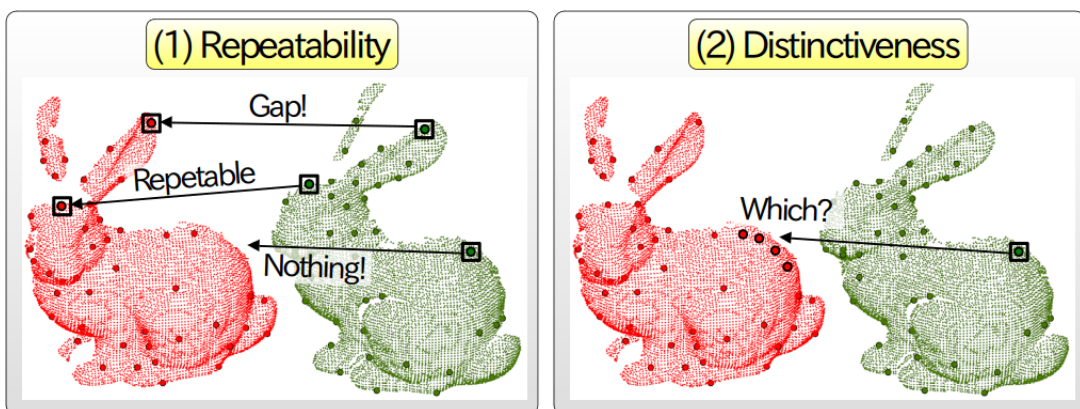


図 2.1 : 特徴点の備えるべき機能

示すように、赤い点群側の特徴点が抽出されていない場合は対応付けが不可能であるし、抽出されていても僅かな位置変動で、変換パラメータを高精度に推定できなくなる。また、弁別性とは、点の周囲の3次元形状に特徴があることをいう。例えば図2.1(2)に示すような、背中のだらかな曲面上といった平坦な位置に特徴点が抽出されている場合、正しい対応がとれる特徴点はどれなのかを判定するのは困難である。

前述の機能を有する特徴点を抽出するため、まずはそのような少数個の点を見つけ出し、次にその周囲にある点分布を活用して特徴量を算出する、2段階の処理が一般的である [28]。前半はキーポイント検出 (Keypoint detection)、後半は特徴量記述 (Feature description) と呼ばれ、この2つをもって特徴点抽出 (Feature point extraction) 処理となる。どちらの処理も局所性を持たせるため、ある点に着目したとき、支持半径 (Support radius) またはスケールと呼ばれる局所領域に含まれる点の分布を処理対象とする。また、先に述べた2つの処理は、更に2つずつに細分できる。キーポイント検出は突出値 (Saliency) の測定と Non Maxima Suppression (NMS) 処理、特徴量記述は Local Reference Frame (LRF) の構築と特徴ベクトルの計算である [47, 48]。これらをまとめた処理フローを図2.2に示す。

2.2.2 キーポイント

(1) キーポイント検出

キーポイント検出は、特徴点として有用な位置の点を決定する処理である (図2.2(1) 及び (2))。一般的にキーポイント検出は、突出した形状の先端に位置する点を探索する手法がよく用いられる [47, 49]。これは、突出した形状が、その他の部分と比較して高い弁別性を有することを考慮したものである。

本論文で取り扱う点群は、3次元位置座標のみを有する情報の集合であるため、ある点群 \mathbb{C} は

$$\mathbb{C} = \{\mathbf{p}_i^c | i = 1, \dots, N_c\}, \mathbf{p}_i^c = (x_i, y_i, z_i)^T \in \mathbb{R}^{3 \times 1} \quad (2.1)$$

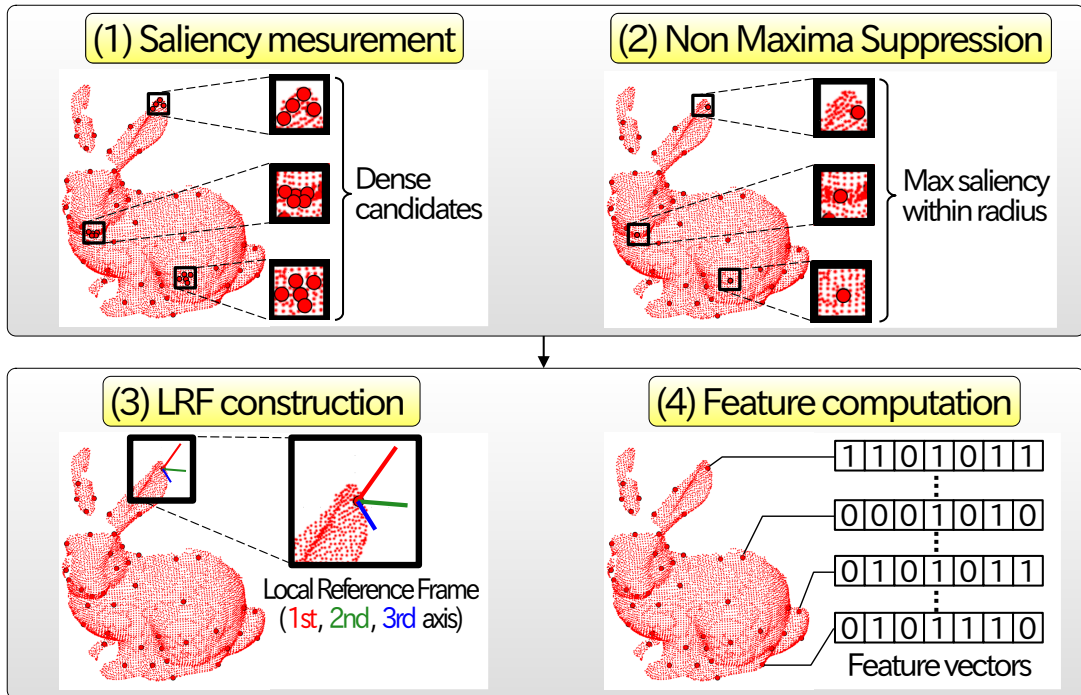


図 2.2 : 特徴点抽出の処理フロー

と定義する. p_i^c は \mathbb{C} を構成する N_c 点のうちの 1 点, x_i, y_i, z_i は 3 軸の位置座標を示す. 以後, 点群 \mathbb{C} から検出したキーポイント $\mathbf{k} = (x, y, z)^T$ の集合を

$$\mathbb{K}^c = \{\mathbf{k}_i^c | i = 1, \dots, N_{ck}\}, \mathbb{K}^c \in \mathbb{C} \quad (2.2)$$

とする.

高い弁別性を有する特徴点抽出のため, 突出しているか否かを定量的に評価した突出値 (Saliency) を全点に付与し, 閾値以上の突出値を有する点をキーポイント候補として残す. 突出値の計算には, 支持半径 Rd_{sal} 内にある点分布について分析し, 特にこれらの点に付与されている曲率を活用することが多い. また, この支持半径を固定せず, 適応的に変化させて突出値を取りまとめることで, 点群のスケールが変化しても再現性を保持することを企図した手法もある. しかしながら, 図 2.2(1) に示すように, 耳や鼻の先端といった領域の点はどれも突出値が高くなるため, キーポイント候補が密集して残る可能性がある. そこで, キーポイント候補ごとに支持半径 Rd_{nms} 内を探索し, そこに含まれる最大の突出値を

示すキーポイント候補だけを残す NMS 処理を用いる。図 2.2(2) の例では，密集していたキーポイント候補のうち，1 つだけが残っている。この NMS 処理後に残った点を，キーポイントとして扱う。

キーポイント検出の代表的な手法である ISS と SIFT3D について，概要を次に示す。

(2) Intrinsic Shape Signatures (ISS)

Intrinsic Shape Signatures (ISS) は，支持半径固定型のキーポイント検出法である [47, 50]。突出値には曲率がそのまま用いられるが，キーポイント検出後の LRF 構築に適した位置に検出するように設計されている。処理の流れとしては，ある点 \mathbf{p}_i に着目したとき，まず Rd_{sal} 内にある点分布から scatter 行列を

$$\frac{1}{N_{Rd}} \sum_{i=1}^{N_{Rd}} (\mathbf{p}_i - \bar{\mathbf{p}})(\mathbf{p}_i - \bar{\mathbf{p}})^T \quad (2.3)$$

により求める。 N_{Rd} は Rd_{sal} 内にある点の数， $\bar{\mathbf{p}}$ は Rd_{sal} 内にある点の重心点である。この行列から，EVD (EigenValue Decomposition) により， $\lambda_1 \geq \lambda_2 \geq \lambda_3$ となる 3 個の固有値を得る。このとき， λ_3 は曲率であり， \mathbf{p}_i の突出値として採用される。一方で，前述した LRF に適したキーポイントを得るため，

$$\frac{\lambda_2}{\lambda_1} < Th_{21}, \quad \frac{\lambda_3}{\lambda_2} < Th_{32} \quad (2.4)$$

を満たす点をキーポイント候補として出力する。 Th_{21}, Th_{32} は各固有値の割合の閾値である。最後に，残ったキーポイント候補に対して NMS 処理を行い，キーポイントを得る。

(3) Scale Invariant Feature Transform (SIFT)

Scale Invariant Feature Transform (SIFT)[29] を拡張した SIFT3D は，支持半径適応型のキーポイント検出法である。また，支持半径を単純に変化させるのではなく，Difference of Gaussians (DoG) の考えに基づき，点群を成す点を段階的にダウンサンプリングしたスケールスペースを構築することで，計算量を低減し

ている。処理の流れとしては、あるスケールスペース（1つのオクターブ）内で、異なる支持半径の DoG により曲率の変化量を求める。次に、曲率の変化量の最大値と最小値の差（コントラスト量）が、設定した閾値を超えたものをキーポイント候補として出力する。すなわち、SIFT3D ではコントラスト量が突出値として用いられる、全オクターブにおけるキーポイント候補の出力が完了した後、コントラスト量を元にした NMS 処理により、キーポイントを得る。

2.2.3 特徴量

(1) 特徴量記述

特徴量記述は、NMS 処理後に残った全キーポイントに、周囲の点分布等を取りまとめて一意となるような特徴量を付与する（図 2.2(3) 及び (4)）。これは、特徴点間の類似性を容易に比較できるようにするため、比較的高次元のベクトルで表現される。

この実現のため、まず回転不変性の獲得を目的として、キーポイントごとに LRF を構築する場合が多い [48]。図 2.2(3) では便宜上、1つのキーポイントに限定して LRF の 3 軸を表示しているが、全てのキーポイントにそれぞれの LRF が与えられている。次に、この LRF を基準の座標系として、支持半径内にある点分布や、付与された法線ベクトル等を用いて特徴量を算出する。特徴量は、支持半径内の点分布をそのまま使用するシグネチャ型と、局所領域ごとの分布をカウントするヒストグラム型の二つに分けられる [51]。近年では、様々な外部要因に対応しやすいヒストグラム型か、両方を組み合わせたハイブリッド型が主に提案されている [48]。以後、あるキーポイント k_i^c には、これに対応する特徴量 f_i^c を与えるものとする。

特徴量記述の代表的な手法である FPFH と SHOT について、概要を次に示す。

(2) Fast Point Feature Histogram (FPFH)

Fast Point Feature Histogram (FPFH) は、支持半径内にある点や法線ベクトルの相対角をカウントする、ヒストグラム型の記述子である [48, 52]。これは、

Point Feature Histogram (PFH)[53] を低計算量化したものであり、LRF が不要な手法である。処理の流れは、まず全点に法線ベクトルを付与し、そしてまた全点に Simplified Point Feature Histogram (SPFH) 特徴量を付与しておく。最終的な特徴量は、キーポイントの支持半径内にある点分布の SPFH 特徴量をとりまとめることで算出する。

まず SPFH 特徴量は、ある点 \mathbf{p}_i と支持半径 Rd_{des} 内にある 1 つの点 \mathbf{p}_j に着目したとき、darboux frame $(\mathbf{u}, \mathbf{v}, \mathbf{w})$ を

$$\begin{cases} \mathbf{u} = \mathbf{n}_i \\ \mathbf{v} = (\mathbf{p}_j - \mathbf{p}_i) \times \mathbf{u} \\ \mathbf{w} = \mathbf{u} \times \mathbf{v} \end{cases} \quad (2.5)$$

により求め、次に 3 個の相対角度 α, ϕ, θ を

$$\begin{cases} \alpha = \mathbf{v} \cdot \mathbf{n}_j \\ \phi = (\mathbf{u} \cdot (\mathbf{p}_j - \mathbf{p}_i)) / \|\mathbf{p}_j - \mathbf{p}_i\| \\ \theta = \arctan(\mathbf{w} \cdot \mathbf{n}_j, \mathbf{u} \cdot \mathbf{n}_j) \end{cases} \quad (2.6)$$

により得る。 \mathbf{n}_i は \mathbf{p}_i の法線ベクトルである。これを Rd_{des} 内にある全点に対して計算し、3 個の角度をカウントした SPFH 特徴量 $SPFH(\mathbf{p}_i)$ を \mathbf{p}_i に付与する。次に、あるキーポイント \mathbf{k}_m に特徴量 \mathbf{f}_m を付与するには、

$$\mathbf{f}_m = SPFH(\mathbf{k}_m) + \frac{1}{N_{Rd}} \sum_{n=1}^{N_{Rd}} \frac{1}{w_n} SPFH(\mathbf{p}_n) \quad (2.7)$$

により得る。 \mathbf{p}_n は Rd_{des} 内にある N_{Rd} 点のうちの 1 つである。 w_n は \mathbf{k}_m と \mathbf{p}_n の距離であり、近い位置の SPFH 特徴量の影響が大きくなるよう重み付けしている。

(3) Signature of Histograms of Orientations (SHOT)

Signature of Histograms of Orientations (SHOT) は、支持半径内の空間を小分けにし、キーポイントとその他の点との相対角度等をカウントするハイブリッ

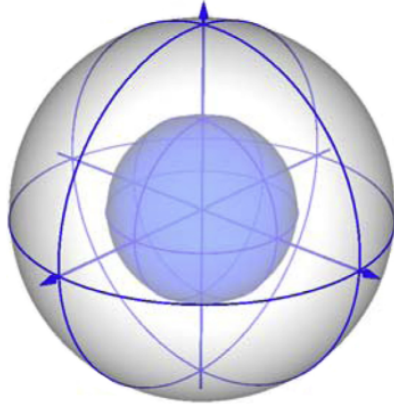


図 2.3 : SHOT 特徴量の空間分割例 [51]

ド型の記述子である [48, 51]. SHOT の LRF は, あるキーポイント \mathbf{k}_i の支持半径 Rd_{lrf} 内の点に対して, 距離に応じて重み付けした scatter 行列を

$$\begin{cases} \frac{1}{\sum_{j=1}^{N_{Rd}} (Rd_{lrf} - d_j)} \sum_{j=1}^{N_{Rd}} (Rd_{lrf} - d_j) (\mathbf{p}_j - \mathbf{k}_i) (\mathbf{p}_j - \mathbf{k}_i)^T, \\ d_j = \|\mathbf{p}_j - \mathbf{k}_i\| \end{cases} \quad (2.8)$$

により求め, 更に前述した EVD により得た 3 個の固有ベクトルを採用する.

特徴量の記述は, まず LRF を基準座標系としたうえで, あるキーポイントの支持半径内にある点分布を, 方位角, 仰角, 半径で分割し, それぞれの空間に分ける. 図 2.3 は, 方位角を 4 つ, 仰角を 2 つ, 半径を 2 つに分割し, 16 個の空間に分けたときの例である. 次に, 各空間内にある点とキーポイントについて, 法線ベクトルの相対角度, 方位角, 仰角, 距離をカウントしてヒストグラムを作成する. 最後に, 各空間のヒストグラムを並べて, SHOT 特徴量を得る.

2.3 特徴点によるレジストレーション

特徴点によるレジストレーション (Feature Point Registration: FPR) は特徴点の活用法の代表例であり, 点群間の特徴点对応から変換パラメータを推定する方式である. 後述する座標系変換を繰り返す方式とは異なり, 反復処理をせずに 1 度だけの座標系変換で処理を終了するため, 計算量を局限できる. 一方で, 1.3.2

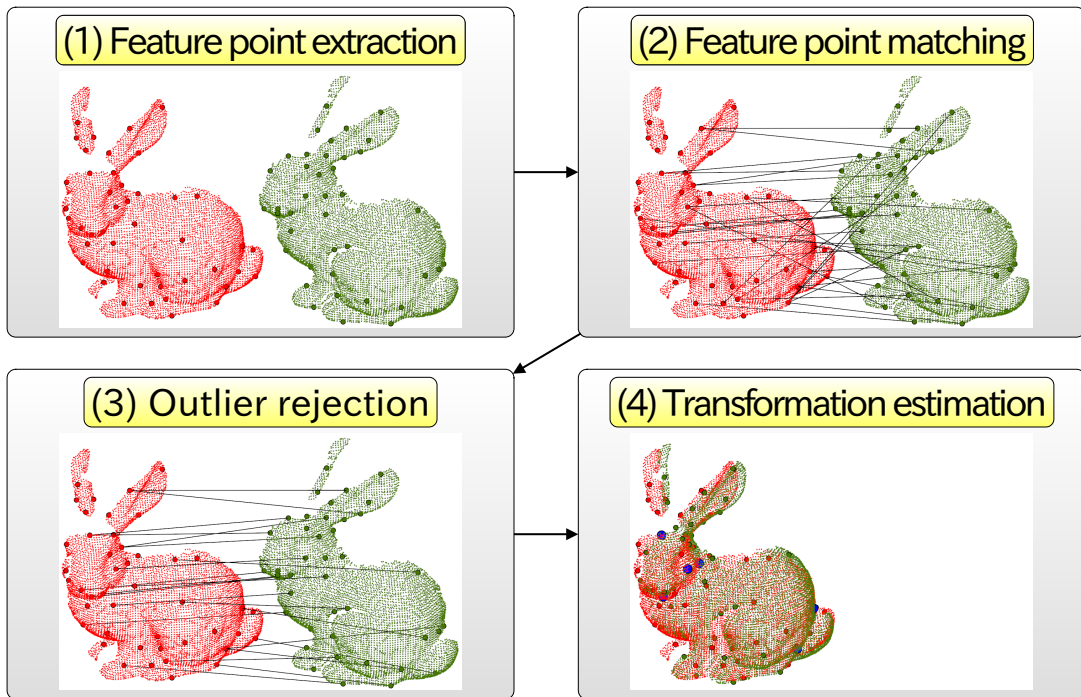


図 2.4 : FPR の処理フロー

項で挙げたとおり各種課題の影響を強く受けるため、高精度なレジストレーションは困難である場合が多い [28, 47, 48].

本論文では、まずレジストレーションを定式化し、その後に一般的な FPR の処理フロー (図 2.4) [54] をもとに詳細を述べる。ただし、特徴点抽出 (図 2.4(1)) は先に述べたとおりであるので省略する。

2.3.1 レジストレーションの定式化

点群は、センシング時におけるセンサの位置・姿勢を基準とした、固有の座標系に属している。すなわち、センサの位置・姿勢を変化させた場合、座標系が異なるため、複数の点群をレジストレーションするには、ある 1 つの点群の座標系を基準として、その他の点群を全て座標系変換する必要がある。このとき、基準となる座標系の点群をターゲット $\mathbb{T} = \{\mathbf{p}_i^t | i = 1, \dots, N_t\}$ 、その他の点群をソース $\mathbb{S} = \{\mathbf{p}_i^s | i = 1, \dots, N_s\}$ とする。

図 2.5 に示す位置関係で 2 回センシングしたとき、2 つの点群をレジストレー

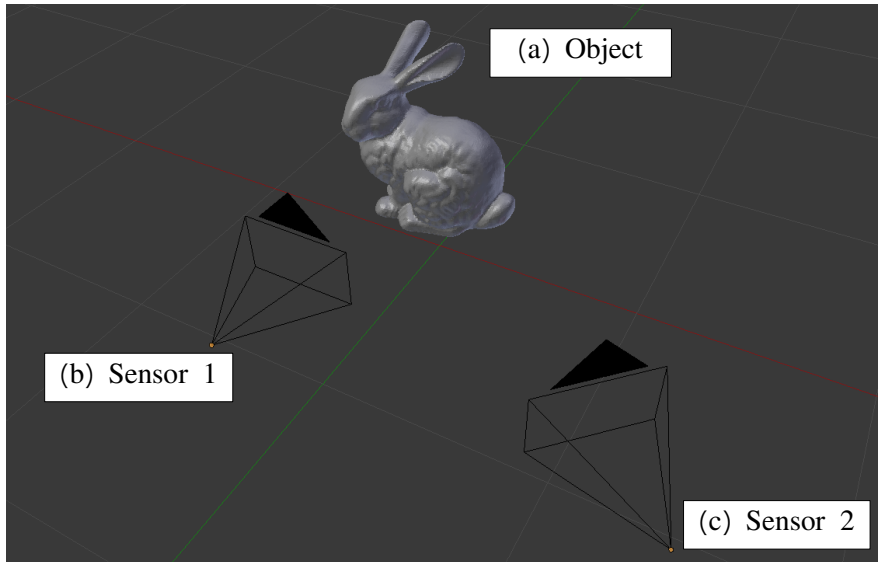


図 2.5 : センシングの位置関係の例

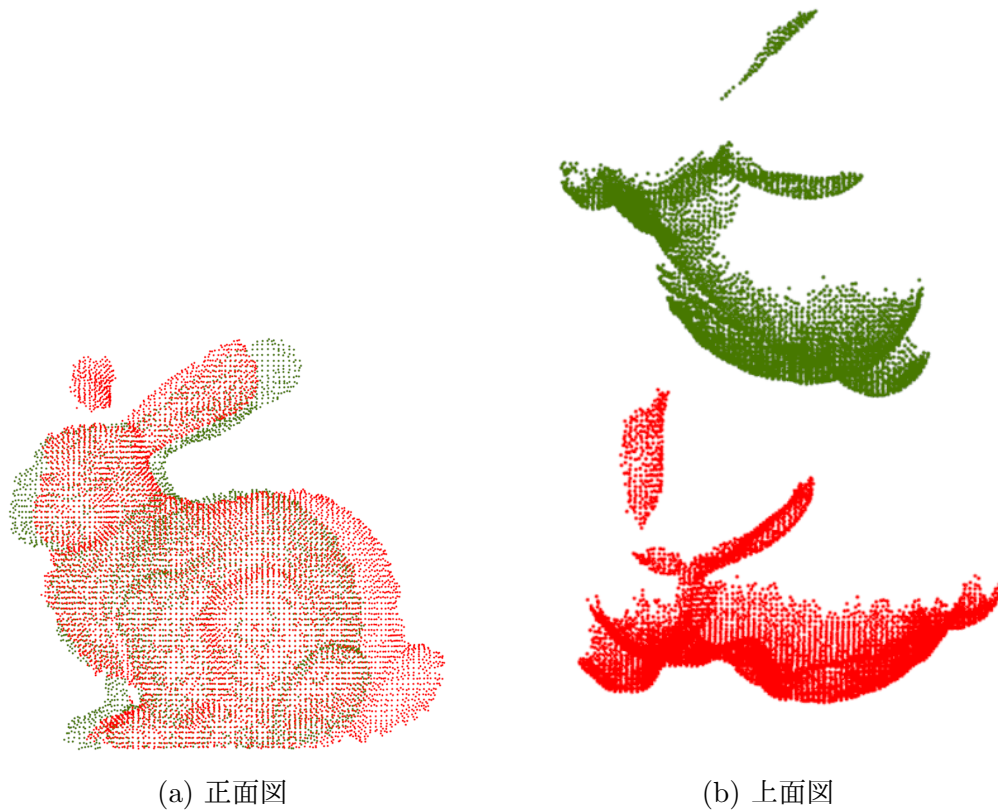


図 2.6 : 2つの点群の位置関係

ションせずに1つの座標系上で表示したときの例を図2.6に示す。図2.5(a)はStanford Bunnyのクローズな3Dモデルであり、(b)でセンシングした点群は図2.6の赤色で、(c)でセンシングした点群は図2.6の緑色で表示している。単純に1つの座標系上で表示しているのに、点群は重ならないが、センサ1と2の座標系は重なった状態にある。センサ1の赤色の点群を \mathbb{T} 、センサ2の緑色の点群を \mathbb{S} とすると、センサ2の座標系を図2.5(c)の位置へ回転・並進することで、 \mathbb{S} が正しい位置に重ね合わせられる。すなわち、レジストレーションは、この座標系変換のための3軸回転・3軸並進を含む6次元の変換パラメータを正しく推定することと等しい。

次に、座標系変換を定式化する。3軸回転量を r_x, r_y, r_z 、3軸並進量を t_x, t_y, t_z とすると、回転パラメータは 3×3 の回転行列

$$\mathbf{R} = \begin{pmatrix} \cos r_z & -\sin r_z & 0 \\ \sin r_z & \cos r_z & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos r_y & 0 & \sin r_y \\ 0 & 1 & 0 \\ -\sin r_y & 0 & \cos r_y \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos r_x & -\sin r_x \\ 0 & \sin r_x & \cos r_x \end{pmatrix} \quad (2.9)$$

で、並進パラメータは3次元のベクトル

$$\mathbf{t} = (t_x, t_y, t_z)^T \quad (2.10)$$

で表現できる[55]。また、 \mathbf{R} と \mathbf{t} は統合して 4×4 の変換行列

$$\mathbf{T} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \quad (2.11)$$

で表現できる。ある点 \mathbf{p} を座標系変換するには、

$$\begin{pmatrix} \mathbf{p}' \\ 0 \end{pmatrix} = \mathbf{T} \begin{pmatrix} \mathbf{p} \\ 0 \end{pmatrix} \quad (2.12)$$

により、任意に回転・並進した点 \mathbf{p}' が得られる。以降は便宜上、点 \mathbf{p} の座標系変換は $\mathbf{T}\mathbf{p}$ と表記する。

最後に、レジストレーションの精度を指標化する。変換パラメータの真値から求めた \mathbf{T}_g と、推定値から求めた \mathbf{T}_e があるとき、そのレジストレーション誤差 RE (Registration Error) は

$$RE(\mathbf{T}_g, \mathbf{T}_e, \mathbb{S}) = \sqrt{\frac{1}{N_s} \sum_{i=1}^{N_s} \|\mathbf{T}_g \mathbf{p}_i^s - \mathbf{T}_e \mathbf{p}_i^s\|^2} \quad (2.13)$$

で求める。これは、同じ点を \mathbf{T}_g と \mathbf{T}_e の両方で座標系変換し、そこで生じた離隔距離の総和平均である。高精度に変換パラメータを推定できた場合、 $RE = 0$ に近づく。

2.3.2 特徴点マッチング

特徴量マッチングは、類似する特徴量をもつ特徴点同士を対応付けする処理である (図 2.4(2))。 \mathbb{S} と \mathbb{T} のキーポイント集合 $\mathbb{K}^s, \mathbb{K}^t$ において、それぞれの特徴量を比較することで、特徴点对応 $(\mathbf{k}_i^s, \mathbf{k}_{cl}^t)$ を得る。 cl は対応付けられた \mathbb{K}^t のインデックスであり、

$$cl = \arg \min_{j \in \{1, \dots, N_{tk}\}} \|\mathbf{f}_i^s - \mathbf{f}_j^t\|^2 \quad (2.14)$$

を満たすものが選択される。ただし、この処理はベクトルの距離が最小の特徴点を対応付けただけで、他の特徴点对応が示す距離と比較し、特徴量の類似性が極端に悪い可能性がある。これは、距離に閾値を設けることで、類似性の低い特徴点同士の対応付けを回避できるが、特徴量が高次元であることが多いため適切な閾値の設定は困難である。そこで、人間の感覚的に理解しやすいコサイン類似度により閾値を設ける手法が用いられる。コサイン類似度 COS は

$$COS(\mathbf{f}_i^s, \mathbf{f}_{cl}^t) = \frac{\mathbf{f}_i^s \cdot \mathbf{f}_{cl}^t}{\|\mathbf{f}_i^s\| \|\mathbf{f}_{cl}^t\|}, \quad (-1 \leq COS \leq 1) \quad (2.15)$$

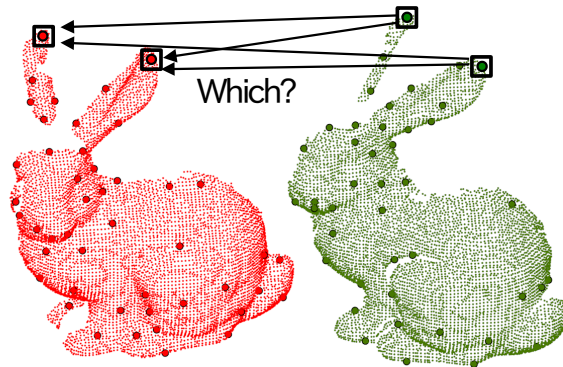


図 2.7 : 類似形状による誤対応

により 2 つの特徴量の向きが一致した場合には $COS = 1$, 直交した場合には $COS = 0$, 逆向きの場合には $COS = -1$ となる. すなわち, 1 に近い値を閾値に設定することで, 高い類似性の特徴点对応のみを残すことができる.

2.3.3 誤対応除去

誤対応除去は, 再現性のない特徴点の対応を除く処理である (図 2.4(3)). 特徴点マッチングにより得た特徴点对応には, 多くの誤対応 (Outlier) が含まれていることが多い. 各種課題によって生じたものはもちろん, 特徴的な形状が同一点群内に複数存在する場合は, 特徴量が近い値を示すためコサイン類似度でも弁別できない. 図 2.7 は, Stanford Bunny における 2 つの耳の先端の特徴点が, 互いに弁別できない例を示す. このようなときには, 互いに別の耳にある特徴点同士を対応付け, 2 個の誤対応が生起する可能性がある.

このような誤対応の除去のため, RANSAC (RANdom SAmple Consensus)[56] が用いられる. これは, モデルフィッティングによく用いられる手法 [57] であり, 無作為なデータ選択とモデルへのフィッティング精度の測定を反復することで, 最も多くモデルにフィットした点 (Inlier) を検出する. キーポイントの集合を RANSAC のモデルに見立てることで, 最大数の正対応を検出し, それ以外は誤対応として除去する.

図 2.8 はこの手法の実行例である. 後述する SVD 処理は, 3 個の対応があれば

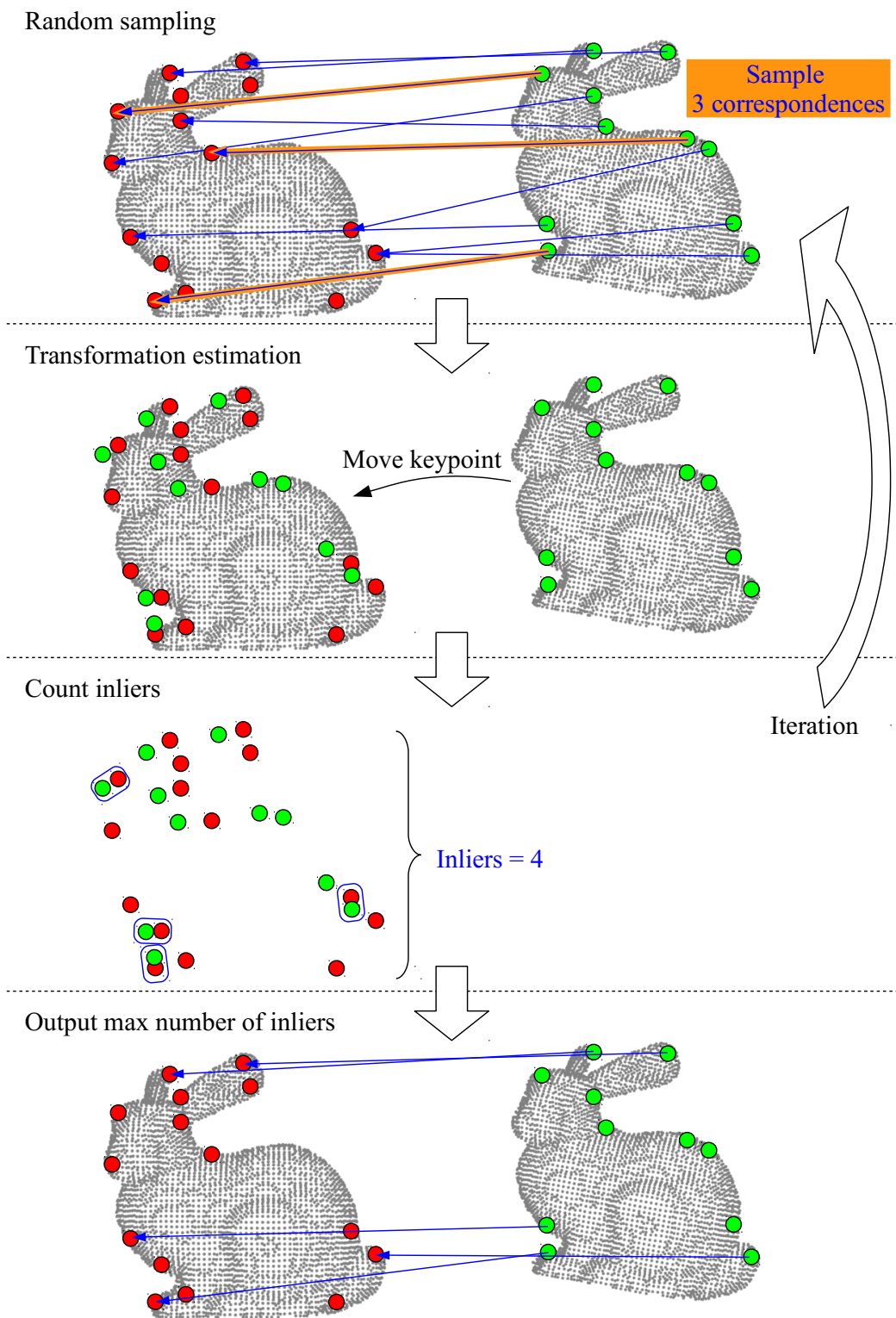


図 2.8 : RANSAC による誤対応除去

変換パラメータを推定可能であるので，無作為に3個の対応を選択し変換行列 \mathbf{T}_e を得る．そして， \mathbb{K}^s を \mathbf{T}_e で座標系変換し，

$$\|\mathbf{T}_e \mathbf{k}_i^s - \mathbf{k}_{cl}^t\|^2 < Th_{inlier}, cl = \arg \min_{j \in \{1, \dots, N_{tk}\}} \|\mathbf{T}_e \mathbf{k}_i^s - \mathbf{k}_j^t\|^2 \quad (2.16)$$

を満たす特徴点の数を数え上げる． Th_{inlier} は正対応とみなす特徴点の位置座標の距離の閾値である．これを反復し，数え上げた数が最大のときの特徴点对応のみを正対応として出力する．

2.3.4 変換パラメータの推定

誤対応除去処理後，残った特徴点对応だけを使用し，座標系変換に必要な回転・並進量を含む変換パラメータを得る（図2.4(4)）．特徴点对応 $(\mathbf{k}_i^s, \mathbf{k}_{cl}^t)$ から相互共分散行列 \mathbf{C} を形成し，SVD (Singular Value Decomposition)[58] により \mathbf{R} を求める． \mathbf{C} は，

$$\mathbf{C} = \frac{1}{N_{cor}} \sum_{i=1}^{N_{cor}} (\mathbf{k}_i^s - \bar{\mathbf{k}}^s)(\mathbf{k}_{cl}^t - \bar{\mathbf{k}}^t)^T \quad (2.17)$$

により求める． N_{cor} は対応の数， $\bar{\mathbf{k}}^s, \bar{\mathbf{k}}^t$ はそれぞれの点群内で対応付けられた全点の重心点を示す．さらに， \mathbf{C} はSVDにより，

$$\mathbf{C} = \mathbf{U} \mathbf{S} \mathbf{V}^T \quad (2.18)$$

に分解できる． \mathbf{U} は \mathbf{C} の入力基底となる正規直交行列， \mathbf{V} は \mathbf{C} の出力基底となる正規直交行列， \mathbf{S} は特異値の対角行列である．このとき，回転行列 \mathbf{R} は

$$\mathbf{R} = \mathbf{V} \mathbf{U}^T \quad (2.19)$$

により求めることができる．

Umeyamaはこの式を変形し，

$$\mathbf{R} = \mathbf{V} \mathbf{S}' \mathbf{U}^T \quad (2.20)$$

$$\mathbf{S}' = \begin{cases} \mathbf{I} & \text{if } \det(\mathbf{U}) \det(\mathbf{V}) = 1 \\ \text{Diag}(1, 1, \dots, 1, -1) & \text{if } \det(\mathbf{U}) \det(\mathbf{V}) = -1 \end{cases}$$

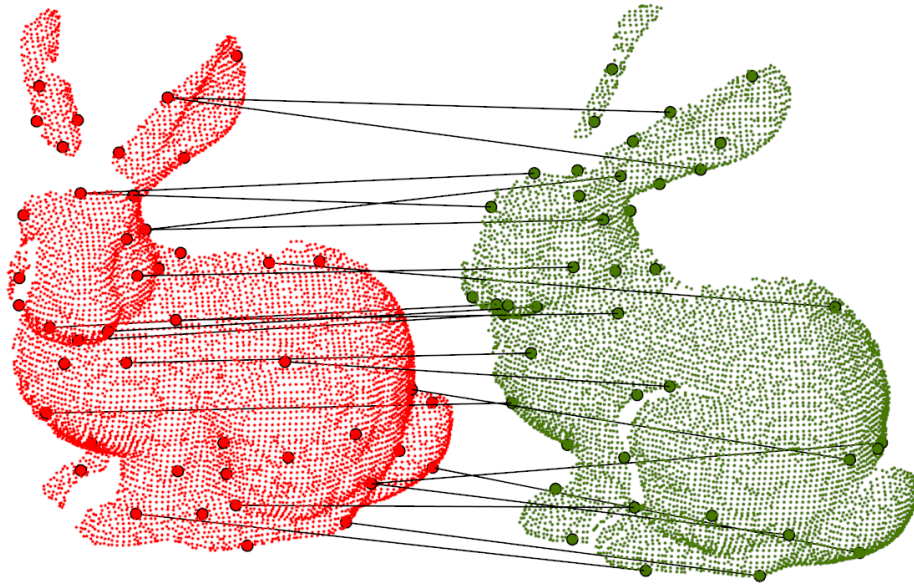


図 2.9 : 特徴点对応の例 (誤対応除去後)

により安定的に R を求める手法を提案した [59]. 残る並進ベクトル t は,

$$t = \bar{k}^t - R\bar{k}^s \quad (2.21)$$

で求める.

2.4 従来の特徴点抽出の課題

FPR の精度は, 全ての処理の起点であるキーポイントの性能, 特に再現性の有無に強く依存するといつてよい. 再現性が全くない特徴点对応が残り続けた場合, この誤対応がレジストレーション精度に与える影響は大きい. 図 2.9 に示す誤対応除去処理後の特徴点对応について, SVD で変換パラメータを求めてレジストレーションした結果を図 2.10 に示す. 図 2.10(a) は正面図, (b) は右からの側面図, (c) はターゲット点群のオクルージョン位置に抽出された特徴点を示す. 図内の太い黒線は, 特徴点对応を結んだものである. すなわち, 黒線が長いキーポイントレジストレーションに強い悪影響を与えているといえる.

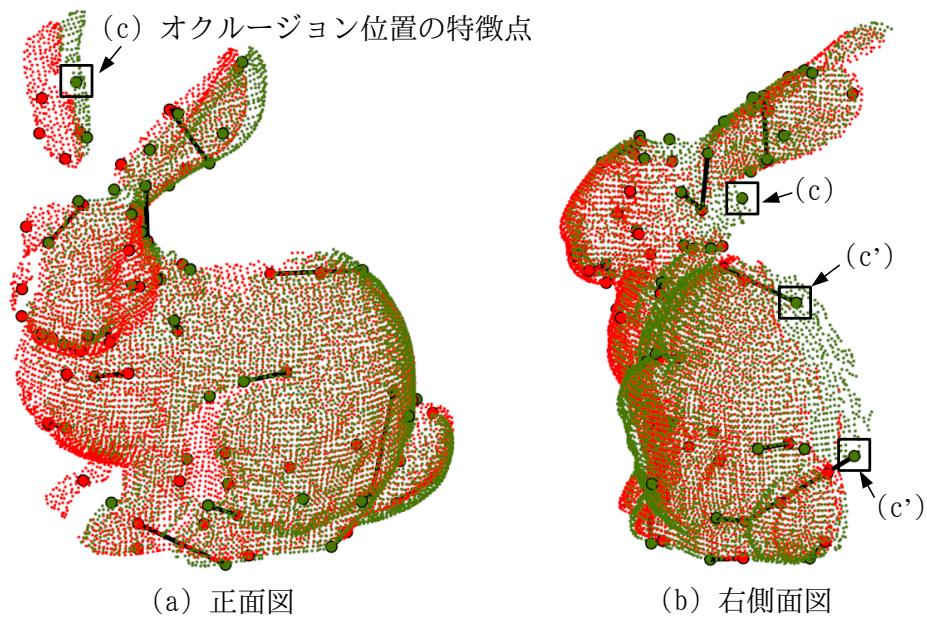


図 2.10 : レジストレーション後の特徴点对応

特に (c') は、オクルージョンの位置に抽出されたにもかかわらず、対応が残ってしまった例である。この位置の特徴点は、正対応の取れるターゲット側の特徴点が存在しないため、そもそも抽出しないように作為する必要がある。また、重なりのある領域であっても、ノイズによる位置変動や、解像度の違いによって、特徴点の位置には必ず差異が生じる。しかも、突出した形状は表面積が小さいためノイズの影響を受けやすく、解像度も荒くなることから、キーポイントの再現性には限界があるといえる。

以上を総合すると、従来の特徴点は弁別性を重視するため、突出した形状の先端の点を探索するアプローチをとっていたが、外部要因により再現性が犠牲になっていた。一方で、レジストレーションに特徴点を活用することを考えると、弁別性よりも再現性を重視する必要があるといえる。

第3章

特徴点对応によらないレジストレーション

特徴点を用いないレジストレーションについて述べ、課題の解決法を考察する。

3.1 はじめに

1.4.1 項で特徴点对応によらないレジストレーションの概要を述べたが、この代表的な方式である、近傍点探索による座標系変換を繰り返す方式と、進化計算により変換パラメータを探索する方式の詳細を述べる。次に、従来方式の長所と短所を比較し、その課題と解決法を考察する。

3.2 近傍点探索による座標系変換を繰り返す方式

Besl らの提案した ICP (Iterative Closest Point) アルゴリズム [40] はレジストレーションの最も代表的な方式であり、 \mathbb{S} から \mathbb{T} への近傍点探索と重ね合わせの精度評価を収束するまで反復することで高精度な変換パラメータを得る。その長所は法線等の特徴量が不要であることや、収束性が保証されていることが挙げられる。一方、短所としては近傍点探索という計算量の大きい処理を繰り返す必要があることと、収束した位置が局所解である可能性があることが挙げられる。このような課題の克服のため、多くの派生方式が提案されている [60–62]。本論文では、図 3.1 に示した Besl の方式 [40] の処理フローをもとに述べる。

3.2.1 初期位置合わせ

初期位置合わせは、反復処理前にある程度近い位置に座標系変換しておく処理である (図 3.1(1))。ICP は単調に収束することが証明されているが、全体の最適解が得られることは保証されておらず、局所解に陥っている可能性がある。どの解に収束するかは ICP 開始時の初期位置に依存するので、初期位置合わせに

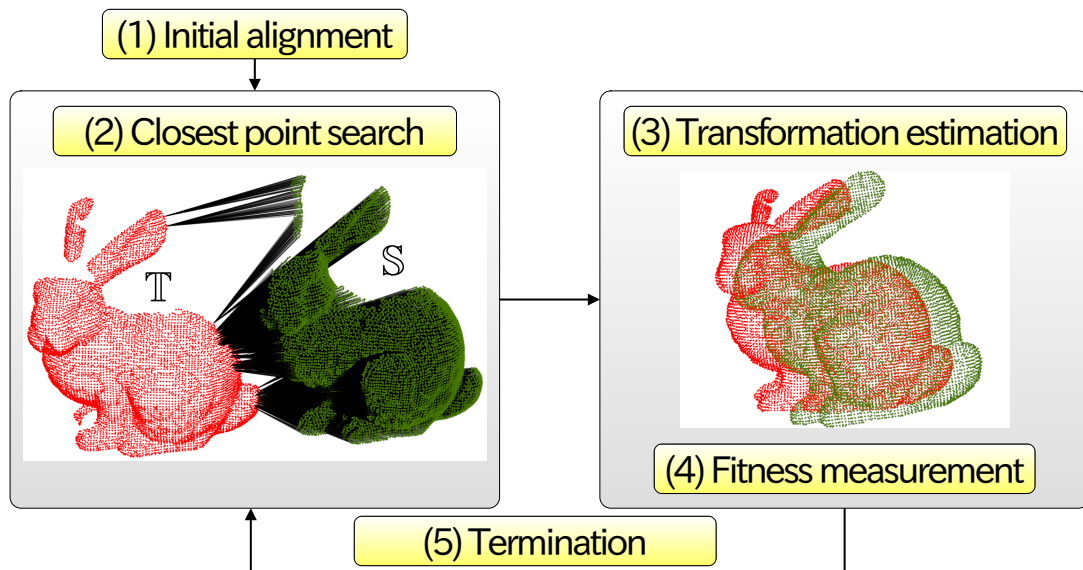


図 3.1 : ICP の処理フロー

よってこの問題を回避する．初期位置合わせのための変換パラメータは，外部システム等から得ることも可能であるが，本研究の範囲からみたときは3次元動きベクトルによる方式 [63–65] や，先に述べた特徴点对応による方式を用いることが一般的である．このように，点群のみから変換パラメータを粗く求め，その後細かく位置合わせする方式は Coarse-to-Fine アプローチと呼ばれる [39]．

初期位置合わせにおいて求めた変換行列は $\mathbf{T}_{k=0}$ とする． k は反復回数であり，次に示す反復処理で \mathbf{T}_k を更新していく．なお，初期位置合わせをしない場合は， $\mathbf{T}_{k=0} = \mathbf{I}$ とおく．

3.2.2 近傍点探索

近傍点探索は， \mathcal{S} の全ての点から \mathcal{T} の最も近い位置にある点を探索し，対応付けする処理である (図 3.1(2))．これは，前述のとおり計算量の大きい処理であるので，FLANN (Fast Library for Approximate Nearest Neighbors)[66, 67] のような高能率な手法が望ましく，ICP の派生方式ではよく用いられている [68–70]．また対応付けについて，Besl らの方式では図 3.2 に示すような point-to-point の

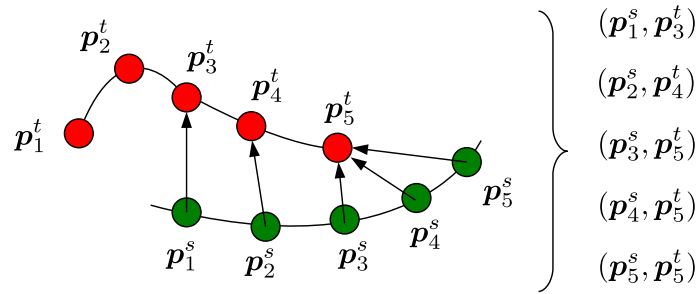


図 3.2 : 近傍点探索の例

対応 $(\mathbf{p}_i^s, \mathbf{p}_{cl}^t)$ を用いている。なお、 cl は対応付けられた \mathbf{p}_j^t のインデックスであり、

$$cl = \arg \min_{j \in \{1, \dots, N_t\}} \|\mathbf{T}_k \mathbf{p}_i^s - \mathbf{p}_j^t\|^2 \quad (3.1)$$

を満たすものが選択される。図 3.1(2) において、 \mathbb{S} から \mathbb{T} へ伸びている黒線は、全ての近傍点对応 $(\mathbf{p}_i^s, \mathbf{p}_{cl}^t)$ を示したものである。

対応付けについて、この発展形として点と接平面を対応付ける point-to-plane[71] や、距離画像の同一画素を対応付ける point-to-projection[72] 等がある。文献 [60] では、point-to-plane が最も高精度であり、point-to-projection が最適解ではないものの低計算量であり、point-to-point はノイズの影響を強く受けるが必ず収束するとしている。

3.2.3 変換パラメータの推定

変換パラメータの推定は、全ての近傍点对応 $(\mathbf{p}_i^s, \mathbf{p}_{cl}^t)$ から座標系変換に必要な回転・並進量を求める処理である (図 3.1(3))。この処理は前述の 2.3.4 項と同じであり、一般的には SVD が用いて推定の変換行列 \mathbf{T}_k を更新する。

3.2.4 適応度の測定

適応度の測定は、 \mathbf{T}_k で \mathbb{S} を座標系変換したとき、どの程度 \mathbb{T} に重なり合っているかを評価する (図 3.1(4))。この値を適応度 FS (Fitness Score) とし、Besl らの方式では

$$FS_{sum}(\mathbf{T}_k, \mathbb{S}, \mathbb{T}) = \sum_{i=1}^{N_s} d_i \quad (3.2)$$

により求める． d_i は $\mathbf{T}_k \mathbf{p}_i^s$ から \mathbb{T} の最近傍点への距離であり，式 (3.1) を満たす

$$d_i = \|\mathbf{T}_k \mathbf{p}_i^s - \mathbf{p}_{cl}^t\|^2 \quad (3.3)$$

である．一般的には FS は小さいほど適応度が高いとみなせるが，真値との誤差 RE とは異なり， FS の最小値が最適な \mathbf{T} であるとは限らないことに注意が必要である．なお，適応度の算出は終了判定に至るまで反復処理し，適応度同士で比較するため，平方根や平均を求める必要はない．

3.2.5 終了判定

終了判定は，反復処理を打ち切るための処理である (図 3.1(5))．真値の変換行列 \mathbf{T}_g が既知である場合は，これに近づくまで反復処理をすればいいが，このためには外部システム等が必要である．そこで Besl らは， FS の変化量に閾値 Th_{FS} を設定し，

$$FS_k - FS_{k-1} < Th_{FS} \quad (3.4)$$

を満たしたとき，推定の変換行列を $\mathbf{T}_e = \mathbf{T}_k$ として終了することとしている．

この他に，最大反復回数を設定することも可能である．Besl らは，実験的に 30 から 50 回程度の反復により， FS が \mathbb{T} の大きさの約 0.1% の誤差でレジストレーションできるとしている [40]．

3.2.6 ICP の課題

ICP によるレジストレーション精度と計算量は，トレードオフの関係にある．ただし，ICP に要する計算量は反復回数に強く依存するので，初期位置合わせや近傍点探索の処理精度を高めることで計算量を削減できる．例えば変換パラメータの推定では，SVD の代わりに LM (Levenverg Marquadt) 法を用いて非線形的に解くことにより，高精度に求めることができる [73]．1 回の反復における個々の処理をみると，近傍点探索と適応度評価の両方の計算量が大きい．これは， S の全点の近傍点探索を行っているためであり，点群を成す点数を削減したりすることで低計算量化に貢献できる．

一方で、これらの手法は局所解に陥るか否かという、初期位置の依存性という課題に対する貢献度が低い。これを回避するには、適切な位置に初期位置合わせする必要があるが、先に述べた特徴点による方式を初期位置合わせに採用した Coarse-to-Fine アプローチが一般的である。すなわち、ICP を高能率化するには、特徴点による方式を高精度化・低計算量化することが求められるといえる。

3.3 進化計算により変換パラメータを探索する方式

進化計算による方式 (Evolutionary Computation Registration: ECR) は、進化計算で更新した変換パラメータを用いる再帰的方式である。解を得る手法が進化計算アルゴリズムによるため、付近の局所解を回避しつつ、全体最適解を探索することができる。すなわち、ECR は ICP などとは異なり、初期位置に依存せず高精度な変換パラメータを推定できる方式である [74]。しかしながら、遺伝的アルゴリズムを採用した ECR [75] は、ICP といった ECR 以外の方式と比較して莫大な処理時間を要することが示されている [61]。これは、*FS* 計測のための近傍点探索の回数が、ICP 等よりもはるかに多いことが要因と考えられる。

本論文では、図 3.3 に示す一般的な ECR の処理フロー [74] をもとに述べる。次に、ECR に適当な進化計算アルゴリズムである Differential Evolution (DE) [76] 及びこの発展形である Self-Adaptive Differential Evolution (SADE) [77] について概要を述べる。

3.3.1 ECR の処理フロー

(1) 個体の生成

個体の生成は、進化させる値を用意する処理である (図 3.3(1))。ランダムな値の要素をもつ個体 \boldsymbol{x}_i^g を N_p 個生成し、母集団を成す。 i は個体数の、 g は世代数のインデックスである。レジストレーションで求めるべき変換パラメータは 3 軸回転・3 軸並進の 6 次元であることから、 $\boldsymbol{x} = \{r_x, r_y, r_z, t_x, t_y, t_z\}$ とする。このとき、生成する \boldsymbol{x} の要素の範囲と、その個数 N_p は大きいほどレジストレー

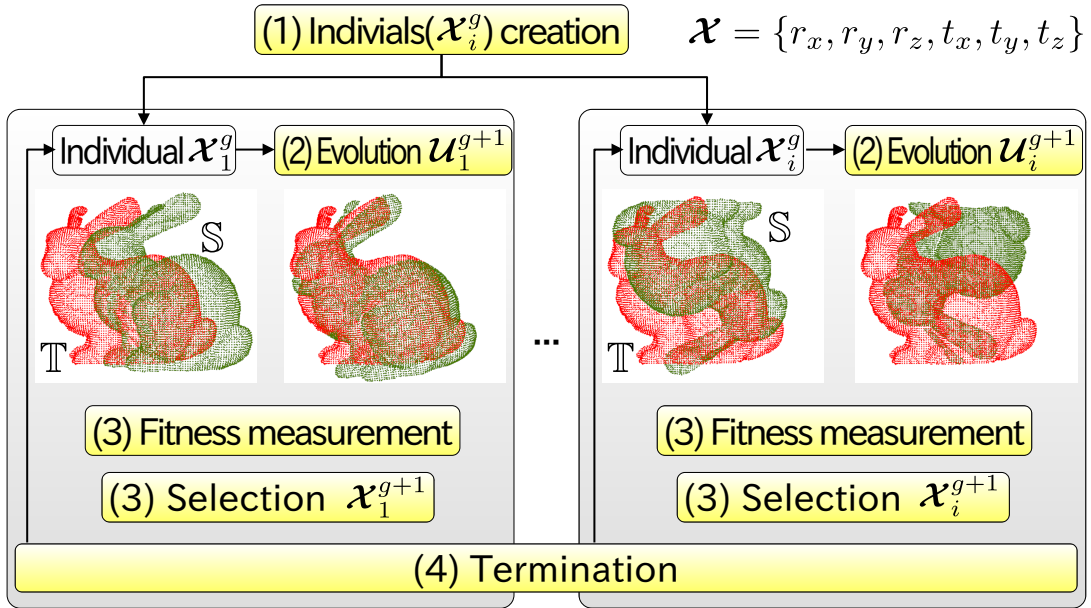


図 3.3 : ECR の処理フロー

シオン精度を高めることができるが、その分の計算量が増大するため、設定には注意が必要である。

(2) 個体の進化

個体の進化は、 \mathbf{x}_i^g の要素の値を後述する進化計算アルゴリズムにより変化させ、進化個体 \mathbf{u}_i^{g+1} を生成する (図 3.3(2))。最も代表的な遺伝的アルゴリズムでは、一定の割合で個体の突然変異や、個体同士の交叉を作為することで値を進化させる。

(3) 適応度の測定と選択

適応度の測定と選択は、進化させた個体が適当かどうかを判定する処理であり、 \mathbf{x}_i^g と \mathbf{u}_i^{g+1} の FS をそれぞれ測定する (図 3.3(3))。ある個体 \mathbf{x} から得る変換行列を \mathbf{T}_x とするとき、式 (3.3) を満たす

$$FS_{med}(\mathbf{T}_x, \mathbf{S}, \mathbf{T}) = \text{Median}(d_i), \quad (i = 1, \dots, N_s) \quad (3.5)$$

により求める [78]。Median() は中央値を返す関数である。ICP と異なり FS_{sum} を用いないのは、オクルージョンによる欠損の影響を低減するためである [79]。

次に、 \boldsymbol{x}_i^g と \boldsymbol{u}_i^{g+1} の FS を比較し、適した方を次世代の個体 \boldsymbol{x}_i^{g+1} とする。このとき、選択した方の FS を保存しておき、計算量が増大しないように取り扱うべきである。

(4) 終了判定

終了判定は、次世代への移行を打ち切るかを判定する (図 3.3(4))。ICP で採用されている手法のほか、ECR の計算量が大きいくこと鑑みて実際の処理時間で打ち切る場合もある。

3.3.2 ECR に適当な進化計算アルゴリズム

(1) Differential Evolution

進化計算アルゴリズムは複数存在するが、そのうちの 1 つである、Differential Evolution (DE)[76] について述べる。DE を用いた ECR は、文献 [74] の最も条件の厳しいデータセットにおいて、レジストレーション精度の平均値が他のアルゴリズムよりも高く、ECR に適しているといえる。DE の個体進化プロセスである突然変異と交叉について、概要を図 3.4 に示す。

DE において、現個体 \boldsymbol{x}_i^g の進化は、同じ世代の他の 3 つの個体 $\boldsymbol{x}_{R_1}^g, \boldsymbol{x}_{R_2}^g, \boldsymbol{x}_{R_3}^g$ 、突然変異係数 $F \in [0, 2]$ 及び交叉係数 $C \in [0, 1]$ によって算出される。 F と C は、

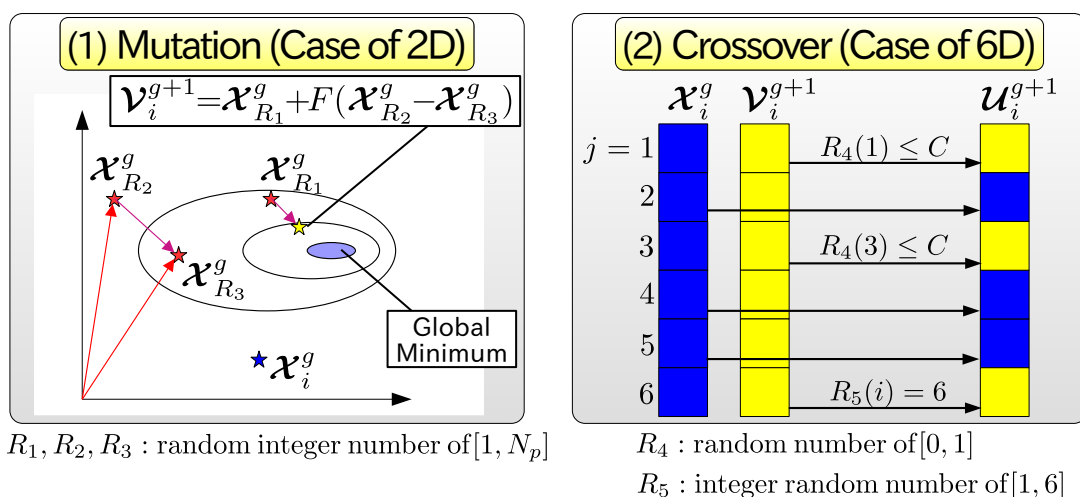


図 3.4 : DE の突然変異・交叉の概要

自由に設定できるパラメータである． $R_1, R_2, R_3 \in [1, N_p]$ は，母集団からランダムに選択された個体のインデックス ($R_1 \neq R_2 \neq R_3 \neq i$) である．このとき，進化個体 \mathbf{u}_i^{g+1} を生成するための突然変異個体 \mathbf{v}_i^{g+1} を，

$$\mathbf{v}_i^{g+1} = \mathbf{x}_{R_1}^g + F(\mathbf{x}_{R_2}^g - \mathbf{x}_{R_3}^g) \quad (3.6)$$

によって得る．

図3.4(1)は， \mathbf{x} の要素数が2次元であるときの \mathbf{v}_i^{g+1} の生成例を示す． \mathbf{x}_i^g とは異なる他の3つの個体と F により，全体最適解に近づく \mathbf{v}_i^{g+1} が生成されている．

次に， \mathbf{x}_i^g と \mathbf{v}_i^{g+1} を交叉して， $\mathbf{u}_i^{g+1} = \{\mathbf{u}_{(j,i)}^{g+1} | j = 1, 2, \dots, 6\}$ を

$$\mathbf{u}_{(j,i)}^{g+1} = \begin{cases} \mathbf{v}_{(j,i)}^{g+1} & \text{if } R_4(j) \leq C \text{ or } R_5(i) = j \\ \mathbf{x}_{(j,i)}^g & \text{otherwise} \end{cases} \quad (3.7)$$

によって得る． j は個体の要素のインデックス， $R_4(j) \in [0, 1]$ は j ごとに得る乱数， $R_5(i) \in \{1, 2, \dots, 6\}$ は i ごとに得る整数乱数である． R_4 で得られた乱数によっては，全て現個体の要素が選択される可能性があるが，他方で R_5 により必ず1つは交差するように作為されている．

図3.4(2)は， \mathbf{u}_i^{g+1} の生成例を示す．この図では， \mathbf{v}_i^{g+1} の第1,3要素が R_4 によって，第6要素が R_5 によって選ばれ，残りは \mathbf{x}_i^g の要素から成る \mathbf{u}_i^{g+1} が生成されている．

(2) Self-Adaptive Differential Evolution

Self-Adaptive Differential Evolution (SADE)は，DEのパラメータである F と C を固定せず，適応的に値を更新するアプローチの手法である [77]．DEの両パラメータは，与えられる課題によって適当な値が変わる [80]ため，SADEの採用はパラメータの変更による試行錯誤を削減できる．また，得られる最適解はDEと同等以上であることが確認されている [77]．

両パラメータは、次式を用いて個体ごとにする。

$$F_i^{g+1} = \begin{cases} F_l + R_6 \times F_u & \text{if } R_7 < \tau_1 \\ F_i^g & \text{otherwise} \end{cases} \quad (3.8)$$

$$C_i^{g+1} = \begin{cases} R_8 & \text{if } R_9 < \tau_2 \\ C_i^g & \text{otherwise} \end{cases} \quad (3.9)$$

$R_6, R_7, R_8, R_9 \in [0, 1]$ は乱数, F_l, F_u はコントロールパラメータ, τ_1, τ_2 は更新率である。これらのパラメータは, $F_l = 0.1, F_u = 0.9, \tau_1 = \tau_2 = 0.1$ が適当であるとされている [77].

3.3.3 ECR の課題

本方式の課題は、他の方式と比較して計算量が莫大であることといえる。計算量の大きい適応度評価を、終了判定に至るまで個体数と世代数の回数分実行しなければならない。計算量を低減するには、点群を成す点の個数を削減したり、または個体数・世代数を限定する必要がある必要がある。しかしながら、個体数の限定は局所解に陥る可能性を高めるとともに、収束性を悪化させる可能性まである。世代数の限定は、全体最適にたどり着く前の収束途中で終了してしまう可能性がある。すなわち、ECR の課題を解決するには、レジストレーション精度が悪化しない程度に点群を成す点の個数を削減する必要があるといえる。

3.4 従来方式の課題整理

ICP の大きな課題は初期位置の依存性であり、これを克服しない限り、局所解に陥り続ける可能性がある。それ以外は、計算量の多い近傍点探索を繰り返すため、計算量は大きくなる一方、外部阻害要因の耐性は向上し、これはトレードオフの関係にある。初期位置依存性の解決には、ICP に先んじて高精度な特徴点対応を用いたレジストレーションで初期位置合わせをする、Coarse-to-Fine アプローチの採用が適当である。

ECR は、進化計算によって適応度評価を個体数・世代数の回数分繰り返すため、ICP と比べても計算量は莫大となる。しかしながら、個体数・世代数の限定は局所解に陥る可能性と、また全体最適にたどり着く前に終了してしまう可能性を高める。すなわち、レジストレーション精度が悪化しない程度に、点群を成す点の個数を削減するのが適当である。

第4章

仮想特徴点抽出法

再現性重視の特徴点を抽出するため，安定的な表面形状から求めた位置座標に特徴点を配置する，仮想特徴点抽出を提案する．

4.1 本章の概要

本章では，特徴点の性能のうち，再現性に着目した．従来の特徴点は，弁別性を重視した位置の点を採用していたため再現性に課題があり，特徴点を活用した各種方式へ悪影響を与えていた．特にFPRは特徴点を活用した代表例であり，ICPの前処理としても使用される（Coarse-to-Fineアプローチ）．レジストレーションのみならず，特徴点を用いた各種応用技術の性能は，正しい特徴点对応をどれだけ取得できるかに依存する．すなわち，特徴点の再現性向上は点群処理全体の精度向上に貢献するといえる．

特徴点の再現性を左右するのは，その位置座標を決定するキーポイント検出処理である．そこで，点群の各種課題を考慮し，点群を成す実在点から探索せず，まずは安定的に得られる形状に着目した．この形状から方程式を取得し，複数の方程式を組み合わせて仮想的な位置座標にキーポイントを配置する，仮想特徴点抽出法を提案した [81–83]．特に点群内の平面形状に着目し，3平面の交点位置に配置する手法 [81, 83] と，これに特徴量を記述するための手法も提案した [82, 83]．

平面を含む環境を想定したシミュレーション実験により，従来の特徴点抽出法と同等の処理時間で，最大2倍程度の高い再現性を示すことを確認した [83]．

4.2 従来法の課題と解決方針

2.4節で述べたとおり，特徴点抽出の課題はキーポイントの再現性の向上である．しかしながら，弁別性を考慮して突出した形状の先端に位置する実在点は，

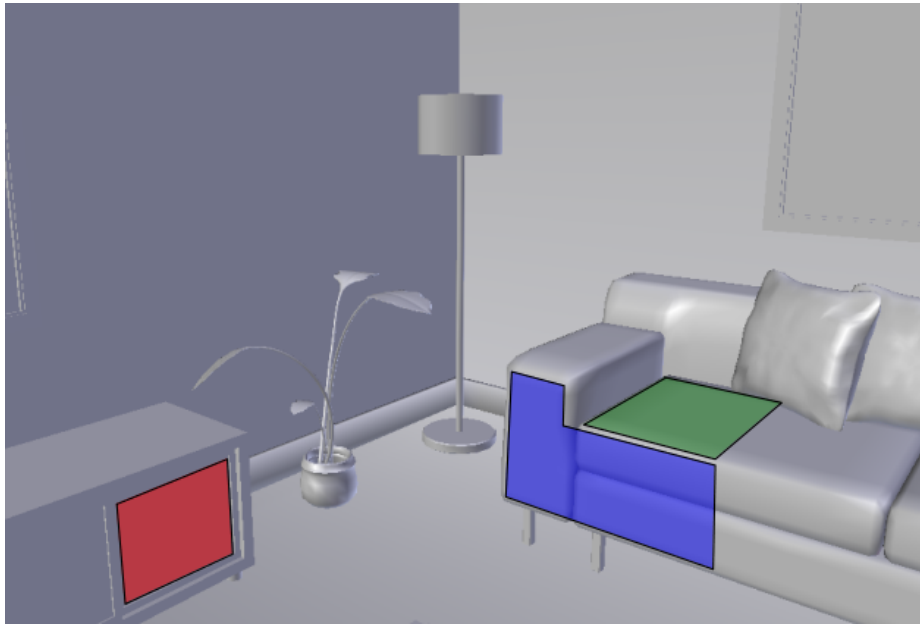
ノイズや解像度により変動が生じており，そもそも再現性が高いとはいえない．そのような位置変動を含んだ点を採用し，少数個の特徴点对応をレジストレーション等に活用するため，誤差の平均化の効果は期待できない．さらに，ターゲット点群のオクルージョン領域に特徴点が抽出されたり，マッチングエラー等により誤対応が生起したりした場合には，この影響がより強くなる．すなわち，実在点を含めて，点群のなかで再現性のあるものを抽出し，キーポイント検出に用いることが課題解決の方針といえる．

4.3 平面方程式による仮想特徴点 (VKOP) 抽出

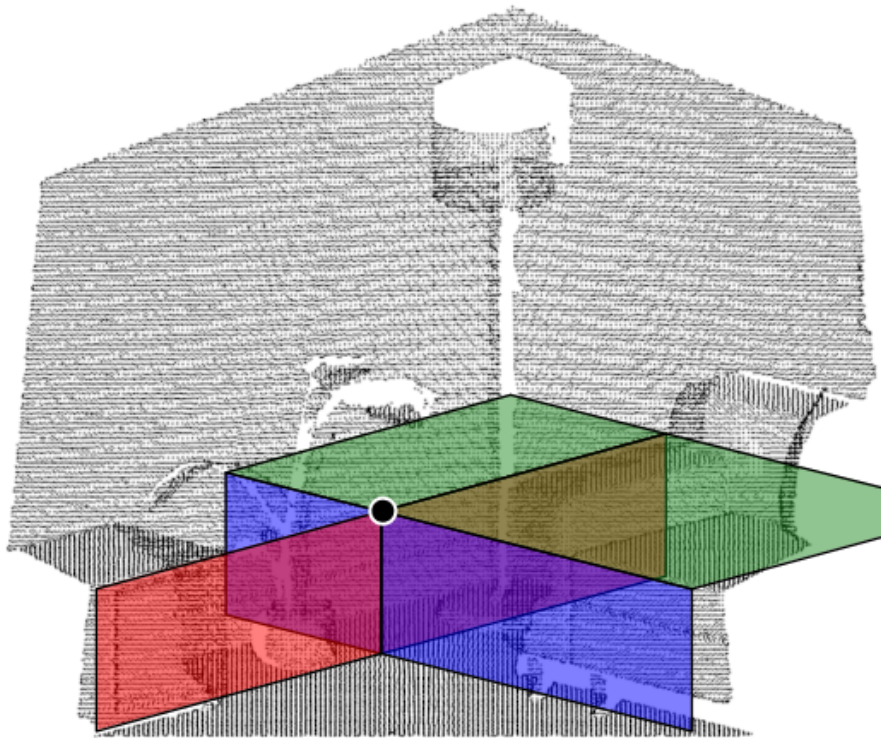
点群のなかで再現性の高いものとして，平面形状を成す点分布に着目した．この領域は，一般的に面積が大きいので点数を多く取得可能であり，またセンシングが容易であるためノイズによる位置変動が小さい [32, 84] という特長がある．また，平面はセンサの視野限界による両端のオクルージョン，または障害物による形状内のオクルージョンが生起しても，幾何学的に変化することはない．すなわち，平面形状を方程式化することにより，同一の平面が僅かでもセンシングされ続ける限り，この平面方程式は高い再現性を示すといえる．そしてこの平面形状は，ほとんどの人工物の表面を形成しており，室内や都市内といった環境に多く存在するため，この場合に多くの平面方程式が得られる．

そこで，この平面方程式の高い再現性を活用し，3つの平面方程式の交点位置にキーポイントを配置する仮想特徴点を提案した．このとき，点群を1つの大きな多面体とみなして点分布を分析することから，この抽出法による特徴点を Virtual Keypoint Of Polyhedron (VKOP) と呼ぶ．

ICL-NUIM RGB-D Benchmark Dataset [85] のリビングを模したモデル (living-room) [86] を用いて取得した点群から，1個のVKOPを検出した例を図4.1に示す．図4.1(a)における赤・緑・青の半透明の平面 (キャビネットの側面，ソファの側面，ソファの上面) は，図4.1(b)の平面と対応しており，この3枚の平面の



(a) ソリッドモデル



(b) VKOP (黒色の丸)

図 4.1 : VKOP の検出例

交点位置に VKOP がある。また、図 4.1(b) の 3 枚の平面は、VKOP 検出位置を超えて左右の壁と床まで伸ばしてあり、すなわち例示した VKOP の位置は点群から完全に離れた位置にある。

4.4 VKOP 抽出の処理フロー

VKOP の性能は、平面方程式の再現性に依存することとなるので、高精度に推定する必要がある。しかしながら、理想的にきれいな平面だけが点群内に存在するわけではなく、なだらかな曲面や、ノイズ等によって起伏の生じた平面も多く点群に含まれることとなる。このようなときの平面推定の例を図 4.2 に示す。この図は、点群からの平面の抽出について、2 次元的に表現した図である。黒丸は 1 つの点、赤線は抽出が予想される平面の例を示す。図 4.2(a) は、きれいな平面部分を含むため、1 枚の平面のみを抽出可能といえる。一方で図 4.2(b) のなだらかな曲面や図 4.2(c) の起伏のある平面では、平面を取りうる部分が数多く存在するため、VKOP の再現性が低下する可能性がある。これを回避するには、後者 2 つの部分から平面を抽出しないように、または安定的に平面が抽出できるように、結果を毎回確認しながらパラメータの微調整を繰り返す必要がある。

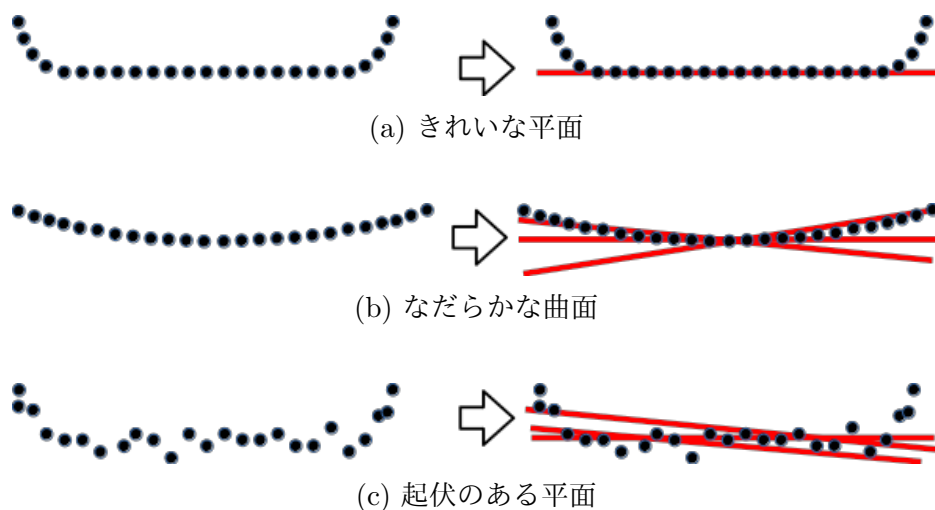


図 4.2 : 平面の抽出例

本章では、RANSAC ベースの高精度な平面推定法を提案する。また、キーポイントに記述する特徴量については、従来方式ではキーポイントの周囲の点分布から計算していたが、VKOP は図 4.1(a) に示すように周囲に点が存在しない場合がある。そこで、平面方程式のみを用いた専用の特徴量記述法もあわせて提案する。各提案方式の処理手順に従って、平面方程式推定、キーポイント検出（仮想的な位置への配置）、特徴量記述の順に詳細を述べる。

4.4.1 平面方程式の推定

VKOP は、まず点群全体から複数の平面部分を抽出し、個々に平面方程式を推定する必要がある [81]。本章では、平面部分の抽出に、法線で重み付けした RANSAC [56, 57] による平面モデルフィッティングを用いることとした。ただし、離れた位置にある点群を同一の平面とみなしたり、滑らかな曲面を平面とみなすことがあるため、その対策として、ユークリッドクラスタリング [87] と段階的な平面モデルフィッティングを組み合わせた高精度推定法を提案した。粗く平面を取得し、次にパラメータを厳しくして平面を推定することから、この提案手法を CFRANSAC (Coarse to Fine RANSAC) と呼ぶ。

CFRANSAC は、次の処理を繰り返し適用する。

- step1** 支持半径 Rd_{nor} を用い、点群 \mathbb{C} の全点に法線を付与する。
- step2** 平面と点の距離閾値 Th_c を用い、法線で重み付けした RANSAC 処理により \mathbb{C} から粗い平面候補の点群 \mathbb{P}^c を抽出する。
- step3** \mathbb{P}^c が 2 つ以上の平面を含む可能性があるため、ユークリッドクラスタリングにより分割する。以後、分割された点群の 1 つを \mathbb{P}_i^c と表記する。
- step4** \mathbb{P}_i^c に対し、距離閾値 $Th_f = C_{cf}Th_c$, ($C_{cf} = 0.5$) を用いて再度 RANSAC 処理し、より高精度な平面候補の点群 \mathbb{P}_i^f を得る。
- step5** \mathbb{P}_i^c の点数を N_{pc} , \mathbb{P}_i^f の点数を N_{pf} とするとき、この点数比 $R_{cf} = N_{pf}/N_{pc}$ が閾値 $Th_{cf} = 0.8$ を下回る場合は、 \mathbb{P}_i^f を除去する。

step6 除去されずに残った \mathbb{P}_i^f から平面方程式

$$a_i x + b_i y + c_i z + d_i = 0 \quad (4.1)$$

を得る.

step7 step3で複数の平面に分割されていた場合はstep4に戻り, 全ての \mathbb{P}_i^c に処理を適用する.

step8 \mathbb{C} から \mathbb{P}^c を除去して \mathbb{C}' とし, step2に戻る. ただし, 元の点数と処理済みの点数の割合 $N_{c'}/N_c$ が閾値 Th_{pp} を下回ったとき, またはループの回数 N_{cp} を超えたとき, この処理を終了する.

step5における, 平面候補の除去判定の例を図4.3に示す. 橙の点は1度目の Th_c で平面とされたもの, 青の点は2度目の Th_f で平面とされたものを表す. きれいな平面の部分が多い図4.3(a)と比較し, 図4.3(b)や図4.3(c)の場合は青の点数が少なくなり, R_{cf} が小さくなる. このとき, Th_{cf} によって除去することで,

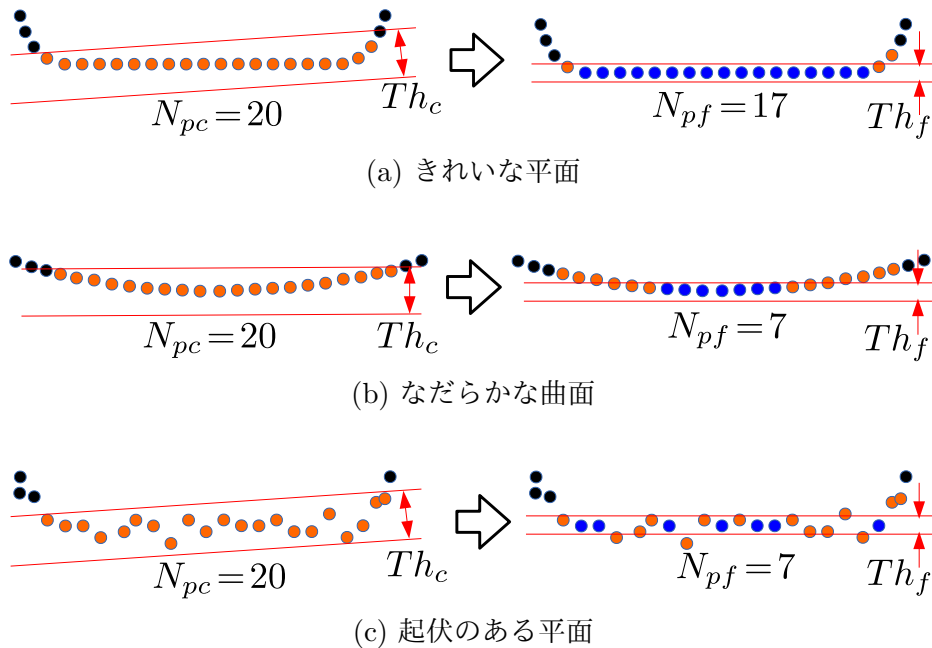


図 4.3 : CFRANSAC による平面候補の変化例

きれいな平面のみを抽出できる。

4.4.2 VKOP によるキーポイント検出

推定した複数の平面方程式のうち、3個の組み合わせから1個のキーポイントを検出する。 $\mathbb{P}_1, \mathbb{P}_2, \mathbb{P}_3$ の3個の平面を成す点群があるとき、これらの交点 $\mathbf{v} = (v_x, v_y, v_z)^T$ は、

$$\begin{cases} a_1 v_x + b_1 v_y + c_1 v_z + d_1 = 0 \\ a_2 v_x + b_2 v_y + c_2 v_z + d_2 = 0 \\ a_3 v_x + b_3 v_y + c_3 v_z + d_3 = 0 \end{cases} \quad (4.2)$$

の解である。すなわち、 \mathbf{v} の位置座標は、

$$\mathbf{v} = \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} = \begin{pmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{pmatrix}^{-1} \begin{pmatrix} -d_1 \\ -d_2 \\ -d_3 \end{pmatrix} \quad (4.3)$$

である。

平行の位置関係にない平面が n 個 ($n \geq 3$) あるならば、その組み合わせ $\binom{n}{3}$ 個の VKOP を検出することができる。ただし、選択した平面のうち2つがほぼ平行の位置関係にある場合は、点群から極端に離れた位置が算出される。そのため、平行となる平面の組み合わせは検出から除外する必要がある。これには、平面方程式の法線を COS 類似度を閾値 Th_{cos} で判定する手法、3平面の重心点と VKOP の距離を閾値 Th_{cen} で判定する手法の2つを用いる。

4.4.3 VKOP 特徴量記述

前述のとおり、VKOP は点群の表面から離れた位置にも存在しうるため、特徴量記述に実在の点分布を使用できない場合がある。そこで、VKOP 検出に用いた平面方程式を用いて、LRF の算出と特徴量を記述する手法を提案した。

図 4.4 は VKOP の LRF 定義の例である。 $\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3$ は、VKOP 検出に用いた3つの平面の法線であり、これらが成す角度を、それぞれ $\theta_{12}, \theta_{23}, \theta_{31}$ とする。こ

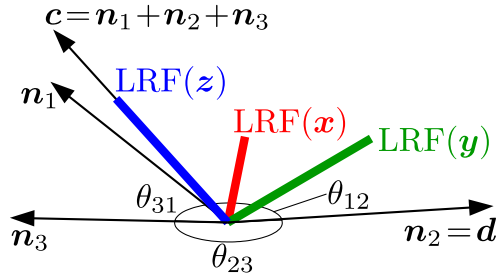


図 4.4 : 3つの法線から定義した VKOP LRF の例

のうち，1番目と2番目に大きい角度を成すベクトルを，方向ベクトル \mathbf{d} とおく．
 図 4.4 の例では， $\theta_{23} > \theta_{12} > \theta_{31}$ の関係にあるので， $\mathbf{n}_2 = \mathbf{d}$ となる．次に，3つ
 の法線の合成ベクトル $\mathbf{c} = \mathbf{n}_1 + \mathbf{n}_2 + \mathbf{n}_3$ を算出し，これを LRF の z 軸とする．
 \mathbf{x} 軸は方向ベクトルと合成ベクトルの外積とし，さらに \mathbf{y} 軸は \mathbf{x} 軸と z 軸の外積
 とする．すなわち，VKOP の LRF は

$$\begin{cases} \mathbf{x} = \mathbf{d} \times \mathbf{c} \\ \mathbf{y} = \mathbf{z} \times \mathbf{x} \\ \mathbf{z} = \mathbf{c} \end{cases} \quad (4.4)$$

により定まる．

記述子を回転不変にするため，まず3つの法線を LRF の座標系に変換し，それ
 ぞれ $\mathbf{n}'_1, \mathbf{n}'_2, \mathbf{n}'_3$ とする．次に，LRF の $\mathbf{x} - \mathbf{y}$ 平面にこれらを投影し， \mathbf{x} 軸から反
 時計回りの順に整列する．この整列した法線ベクトルを $\{\mathbf{n}'_a, \mathbf{n}'_b, \mathbf{n}'_c\}$ とする． $|\mathbf{c}|$
 は，3平面の位置関係によって変化するので，これを突出性の尺度として用いる
 こととした．すなわち，提案方式では $\{|\mathbf{c}|, \mathbf{n}'_a, \mathbf{n}'_b, \mathbf{n}'_c\}$ を並べた，シグネチャ型
 の 10次元のベクトルを VKOP の特徴量とする．

4.5 評価実験

提案方式を評価するため，特徴点の使用が想定され，かつ平面を多く含む点群
 を対象に実験を行った．実験結果の数値を正確に測定するため，真値の変換パラ

表 4.1 : ハードウェア及びソフトウェア

CPU	Intel Core i5-4570S (2.90GHz)
Memory	8.0 [GByte]
OS	Ubuntu ver. 14.04.4 (64bit)
Compiler	Clang++ ver. 3.4
Program library	PCL ver. 1.7.2
Simulator	Blender ver. 2.69
	Blensor ver. 1.0.18 RC1

メータを得られるソフトウェアを用いたシミュレーションセンシングにより，各種点群を作成した．

4.5.1 実験環境

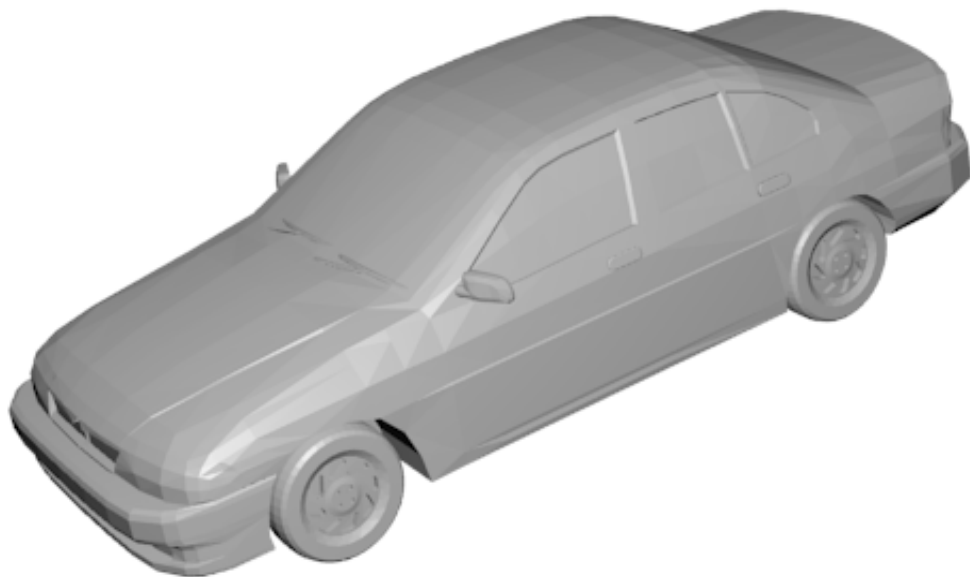
(1) ハードウェア及びソフトウェア

本実験に用いたハードウェア及びソフトウェアを表 4.1 に示す．実験環境の実装はすべて Point Cloud Library (PCL)[88] を用いた．シミュレーションソフトウェアの詳細は次で述べる．

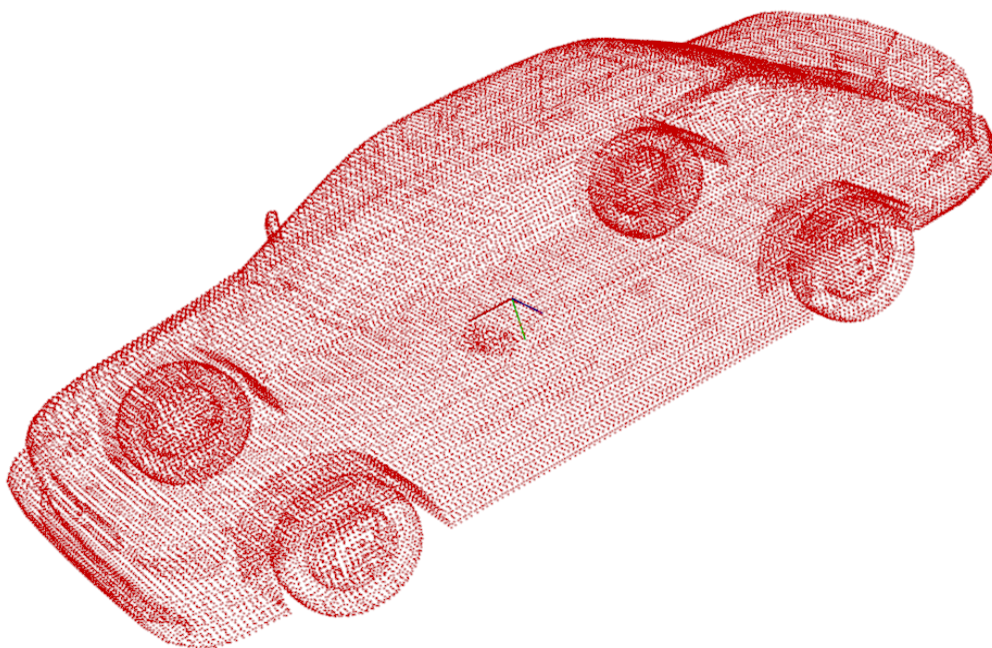
(2) データセット

本章では，CAD で設計されたような完全モデルを点群化したデータ，距離画像センサを模してシミュレーションセンシングした 2.5D データ，LIDAR を模してシミュレーションセンシングした全周データの 3 つの形態で実験する．データセットの外観を図 4.5，図 4.6，図 4.7 に，詳細を表 4.2 に示す．

“CAD” は車，“Kinect” はリビング，“LIDAR” は大学の構内を模した 3D モデルから得た点群である．これらのモデルから，1 個のターゲット点群と 10 個のソース点群を作成し，10 回試行できるデータセットを得た．また，ノイズのないものと，ノイズを付加した 2 種類のデータセットを作成した．なお表内，“Voxel” はボクセルサンプリングに用いた 1 辺の長さ，“BBDD” は点群を包み込む箱の対

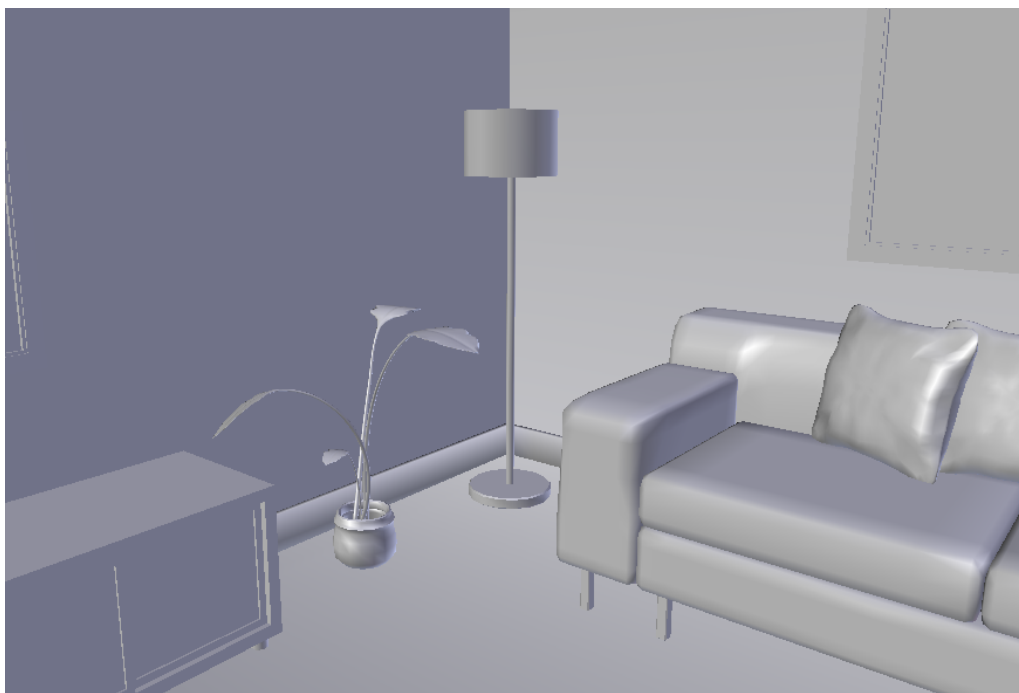


(a) ソリッドモデル

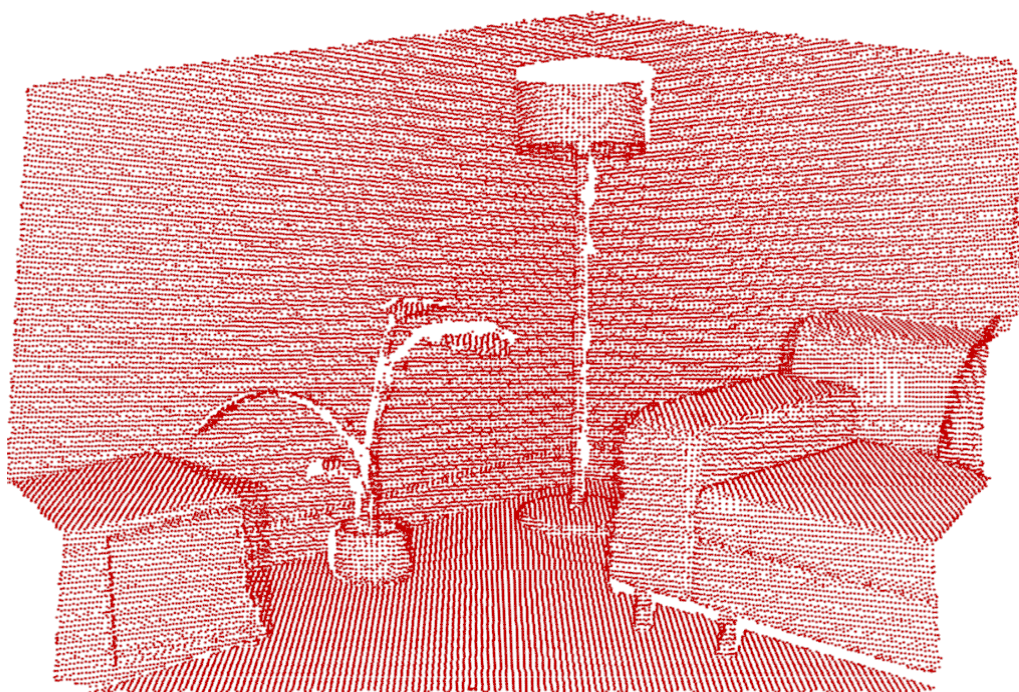


(b) 点群

図 4.5 : CAD データセット

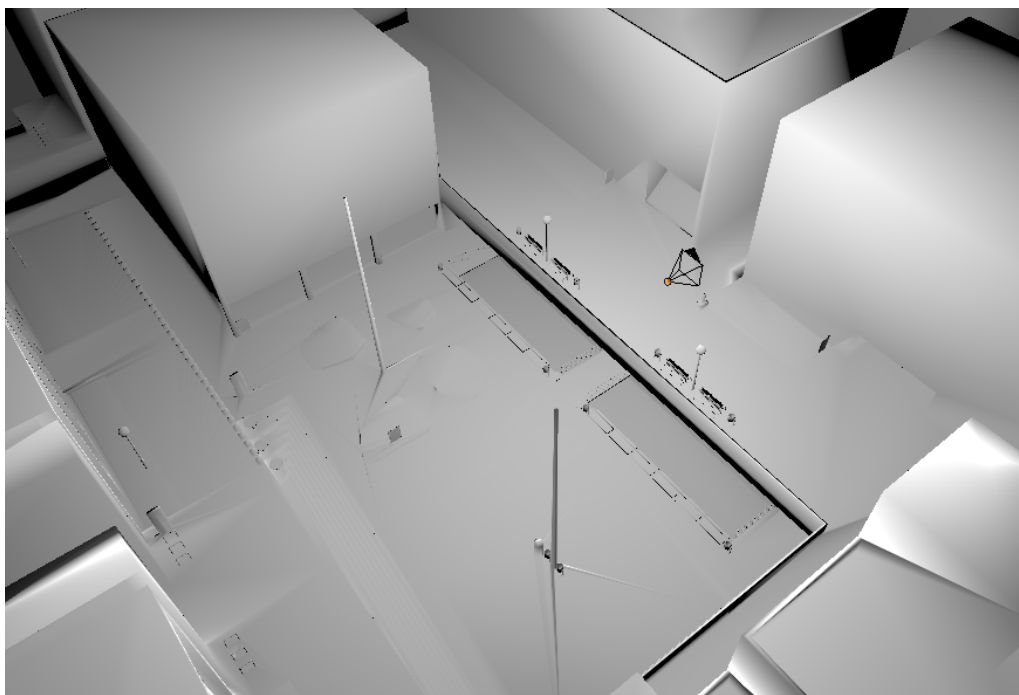


(a) ソリッドモデル

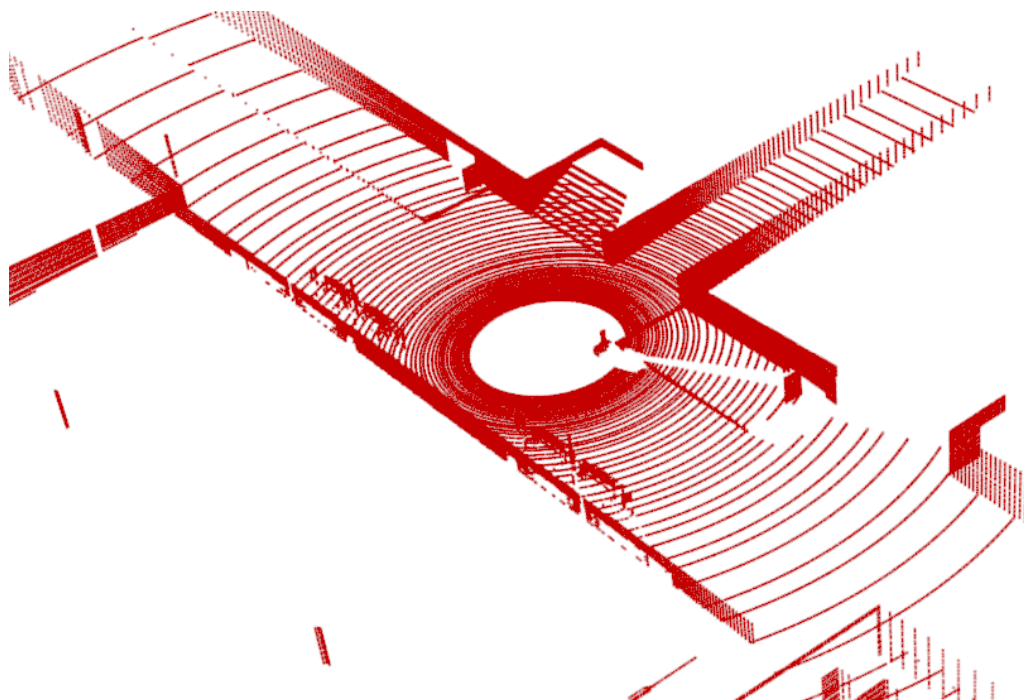


(b) 点群

図 4.6 : Kinect データセット



(a) ソリッドモデル



(b) 点群

図 4.7 : LIDAR データセット

表 4.2 : データセットの詳細

ID	Repository /filename	Acquisition	Voxel [mm]	mr [mm]	#Pts	BBDD [m]	View
CAD	CAPOD/m169	Blender remesh	11.0	8.3	50,061	2.2	Fig.4.5
Kinect	ICL-NUIM/living-room	Blensor kinect	15.0	10.4	48,732	4.1	Fig.4.6
LIDAR	BRLD/BRL_final	Blensor HDL-64E	80.0	82.6	52,859	103.1	Fig.4.7

角線の長さ (Boundary Box Diagonal Distance) を示す.

CAD データセットは, The Canonically Posed 3D Objects Dataset から, 車を模したモデル (m169) を選択した [89]. このモデルを, Blender のリメッシュ機能で十分な点数の点群を作成し, PCL のボクセルサンプリングで約 5 万点とした. ノイズは, 0.5[mr] のガウシアンノイズを付加した. CAD データセットは閉じた点群であるので, オクルージョンによる欠損部分は存在しない.

Kinect データセットは, ICL-NUIM RGB-D Benchmark Dataset から, リビングを模したモデル (living-room) を選択した [86]. このモデルを, Blensor の Microsoft Kinect モードでシミュレーションセンシングし, 次に PCL のボクセルサンプリングによって約 5 万点の点群とした. センサの位置と姿勢は, ターゲット点群を基準として, 各軸回転 ± 10 [deg] (1.0[deg] 刻み), 各軸並進 ± 0.1 [m] (0.01[m] 刻み) の範囲でランダムに変化させた. この値から, 真値の変換パラメータを取得した. ノイズは, ターゲット点群も含めて, Blensor のデフォルトパラメータによって付加したものをを用いた. センシングされた点群なので, このデータセットにはオクルージョンによる欠損領域が含まれる. 本来, センサとオブジェクトの位置が離れるほど点の解像度が大きくなるが [84], このデータセットではボクセルサンプリングによってセンサとの距離に関係なくほぼ一定の解像度となった.

LIDAR データセットは, Barcelona Robot Lab Dataset の大学の構内を模した

モデル (BRL_final) を選択した [37]. このモデルを, Blensor の ARGO HDL-64E モードでシミュレーションセンシングし, 次に PCL のボクセルサンプリングによって約 5 万点の点群とした. シミュレーションエラーを除くため, センサから 40[m] の半径内にある点のみを処理対象とした. センサの姿勢と位置はターゲット点群を基準に, Kinect データセットと同様の範囲でランダムに変化させた. また, 同様に真値の変換パラメータを取得した. ノイズは, ARGO HDL-64E モードのデフォルトパラメータで付加した. このデータセットは, センシングされた範囲が広大であるため, センサから離れた部分の点の解像度は, 近い部分よりも荒くなった. すなわち, このデータセットは, 単一の点群内で解像度が変化している.

(3) 評価指標

キーポイントの性能は, 文献 [47] と同様に, 2 つの再現性で評価する. \mathbb{S} から検出した特徴点を \mathbf{k}_i^s と, \mathbb{T} から検出した特徴点 \mathbf{k}_j^t があり, \mathbb{S} から \mathbb{T} への真値の変換パラメータから求めた変換行列を \mathbf{T}_g とするとき,

$$\|\mathbf{T}_g \mathbf{k}_i^s - \mathbf{k}_{cl}^t\| < Th_R, cl = \arg \min_{j \in \{1, \dots, N_{tk}\}} \|\mathbf{T}_g \mathbf{k}_i^s - \mathbf{k}_j^t\|^2 \quad (4.5)$$

を満たすならば \mathbf{k}_i^s は再現性があるとみなす. Th_R は再現性判定の閾値であり, 文献 [47] と同様に $Th_R = 2[\text{mr}]$ を用いる. また, 再現性の評価には絶対再現性と相対再現性の 2 つが用いられる [47]. 絶対再現性 AR (Absolute Repeatability) は, 式 (4.5) を満たすキーポイントの数である. 相対再現性 RR (Relative Repeatability) については, まず \mathbf{k}_i^s が \mathbb{T} の表面上にあることを

$$\|\mathbf{T}_g \mathbf{k}_i^s - \mathbf{p}_{cl}^t\| < Th_R, cl = \arg \min_{j \in \{1, \dots, N_t\}} \|\mathbf{T}_g \mathbf{k}_i^s - \mathbf{p}_j^t\|^2 \quad (4.6)$$

により確認する. 次に, RR は式 (4.6) を満たす点数 K_{st} と AR の割合

$$RR = \frac{AR}{K_{st}} \quad (4.7)$$

で算出される. なお VKOP は, 前述のとおり \mathbb{T} の表面上に検出されないことが多いので, $K_{st} = N_{sk}$ と定義して RR を求めた.

特徴量は、再現性のある特徴点に付与された記述子の類似度を測定し、その平均値で評価する。本実験では、式 (2.15) のコサイン類似度を用いた。

(4) 使用方式とパラメータ

VKOPに用いたパラメータを表 4.3 に示す。“*” の付いているパラメータはPCL に実装されているもの、 C_{nor} は RANSAC 処理時の法線の重み付け係数、“Pre. ex.” は、ノイズのないときに特徴点が抽出できるよう調整した経験値である。また、ノイズがあるときは平面方程式の推定精度が低下することから、 Th_c をもとの2倍、3倍とした実験も行った。それぞれ、基準値のものはVKOP($\times 1$)、2倍はVKOP($\times 2$)、3倍はVKOP($\times 3$)と表記する。

キーポイントの比較対象としてISS-BRとSIFT3Dを選択し、PCLで実装されている関数をそのまま用いた。ISS-BRのパラメータは表 4.4 に、SIFT3Dは表 4.5 に示す。どちらも突出値の大きいキーポイントを残すよう、ISS-BRはNMS

表 4.3 : VKOP のパラメータ

Parameter	Value	Unit	Remark
Normal est. radius* Rd_{nor}	10.0	mr	Pre. ex.
Normal weight* C_{nor}	0.1	-	PCL default
Dist. thresh.* $Th_c (\times 1)$	20.0	mm	Pre. ex.
Dist. thresh.* $Th_c (\times 2)$	40.0	mm	For noise
Dist. thresh.* $Th_c (\times 3)$	60.0	mm	For noise
Coefficient CF C_{cf}	0.5	-	Sec. 5.4.1 (Pre. ex.)
CF ratio threth. Th_{cf}	0.8	-	Sec. 5.4.1 (Pre. ex.)
Processed ratio threth. Th_{pp}	0.2	-	Pre. ex.
# coarse plane est. N_{cp}	30	times	Pre. ex.
Plane's similar thresh. Th_{cos}	0.9	-	Pre. ex.
Keypoint dist. thresh. Th_{cen}	4.0	BBDD	Pre. ex.

表 4.4 : ISS-BR のパラメータ

Parameter	Value	Unit	Remark
Normal est. radius* Rd_{nor}	10.0	mr	Pre. ex.
Boundary est. radius* Rd_{bor}	4.0	mr	Pre. ex.
Boundary est. angle* Th_{bor}	90	deg	[87]
Saliency est. radius* r_{sal}	10.0	mr	Pre. ex.
Thresh. 21* Th_{21}	0.6	-	Pre. ex.
Thresh. 32* Th_{32}	0.975	-	[50]
NMS radius* Rd_{nms}	4.0	mr	$AR \approx 50$

表 4.5 : SIFT3D のパラメータ

Parameter	Value	Unit	Remark
Curvature est. radius*	10.0	mr	Pre. ex.
Min. scale*	1.6	mr	[29]
# scales per octave*	3	-	[29]
Min. contrast*	0.006	-	$AR \approx 50$

処理で, SIFT3D は最小コントラスト閾値を調整して $AR \approx 50$ となるように検出した.

ISS は主要な 2 つのサーベイ [47, 49] で比較評価されている. 文献 [47] では, ISS は 2 つの再現性のバランスがよく, しかも計算量は少ないと評価されていた. 特に, SHOT 特徴量を記述したとき, 他の手法による特徴点よりもマッチング精度が高かった. 文献 [49] でも, ISS は他の手法よりも高い再現性を示し, しかも計算量が少ないと評価された. ISS-BR については, 文献 [48] において, このマッチング精度は, オリジナルの ISS, ユニフォームサンプリング, Harris3D[90] よりも高いと評価された.

SIFT3Dは、比較評価に用いられていないが、2次元画像の分野の代表的なキーポイントであり、PCLにも実装されている。各オクターブあたりの処理スケール数は3と設定したが、内部処理で前後2スケールが追加されるので、実際のスケール数は5である。文献[29]に基づき、終了条件はオクターブの数ではなく、点数が25点以下になったときとした。

特徴量の比較対象は、FPFHとSHOTを選択した。支持半径による影響を排除するため、LRFの支持半径 Rd_{lrf} 及び特徴量の支持半径 Rd_{des} は、10.0[mm]に条件を揃えた。特徴量のビン数と、これに伴う次元数はPCLデフォルト値を用い、FPFHは33次元、SHOTは352次元である。

4.5.2 VKOPの性能評価実験

ここでは、各データセットにおいて、各キーポイントがどのような性能を示すかを確認する。表4.6はCADデータセット、表4.7はKinectデータセット、表4.8はLIDARデータセットの実験結果である。これらの数値は、10回の試行の平均値である。“Noise”はノイズ付加の有無、 COS はその下の特徴量を用いたときの類似度を示す。

CADデータセットでは、VKOP($\times 1$)がノイズの有無にかかわらず他の検出器よりも高い RR を示したが、ISS-BRと比較して約2倍の処理時間を要した。これは、平面ではない部分の点数が多いため終了条件の Th_{pp} に達せず、 $N_{cp} = 30$ 回の平面推定を実施したためと考えられる。VKOP($\times 2$)とVKOP($\times 3$)は、うまく平面部分を抽出できず、 RR が低下していた。このときの平面推定の例を図4.8に示す。図4.8(a)のように、(c)のボンネットと(d)のフロントガラスから1枚ずつ平面を得るのが理想的であるが、(e)に示すように、この2枚を1枚とみなしてしまうことが多数あった。このような場合は平面の再現性が低下するため、当然VKOPの再現性も低下する。この原因は、 Th_c の値が大きすぎたためであると考えられる。

Kinectデータセットの RR と COS は、CADデータセットと近い傾向の結果で

表 4.6 : CAD データセットの実験結果

Detector	Noise	AR	RR	COS			Time [sec]
				FPFH	SHOT	VKOP	
ISS-BR	OFF	35.0	0.584	0.999	0.840	-	4.184
	ON	27.5	0.320	0.986	0.544	-	3.695
SIFT3D	OFF	89.5	0.502	0.993	0.760	-	3.840
	ON	54.0	0.239	0.968	0.499	-	4.647
VKOP($\times 1$)	OFF	44.4	0.536	-	-	0.958	10.421
	ON	21.6	0.378	-	-	0.964	10.375
VKOP($\times 2$)	ON	9.2	0.106	-	-	0.980	7.691
VKOP($\times 3$)	ON	0.8	0.008	-	-	0.973	5.618

表 4.7 : Kinect データセットの実験結果

Detector	Noise	AR	RR	COS			Time [sec]
				FPFH	SHOT	VKOP	
ISS-BR	OFF	26.7	0.351	0.994	0.699	-	2.836
	ON	7.6	0.131	0.989	0.605	-	2.443
SIFT3D	OFF	18.7	0.299	0.961	0.514	-	3.159
	ON	9.9	0.138	0.914	0.416	-	3.617
VKOP($\times 1$)	OFF	10.5	0.468	-	-	1.000	3.990
	ON	2.9	0.266	-	-	0.998	6.576
VKOP($\times 2$)	ON	5.5	0.251	-	-	0.998	3.356
VKOP($\times 3$)	ON	4.0	0.713	-	-	1.000	3.033

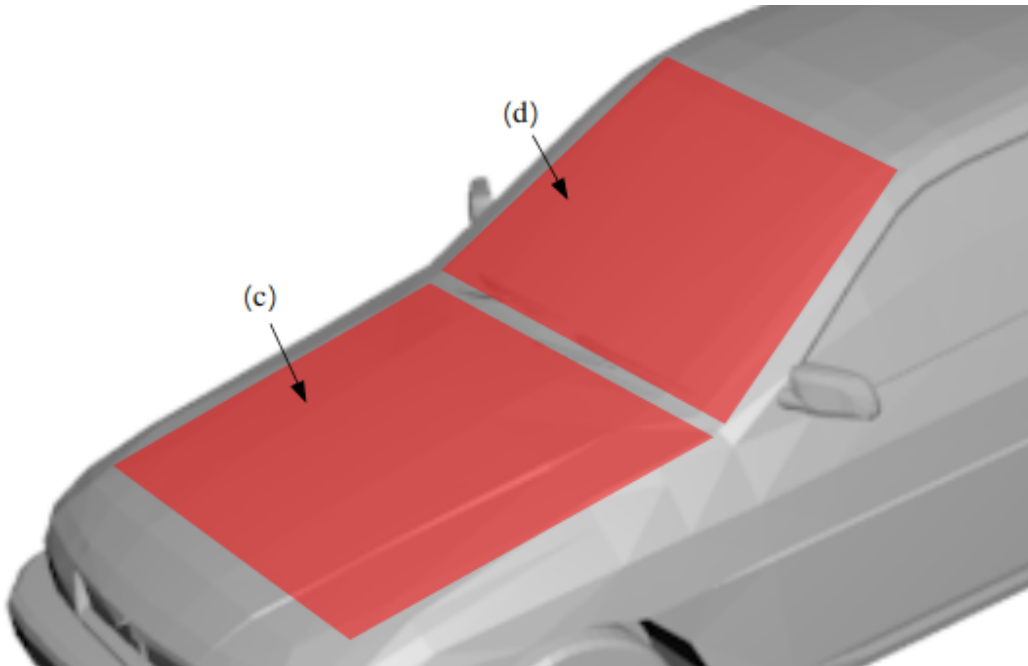
表 4.8 : LIDAR データセットの実験結果

Detector	Noise	AR	RR	COS			Time [sec]
				FPFH	SHOT	VKOP	
ISS-BR	OFF	24.2	0.220	0.977	0.678	-	2.021
	ON	11.2	0.095	0.984	0.612	-	1.998
SIFT3D	OFF	34.3	0.163	0.884	0.334	-	2.756
	ON	36.5	0.155	0.889	0.366	-	3.401
VKOP($\times 1$)	OFF	27.3	0.486	-	-	0.995	6.429
	ON	0.0	0.000	-	-	0.000	9.381
VKOP($\times 2$)	ON	31.4	0.254	-	-	0.978	7.595
VKOP($\times 3$)	ON	13.1	0.097	-	-	0.953	6.491

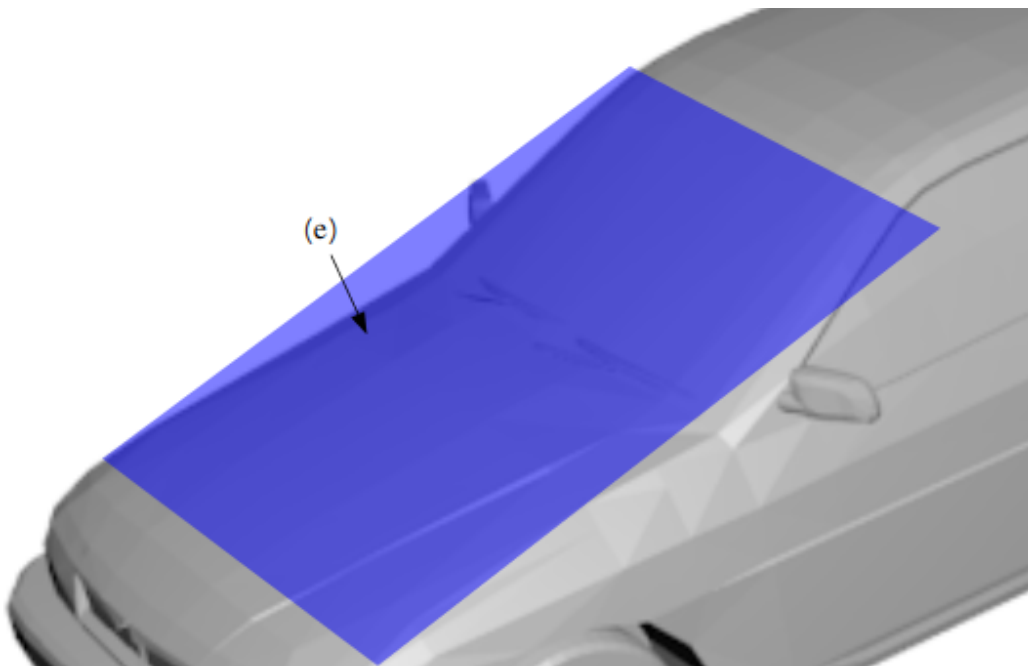
あったが, VKOP は Th_c の大きさによって結果が大きく異なり, これが最大である VKOP($\times 3$) が最も良い RR と処理時間を示した. ノイズのある Kinect データセットに対しては, VKOP($\times 3$) の Th_c が最も適したパラメータであったといえる. VKOP の処理時間は, Th_{pp} に達して平面推定が早期に終了したため, ISS-BR や SIFT3D と同程度にまで短縮された.

LIDAR データセットでは, ノイズのないときは VKOP($\times 1$), ノイズのあるときは VKOP($\times 2$) が高い RR を示した. 一方, 従来手法による特徴点の RR は, 他のデータセットと比較して全体的に低かった. これは, 点群の中で点の解像度が変化したため, 支持半径に依存する従来方式が上手く機能しなかったためと考えられる. VKOP では解像度の変化には依存せず, Th_c の適否に結果が依存しているといえる. VKOP の処理時間は, CAD データセットと同様に $N_{cp} = 30$ 回の平面推定をを実施したため, ISS-BR よりも 2 倍以上を要した.

特徴量は, 全てのデータセットで FPFH と VKOP がほぼ同等, かつ完全一致に近い値を示していた. 一方で, SHOT は類似度が低かった.



(a) 理想的な平面推定



(b) 平面推定のエラー

図 4.8 : 平面推定の例

以上から、本データセットに関する VKOP の性能は、従来方式よりも優れていることが確認できた。処理時間は、点群に適したパラメータを用いた場合、ISS-BR と同等であった。ただし、これらの VKOP の結果はどのデータセットにおいても、使用した Th_c によって RR が大きく変動していることから、VKOP は平面推定の成否に強く依存している。これは、図 4.8 のように複数枚の平面を 1 枚とみなしたりすることで、平面の再現性が大きく低下したことに起因すると考えられる。すなわち、平面推定法の選択と、これに用いるパラメータには注意が必要といえる。

4.5.3 検出された特徴点位置に関する考察

ここでは、各キーポイントの検出位置を目視で確認することにより、その傾向について考察する。検出対象の点群は、先の実験で使用した Kinect データセットにおける、ノイズのないターゲット点群である。パラメータはすべて前節で使用したものをし、VKOP は VKOP($\times 1$) のパラメータを適用した。

図 4.9 は ISS-BR、図 4.10 は SIFT3D の検出結果である。検出された特徴点の位置はほぼ同じであり、突出した表面上（あるいは凹んだ表面上）にあることが確認できた。ISS-BR では境界点除去機能により、境界点から離れた位置に特徴点が検出されている。

一方、図 4.11 は VKOP の検出結果である。キーポイント数は、他の手法よりも 50 点以上少なかった。図 4.11(a) の VKOP は、従来方式とは異なり点群から完全に離れた空間に検出されている。キーポイントが複数重なって検出されている理由は、キャビネットとソファの上面が、ほぼ同じ高さに存在していたためである。図 4.11(b) と (c) は、オクルージョンによる欠損部分にある VKOP である。(b) は床、左の壁、ソファの側面の組み合わせ、(c) は床、右の壁、キャビネットの側面の組み合わせから検出された。点群内にある平面部分は、ほぼ正しく抽出できたが、図 4.11(d) に示すソファの肘掛けにある平面部分は抽出できなかった。このような部分の平面も正しく推定できれば、より多くの VKOP を検出できる

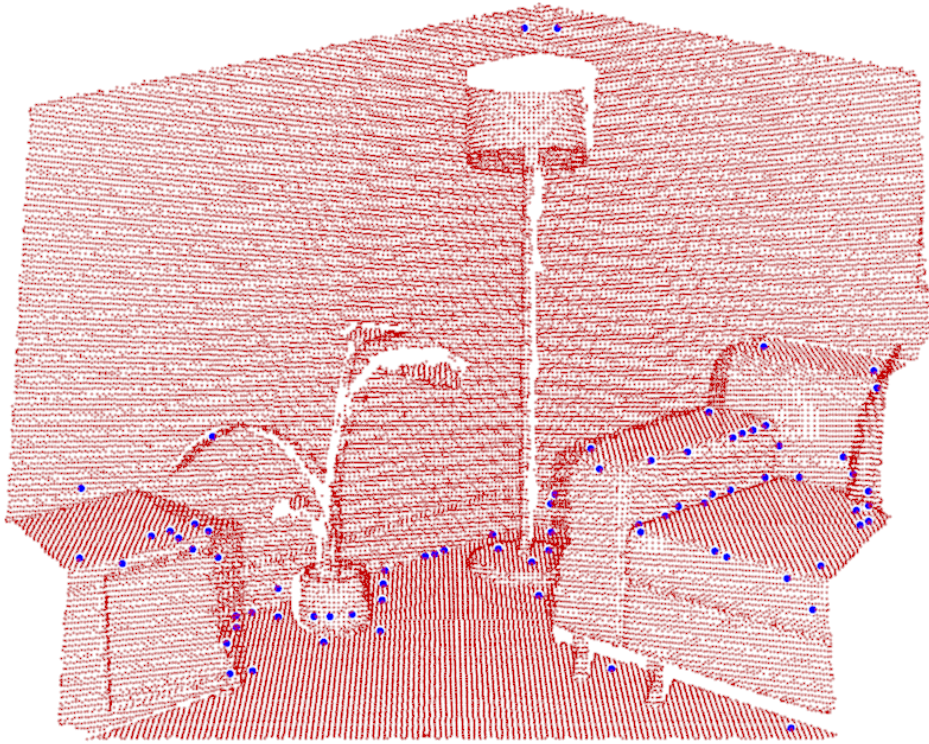


図 4.9 : ISS-BR の位置 (76 点)

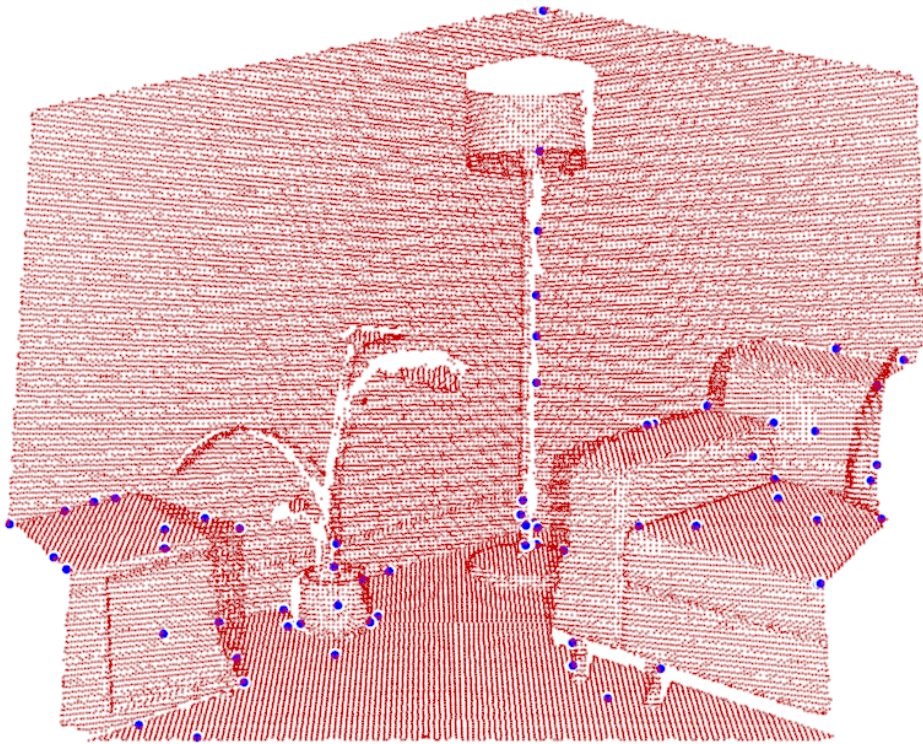


図 4.10 : SIFT3D の位置 (69 点)

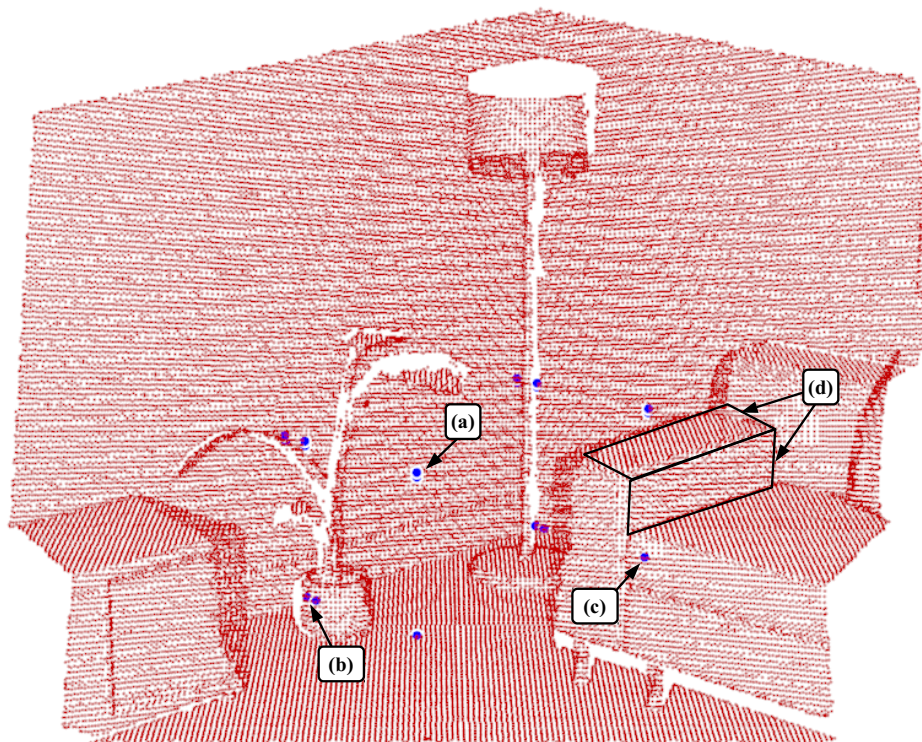


図 4.11 : VKOP の位置 (18 点)

ので，平面推定法の依存性がここでも確認できる．

4.6 まとめ

本章では，再現性を重視したキーポイントを検出するため，そもそも再現性の疑わしい実在する点から選ぶのではなく，再現性が高いと見込まれる形状から，仮想的な位置にキーポイントを配置する仮想特徴点抽出法を提案した．特に，平面形状はセンシングが容易であるため点の再現性が高く，しかも幾何学的にオクルージョン耐性を有することに着目し，点群全体から複数の平面方程式を得て交点にキーポイントを配置する，VKOP (Virtual Keypoint Of Polyhedron) を提案した．また，VKOP へ記述するのに適当な，専用の LRF と特徴量をあわせて提案した．さらに，平面方程式を高精度に推定するため，RANSAC による平面推定を 2 段階に分けて実行する CFRANSAC 平面推定法も提案した．

平面を多く含む環境を想定したシミュレーション実験で、ノイズを含まない場合には他の従来方式と同等以上の性能を示すことを確認した。ノイズを含む場合では、適切に平面方程式を推定できたとき、従来方式よりも高い特徴点抽出性能を示した。しかしながら、平面推定のパラメータが適当ではなかった場合は特徴点の性能が低下し、処理時間も遅延した。すなわち、VKOP の性能と計算量は、平面推定に使用する手法と、そのパラメータに強く依存することから、この改善が大きな課題といえる。

第5章

平面尤度を用いたVKOP

VKOPの平面推定精度の依存性低減のため、平面の安定性、すなわち尤度を評価し、最適なVKOPの検出法を提案する。

5.1 はじめに

本章では、VKOPに用いる平面方程式の安定性、すなわち平面の尤度に着目した。CFRANSACによる平面方程式の推定と同時に、これに適した尤度を計測し、高い尤度をもつ平面をだけを使用する手法を提案した [91, 92]。この尤度をVKOPにも統合して付与し、高い尤度を示すVKOPを優先的に採用する手法を提案した [92–94]。さらに、この尤度を算出するのに最適なパラメータを事前実験により求めておく手法を提案した [92, 95]。

平面を多く含む室内環境を想定したシミュレーション実験により、従来のキーポイント、または改善前のVKOPより短い処理時間で、最大4倍程度の再現性向上が可能であることを確認した [92]。

5.2 従来法の課題と解決方針

4.5.2項及び4.6節で述べたとおり、VKOPの課題は平面推定に使用する手法と、そのパラメータに強く依存することである。特に、許容する平面の厚みを指定するパラメータ、CFRANSACでいえば Th_c が適切ではなかった場合は、図4.2のようになだらかな曲面や起伏のある平面では毎回異なる平面を得たり、または図4.8のように複数の平面を1枚とみなしたりなど、平面方程式の再現性を大きく低下させる。これに伴って、VKOPの再現性が大きく低下するのは自明といえる。そこで、平面形状の安定性を評価し、そのような平面方程式、またはVKOPを除くことが、平面推定法の依存度低減の解決方針といえる。

5.3 平面の尤度を活用した VKOP 検出

VKOP の再現性は、位置座標を決めるのに使用した 3 枚の平面の再現性に依存するため、平面推定法のパラメータを厳しくして、きれいな平面のみを使用するのが理想的といえる。しかしながら、センサのノイズによる点分布の変動や、オブジェクトの雑然さ等の要因により、パラメータの調整だけできれいな平面だけを取り出すことは困難である。VKOP の平面推定に用いた CFRANSAC の主要なパラメータは、許容する厚みの閾値である Th_c であるが、図 4.2 の (a) と (b) を厚みだけで比較したとき、(a) は丸みを帯びた縁が存在するため (b) のほうが厚みが薄いとみなす場合がある。すなわち、パラメータを厳しくしても曲面のほうが残る場合も多くある。そこで、平面の安定性、すなわち平面の尤度を定量的に測定し、これを VKOP の検出に導入する方式を提案する。

5.3.1 平面のフィルタリング法

まず第 1 に、平面の尤度に閾値を設定し、これを超えない尤度の平面方程式を除去する平面のフィルタリング法を提案する。VKOP 検出前に平面方程式を除去しておくことで、位置座標の決定に要する計算量も削減できるといえる。図 4.2 の例では、(a) から (c) の点群のうち、(a) の点群が理想的な平面であるため、これだけを残す閾値を設定できれば再現性向上が見込める。一方で、(b) と (c) を比較したとき、起伏のある平面については平均化の効果により、安定的に平面方程式を得られる可能性がある。そこで、尤度の測定には、曲面が最も低い値となるように設計する必要がある。

5.3.2 VKOP のランキング法

平面のフィルタリング法で VKOP の再現性向上が見込める一方、残る平面方程式の数が推測できないため、場合によっては VKOP が 1 点も得られないことも起こりうる。レジストレーション等への活用を考慮した場合、2.3.4 項で述べたとおり SVD で変換パラメータを推定するため、再現性のあるキーポイントが少なくとも 3 点必要となる。このとき、出力結果を見ながら閾値の微調整をしな

なければならない。そこで、キーポイントの数を確保する観点から VKOP にも尤度を付与し、これが高い順に必要な数を取り出すランキング法を提案する。VKOP に付与する尤度には、位置座標を計算する際に用いた 3 枚の平面の尤度を乗算した値を使用することとした。

この VKOP に付与する尤度は、ISS 等の従来法に用いられている突出値と同じ位置づけとみなせる。ISS 等は有用なキーポイントを判別する手法に NMS を用いているが、これも何点残るかが事前に予測できないため、支持半径の調整が必要となる。そのため、VKOP への NMS 適用は、図 2.2(1) のように極めて近い位置に検出されたものだけにとどめ、ランキング法にて必要数を検出することとした。

5.4 平面の尤度測定法

VKOP の平面推定に用いた CFRANSAC は、4.4.1 項で述べたとおり 2 段階の閾値 Th_c, Th_f を用いている。各段階における点数比 R_{cf} に閾値 $Th_{cf} = 0.8$ を固定的に使用して、きれいな平面を残すように作為していたが、4.5 節の実験結果からは Th_c の値に再現性が依存し、この閾値はうまく機能していないといえる。そこで本章では、平面候補の点群をこの段階で除去せずに平面方程式を求め、これに R_{cf} を平面の尤度として一律に付与することとした。このときに付与する R_{cf} は、平面の CF 尤度と呼ぶ。

さらに、なだらかな曲面と起伏のある平面の違いにも着目した。図 4.3 の例では、どちらも同程度の $R_{cf} = 7/20 = 0.35$ の CF 尤度が算出されることとなるが、起伏のある平面は平均化によって再現性のある平面方程式を得られる可能性がある。一方でなだらかな曲面は、どの部分を平面とみなすかが一定ではなく、その再現性は低いといえる。すなわち、曲面の CF 尤度が小さくなるよう、点分布の偏りを考慮した新たな CF 尤度を提案する。例として、図 4.3 の Th_c で得た橙色の点に着目すると、なだらかな曲面は図を正面から見て上側に偏っているの

に対し、起伏のある平面は両側に分布していることがわかる。そこで、 Th_f で得られた平面方程式を基準に、橙色の点がどちら側に分布しているかを測定し、偏っている側の点数 $N_{pc'}$ を用いて $R'_{cf} = N_{pf}/(N_{pf} + N_{pc'})$ により改良した CF 尤度を得る。これによると、図 4.3(b) は $R'_{cf} = 7/(7 + 13) = 0.35$ 、図 4.3(c) は $R'_{cf} = 7/(7 + 8) \approx 0.47$ となり、曲面のほうが低い値となる。この CF 尤度は、曲面だけではなく、複数枚の平面を 1 枚とみなしたときも同様に値が低下するので、平面の再現性判定に有用といえる。

また、CFRANSAC の処理フローでは初回の閾値 Th_c が大きすぎるとき、CF 尤度に差が生じない場合がある。そこで次回の処理前に、 Th_c で得た平面の厚みを実測して Th'_c とし、 $Th_f = C_{cf}Th'_c$ を用いて尤度を算出することとした。図 5.1 はこれを追加したときの例であり、緑の線で示す平面の厚みの実測値 Th'_c を用いることで、図 5.1(a) は $R_{cf} = 19/21 \approx 0.905$ であるのに対し、図 5.1(b) は $R_{cf} = 12/21 \approx 0.571$ となり、CF 尤度がより機能的となるといえる。

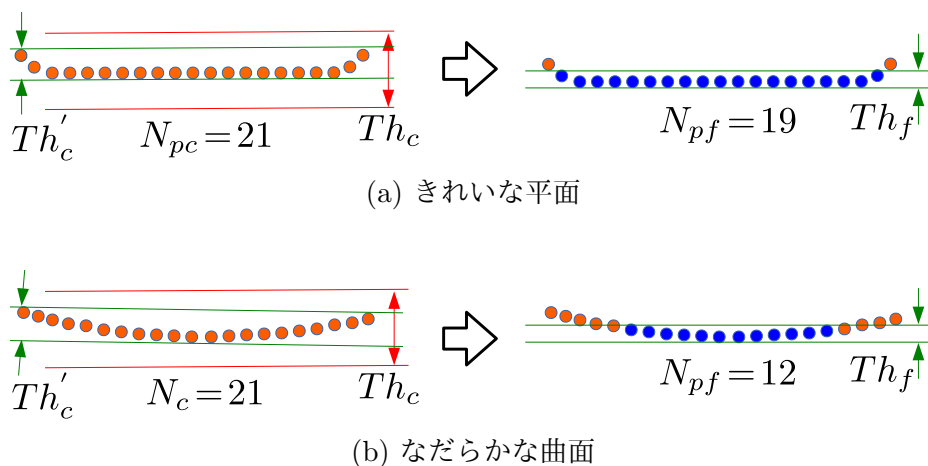


図 5.1 : Th_c を実測した場合の例

5.5 センサパラメータ決定法

パラメータ C_{cf} は、2回目の RANSAC 処理における Th_f を決定するとともに、CF 尤度の測定にも影響を与える。前章では $C_{cf} = 0.5$ を用いたが、この値が適当かは議論していない。CF 尤度へ与える影響を考えた場合、 C_{cf} が小さすぎると適当な平面部分であっても CF 尤度が低下し、閾値 Th_{cf} によって除去される可能性がある。逆に大きすぎると CF 尤度に差が生じず、フィルタリング法やランキング法に適用できなくなる。すなわち、理想的な平面は CF 尤度が大きく、それ以外の場合は小さくなるように C_{cf} を設定する必要がある。そこで、文献 [32, 84] と同様の事前実験によって、比較的きれいな平面と、最も厚みの生じた平面から得られる CF 尤度を推測し、当該実験環境において VKOP 検出に最適なセンサパラメータを求めておく手法を提案する。

提案する事前実験のフローは、以下のステップに示す。

- step1** センサの仕様上、点群が取得可能な距離の範囲を均等に分割し、平面状のオブジェクトを配置して点群を得る。このときの分割数は、後述する厚みの方程式を高精度に得られるように設定する。
- step2** 極端な外れ値を除外するため、ユークリッドクラスタリングによって点群を分割し、元の平面の面積の 1.0% に満たないクラスタを除去する。
- step3** センサとの距離と姿勢から理想的な平面方程式を求め、点群をなす全点との距離を測定する。この距離は片面側の距離であるので、最大の距離を 2 倍し、当該距離と角度で生じた平面の厚みを得る。
- step4** 角度を適宜に変化させて step2 から 4 を繰り返す、当該距離における最大の厚みを得る。
- step5** 最大の厚みを示した角度のとき、この平面の厚みを均等に片側 10 分割したビンを設定し、ビンの範囲内に含まれる点数をカウントする。

step6 積み上げ式のヒストグラムを作成し、全点数で割る。これらの割合の値は、 C_{cf} を設定したときに得られる CF 尤度に等しい。

step7 オブジェクトを配置する距離を変え、step2 から 7 を繰り返して各距離の最大の厚みのときにおける積み上げ式のヒストグラムを得る。このとき、遠距離になるほど得られる点群が少なくなることを考慮し、距離に応じて平面の面積を大きくする。

step8 同じビンに属する値を距離ごとに比較し、最大値と最小値の差を得る。一般的には、最近距離の値が最大であり、逆に再遠距離の値が最小である場合が多い。

step9 step9 における差が最大であるビンの割合を C_{cf} に採用する。

step9 で得られたビンの割合が 0.5 であった場合において、厚みが小さい平面と大きい平面で得られる R_{cf} の例を図 5.2 に示す。図内の記号の上部添字は、それぞれ (a) と (b) における値を示す。また、 Th_c は各例における実測の厚みを示しており、 $Th_c^a < Th_c^b$ である。図 5.2(a) は厚みが小さい平面の例であり、ある厚みの部分に点が密集しているため、 $Th_f^a = 0.5Th_c^a$ となっても N_{pf}^a の変化が少なく、 $R_{cf} = 17/22 \approx 0.773$ と高い値を示す。一方で、厚みの大きい平面の例であ

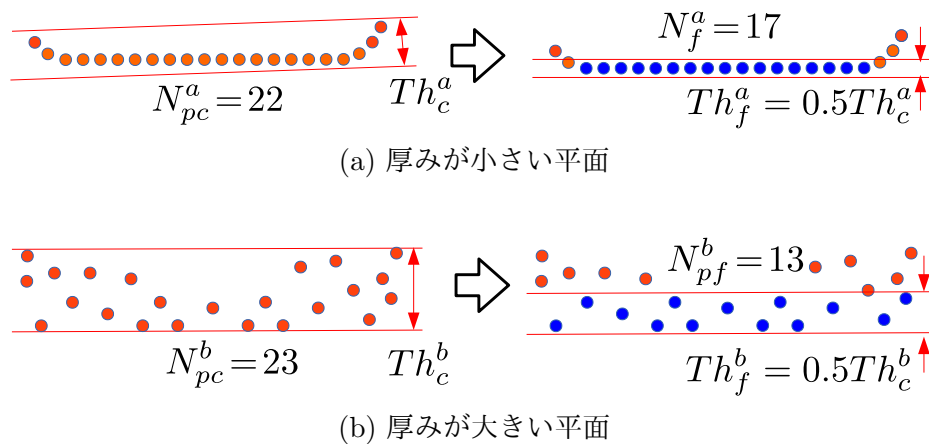


図 5.2 : $C_{cf} = 0.5$ における R_{cf} の例

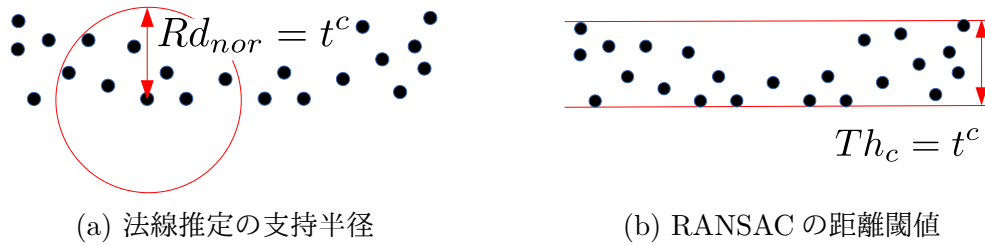


図 5.3 : 平面の厚みによるパラメータの決定

る図 5.2(b) では, $R_{cf} = 13/23 \approx 0.565$ となり, ノイズの少ない場合より値が小さくなる. すなわち, $C_{cf} = 0.5$ を用いたとき, R_{cf}^a と R_{cf}^b の差が大きくなるため, Th_{cf} でノイズの強い平面の除去が容易となる.

更に, この事前実験を活用し, CFRANSAC のパラメータであり, 平面の厚みを考慮する必要のある Th_c と, 法線推定の支持半径 Rd_{nor} の最適値を設定できる. まず, 上記の step4 で得られた各距離の最大の厚みから, 回帰分析によりセンサの距離と生じうる厚みの方程式を得る. step1 における距離の分割数は, 高い R 値を示すまで分割を繰り返す. 本処理の際には, 処理対象の点群 \mathbb{C} をなす全点とセンサとの距離を求め, この中央値を厚みの方程式に代入することで, 代表的な平面の厚み t^c を得る. すなわち, \mathbb{C} には t^c の厚みをもつ平面が多く含まれるといえるため, $Rd_{nor} = Th_c = t^c$ を用いるのが適当である.

$Rd_{nor} = Th_c = t^c$ を用いた場合における, 法線推定の支持半径と, 初回の RANSAC の距離閾値の例を図 5.3 に示す. 法線推定の図 5.3(a) では, 厚みの端の位置にある点であっても, 支持半径が広いので適切に法線を推定できる. RANSAC の図 5.3(b) は, 生じうる厚みに適した距離閾値を指定可能である.

5.6 評価実験

提案方式を評価するため, 前章と同様に平面を多く含む点群を対象に実験を行った. また, 点群の取得についても同様に, 真値の変換パラメータを得るためシミュレーションセンシングによりデータセットを作成した.

5.6.1 実験環境

(1) ハードウェア及びソフトウェア

本実験に用いたハードウェア及びソフトウェアを表 5.1 に示す。実験環境の実装はすべて PCL[88] を用いた。

(2) データセット

本章では、ノイズのあるときでも再現性が高く、比較的容易に平面推定ができると考えられる Kinect データセットを対象とした。Blensor のデフォルトパラメータでノイズを付加した点群だけを実験に用いる。また、条件を緩和したときの効果を確認するため、センサの位置と姿勢の変化量を低減したデータセットも作成した。

変化量の小さいデータセットを SDS, 大きいものを BDS を呼び、その詳細を表 5.2 に示す。Rot. はセンサの姿勢の最大の回転量, Trs. は位置の最大の並進量を

表 5.1 : ハードウェア及びソフトウェア

CPU	Intel Core i7-8750H (2.20GHz)
Memory	8.0 [GByte]
OS	Ubuntu ver. 18.04.5 (64bit)
Compiler	Clang++ ver. 6.0.0
Program library	PCL ver. 1.11.1
Simulator	Blensor ver. 1.0.18 RC10

表 5.2 : データセットの詳細

ID	#Pts	BBDD [m]	mr [mm]	Rot. [deg]	Trs. [m]	Median dist. [m]
SDS	48,576	4.2	10.2	± 5.0	±0.05	3.0
BDS	48,309	4.2	10.2	± 10.0	±0.10	3.0

示し、初回に取得した点群を基準として、3軸回転（1.0[deg]刻み）と3軸並進（0.01[m]刻み）でランダムに30回変化させ、データセットごとに31個の点群を得た。これらを順番に取り出してターゲット点群 \mathbb{T} とソース点群 \mathbb{S} の異なるペアを作り、30回試行できるデータセットとした。最後に、実験の高速化のため、全ての点群を1辺0.015[m]でボクセルサンプリングした。なお、表内の Median dist. は各点とセンサの距離の中央値である。

(3) 評価指標

本実験では、4.5.1項(3)で示した再現性のうち RR について、VKOPと他の手法を同じ基準で評価できるよう、つぎのとおり一部修正した。

AR の測定は変わらず、式(4.5)を満たすキーポイントの数である。 RR の測定において、式(4.6)で \mathbb{T} のオクルージョンに領域にある \mathbf{k}_i^s を無視し、マッチングの取れる可能性のある領域、すなわち \mathbb{T} と \mathbb{S} の重なりがある領域内で性能を評価している。一方で、どのキーポイントが欠けた領域にあるかは不明であり、実用上で必要な性能を測定できていないといえる。また、VKOPは前述のとおり、仮想的な位置にキーポイントを配置するため、点群の表面上に検出されるとは限らないことから、この性能測定は適さない。そこで、VKOPに限らず他の手法のキーポイントも $K'_{st} = N_{ck}$ と定義し、相対再現性を

$$RR' = \frac{AR}{K'_{st}} \quad (5.1)$$

ですべて評価することとした。

(4) 事前実験

前節で提案したとおり、平面状のオブジェクトを用いてVKOPの検出に必要なパラメータを決定する。ここで決定するのは、法線推定の支持半径 Rd_{nor} 、初回の距離閾値 T_c 、2回目の距離閾値を決定する係数 C_{cf} である。

BlensorのMicrosoft Kinectモードは仕様上、約0.4[m]から4.8[m]の範囲でセンシング可能であることから、本稿では1.0[m]から4.0[m]までの間、1.0[m]刻み

でオブジェクトを配置することとした。オブジェクトは、各距離に応じて辺の長さを増加させた正方形のもの配置した。表 5.3 に、センサからの距離 (dist.), 1 辺の長さ (length), 角度をつけずにセンシングしたときの平均点数 (#Pts) を示す。次に、各距離に配置したオブジェクトを 1.0[deg] ずつ回転させ、センサとの角度をつけた上で 10 回センシングし、各角度で 10 個の点群を得た。これらの点群は、ユークリッドクラスタリングによってクラスタに分類し、元の平面の面積の 1.0% に満たない小さなクラスタは除去した。最後に、平面方程式と各点の距離を測定し、最大距離のものから平面の厚みを得た。

まず、厚みの方程式を求める。図 5.4 に各距離における最大の厚みをプロット

表 5.3 : 事前実験用平面の詳細

dist. [m]	length [m]	#Pts
1.0	0.2	14,373
2.0	0.4	14,119
3.0	0.6	14,206
4.0	0.8	14,349

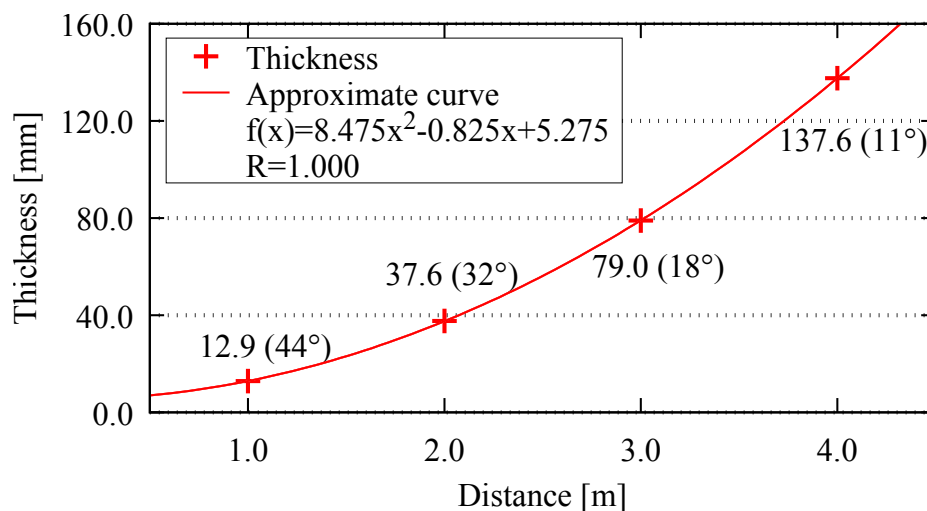


図 5.4 : 距離と厚みの関係

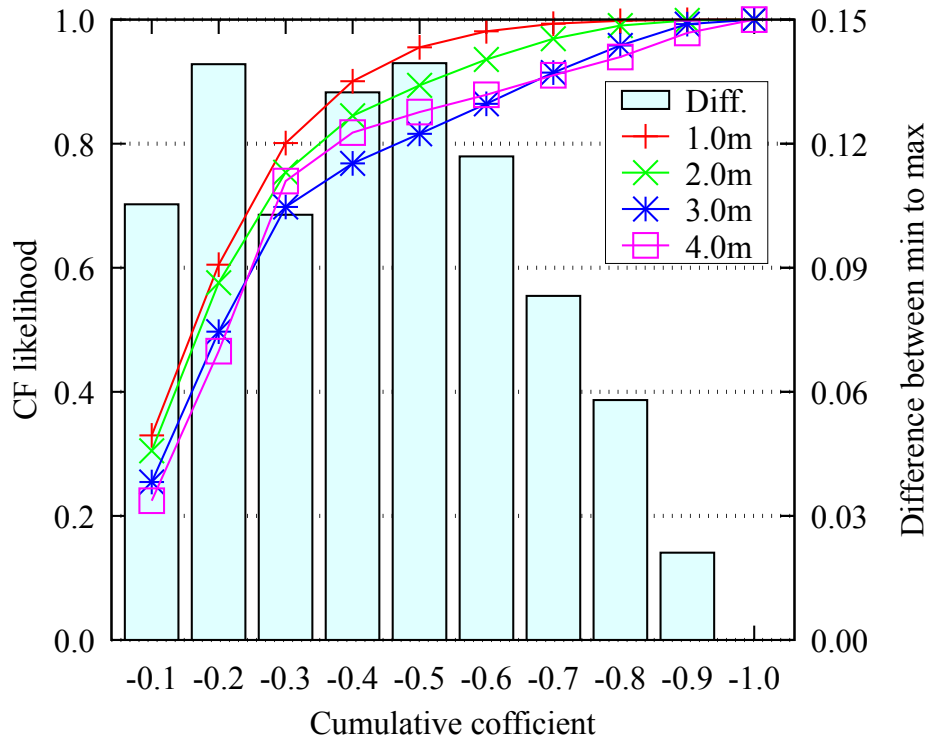


図 5.5 : CF 尤度の変化と最大の差

し，近似曲線を引いたものを示す．括弧内は最大厚みを示した角度を示し，近似曲線は回帰分析により求めた．なお，本結果は10個の点群から求めた平均値である．また，回転の角度は回転軸や回転方向を変化させても，同様の結果を示したため，ここでは1軸の回転角度のみを示す．この近似曲線と，表5.2のセンサとの距離の中央値 (3.0[m]) から， $Rd_{nor} = Th_c = 0.079[m]$ とした．

次に， C_{cf} を求める．前節のステップに基づき作成した，各距離における厚みの積み上げ式ヒストグラムの値を，グラフに表示したものを図5.5に示す．左軸の折れ線グラフは各距離におけるCF尤度，右軸の棒グラフは最大値と最小値の差である．棒グラフが最大値を示したのは0.2及び0.5であり，ほぼ同値であったことから，以後の実験では両方の値を用いて比較評価することとした．

表 5.4 : VKOP のパラメータ

Parameter	Value	Unit	Remark
Normal est. radius* Rd_{nor}	0.079	m	Proposed
Normal weight* C_{nor}	0.1	-	PCL default
Dist. thresh.* Th_c	0.079	m	Proposed
Coefficient CF C_{cf}	0.2	-	Proposed
	0.5		
Processed ratio threth. Th_{pp}	0.1	-	-
# coarse plane est. N_{cp}	100	times	-
Plane's similar thresh. Th_{cos}	0.9	-	-
Keypoint dist. thresh. Th_{cen}	1.0	BBDD	-
NMS radius Rd_{nms}	4.0	mr	[47]

(5) 使用方式とパラメータ

VKOP に用いたパラメータを表 5.4 に示す. “*” の付いているパラメータは PCL に実装されているものを示す. 4.5.1 項 (4) からのパラメータの入れ替えとして, CF 尤度の採用による Th_{cf} の削除, 事前実験から Rd_{nor}, Th_c と, 2 つの C_{cf} の使用, NMS 追加のため Rd_{nms} の追加がある. また, “Proposed” は先の事前実験で得たセンサパラメータを示し, ほかに Th_{pp} と N_{cp} を変更して多くの平面をとることとした. すなわち, 理想的ではない平面方程式も多く推定し, 提案方式の効果を確認する. 一方で, Th_{cen} を変更して遠い位置の VKOP は検出しないこととした.

キーポイントの比較対象は, 4.5.1 項 (4) と同じく ISS-BR と SIFT3D を用い, それぞれ表 5.5, 表 5.6 を用いた. 各種支持半径については, VKOP の法線推定と同じ事前実験で得た値を用いる. ただし, Rd_{nms} は VKOP と同じく, 文献 [47] で採用されている値とした. なお, PCL には SIFT3D の NMS 処理がなかったため, 自前で実装した. さらに, 突出値の閾値にあたる ISS の Th_{iss} と SIFT3D の

表 5.5 : ISS-BR のパラメータ

Parameter	Value	Unit	Remark
Normal est. radius* Rd_{nor}			
Boundary est. radius* Rd_{bor}	0.079	m	Proposed
Saliency est. radius* Rd_{sal}			
Boundary est. angle* Th_{bor}	90.0	deg	[87]
	0.975	-	[50]
	0.900	-	-
Thresh. 21*, Thresh. 32*	0.800	-	-
Th_{iss}	0.700	-	-
	0.600	-	-
	0.500	-	-
NMS radius* Rd_{nms}	4.0	mr	[47]

表 5.6 : SIFT3D のパラメータ

Parameter	Value	Unit	Remark
Curvature est. radius*	0.079	m	Proposed
Min. scale*	1.6	mr	[29]
# scales per octave*	3	-	[29]
	0.030	-	[29]
	0.025	-	-
Min. contrast*	0.020	-	-
Th_{sift}	0.015	-	-
	0.010	-	-
	0.005	-	-
NMS radius Rd_{nms}	4.0	mr	[47]

Th_{sift} は、両再現性の変化を確認するため、原著論文のパラメータに加えてそれぞれ5段階の強度を加えた。

5.6.2 センサパラメータ決定法の効果確認

ここでは、尤度の導入効果ではなく、センサパラメータ決定法で得た値を使用したときの効果を確認する。

表 5.7(a) は SDS での、表 5.7(b) は BDS での結果をまとめた表である。表中の Param. は使用したパラメータを示し、ISS-BR は Th_{iss} 、SIFT3D は Th_{sift} 、VKOP は C_{cf} で用いた数値のときの結果である。また、#KP は検出したキーポイントの数、Time は法線推定からキーポイントの検出までに要した秒数を示す。

全体の RR' を見ると、どちらのデータセットにおいても VKOP が2倍から4倍程度高いことが示されている。また、処理時間も VKOP が最短であり、最も高能率な方式であるとともに、最適なパラメータを設定できているといえる。VKOP 同士を比較すると、BDS の RR' が僅かに下回っているのを除き、 $C_{cf} = 0.5$ の両再現性が高いことがわかる。ただし、 $C_{cf} = 0.2$ のときは両データセットともに AR が 1.0 程度と小さく、レジストレーション等の応用技術への適用は困難といえる。すなわち、本実験に用いた環境下では、 $C_{cf} = 0.5$ の使用が適当である。

その他の従来方式をみると、ISS-BR と SIFT3D は原著論文に示されるパラメータとは異なる値を設定するほうが適当であった。どの値を採用すべきかは、データセットによって毎回確認しなければならないが、VKOP は事前実験により適当なパラメータを得ることができるため、何度も試行する必要はない。

なお、前章の表 4.7 における VKOP($\times 3$) と比較すると、 RR' が低下しているように見えるが、これは以下の条件が異なる。まず、前章の VKOP は NMS が未適用であり、非常に近い位置に VKOP が複数検出されることがある。そのため、 AR をダブルカウントしている場合があった。また、 $Th_{cf} = 0.8$ を適用しているため、本実験では未使用の CF 尤度による平面のフィルタリング法が既に適用されているに等しい。さらに、本実験では平面推定法の終了条件を緩め、理想的で

表 5.7 : キーポイント検出の比較結果

(a) SDS の結果

Method	Param.	#KP	AR	RR'	Time [sec]
ISS-BR (Param.: Th_{iss})	0.975	1356.3	149.0	0.110	2.829
	0.900	1317.5	141.7	0.108	2.840
	0.800	791.9	72.8	0.092	2.969
	0.700	244.8	25.4	0.104	2.886
	0.600	69.8	8.6	0.123	2.914
	0.500	7.4	0.3	0.040	2.910
SIFT3D (Param.: Th_{sift})	0.030	0.8	0.0	0.017	2.271
	0.025	2.5	0.4	0.161	2.222
	0.020	5.9	0.6	0.103	2.194
	0.015	18.9	1.9	0.104	2.231
	0.010	57.6	5.5	0.097	2.230
	0.005	175.7	19.3	0.110	2.260
VKOP (Param.: C_{cf})	0.2	3.1	1.0	0.325	1.800
	0.5	11.4	4.0	0.386	1.791

(b) BDS の結果

Method	Param.	#KP	AR	RR'	Time [sec]
ISS-BR (Param.: Th_{iss})	0.975	1319.3	117.7	0.089	3.204
	0.900	1298.6	116.6	0.090	3.095
	0.800	799.3	61.3	0.077	2.939
	0.700	252.8	21.9	0.086	2.762
	0.600	69.9	6.8	0.098	2.695
	0.500	7.9	0.2	0.033	2.622
SIFT3D (Param.: Th_{sift})	0.030	0.7	0.0	0.033	2.343
	0.025	2.2	0.2	0.064	2.297
	0.020	5.5	0.4	0.057	2.270
	0.015	17.2	1.6	0.103	2.175
	0.010	56.2	5.7	0.103	2.118
	0.005	173.0	16.6	0.097	2.065
VKOP (Param.: C_{cf})	0.2	3.0	0.9	0.433	1.748
	0.5	10.4	3.4	0.407	1.743

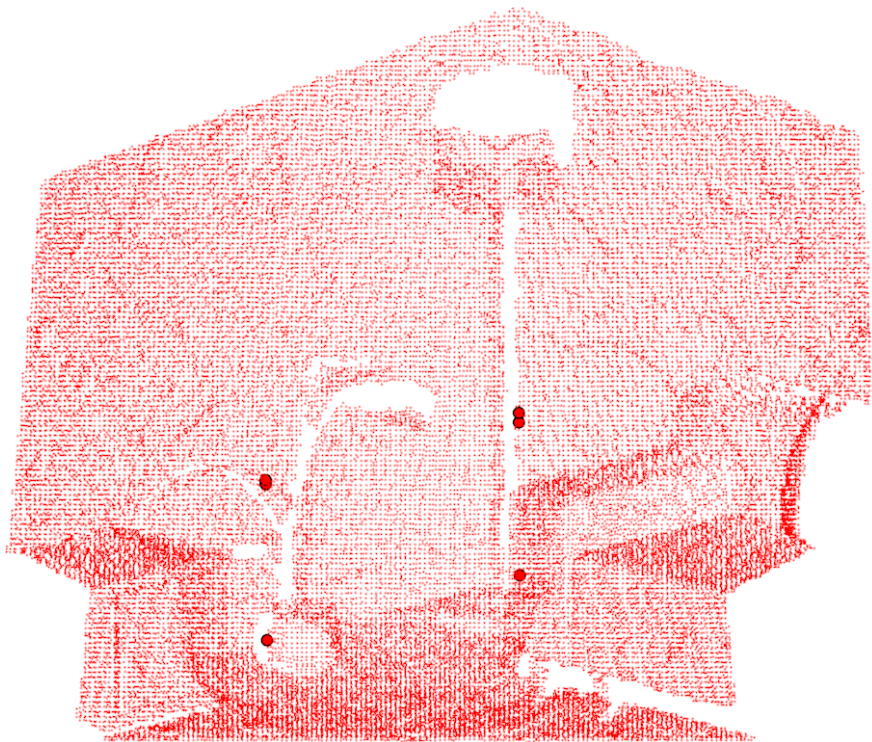


図 5.6 : 4 章の実験の VKOP 検出例

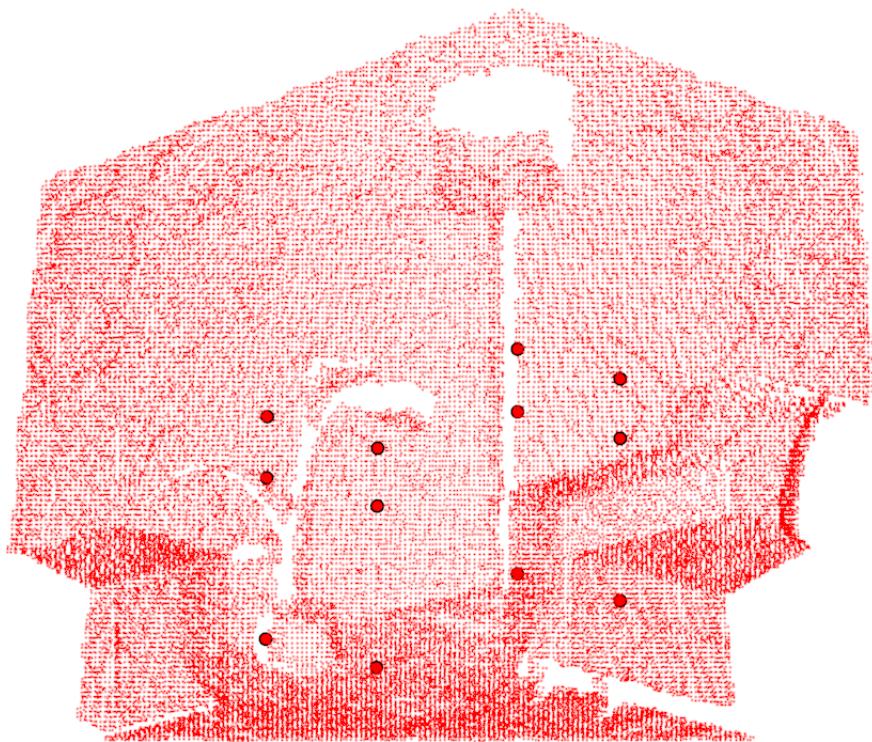


図 5.7 : 5.6.2 項の実験の VKOP 検出例 (尤度は不使用)

はない平面方程式も多く含んだ状態である。すなわち、平面のフィルタリング法の適用と、その他のパラメータが偶然最適値であったことから、高い RR' を示していたといえる。

前章の実験の VKOP 検出例を図 5.6 に、本実験の VKOP 検出例を図 5.7 に示す。図 5.6 の検出位置のように、左右の壁、床、ソファの上面の 4 枚を得たところで平面推定を終了することが多かった。また、ほぼ重なった位置にも VKOP が検出される場合があった。一方で本実験では、それ以外の平面も積極的に得るようしており、重なった位置の VKOP は存在しない。

以上から、提案したセンサパラメータの事前決定は、高能率な VKOP の検出に貢献することを確認できた。同じセンサを使用する限りはパラメータの微調整が不要で、高い再現性を有する VKOP を検出できる。

5.6.3 尤度を用いた VKOP の性能評価実験

ここでは、CF 尤度を活用した平面のフィルタリング法と、ランキング法で得られる VKOP の性能を確認する。VKOP のパラメータは前実験と同様であり、また $C_{cf} = 0.5$ のみを用いた。この 2 つの提案法は同時に適用することが可能であるが、それぞれの効果を確認するため独立して試行した。

フィルタリング法は、CF 尤度の閾値 Th_{cf} を 0.00 から 1.00 まで 0.01 刻みで変化させ、両再現性を得た。ランキング法は、フィルタリング法と同様に再現性の変化を詳細に確認するため、必要点数を直接的に指定するのではなく、除去する点数を割合で指定することとした。例として閾値を 0.3 とした場合には、CF 尤度が小さい順に 30% の点数を除去する。この閾値は、フィルタリング法と同様に 0.00 から 1.00 まで 0.01 刻みで変化させ、両再現性を得た。なお処理時間は、両手法ともに追加される処理の計算量が極めて微小であり、結果として前項の結果と同等であったため、ここでは評価しない。

図 5.8 と図 5.9 は、両手法の閾値を 0.01 刻みで変化させた全ての結果を、フィルタリング法は赤線で、ランキング法は青線でプロットした図である。なお、縦

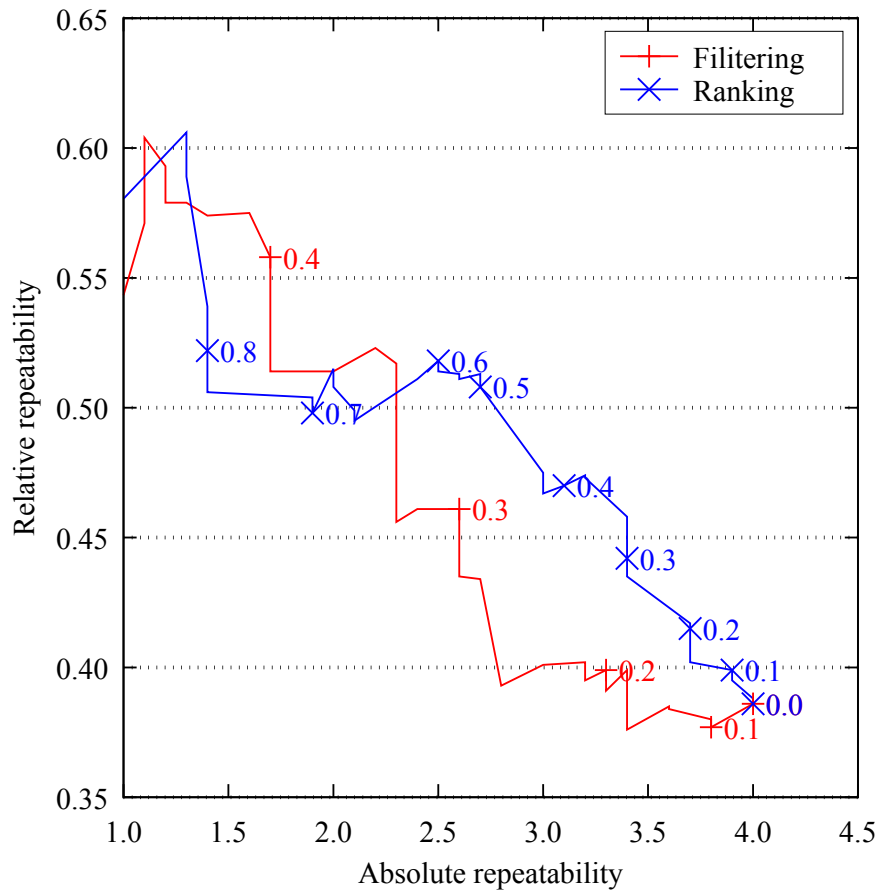


図 5.8 : SDS における両再現性の変化

表 5.8 : SDS における代表値

Param.	Filtering			Ranking		
	#KP	AR	RR	#KP	AR	RR'
0.0	11.4	4.0	0.386	11.4	4.0	0.386
0.1	11.2	3.8	0.377	10.7	3.9	0.399
0.2	9.8	3.3	0.399	9.5	3.7	0.415
0.3	7.2	2.6	0.461	8.4	3.4	0.442
0.4	3.8	1.7	0.558	7.2	3.1	0.470
0.5	1.6	0.8	0.488	5.7	2.7	0.508
0.6	0.4	0.0	0.017	5.0	2.5	0.518
0.7	0.0	0.0	0.000	3.9	1.9	0.498
0.8	0.0	0.0	0.000	2.7	1.4	0.522
0.9	0.0	0.0	0.000	1.5	0.8	0.544

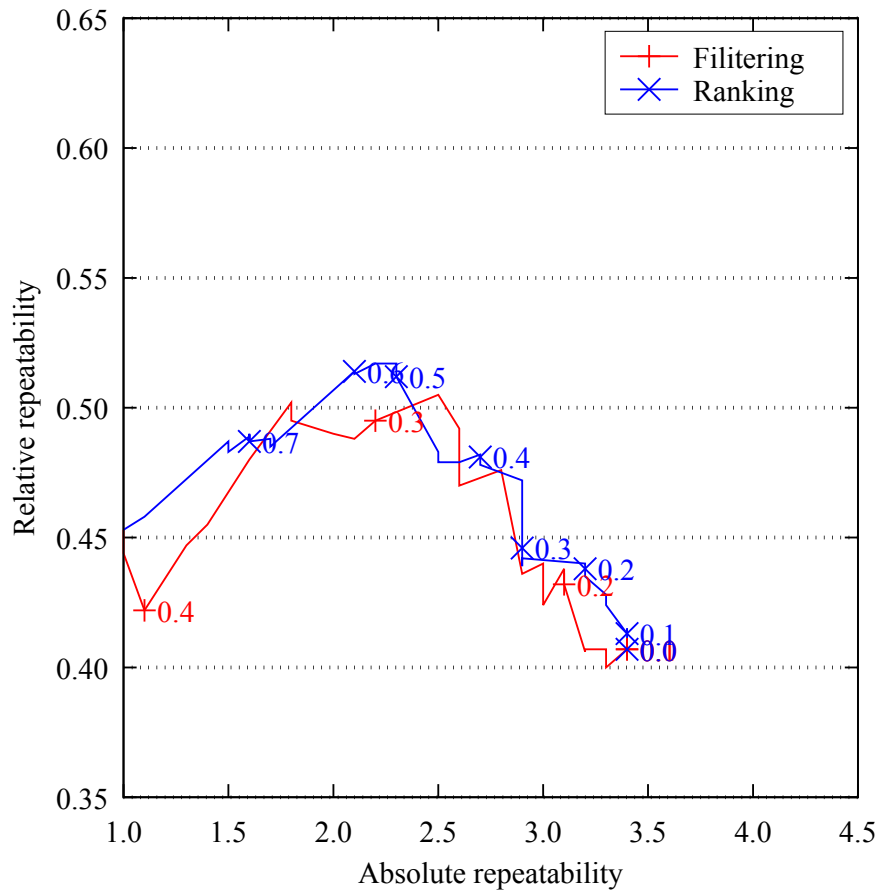


図 5.9 : BDS における両再現性の変化

表 5.9 : BDS における代表値

Param.	Filtering			Ranking		
	#KP	AR	RR	#KP	AR	RR'
0.0	10.4	3.4	0.407	10.4	3.4	0.407
0.1	10.4	3.4	0.407	9.9	3.4	0.413
0.2	9.1	3.1	0.432	8.9	3.2	0.438
0.3	6.6	2.2	0.495	7.7	2.9	0.446
0.4	3.3	1.1	0.422	6.7	2.7	0.481
0.5	1.3	0.5	0.333	5.3	2.3	0.512
0.6	0.4	0.0	0.033	4.6	2.1	0.514
0.7	0.0	0.0	0.000	3.7	1.6	0.487
0.8	0.0	0.0	0.000	2.5	0.9	0.427
0.9	0.0	0.0	0.000	1.5	0.6	0.467

軸は相対再現性，横軸は絶対再現性を示す．また，これらのうち閾値が0.1刻みであるときの結果を代表値として，図中に記号と添字を追加した．表 5.8 と表 5.9 は，閾値が0.1刻みのときにおける両再現性を抜粋し，まとめた表である．表中の両再現性の数値は図中の代表点と同じものであり，例として図中の右端の添字が0.0の記号は，表中の閾値が0.0のときの結果と対応している．

両データセットともに，閾値を強くするほど平面の数やVKOPの数が減り，ARの低下とともに RR' が増加する傾向にあることが確認できる．フィルタリング法の閾値が0.0から0.2の区間内は，未適用よりも RR' が低下している部分があるが，これは再現性のある平面のCF尤度が低く算出され，これをフィルタリングで除去してしまったためと考えられる．一方でランキング法は，他の2つの平面のCF尤度を考慮したことでVKOPが除去されず，未適用の場合を下回ることにはなかった．図 5.10 は，ランキング法で閾値を0.5としたときの検出例である．図 5.7 と比較し，VKOP数が12点から半分の6点に減少している．

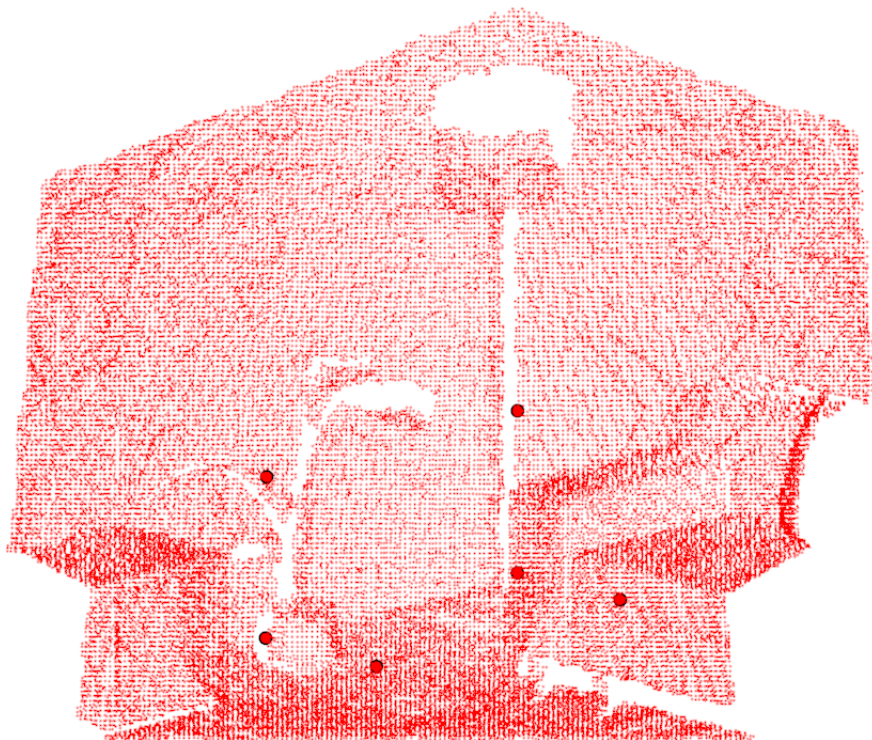


図 5.10 : ランキング法を用いた VKOP の検出例

レジストレーション等の応用技術への適用を考慮すると AR が 3.0 以上であることが望ましいため、この範囲内ではランキング法の RR' が上回っていることが確認できる。VKOP の性質上、両手法の未適用時に得られる VKOP 数と AR が小さいことを考慮すると、与えるパラメータが小さい範囲で RR' が高いほうが望ましい。すなわち、実用上はランキング法の使用が適当であると考えられる。

以上から、両手法とも AR とのトレードオフであるものの、 RR' を向上できることが確認できた。ただし、VKOP の数が少ないことを考慮すると、閾値が小さくても RR' の向上が見込めるランキング法の使用が適当であるといえる。

5.7 まとめ

本章では、高精度な平面推定法の採用が必要な VKOP 検出法について、その依存性を低下させるため、平面の尤度を活用することとした。推定した平面の尤度を測定し、閾値を設けて再現性の高い VKOP だけを残すことを企図している。提案した平面の尤度は、厚みを薄くしたときにおける平面を成す点数の減少割合から算出するものであり、CFRANSAC を平面推定法に用いている場合は追加処理が不要である。また、CF 尤度の測定や CFRANSAC に必要なパラメータについて、事前実験により当該環境における適当な値を決定する手法も提案した。

ノイズを含む点群を対象としたシミュレーション実験から、提案した VKOP の相対再現性は、パラメータの微調整が不要かつ、従来のキーポイントよりも 4 倍程度高いことを確認した。また、尤度の導入により、絶対再現性とのトレードオフであるものの、相対再現性を更に向上できることが確認できた。

今後の課題は、CFRANSAC 以外の平面推定法を用いた場合の尤度の検討や、その他の尤度を活用した場合における性能の比較評価が挙げられる。例えば、平面の面積は尤度として活用可能と考えられる [96]。また、複数の尤度を扱う場合、どのように連携すべきか検討が必要である [93, 94]。最後に、VKOP をレジストレーションに用いたときの精度を確認し、その貢献度を確認する必要がある。

(intentionally blank)

第6章

キーポイントパッチを用いた進化計算レジストレーション

進化計算レジストレーションの莫大な計算量を低減するため、特徴点の周囲の点だけを抽出したキーポイントパッチの導入を提案する。

6.1 本章の概要

本章では、高能率なレジストレーション達成のため、ECRに着目した。ECRの特長は、初期位置の依存性がなく、しかも高精度なレジストレーションが可能であることである。一方で課題は莫大な計算量にあるため、精度が悪化しない程度に点群を成す点の個数を削減する必要がある。

そこで、特徴点の性質も考慮し、この周囲の点だけを抽出したキーポイントパッチ (KeyPoint Patches: KPP) を ECR に導入することを提案した [97, 98]。また、点群の境界点付近から KPP の抽出を避けることで、オクルージョンの影響を低減した [98–100]。更に、KPP の個数と、KPP を成す点数の削減効果を確認し、計算量の低減を企図した [98–100]。

評価実験により、KPP を導入することで、従来の ECR よりも精度を維持しながら、最大で約 100 倍以上高速処理できることを確認した [98]。

6.2 従来法の課題と解決方針

3.3.3 項で述べたとおり、ECR の課題は莫大な計算量にある。計算量を増やす要素として、適応度 (FS) 評価のための近傍点探索、個体数、世代数が挙げられる。しかしながら、個体数・世代数の削減は、進化計算の特長そのものを抑制することに等しい。そのため、 FS 評価に用いる点数を削減することが、課題解決

の基本方針となる。

リサンプリングによる点数の削減は、レジストレーション高速化の前処理としてよく用いられる [39]。これは FS 評価に用いる点数も同様に削減されるため、レジストレーション精度とのトレードオフとなる。しかしながら、 ECR は FS の値によって個体を進化させるか否かを決定するため、単純な点数削減では適切な進化を阻害し、収束性の悪化や、局所解への収束の可能性を高めると考えられる。

6.3 KPP 抽出法

FS 評価に極力影響を与えず点数を削減するために、特徴点の性質に着目した。2.2.1 項で述べたように、特徴点は再現性と弁別性を有する点を点群から探索し、採用するものである。この特徴点の用途は主にマッチングにより点群間の対応を取ることであるが、これを点数削減のために用いることで、計算量を低減しながらも適切に FS 評価ができると考えられる。しかしながら、特徴点の点数は、元の点群の点数と比較して極めて少ないため、これだけでは FS 評価に悪影響を及ぼす可能性がある。そこで、特徴点の周囲にある点を抽出し、ある程度の点数を確保することを企図した KPP 抽出法を提案する。KPP を抽出する支持半径は、 Rd_{kpp} により指定する。

図 6.1 は、KPP 抽出の例を示す。(1) は、まずキーポイントを検出しており、緑の丸で示す 10 個のキーポイントが検出されている。(2) が KPP 抽出であり、黄緑色の丸が抽出された KPP である。元の点群が 6813 点であるのに対し、KPP は 267 点であり、全体の約 3.9% に点数が削減されている。

6.4 KPP 抽出の主要素

KPP 抽出法は、キーポイント検出とその周囲の点の抽出から成る。前者は KPP の位置と個数を定め、後者は KPP を成す点数を定める。これら 3 つの主要素が KPP の性能を左右するといえるので、各要素ごと KPP に与える影響を考察し、

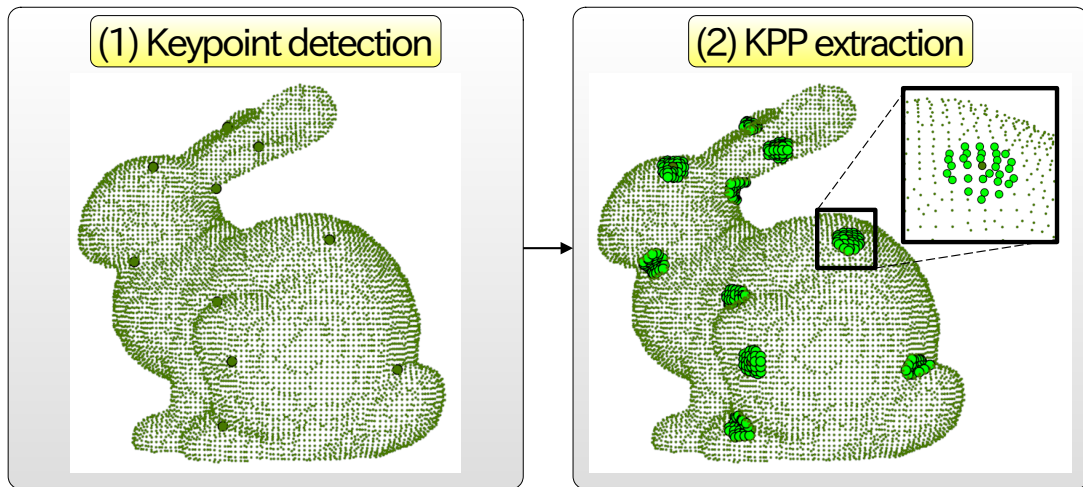


図 6.1 : KPP 抽出の例

それぞれの改良手法を示す.

6.4.1 KPP の位置

抽出した KPP を用いることで, ECR における FS 評価の近傍点探索回数を削減できるが, 式 (3.5) の d_i の数も減少するため, オクルージョンの影響が強くなる. 特に, ある KPP が \mathbb{T} のオクルージョン領域に抽出された場合には, この KPP の d_i の値が大きくなるため, 局所解に収束する可能性が高くなるといえる.

そこで, オクルージョンの性質に着目した. 図 1.2 に示したように, オクルージョンはセンサへの表面露出が遮られることによって生じる. そのため, 点群がオープンとなっている領域, すなわち境界付近にはオクルージョンが存在する可能性が高いといえる. この性質を利用し, 境界付近の点をキーポイント検出に用いないことで, オクルージョンの影響を回避する. 本手法を適用した KPP 法を, KPP-BR (Boundary Removal) 法と呼称する.

Stanford Dragon の KPP 抽出例を図 6.2 に示す. 図 6.2(a) のような, 境界に囲まれた部分に抽出された KPP は, \mathbb{T} 側もオクルージョンである (重なる点がない) 場合が多い. 図 6.3 は, KPP-BR の抽出位置を示す. 境界点は Rusu の手法 [87] で抽出し, 黒色の丸で表示した. これは, 支持半径 Rd_{nor} から得た法線ベ

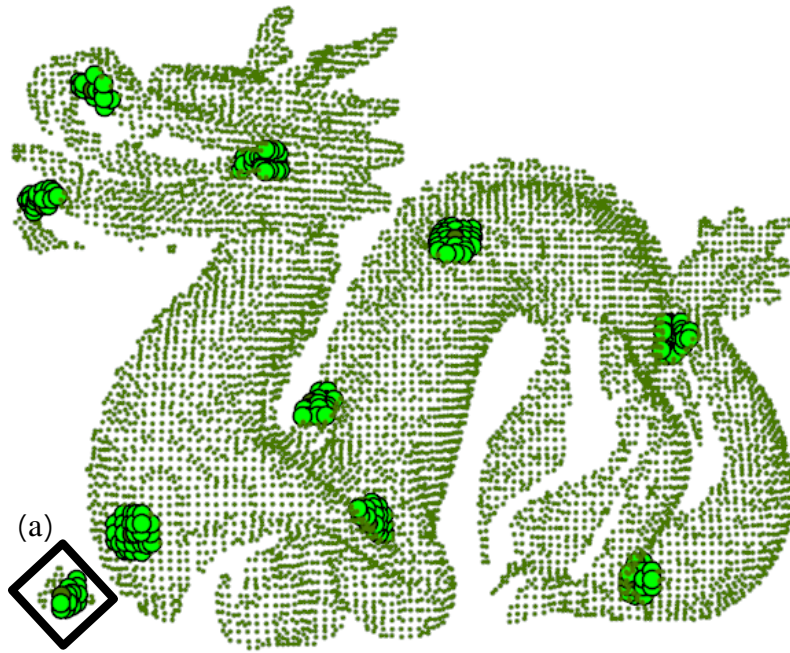


図 6.2 : Stanford Dragon の KPP 抽出例

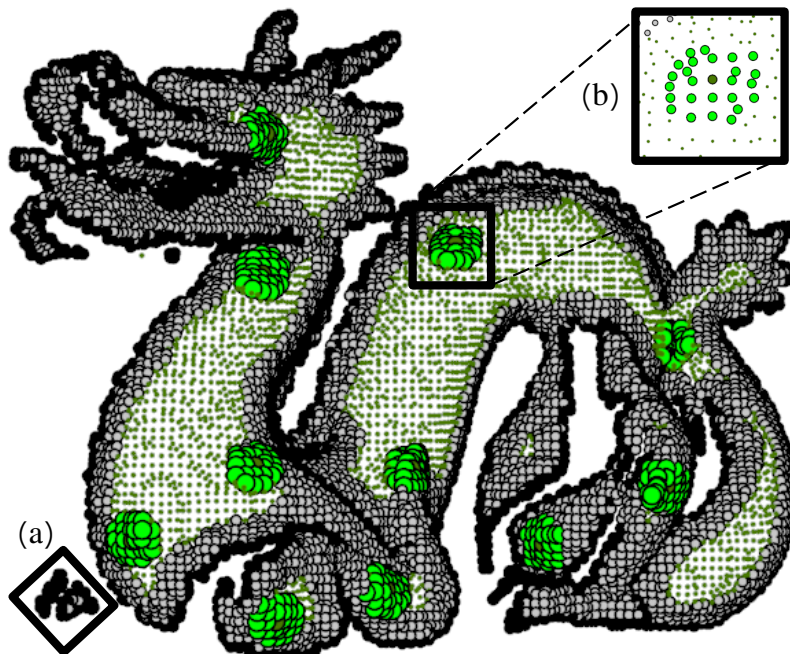


図 6.3 : Stanford Dragon の KPP-BR 抽出例

クトルを活用し、ある点の支持半径 Rd_{bor} 内にある点分布、角度閾値 Th_{bor} を用いて全点を境界か否か判定する手法である。オクルージョンから KPP を離隔するため、境界点から半径 Rd_{off} にある点はオフセットとして、これらの点も境界の点と同等に扱う。図 6.3 の灰色の丸が、オフセットのための点である。その後、境界を除去した点群からキーポイントを検出し、KPP は形状の完全性を確保するため元の点群から抽出する。図 6.2 と図 6.3 を比較すると、両図の (a) に示すような位置のほか、境界点付近に KPP がいないことが確認できる。KPP 法で抽出した KPP の点数は 198 点（元の約 2.6%）、KPP-BR 法は 312 点（元の約 4.1%）であった。KPP-BR 法の点数が増加した理由は、KPP がオクルージョンによって欠けることなく Rd_{kpp} 内の点を抽出したためである。

6.4.2 KPP の個数

2.2.2 項で述べたとおり、キーポイントは、点の周囲の形状から算出される突出値をもとに検出される。単に突出値が大きい順に取り出した場合、ある箇所にキーポイントが集中する可能性がある。これを避けるため、指定した支持半径内で最大の突出値をもつ点のみをキーポイントとして検出する NMS 処理がよく用いられる [47]。同様に KPP も、少ない数で全体に分散していることが望ましいことから、 Rd_{nms} の支持半径で NMS 処理することで、KPP の個数を削減する。ただし、探索範囲を広げることとなるため、KPP 抽出に要する計算量は増加することとなる。

図 6.4 は、図 6.3 の Rd_{nms} を広げて KPP を 4 個に制限した例である。図 6.2 では抽出されていた KPP が、図 6.4 の赤枠の領域に抽出されていない。これは、図 6.4 で残った KPP のもととなったキーポイントの突出値が、消えた KPP のものよりも高かったためである。図 6.4 における KPP の点数は 115 点であり、元の点群の約 1.5% だった。

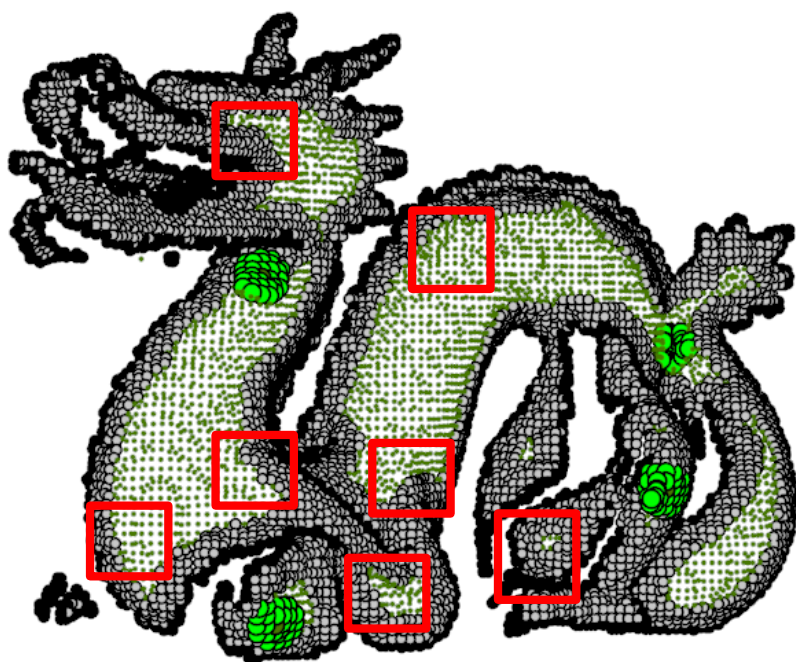


図 6.4 : KPP の個数削減例

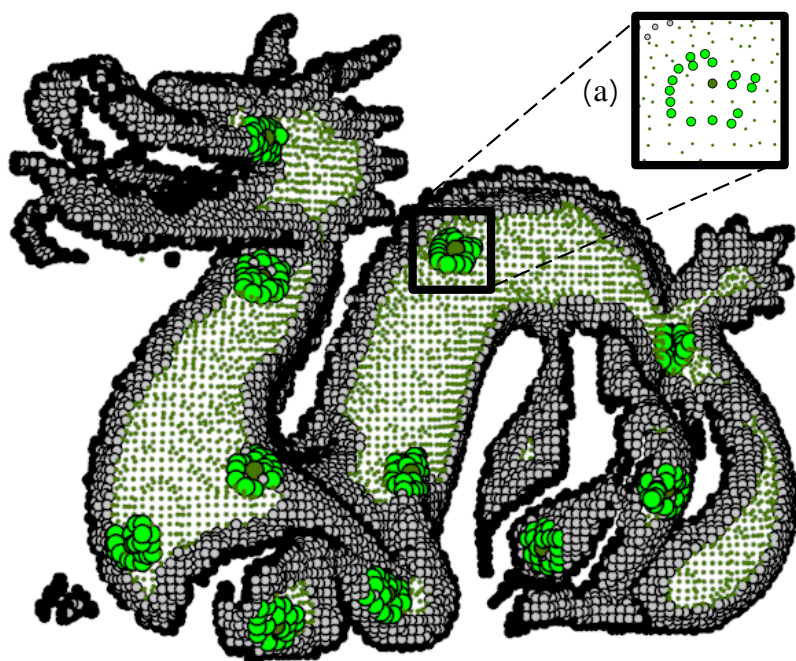


図 6.5 : KPP を成す点のリサンプリング例

6.4.3 KPP を成す点数

KPP を成す点数は、KPP を抽出する範囲や、抽出範囲に含まれる点の解像度によって変化する。ここでは抽出する範囲を一定とし、そのなかの点をリサンプリングすることでレジストレーションの計算量を低減する。

図 6.5 は、図 6.3 の KPP を 50% の割合でランダムに間引いた例である。図 6.5(a) は、図 6.3(b) と同じ位置の KPP であるが、KPP を成す点である黄緑色の丸が減っていることが確認できる。このときの KPP の点数は 205 点であり、元の点群の約 2.7% であった。

6.5 KPP を用いた ECR

図 6.6 は、KPP を用いた ECR の処理フローを示す。元の ECR と異なる箇所は、 S から抽出した KPP を ECR のソース点群 S' として用いることと、 FS の算出法を ICP と同様の FS_{sum} に変更したことである。 T から KPP を抽出しないの

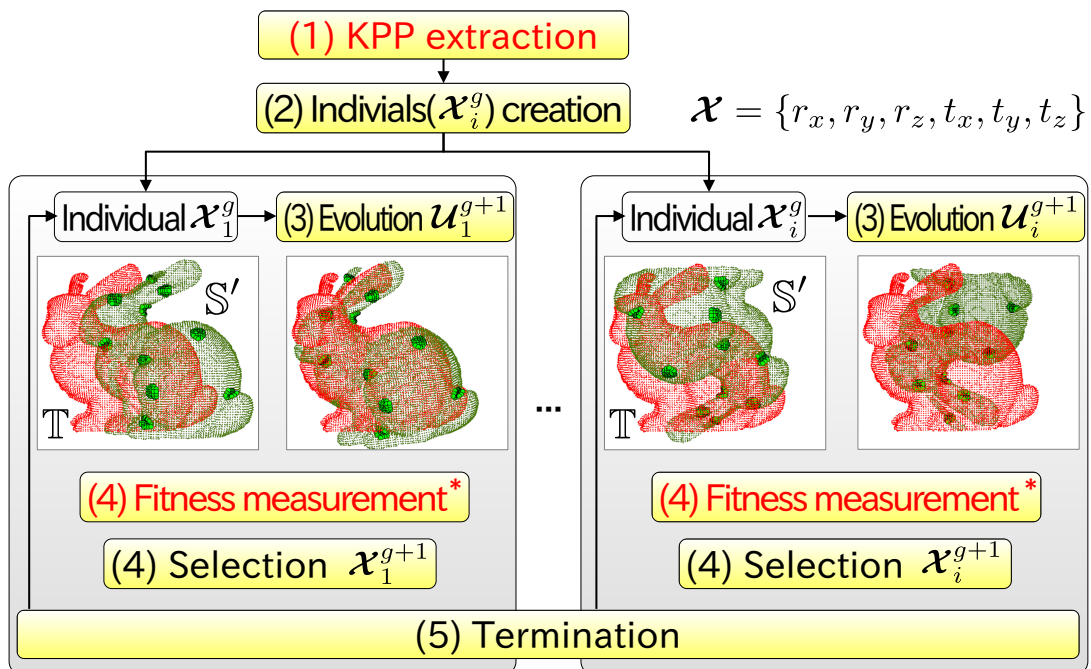


図 6.6 : KPP を用いた ECR の処理フロー

は、重なりのある領域を減らさないためである。キーポイントの再現性が極めて高い場合は、 \mathbb{T} からも KPP を抽出して計算量を削減可能であるが、再現性に課題があることは 2.4 節でも述べたとおりである。また FS の算出に他の ECR と同様 FS_{med} を用いないのは、一部の KPP だけが \mathbb{T} に重なり、そのまま局所解に陥ることを避けるためである。どちらも、安定的に FS を算出するための処置であり、まとめると式 (3.3) を満たす

$$FS_{sum}(\mathbf{T}_{\mathcal{X}}, \mathcal{S}', \mathbb{T}) = \sum_{i=1}^{N_{s'}} d_i \quad (6.1)$$

により FS を求める。 $N_{s'}$ は、 \mathcal{S}' の点数である。なお、文献 [98] では d_i の平均値を使用することとしているが、3.2.4 項と同じ理由で追加計算の必要はない。

6.6 評価実験

提案方式を評価するため、変換パラメータが公開されており、ノイズのほとんどない点群を用いて実験を行った。まず、KPP を用いた ECR の性能を従来方式と比較し、次に KPP-BR によるオクルージョンの影響低減効果を確認する。さらに、KPP の個数、KPP を成す点数を削減することによる計算量削減効果を確認する。最後に、点群にノイズを付加し、その耐性を確認する。

6.6.1 実験環境

(1) ハードウェア及びソフトウェア

本実験に用いたハードウェア及びソフトウェアを表 6.1 に示す。なお、並列計算は行っていないので、GPU は省略した。ソフトウェアについて、実験環境の実装はすべて Point Cloud Library (PCL)[88] を用いた。

(2) データセット

実験対象の点群は、The Stanford 3D Scanning Repository[23] の Armadillo(A), Stanford Bunny(B), Dragon(D), Happy Buddha(H) を用いた。正面からセンシングされたものをターゲット点群とし、その左右の異なる位置からセンシングさ

表 6.1 : ハードウェア及びソフトウェア

CPU	Intel Core i5-4570SK (3.50GHz)
Memory	8.0 [GByte]
OS	Ubuntu ver. 14.04.3 (64bit)
Compiler	Clang++ ver. 3.4
Program library	PCL ver. 1.7.2

れた2つをソース点群とする。これらは、実験上の処理時間短縮のため、事前にすべての点群を1辺2.0[mm]のボクセルでリサンプリングした。ボクセルサンプリングはPCLに実装されている関数をそのまま用いた。これは、ボクセル内にある点の重心点を取り出すものである。

データセットの点群の外観を図6.7に、詳細を表6.2に示す。“Attr.”は点群の属性を示し、“tgt”はターゲット点群，“src”はソース点群である。“mr”はMesh Resolutionの略であり、最近傍点間の距離の中央値により求めた。これが小さいほど点の解像度が高くなり、形状の表現が精緻となる。以後は、mrを基本単位として扱う。“#Pts”は、点群を構成する全点数である。“Angle”は、センシング時におけるオブジェクトのY軸回転度数（ターゲットとの相対角度）である。“%Over”は、ソース点群を真値で変換したときのターゲット点群に対する重なる点数の割合であり、値が大きいほどオクルージョン領域が少ないことを示す。

(3) 評価指標

本実験では、レジストレーションの成功回数、成功時のレジストレーション精度、処理時間の3項目で評価する。レジストレーション精度は、式(2.13)に示したレジストレーション誤差 RE を用いた。 S の解像度よりも RE が小さい($RE < 1.0[\text{mr}]$)ときを、レジストレーション成功とみなし、このときの RE をレジストレーション精度とした。処理時間は、成否にかかわらず全試行回数の平均値とした。

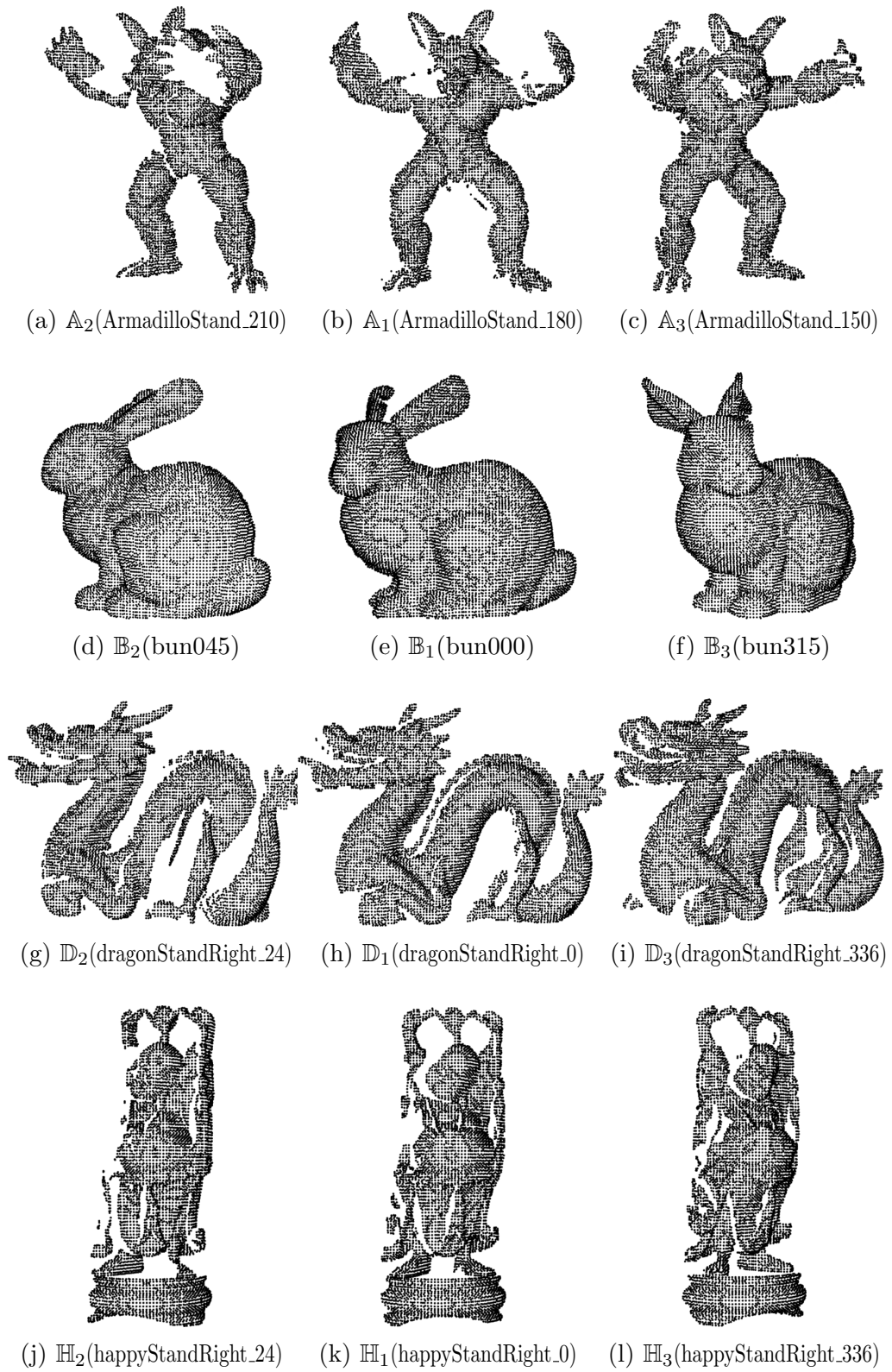


図 6.7 : 実験に用いた点群

表 6.2 : データセットの詳細

ID	Attr.	Filename	mr [mm]	#Pts	Angle [deg]	%Over	View
A ₁	tgt	ArmadilloStand_180	1.3	5,292	-	-	Fig.6.7(b)
A ₂	src	ArmadilloStand_210	1.3	5,095	+30	76.5	Fig.6.7(a)
A ₃	src	ArmadilloStand_150	1.4	5,559	-30	68.5	Fig.6.7(c)
B ₁	tgt	bun000	1.4	7,133	-	-	Fig.6.7(e)
B ₂	src	bun045	1.4	6,813	+45	86.1	Fig.6.7(d)
B ₃	src	bun315	1.3	6,831	-45	74.5	Fig.6.7(f)
D ₁	tgt	dragonStandRight_0	1.3	7,155	-	-	Fig.6.7(h)
D ₂	src	dragonStandRight_24	1.3	6,312	+24	89.1	Fig.6.7(g)
D ₃	src	dragonStandRight_336	1.3	7,512	-24	81.1	Fig.6.7(i)
H ₁	tgt	happyStandRight_0	1.3	5,560	-	-	Fig.6.7(k)
H ₂	src	happyStandRight_24	1.3	5,304	+24	79.6	Fig.6.7(j)
H ₃	src	happyStandRight_336	1.3	5,600	-24	79.7	Fig.6.7(l)

(4) 使用方式とパラメータ

本実験では、KPPの抽出元となるキーポイント検出にISSを用いた。これは、計算量が他のキーポイント検出器よりも小さく、再現性もあるバランスの良いキーポイントであると評価されている[47]。ISS以外にも、KPP抽出とレジストレーションに必要な法線推定、境界抽出、近傍点探索は、PCLに実装されている関数をそのまま用いた。進化計算にSADEを用いたECRは、自前で実装した。このとき、乱数の生成には、Boostライブラリで実装されているメルセンヌ・ツイスタ[101]を用いた。ECRの比較対象としては、PCLに実装されているICPをそのまま使用した。

各手法のパラメータを表6.3に示す。“*”の付いているパラメータは、PCLに

表 6.3 : 実験に用いたパラメータ

Method	Parameter	Value	Unit	Remark
Boundary	Rd_{nor}^*	10.0	mr	Pre. ex.
	Rd_{bor}^*	4.0	mr	Pre. ex.
	Th_{bor}^*	90.0	deg	[87]
ISS	Rd_{sal}	10.0	mr	Pre. ex.
	Th_{21}^*	0.600	-	Pre. ex.
	Th_{32}^*	0.975	-	[50]
	Rd_{nms}^*	-	mr	Various
	Keypoints	10	-	Fix
KPP	Rd_{kpp}	4.0	mr	Pre. ex.
	Rd_{off}	5.0	mr	Pre. ex.
ECR	Population	30	-	Pre. ex.
	Rotation	± 180	deg	[74]
	Translation	± 0.04	m	[74]
SADE	F_l	0.1	-	[77]
	F_u	0.9	-	[77]
	τ_1	0.1	-	[77]
	τ_2	0.1	-	[77]
Termination	Th_{FS}	$1.0e^{-10}$	m^2	Pre. ex.
Experiment	Trial times	30	-	[74]

実装されているものである。“Pre. ex.”は予備実験で得た最適の値である。ISSで指定可能なパラメータは、検出に用いる支持半径 Rd_{sal} と、各固有値の割合の閾値である Th_{21}, Th_{32} 、そしてNMSの支持半径 Rd_{nms} の4つがある。文献 [50] では $Th_{21} = 0.975$ が推奨されているが、この値では平面上からも数多くのキー

ポイントが検出されたため、本実験では $Th_{21} = 0.6$ として、より突出した位置のキーポイントのみを検出した。また ISS の Rd_{nms} は固定せず、各点群で 10 個のキーポイントが検出されるように調整した。ECR の回転と並進は、表 6.3 に示す範囲内でランダムに値を取得し、30 個の個体を生成した。終了条件は、式 (3.4) に示した FS の変化量とした。以上の条件で、次に述べる実験をそれぞれ 30 回試行する。

6.6.2 KPP の性能評価実験

ここでは、KPP の性能を確認する。表 6.4 は、ICP、従来の ECR（表中では ECR と表記）及び KPP 法を用いた ECR（表中では KPP と表記）の試行結果である。“#Pts” はソース点群における処理対象点数、すなわち従来方式はソース点群そのものの構成点数 N_s 、提案方式は KPP の構成点数 $N_{s'}$ を示した。“Success Num.” は成功とみなす RE にまで収束した回数，“Success RE ” は成功時の RE の平均値である。処理時間は、“Prep. Time” が KPP 抽出までに要した処理時間の平均，“ECR Time” が ECR 開始から終了条件に至るまでの処理時間の平均である。一度も成功に至らなかったデータセットの RE には、全体の中央値を括弧内に示した。図 6.8 は、KPP によらない ECR の平均収束時間である 200[sec] ま

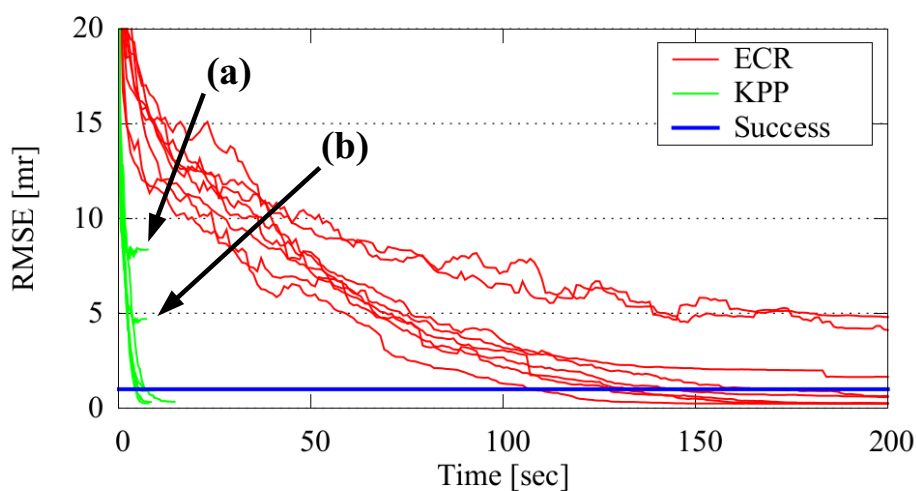


図 6.8 : ECR の収束の様子

表 6.4 : KPP の性能評価実験結果

Dataset	Method	#Pts	Success		Time [sec]	
			Num.	<i>RE</i>	Prep.	ECR
$\{\mathbb{A}_1, \mathbb{A}_2\}$	ICP	5095	0/30	(13.945)	-	0.354
	ECR	5095	30/30	0.238	-	176.362
	KPP	234	30/30	0.329	0.207	4.865
$\{\mathbb{A}_1, \mathbb{A}_3\}$	ICP	5559	0/30	(12.957)	-	0.498
	ECR	5559	28/30	0.472	-	221.067
	KPP	252	29/30	0.901	0.242	4.761
$\{\mathbb{B}_1, \mathbb{B}_2\}$	ICP	6813	0/30	(9.931)	-	1.116
	ECR	6813	22/30	0.644	-	330.911
	KPP	267	30/30	0.337	0.263	7.303
$\{\mathbb{B}_1, \mathbb{B}_3\}$	ICP	6831	0/30	(9.765)	-	0.853
	ECR	6831	29/30	0.404	-	227.242
	KPP	237	0/30	(4.696)	0.246	4.606
$\{\mathbb{D}_1, \mathbb{D}_2\}$	ICP	6312	10/30	0.352	-	0.702
	ECR	6312	29/30	0.211	-	173.601
	KPP	201	30/30	0.253	0.272	5.022
$\{\mathbb{D}_1, \mathbb{D}_3\}$	ICP	7512	7/30	0.868	-	0.992
	ECR	7512	25/30	0.580	-	275.054
	KPP	198	0/30	(8.354)	0.329	4.727
$\{\mathbb{H}_1, \mathbb{H}_2\}$	ICP	5304	0/30	(8.567)	-	0.643
	ECR	5304	30/30	0.270	-	139.846
	KPP	248	29/30	0.596	0.211	5.426
$\{\mathbb{H}_1, \mathbb{H}_3\}$	ICP	5600	11/30	0.887	-	1.262
	ECR	5600	30/30	0.237	-	130.812
	KPP	228	29/30	0.754	0.206	4.522

での、処理時間と RE の平均値の関係を示す。8個のデータセットにおける収束の様子を、従来方式は赤線、提案方式は緑線で表示した。青線は成否の判定基準 ($1.0[\text{mr}]$) である。

ICP は他の方式よりも最も高速に収束したが、その成功率は2つの ECR よりも大きく下回り、レジストレーションはほとんど成功しなかった。実際は、最適解付近 ($RE \approx 1.5[\text{mr}]$) に収束したものも存在するが、これを成功とみなしても全体の成功回数は10回程度であった。すなわち、本実験に用いた範囲で回転及び並進した初期位置にあるデータセットの場合、ICPの使用は適切ではないといえる。

従来の ECR は、約93%の割合で成功していた。しかし、処理時間は平均して約200[sec] かかり、文献 [61] のとおり膨大な計算量を必要とした。提案方式の KPP を用いた ECR は、 $\{\mathbb{B}_1, \mathbb{B}_3\}$ と $\{\mathbb{D}_1, \mathbb{D}_3\}$ を除き、約98%の割合で成功し、その処理時間は5.1[sec] 程度であった。図 6.8 にも示すとおり、収束までの時間的な優位性は明らかである。ただし、図 6.8(a) は $\{\mathbb{D}_1, \mathbb{D}_3\}$ 、図 6.8(b) は $\{\mathbb{B}_1, \mathbb{B}_3\}$ の収束結果であり、どちらも成否判定基準には至らずに Th_{FS} を満たして終了した。この2つのデータセットは、終了条件を設定せずに処理を継続しても RE の変動は0.001程度であり、成功に転じることはなかった。

成功時の RE を比較すると、 $\{\mathbb{B}_1, \mathbb{B}_2\}$ を除き、提案方式のほうが低精度であった。これは、KPPの抽出位置がターゲット点群側でオクルージョンの欠損があり、 $N_{s'}$ が小さいため影響を低減できなかったと考えられる。 $\{\mathbb{B}_1, \mathbb{B}_2\}$ は、全KPPがオクルージョン領域に抽出されなかったため、提案方式が最も高精度であった。

$\{\mathbb{B}_1, \mathbb{B}_3\}$ と $\{\mathbb{D}_1, \mathbb{D}_3\}$ のデータセットについては、KPPの一部がオクルージョン領域に抽出されていたことを確認した。図 6.9 は、 $\{\mathbb{D}_1, \mathbb{D}_3\}$ を用いた試行結果の一例である。赤点は T、緑点は S、緑色の大きい丸はキーポイント、黄緑色の大きい丸は KPP を示しており、黒線は T の最近傍点への距離を示す。黒線が短いほど FS が小さくなり、KPP の位置が適正であればこれに伴って RE も小さ

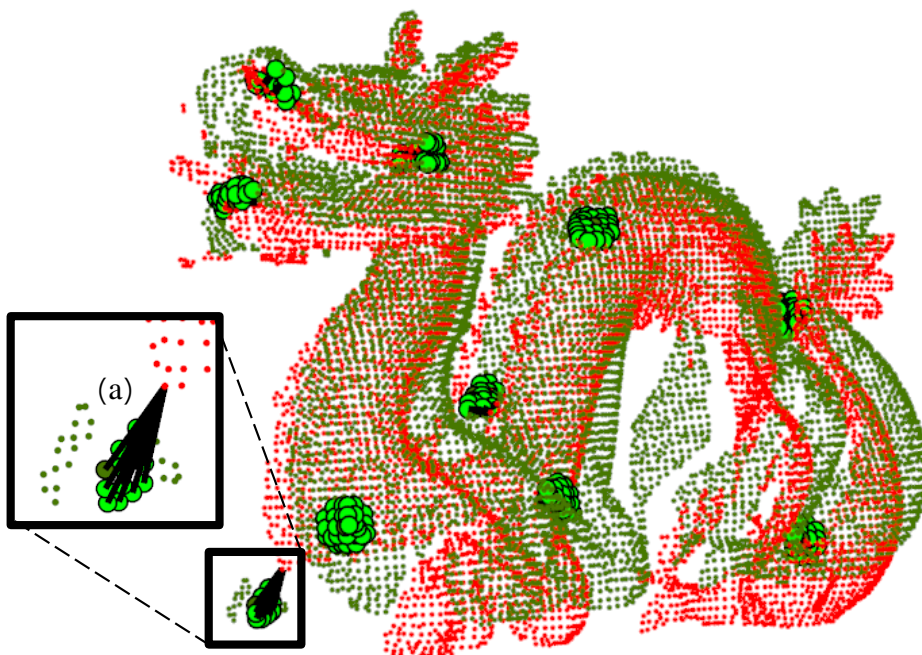


図 6.9 : オクルージョンの影響

くなる．しかしながら，図 6.9(a) に示す KPP が T のオクルージョンの位置にあり，長い黒線が多くあることがわかる．すなわち，ある程度まで最適解に収束した段階で，オクルージョンの影響を排除しきれず局所解に収束したといえる．

以上から，ICP ではレジストレーションが成功できないデータセットであっても，ECR では概ね成功できることを確認した．また，KPP を用いた ECR は 2 個のデータセットを除き，従来の ECR と同等の成功率を示し，かつ約 40 倍高速であることを確認した．成功率を考慮すると，KPP を用いた ECR が最も優れたレジストレーション方式であるといえる．ただし，成功時の RE は従来方式よりもやや低く，KPP の位置によっては局所解に至るデータセットもあった．これらの原因は，オクルージョンの影響を強く受けてしまったためである．

6.6.3 KPP-BR の性能評価実験

ここでは，KPP-BR の性能を確認した．KPP を用いた ECR と，KPP-BR を用いた ECR の試行結果を表 6.5 に，収束時間と RE の関係を図 6.10 に示す．

表 6.5 : KPP-BR の性能評価実験結果

Dataset	Method	#Pts	Success		Time [sec]	
			Num.	<i>RE</i>	Prep.	ECR
$\{A_1, A_2\}$	KPP	234	30/30	0.329	0.207	4.865
	KPP-BR	268	30/30	0.215	0.072	5.842
$\{A_1, A_3\}$	KPP	252	29/30	0.901	0.242	4.761
	KPP-BR	287	28/30	0.473	0.086	5.709
$\{B_1, B_2\}$	KPP	267	30/30	0.337	0.263	7.303
	KPP-BR	302	28/30	0.266	0.194	7.055
$\{B_1, B_3\}$	KPP	237	0/30	(4.696)	0.246	4.606
	KPP-BR	284	30/30	0.639	0.162	8.665
$\{D_1, D_2\}$	KPP	201	30/30	0.253	0.272	5.022
	KPP-BR	269	27/30	0.147	0.101	7.890
$\{D_1, D_3\}$	KPP	198	0/30	(8.354)	0.329	4.727
	KPP-BR	284	30/30	0.228	0.133	6.772
$\{H_1, H_2\}$	KPP	248	29/30	0.596	0.211	5.426
	KPP-BR	282	28/30	0.437	0.074	7.406
$\{H_1, H_3\}$	KPP	228	29/30	0.754	0.206	4.522
	KPP-BR	285	29/30	0.527	0.081	5.843

$\{B_1, B_3\}$ と $\{D_1, D_3\}$ の結果が大きく改善されており、全体でも約97%の割合でレジストレーションが成功した。境界点を除去することによって、オクルージョンの位置に KPP を抽出しなかった結果といえる。成功時の *RE* は全て改善され、これらの平均値 0.367 は、KPP によらない従来の ECR の平均値 0.382 よりも低かった。図 6.10 における KPP-BR の結果（紫線）をみても、成否判定基準である青線を大きく上回るものがないことがわかる。わずかに青線を上回っているものは、失敗時における *RE* の影響を受けた結果である。

6.4.1 項で述べたように、KPP を成す点が KPP-BR では増加しており、ECR の

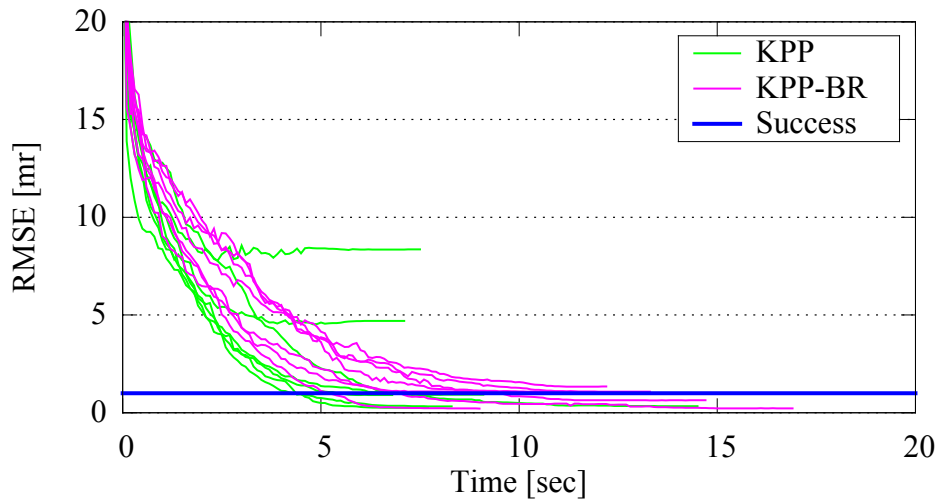


図 6.10 : KPP と KPP-BR の収束の様子

処理時間は平均して約 1.7[sec] 増加した。 $\{\mathbb{B}_1, \mathbb{B}_2\}$ のとき KPP-BR の方が高速であった理由は、収束に至るまでの世代数が少なかったためである。KPP-BR の前処理時間が全体的に短くなっている理由は、境界点を除去したことにより、ISS 検出に要する計算量が削減されたためである。

以上から、KPP-BR による ECR が最も効果的であることが確認できた。すなわち、KPP を用いた ECR には、KPP-BR の適用が必須といえる。KPP によらない従来の ECR の成功率と精度と比較しても、KPP-BR を用いた ECR のほうが優れており、処理時間は約 30 倍高速である。

6.6.4 KPP の個数と KPP を構成する点数の評価実験

ここでは、KPP-BR の抽出結果を基準に、KPP の個数と、KPP を構成する点数をそれぞれ削減して効果を確認した。個数削減については、 Rd_{nms} を手動で微調整し、 $\{10, 8, 6, 4\}$ 個の KPP を抽出した。点数削減については、 $\{100, 75, 50, 25\}$ [%] の割合でランダムにリサンプリングした。個数削減の結果は表 6.6(a)、点数削減の結果は表 6.6(b) に示す。

表 6.6(a) をみると、個数削減により処理時間を削減できているが、成功率と成功時の RE は悪化した。特に、 $\{\mathbb{H}_1, \mathbb{H}_2\}$ と $\{\mathbb{H}_1, \mathbb{H}_3\}$ において、KPP が 6 個以下

表 6.6 : KPP 高速化の評価実験結果

(a) KPP の個数削減							(b) KPP を成す点の削減						
DS	Num.	#pts	Success		Time [sec]		DS	Rate [%]	#pts	Success		Time [sec]	
	KPP		Num.	RE	Prep.	ECR				Num.	RE	Prep.	ECR
$\{A_1, A_2\}$	10	268	30/30	0.215	0.072	5.842	$\{A_1, A_2\}$	100	268	30/30	0.215	0.072	5.842
	8	211	27/30	0.251	0.078	4.748		75	200	30/30	0.251	0.072	4.069
	6	156	29/30	0.309	0.080	3.262		50	136	30/30	0.349	0.073	3.038
	4	106	10/30	0.234	0.107	2.712		25	71	28/30	0.410	0.071	1.645
$\{A_1, A_3\}$	10	287	28/30	0.473	0.086	5.709	$\{A_1, A_3\}$	100	287	28/30	0.473	0.086	5.709
	8	228	24/30	0.473	0.094	4.583		75	214	28/30	0.464	0.090	4.385
	6	173	29/30	0.575	0.096	3.672		50	144	26/30	0.490	0.090	2.868
	4	114	21/30	0.628	0.123	2.506		25	76	24/30	0.482	0.088	1.558
$\{B_1, B_2\}$	10	302	28/30	0.266	0.194	7.055	$\{B_1, B_2\}$	100	302	28/30	0.266	0.194	7.055
	8	242	27/30	0.361	0.233	6.281		75	226	30/30	0.307	0.194	5.857
	6	180	28/30	0.441	0.274	5.184		50	152	30/30	0.355	0.196	4.071
	4	120	9/30	0.558	0.364	4.474		25	80	29/30	0.387	0.192	2.081
$\{B_1, B_3\}$	10	284	30/30	0.639	0.162	8.665	$\{B_1, B_3\}$	100	284	30/30	0.639	0.162	8.665
	8	233	30/30	0.594	0.164	6.772		75	212	30/30	0.639	0.161	6.390
	6	172	30/30	0.704	0.199	4.937		50	143	30/30	0.629	0.160	4.319
	4	115	9/30	0.785	0.234	3.949		25	75	30/30	0.654	0.159	2.382
$\{D_1, D_2\}$	10	269	27/30	0.147	0.101	7.890	$\{D_1, D_2\}$	100	269	27/30	0.147	0.101	7.890
	8	210	26/30	0.110	0.101	5.391		75	200	29/30	0.117	0.099	5.496
	6	158	25/30	0.134	0.110	4.145		50	136	29/30	0.151	0.098	3.596
	4	101	16/30	0.322	0.116	3.000		25	71	28/30	0.206	0.098	2.241
$\{D_1, D_3\}$	10	284	30/30	0.228	0.133	6.772	$\{D_1, D_3\}$	100	284	30/30	0.228	0.133	6.772
	8	227	28/30	0.263	0.136	5.755		75	212	30/30	0.230	0.131	4.941
	6	170	22/30	0.285	0.140	4.373		50	143	28/30	0.271	0.131	3.280
	4	115	14/30	0.684	0.149	3.084		25	75	28/30	0.299	0.132	1.990
$\{H_1, H_2\}$	10	282	28/30	0.437	0.074	7.406	$\{H_1, H_2\}$	100	282	28/30	0.437	0.074	7.406
	8	229	17/30	0.453	0.074	5.643		75	210	25/30	0.380	0.075	5.242
	6	175	3/30	0.439	0.079	3.472		50	142	25/30	0.427	0.073	3.636
	4	121	5/30	0.605	0.091	2.884		25	74	23/30	0.436	0.073	1.802
$\{H_1, H_3\}$	10	285	29/30	0.527	0.081	5.843	$\{H_1, H_3\}$	100	285	29/30	0.527	0.081	5.843
	8	228	25/30	0.556	0.083	4.907		75	213	26/30	0.528	0.082	4.222
	6	164	11/30	0.875	0.087	3.945		50	144	26/30	0.532	0.080	3.073
	4	111	0/30 (9.651)	0.090	0.090	2.879		25	75	25/30	0.570	0.081	1.694

のときの性能低下は顕著である。図 6.7 より、これらのデータセットが他と比較して突出した部分が少なく、円柱の表面をセンシングしたような形状となっているためと考えられる。点群の形状に適した KPP の数量と位置は異なるといえるが、事前にこれを決定することは困難である。すなわち、KPP の個数削減による高速化は、成功率とのトレードオフになるといえる。

点数削減の結果は、成功率と精度を概ね保ちつつ、処理時間を向上できるため、有効であることを確認できた。サンプリング率 100% と 25% を比較すると、成功率は 96% から 90%、精度は 0.367 から 0.411 程度の悪化で、処理時間は約 3.5 倍改善できた。

以上から、個数削減及び点数削減ともに、高速化の効果があることが確認できた。個数削減は、成功率と処理時間のトレードオフであるが、点数削減は成功率と成功時の精度を極端に落とすことなく処理時間を向上できた。特に、点数削減の 25% リサンプリングでは、リサンプリングしない場合よりも約 3.5 倍高速であった。これは、KPP によらない従来の ECR と比較すると、約 100 倍高速である。点数の削減が効果的であるのは、KPP を用いた ECR の精度が、KPP の位置に強く依存しているためであると考えられる。

6.6.5 ノイズ耐性の評価実験

ここでは、前項で性能を確認した KPP-BR と点数削減の結果を基準に、ノイズを付加したデータセットを用いることで、KPP 法のノイズ耐性を確認した。ターゲット点群及びソース点群の両方に対し、試行の度に異なるシードで 0.5[μm] のレベルのガウシアンノイズを付加した。0.5[μm] レベルは、文献 [47] において最も強いレベルである。その他のパラメータは、前項と全く同じものを使用した。また、本実験ではノイズによりレジストレーション成功数の低下が予想されるため、全試行の RE の中央値も確認することとした。

この実験結果を、表 6.7 に示す。各データセットの第 1 行目は、ノイズのないデータセットの結果を示しており、表 6.6(b) と同値である。表中の #KPP は、抽

表 6.7 : ノイズ耐性の評価実験結果

Dataset (#KPP)	Rate [%]	Median <i>RE</i>	Success		Time [sec]	
			Num.	<i>RE</i>	Prep.	ECR
	100	0.210	30/30	0.215	0.072	5.842
$\{\mathbb{A}_1, \mathbb{A}_2\}$ (11.2)	100	1.110	12/30	0.632	0.069	5.149
	75	1.200	10/30	0.580	0.069	3.434
	50	1.320	10/30	0.793	0.068	2.296
	25	1.213	9/30	0.829	0.069	1.657
	100	0.468	28/30	0.473	0.086	5.709
$\{\mathbb{A}_1, \mathbb{A}_3\}$ (11.1)	100	0.931	19/30	0.779	0.081	4.833
	75	0.891	21/30	0.776	0.078	3.901
	50	0.912	19/30	0.769	0.081	2.689
	25	1.047	12/30	0.748	0.078	1.447
	100	0.265	28/30	0.266	0.194	7.055
$\{\mathbb{B}_1, \mathbb{B}_2\}$ (16.4)	100	1.042	14/30	0.597	0.162	9.749
	75	1.056	14/30	0.662	0.165	7.801
	50	1.044	14/30	0.614	0.165	4.932
	25	1.133	12/30	0.689	0.163	2.922
	100	0.645	30/30	0.639	0.161	8.665
$\{\mathbb{B}_1, \mathbb{B}_3\}$ (21.0)	100	1.031	14/30	0.768	0.135	11.399
	75	1.020	14/30	0.810	0.136	8.059
	50	1.050	12/30	0.772	0.135	6.446
	25	1.078	12/30	0.731	0.135	3.382
	100	0.113	27/30	0.147	0.101	7.890
$\{\mathbb{D}_1, \mathbb{D}_2\}$ (9.9)	100	0.658	27/30	0.594	0.089	5.119
	75	0.738	23/30	0.666	0.089	4.350
	50	0.727	19/30	0.605	0.090	2.655
	25	0.855	20/30	0.723	0.090	1.454
	100	0.221	30/30	0.228	0.133	6.772
$\{\mathbb{D}_1, \mathbb{D}_3\}$ (11.4)	100	0.684	26/30	0.667	0.120	5.522
	75	0.685	25/30	0.667	0.118	3.674
	50	0.730	21/30	0.634	0.118	2.554
	25	0.853	21/30	0.687	0.118	1.358
	100	0.450	28/30	0.437	0.074	7.406
$\{\mathbb{H}_1, \mathbb{H}_2\}$ (13.7)	100	0.997	25/30	0.990	0.065	5.042
	75	1.088	12/30	0.811	0.063	4.189
	50	1.067	13/30	0.809	0.064	2.805
	25	1.155	9/30	0.886	0.064	1.625
	100	0.525	29/30	0.527	0.081	5.843
$\{\mathbb{H}_1, \mathbb{H}_3\}$ (10.1)	100	0.671	25/30	0.585	0.070	4.192
	75	0.706	24/30	0.607	0.068	2.720
	50	0.709	23/30	0.603	0.070	2.020
	25	0.850	20/30	0.718	0.068	1.115

出された KPP の個数の平均値である。KPP の個数が変動した理由は、ノイズによる形状の変化のためである。特に Bunny(B) では、滑らかな表面がノイズによって突出した部分となり、この位置にキーポイントが検出されたため、KPP の個数の増加が顕著であった。Median RE は、全試行の RE の中央値を示す。

サンプリング率が 100% (KPP 点数削減の適用なし) のとき同士を比較したとき、全体的に RE の中央値、成功回数と成功時の RE が悪化したことが確認できる。 RE の中央値に着目すると、ノイズのある場合は全体で 0.528 程度の増加が確認された。ただし、 RE の中央値は 1.0 付近で収束しており、全データセットで最適解に近い解を得られているといえる。成功率に着目すると、 $\{A_1, A_2\}$ 、 $\{B_1, B_2\}$ 及び $\{B_1, B_3\}$ はノイズのある場合、成功率が約 98% から 44% 程度にまで大きく低下していた。これらの RE の中央値が、成功とみなす閾値の 1.0 をわずかに超えており、これに伴って成功回数も半減したと考えられる。また、 $\{A_1, A_3\}$ のサンプリング率 25% と、 $\{H_1, H_2\}$ のサンプリング率 75% 以下の場合の成功率低下も、同様の要因といえる。一方で $\{D_1, D_2\}$ 、 $\{D_1, D_3\}$ 及び $\{H_1, H_3\}$ は、サンプリング率を低下させても成功率を維持していた。これは、KPP の個数と位置がノイズのない場合と同様であったためであり、結果が同じ傾向を示したものと考えられる。処理時間については、点数削減の割合に比例して改善されており、前項の実験と同様の傾向を確認した。

以上から、ノイズのある場合においても、 RE の中央値は成功とみなす精度の解付近にまで収束しており、そのノイズ耐性を確認できた。また、点数削減による効果はノイズのない場合と同様に、成功率と成功時の精度を極端に落とすことなく処理時間を向上できている。

6.6.6 まとめ

本章では、ECR の初期位置に依存せずに高精度な結果を得られるという特長を保持しつつ、計算量を低減するための KPP 抽出法を提案した。KPP は、キーポイントの周囲をパッチ状に抽出した点群である。この提案は ECR の近傍点探

索の回数を大きく削減でき、処理時間の向上に有用である。さらに、境界点除去によるオクルージョンの影響の低減、NMS 処理による KPP 個数の削減、KPP のリサンプリングによる点数の削減といった改善手法も提案した。

8組のデータセットを用いた評価実験により、KPP を用いた ECR は、ICP のように初期位置の影響を受けない特長を有したまま、KPP によらない従来の ECR と比較して約 40 倍高速化できることを確認した。また、境界点を除去する KPP-BR は、オクルージョンを多く含むデータセットであっても、従来の ECR と同等以上の成功率と精度を示した。処理時間の削減については、個数削減手法は成功率とのトレードオフであるが、点数削減は成功率と精度を維持できることを確認した。点数削減にて、元の 25%まで点数を削減してもレジストレーション精度に与える影響は小さかったことから、ECR の性能は KPP の抽出位置に依存するといえる。すなわち、KPP は境界点除去の併用が必須であり、更に高速化が必要な場合には点数削減の採用が適切である。これら提案方式の組み合わせは、KPP によらない従来の ECR と同等以上の成功率と精度を示しつつ、最大で約 100 倍以上の高速化を実現した。

一方で課題としては、KPP に用いる適当な手法や、各種パラメータの比較評価が挙げられる。特に、ISS 以外のキーポイント検出や、SADE 以外の進化計算アルゴリズムの適用は、KPP-ECR に強く影響を与えるといえる。また、境界点抽出、キーポイント検出、進化計算アルゴリズムに必要なパラメータは今回の実験では検討していないので、最適化をする必要がある。

(intentionally blank)

第7章

結論

本論文の成果をまとめ、目的に対する達成度と、残された課題に言及する。

本論文は、オブジェクトの表面形状を疎な点の集まりで表現する3次元点群について、重要な中間処理に位置づけられている特徴点抽出処理の高性能化を目指した。また、特徴点抽出の代表的な活用例であるレジストレーション処理について、低計算量かつ高精度化することを目指した。本論文の対象は、個々の点の関係性や色情報等を持たない未整理の点群であり、また外部システム等は一切使用しない、最も原始的な状態を想定している。このような環境では、オクルージョンによる形状の欠損、ノイズによる点位置の変動、また単純に処理すべき点の個数が多いといった、克服すべき課題が多数存在する。そこで特徴点抽出は、その位置を決定するキーポイント検出に焦点を絞り、従来とは異なる枠組みの抽出方法を実現した。レジストレーションに関しては、特徴点の機能である再現性と弁別性に着目し、従来とは異なる活用方法により高能率化を実現した。

以下、その成果を具体的に述べる。

(1) 再現性を重視した仮想特徴点と尤度評価による高性能化

特徴点は、点群内から抽出した少数個の特徴点同士を対応付けることで、点群間の対応関係を用いた各種技術を低計算量化するために用いられる。特に、特徴点対応によるレジストレーションは、1度の座標系変換でレジストレーションを完了できるため、座標系変換を繰り返す方式や進化計算による方式と比較して高速である。しかしながら、従来の特徴点抽出は弁別性を重視しており、再現性は低調なものが多く、発生した誤対応の影響を強く受けて各種の応用技術の精度には限界があった。

そこで、まずは再現性を重視した特徴点を抽出することを目指した。センシング時に生じる諸問題を抱えた実在点は、いわば最初から再現性が低いことから、これをそのまま用いずに幾何形状の方程式化により再現性を高めることを企図した、仮想特徴点を提案した。この具体化として、センシング時でも安定的に点を取得できる平面形状に着目し、複数平面の交点位置に特徴点を配置する、VKOP (Virtual Keypoint Of Polyhedron) を提案した。シミュレーション実験により、VKOP は従来の特徴点抽出法と比較して、同等の処理時間で最大2倍程度の高い再現性を示すことを確認した。

VKOP の性能が確認された一方で、平面形状の抽出法と、これに用いるパラメータに性能が強く依存する課題が残った。この課題に対しては、平面の安定性を指標化した尤度を、平面抽出と同時に、または事後的に測定し、尤度の低いものは除去する手法を提案した。また、この尤度の測定に適したセンサパラメータを、事前に求めておく手法を提案した。これにより、平面抽出法の依存性を低減するとともに、従来の特徴点よりも短い処理時間で、4倍以上の高い再現性を示すことを確認した。

今後の課題としては、本論文で用いた平面抽出法以外の手法の適用や、そのときの最適な尤度、または全く異なる尤度の導入が挙げられる。さらに、複数の尤度を扱う場合において、その連携の手法には検討が必要である。また、VKOP を用いたレジストレーションを評価し、その貢献度を確認する必要がある。

(2) キーポイントパッチを用いた進化計算レジストレーション

進化計算によるレジストレーションは、点群間の座標系変換に必要な変換パラメータを、進化計算により更新して高精度にレジストレーションする方式である。解を得る手法が進化計算アルゴリズムによるため、局所解を回避して全体最適解を探索することができる。しかしながら、計算量の大きい点群間の近傍点探索処理を、個体数・世代数の必要分だけ繰り返すため、他の方式と比較して処理に時間がかかるという課題がある。

この課題解決のため、特徴点の周囲の点を抽出した KPP (KeyPoint Patches) を進化計算レジストレーションに用いることを提案した。KPP は弁別性の高い形状を有しており、この部分だけを処理に組み込むことで精度を維持しつつ、低計算量化を企図したものである。また、KPP の位置、抽出する個数、KPP を成す点数を考慮することで、更なる高能率化を目指した。

評価実験により、従来の進化計算レジストレーションと比較して、提案方式は同等の精度を維持し、約 40 倍高速に処理できることを確認した。また、KPP を成す点数を削減することで、精度を維持しつつ、従来よりも最大で約 100 倍高速に処理できることを確認した。

今後の課題として、進化計算の手法・特徴点抽出の手法を変更したときの比較や、KPP 抽出範囲等の使用するパラメータの最適化が挙げられる。また、先に提案した VKOP は点群から離れた位置に抽出されるため、このまま KPP に使用できないことから、この 2 つを連携するための方策を検討する必要がある。

謝 辞

本論文は、信州大学学術研究院（工学系） 田中清教授のご指導のもと執筆されたものです。田中教授には、博士課程の在学中、長期間にわたって情熱あふれるご教示・ご鞭撻を賜りました。本論文の執筆に際しても、研究当初からの方向性や方法論について様々なご示唆を賜りました。ここに謹んで厚くお礼を申し上げます。

防衛大学校情報工学科 岩切宗利准教授には、防衛大学校の卒業研究から始まり現在に至るまで、貴重なご示唆を何度も賜りました。特に、東京船堀での月例ゼミにおいて、田中教授とともにお二人から非常に濃厚なご指導を賜ったことは、研究活動以外にも著者自身の血肉となっております。心から厚くお礼を申し上げます。

Sandoval Galvez Jaime Alberto 氏には、3次元点群処理という同じ分野を研究する同僚であるにも関わらず、多くのご助言をいただきました。特に、協力しながら個別に著した論文が、同時に両方とも表彰されたことは、何ものにも代えがたい喜びでした。

同じ分野を研究する研究室の諸氏には、議論に参加していただくとともに、多くのご協力をいただきました。また、研究室事務の渡辺雅子氏と高柳梓氏には、長野県より離れた場所での研究活動であったにも関わらず、きめ細やかなご支援をいただきました。皆様に心から感謝いたします。

最後に、著者の研究活動を励まし続けてくれた職場の方々と、家族に深く感謝いたします。

参考文献

- [1] “行政手続における特定の個人を識別するための番号の利用等に関する法律”, 平成二十五年法律第二十七号 (2020).
- [2] 内閣府: “地方公共団体における押印見直しマニュアル”, 令和2年12月18日 (2020).
- [3] 首相官邸: “世界最先端デジタル国家創造宣言・官民データ活用推進基本計画”, 令和2年7月17日 全部変更の閣議決定 (2020).
- [4] 首相官邸: “デジタル社会の実現に向けた改革の基本方針”, 令和2年12月25日閣議決定 (2020).
- [5] 経済産業省: “デジタルトランスフォーメーションを推進するためのガイドライン”, 平成30年12月 (2018).
- [6] デジタルトランスフォーメーションの加速に向けた研究会: “DXレポート2”, 令和2年12月28日 (2020).
- [7] R. Koch, N. Pears and Y. Liu: “Introduction”, 3D Imaging, Analysis and Applications, Springer, London, pp. 1–31 (2012).
- [8] 碓井茂夫, 野村直樹, 熊谷光, 関根浩史: “運転支援システム“アイサイト”の事故低減への取り組み”, 日本 AEM 学会誌, **25**, 4, pp. 383–389 (2017).
- [9] 内閣府: “戦略的イノベーション創造プログラム (SIP) 自動運転 (システムとサービスの拡張) 研究開発計画”, 令和2年5月14日 (2020).
- [10] 二宮芳樹: “自動運転への適用”, LiDAR の最前線, S&T 出版, 東京都, p. 101 (2019).

- [11] 3D Network Japan: “東日本大震災における 3D 計測関連の初動利活用調査報告書 (2011.3~4 期間) 暫定版”, http://www.3dnet-j.com/other/higashinohon_report.pdf. 最終確認: 2021 年 4 月 20 日.
- [12] 山崎文雄: “リモートセンシングによる広域災害の把握”, 予防時報, **248**, pp. 12–18 (2012).
- [13] H. Wei and M. Bartels: “3D Digital Elevation Model Generation”, 3D Imaging, Analysis and Applications, Springer, London, pp. 367–416 (2012).
- [14] 山崎文雄: “総論 地震防災におけるリモートセンシングの最新動向”, 日本地震工学会誌, **33**, pp. 2–5 (2018).
- [15] 国土交通省: “Project PLATEAU ver1.0 をリリース”, 令和 3 年 3 月 26 日プレスリリース (2021).
- [16] 国土交通省: “3 次元データ利活用方針”, 平成 29 年 11 月 (2017).
- [17] A. Mian and N. Pears: “3D Face Recognition”, 3D Imaging, Analysis and Applications, Springer, London, pp. 311–366 (2012).
- [18] P. G. Batchelor, P.J. “Eddie”, A. P. King: “3D Medical Imaging”, 3D Imaging, Analysis and Applications, Springer, London, pp. 445–495 (2012).
- [19] 映像産業振興機構: “VR 等のコンテンツ制作技術活用ガイドライン 2018”, 平成 30 年 2 月 26 日 (2018).
- [20] 文化庁: “埋蔵文化財保護行政におけるデジタル技術の導入”, 令和元年 2 月 28 日 (2019).
- [21] 井村篤, 竹前和幸, 畔上宏夫, 岩切宗利, 田中清: “RGB-D 画像を用いた遺構図作成に資する特徴抽出”, 第 292 回画像電子学会研究会予稿集 (2020).

- [22] W. A. P. Smith: “Representing, Storing and Visualizing 3D Data”, 3D Imaging, Analysis and Applications, Springer, London, pp. 139–182 (2012).
- [23] The Stanford 3D Scanning Repository: <http://graphics.stanford.edu/data/3Dscanrep/>. 最終確認 : 2021 年 4 月 20 日.
- [24] G. Bradski and A. Kaehler: “Learning OpenCV: Computer vision with the OpenCV library”, O’reilly, California (2008).
- [25] S. Se and N. Pears: “Passive 3D Imaging”, 3D Imaging, Analysis and Applications, Springer, London, pp. 35–94 (2012).
- [26] H. Sarbolandi, D. Lefloch and A. Kolb: “Kinect range sensing: Structured-light versus time-of-flight kinect”, Computer vision and image understanding, **139**, pp. 1–20 (2015).
- [27] M. A. Drouin and J. A. Beraldin: “Active 3D Imaging Systems”, 3D Imaging, Analysis and Applications, Springer, London, pp. 95–138 (2012).
- [28] A. Bronstein, M. Bronstein and M. Ovsjanikov: “Feature-Based Methods in 3D Shape Analysis”, 3D Imaging, Analysis and Applications, Springer, London, pp. 185–220 (2012).
- [29] D. G. Lowe: “Distinctive image features from scale-invariant keypoints”, International journal of computer vision, **60**, 2, pp. 91–110 (2004).
- [30] B. Bustos and I. Sipiran: “3D Shape Matching for Retrieval and Recognition”, 3D Imaging, Analysis and Applications, Springer, London, pp. 265–308 (2012).
- [31] H. Gonzalez-Jorge, B. Riveiro, E. Vazquez-Fernandez, J. Martínez-Sánchez

- and P. Arias: “Metrological evaluation of Microsoft Kinect and Asus Xtion sensors”, *Measurement*, **46**, 6, pp. 1800–1806 (2013).
- [32] H. Gonzalez-Jorge, P. Rodríguez-González, J. Martínez-Sánchez, D. González-Aguilera, P. Arias, M. Gesto and L. Díaz-Vilariño: “Metrological comparison between kinect I and kinect II sensors”, *Measurement*, **70**, pp. 21–26 (2015).
- [33] Verodyne HDL-64E : <https://velodynelidar.com/products/hdl-64e/>.
最終確認 : 2021 年 4 月 20 日.
- [34] J. C. Chow and D. D. Lichti: “Photogrammetric bundle adjustment with self-calibration of the PrimeSense 3D camera technology: Microsoft Kinect”, *IEEE Access*, **1**, pp. 465–474 (2013).
- [35] B. Choo, M. Landau, M. DeVore and P. A. Beling: “Statistical analysis-based error models for the microsoft kinecttm depth sensor”, *Sensors*, **14**, 9, pp. 17430–17450 (2014).
- [36] Barcelona Robot Lab Dataset: <http://www.iri.upc.edu/research/webprojects/pau/datasets/BRL/>. 最終確認 : 2021 年 4 月 20 日.
- [37] J. Xiao, J. Zhang, B. Adler, H. Zhang and J. Zhang: “Three-dimensional point cloud plane segmentation in both structured and unstructured environments”, *Robotics and Autonomous Systems*, **61**, 12, pp. 1641–1652 (2013).
- [38] M. Gschwandtner, R. Kwitt, A. Uhl and W. Pree: “Blensor: blender sensor simulation toolbox”, *Advances in Visual Computing*, Springer, pp. 199–208 (2011).

- [39] U. Castellani and A. Bartoli: “3D Shape Registration”, 3D Imaging, Analysis and Applications, Springer, London, pp. 367–416 (2012).
- [40] P. Besl and H. McKay: “A method for registration of 3-D shapes”, IEEE Transactions on Pattern Analysis and Machine Intelligence, **14**, pp. 239–256 (1992).
- [41] G. Turk and M. Levoy: “Zippered polygon meshes from range images”, Proceedings of the 21st annual conference on Computer graphics and interactive techniques ACM, pp. 311–318 (1994).
- [42] S. Hong, G. Saavedra and M. Martinez-Corral: “Full parallax three-dimensional display from kinect v1 and v2”, Optical Engineering, **56**, 4, p. 041305 (2016).
- [43] R. J. Campbell and P. J. Flynn: “A survey of free-form object representation and recognition techniques”, Computer Vision and Image Understanding, **81**, 2, pp. 166–210 (2001).
- [44] U. Imdad, M. Asif, M. T. Ahmad, O. Sohaib, M. K. Hanif and M. H. Chaudary: “Three dimensional point cloud compression and decompression using polynomials of degree one”, Symmetry, **11**, 2, p. 209 (2019).
- [45] J. L. Bentley: “Multidimensional binary search trees used for associative searching”, Communications of the ACM, **18**, 9, pp. 509–517 (1975).
- [46] D. J. Meagher: “Octree encoding: A new technique for the representation, manipulation and display of arbitrary 3-D objects by computer”, Electrical and Systems Engineering Department Rensselaer Polytechnic Institute Image Processing Laboratory (1980).

- [47] F. Tombari, S. Salti and L. Di Stefano: “Performance Evaluation of 3D Keypoint Detectors”, *International Journal of Computer Vision*, **102**, 1-3, pp. 198–220 (2013).
- [48] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan and N. M. Kwok: “A Comprehensive Performance Evaluation of 3D Local Feature Descriptors”, *International Journal of Computer Vision*, pp. 1–24 (2015).
- [49] S. Filipe and L. A. Alexandre: “A Comparative Evaluation of 3D Keypoint Detectors in a RGB-D Object Dataset”, *Proc. of International Conference on Computer Vision Theory and Applications*, pp. 5–8 (2014).
- [50] Y. Zhong: “Intrinsic shape signatures: A shape descriptor for 3D object recognition”, *Proceedings of International Conference on Computer Vision Workshops*, pp. 689–696 (2009).
- [51] S. Salti, F. Tombari and L. Di Stefano: “SHOT: Unique signatures of histograms for surface and texture description”, *Computer Vision and Image Understanding*, **125**, pp. 251–264 (2014).
- [52] R. B. Rusu, N. Blodow and M. Beetz: “Fast point feature histograms (FPFH) for 3D registration”, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 3212–3217 (2009).
- [53] R. B. Rusu, Z. C. Marton, N. Blodow and M. Beetz: “Persistent point feature histograms for 3d point clouds”, *Proc 10th Int Conf Intel Autonomous Syst (IAS-10)*, Baden-Baden, Germany, pp. 119–128 (2008).
- [54] L. Yongshan, K. Dehan, Z. Dandan, G. Xiang and H. Guichun: “A Point Cloud Registration Algorithm Based on Feature Extraction and Matching”, *Mathematical Problems in Engineering*, **2018**, (2018). Article ID 7352691.

- [55] 玉木徹：“回転行列と姿勢推定”，電子情報通信学会技術報告, **109**, 202, pp. 59–64 (2009).
- [56] M. A. Fischler and R. C. Bolles: “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography”, *Communications of the ACM*, **24**, 6, pp. 381–395 (1981).
- [57] R. Schnabel, R. Wahl and R. Klein: “Efficient RANSAC for Point-Cloud Shape Detection”, **26**, 2, pp. 214–226 (2007).
- [58] K. S. Arun, T. S. Huang and S. D. Blostein: “Least-Squares Fitting of Two 3-D Point Sets”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **9**, 5, pp. 698–700 (1987).
- [59] S. Umeyama: “Least-squares estimation of transformation parameters between two point patterns”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**, 4, pp. 376–380 (1991).
- [60] S. Rusinkiewicz and M. Levoy: “Efficient variants of the ICP algorithm”, *Proceedings of International Conference on 3D Digital Imaging and Modeling*, *IEEE Comput. Soc.*, pp. 145–152 (2001).
- [61] J. Salvi, C. Matabosch, D. Fofi and J. Forest: “A Review of Recent Range Image Registration Methods with Accuracy Evaluation”, *Image and Vision computing*, **25**, 5, pp. 578–596 (2007).
- [62] O. van Kaick, H. Zhang, G. Hamarneh and D. Cohen-Or: “A Survey on Shape Correspondence”, *Computer Graphics Forum*, **30**, 6, pp. 1681–1707 (2011).
- [63] 植西一馬, 岩切宗利: “3次元動きベクトルを用いた複数点群間の特徴点マッチング”, 電子情報通信学会技術研究報告, MVE2012-132, pp. 209–214 (2013).

- [64] K. Uenishi and M. Iwakiri: “Efficient Registration Based on Global 3D Motion Vector for Sequential Point Clouds”, Proceedings of IEEE Global Conference on Consumer Electronics, pp. 339–340 (2013).
- [65] 植西一馬, 岩切宗利: “3次元動きベクトル推定による接続点群間レジストレーションの高速化”, 画像電子学会誌, **42**, 5, pp. 642–649 (2013).
- [66] M. Muja and D. G. Lowe: “Fast approximate nearest neighbors with automatic algorithm configuration”, Proceedings of International Conference on Computer Vision Theory and Application, pp. 331–340 (2009).
- [67] M. Muja and D. G. Lowe: “Fast matching of binary features”, Proceedings of Computer and Robot Vision, pp. 404–410 (2012).
- [68] D. A. Simon, M. Hebert and T. Kanade: “Real-time 3-D pose estimation using a high-speed range sensor”, Proceedings of IEEE International Conference on Robotics and Automation, pp. 2235–2241 (1994).
- [69] M. Greenspan and G. Godin: “A nearest neighbor method for efficient ICP”, Proceedings of International Conference on 3-D Digital Imaging and Modeling, pp. 161–168 (2001).
- [70] R. Benjemaa and F. Schmitt: “Fast Global Registration of 3D Sampled Surfaces Using a Multi-z-buffer Technique”, Image and Vision Computing, **17**, 2, pp. 113–123 (1999).
- [71] Y. Chen and G. Medioni: “Object modeling by registration of multiple range images”, Proceedings of IEEE International Conference on Robotics and Automation, pp. 2724–2729 (1992).
- [72] P. J. Neugebauer: “Reconstruction of real-world objects via simultaneous

- registration and robust combination of multiple range images”, *International Journal of Shape Modeling*, **3**, 01n02, pp. 71–90 (1997).
- [73] A. W. Fitzgibbon: “Robust registration of 2D and 3D point sets”, *Image and Vision Computing*, **21**, 13, pp. 1145–1153 (2003).
- [74] J. Santamaría, O. Cordón and S. Damas: “A Comparative Study of State-of-the-art Evolutionary Image Registration Methods for 3D Modeling”, *Computer Vision and Image Understanding*, **115**, 9, pp. 1340–1354 (2011).
- [75] C. K. Chow, H. T. Tsui and T. Lee: “Surface Registration using a Dynamic Genetic Algorithm”, *Pattern recognition*, **37**, 1, pp. 105–117 (2004).
- [76] R. Storn and K. Price: “Differential Evolution– A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces”, *Journal of global optimization*, **11**, 4, pp. 341–359 (1997).
- [77] J. Brest, S. Greiner, B. Boskovic, M. Mernik and V. Zumer: “Self-Adapting Control Parameters in Differential Evolution: A Comparative Study on Numerical Benchmark Problems”, *IEEE Trans. on Evolutionary Computation*, **10**, 6, pp. 646–657 (2006).
- [78] M. Rodrigues, R. Fisher and Y. Liu: “Special Issue on Registration and Fusion of Range Images”, *Computer Vision and Image Understanding*, **87**, 1, pp. 1–7 (2002).
- [79] E. Trucco, A. Fusiello and V. Roberto: “Robust motion and correspondence of noisy 3-d point sets with missing data”, *Pattern recognition letters*, **20**, 9, pp. 889–898 (1999).
- [80] S. Das and P. N. Suganthan: “Differential evolution: A survey of the state-

- of-the-art”, IEEE transactions on evolutionary computation, **15**, 1, pp. 4–31 (2010).
- [81] K. Uenishi and M. Iwakiri: “Virtual Feature Point Extraction from Polyhedral Structure”, Proceedings of IEEE International Symposium on Intelligent Signal Processing and Communication Systems, pp. 519–524 (2013).
- [82] 植西一馬, J.Sandoval, 岩切宗利, 田中清: “平面形状を活用した3次元仮想特徴点のための特徴量記述法”, 第44回画像電子学会年次大会予稿集, pp. R1–2 (2016).
- [83] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “VKOP: 3次元幾何構造に適した仮想特徴点検出器及びその特徴記述子”, 画像電子学会誌, **46**, 2, pp. 283–297 (2017).
- [84] H. Gonzalez-Jorge, B. Riveiro, E. Vazquez-Fernandez, J. Martínez-Sánchez and P. Arias: “Metrological evaluation of microsoft kinect and asus xtion sensors”, Measurement, **46**, 6, pp. 1800–1806 (2013).
- [85] ICL-NUIM RGB-D Benchmark Dataset: <http://www.doc.ic.ac.uk/~ahanda/VaFRIC/iclnuim.html>. 最終確認: 2021年4月20日.
- [86] A. Handa, T. Whelan, J. McDonald and A. J. Davison: “A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM”, Proc. of IEEE International Conference on Robotics and Automation, pp. 1524–1531 (2014).
- [87] R. B. Rusu: “Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments”, PhD thesis, Computer Science department, Technische Universitaet Muenchen, Germany (2009).
- [88] R. B. Rusu and S. Cousins: “3D is here: Point Cloud Library (PCL)”, Pro-

- ceedings of IEEE International Conference on Robotics and Automation, IEEE, pp. 1–4 (2011).
- [89] P. Papadakis: “The canonically posed 3D objects dataset”, Eurographics Workshop on 3D Object Retrieval, pp. 33–36 (2014).
- [90] I. Sipiran and B. Bustos: “Harris 3D: a robust extension of the harris operator for interest point detection on 3D meshes”, *The Visual Computer*, **27**, 11, pp. 963–976 (2011).
- [91] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “平面形状の安定性を活用した尤度付き VKOP”, 第 45 回画像電子学会年次大会予稿集, pp. R2–1 (2017).
- [92] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “平面尤度を用いた VKOP の再現性評価”, *画像電子学会誌*, **50**, 3, pp. 351–361 (2021).
- [93] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “尤度付き VKOP 抽出法の性能に関する一考察”, 第 286 回画像電子学会研究会予稿集, pp. 41–44 (2018).
- [94] K. Uenishi, J. Sandoval, M. Iwakiri and K. Tanaka: “Likelihood-Based VKOP Detection with Reliability of Estimated Planes”, *Proc. of International Workshop on Image Electronics and Visual Computing*, pp. 1A–1 (2019).
- [95] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “VKOP 特徴点に適した平面推定のパラメータ決定法”, 第 48 回画像電子学会年次大会予稿集, pp. R2–2 (2020).
- [96] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “平面の面積を活用した

- 尤度付き VKOP 抽出法”, 第 284 回画像電子学会研究会予稿集, pp. 14–18 (2018).
- [97] K. Uenishi, M. Iwakiri and K. Tanaka: “Efficient Point Clouds Registration Based on Differential Evolution with Keypoint Patches”, Proc. of International Workshop on Image Electronics and Visual Computing, pp. 1C–1 (2014).
- [98] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “キーポイントパッチ抽出法を用いた高能率な進化計算による 3 次元点群レジストレーション”, 画像電子学会誌, **47**, 2, pp. 154–166 (2018).
- [99] 植西一馬, 岩切宗利, 田中清: “進化計算による 3 次元レジストレーションのための高能率なりサンプリング”, 情報処理学会第 78 回全国大会講演論文集, pp. 2C–4 (2016).
- [100] K. Uenishi, J. Sandoval, M. Iwakiri and K. Tanaka: “Improved Keypoint Patches in Evolutionary Point Clouds Registration”, Proc. of International Workshop on Image Electronics and Visual Computing, pp. 4A–1 (2017).
- [101] M. Matsumoto and T. Nishimura: “Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator”, ACM Transactions on Modeling and Computer Simulation, **8**, 1, pp. 3–30 (1998).

研究業績

学会誌論文

- [1] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “VKOP: 3次元幾何構造に適した仮想特徴点検出器及びその特徴記述子”, 画像電子学会誌, **46**, 2, pp. 283–297 (2017). 【優秀論文賞】
- [2] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “キーポイントパッチ抽出法を用いた高能率な進化計算による3次元点群レジストレーション”, 画像電子学会誌, **47**, 2, pp. 154–166 (2018). 【最優秀論文賞】
- [3] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “平面尤度を用いたVKOPの再現性評価”, 画像電子学会誌, **50**, 3, pp. 351–361 (2021).

国際会議（口頭発表）

- [1] K. Uenishi, M. Iwakiri and K. Tanaka: “Efficient Point Clouds Registration Based on Differential Evolution with Keypoint Patches”, Proc. of International Workshop on Image Electronics and Visual Computing, pp. 1C–1 (2014). 【Excellent paper award】
- [2] K. Uenishi, J. Sandoval, M. Iwakiri and K. Tanaka: “Improved Keypoint Patches in Evolutionary Point Clouds Registration”, Proc. of International Workshop on Image Electronics and Visual Computing, pp. 4A–1 (2017).
- [3] K. Uenishi, J. Sandoval, M. Iwakiri and K. Tanaka: “Likelihood-Based VKOP Detection with Reliability of Estimated Planes”, Proc. of International Workshop on Image Electronics and Visual Computing, pp. 1A–1 (2019).

国内学会等（口頭発表）

- [1] 植西一馬, 岩切宗利, 田中清: “進化計算による 3 次元レジストレーションのための高能率なりサンプリング”, 情報処理学会第 78 回全国大会講演論文集, pp. 2C-4 (2016).
- [2] 植西一馬, J.Sandoval, 岩切宗利, 田中清: “平面形状を活用した 3 次元仮想特徴点のための特徴量記述法”, 第 44 回画像電子学会年次大会予稿集, pp. R1-2 (2016). 【研究奨励賞】
- [3] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “平面形状の安定性を活用した尤度付き VKOP”, 第 45 回画像電子学会年次大会予稿集, pp. R2-1 (2017).
- [4] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “平面の面積を活用した尤度付き VKOP 抽出法”, 第 284 回画像電子学会研究会予稿集, pp. 14-18 (2018).
- [5] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “尤度付き VKOP 抽出法の性能に関する一考察”, 第 286 回画像電子学会研究会予稿集, pp. 41-44 (2018).
- [6] 植西一馬, サンドバル・ハイメ, 岩切宗利, 田中清: “VKOP 特徴点に適した平面推定のパラメータ決定法”, 第 48 回画像電子学会年次大会予稿集, pp. R2-2 (2020). 【研究優秀賞】